# Computational Mechanisms of Language Understanding and Use in the Brain and Behaviour

by

Ivana Kajić

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2020

## Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

| | |
|---|---|
| External Examiner | Suzanne Stevenson<br>Professor,<br>Department of Computer Science,<br>University of Toronto |
| Supervisor | Chris Eliasmith<br>Professor,<br>Deptartment of Philosophy and<br>Department of Systems Design Engineering,<br>University of Waterloo |
| Internal Member | Jesse Hoey<br>Professor,<br>David R. Cheriton School of Computer Science,<br>University of Waterloo |
| Internal Member | Kate Larson<br>Professor,<br>David R. Cheriton School of Computer Science,<br>University of Waterloo |
| Internal-External Member | Olga Vechtomova<br>Associate Professor,<br>Faculty of Engineering,<br>University of Waterloo |

## Author’s Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

## Statement Of Contributions

Chapters in this thesis that are based on published research that I conducted in collaboration with researchers in the Computational Neuroscience Research Group (CNRG) at the University of Waterloo, University of Plymouth, and DeepMind are outlined below. All researchers whose affiliations are not mentioned are current or former members of CNRG, headed by my supervisor, Dr. Chris Eliasmith. In all cases, I was leading the research efforts, and was involved in all aspects of research including, but not limited to, formulation and identification of the scientific problem, devising and implementation of methods, conceiving of and conducting experiments, analyzing data, interpreting results, writing, and finalization of respective research papers.

The work presented in Chapter 3 on properties of semantic networks is based on the technical report Kajić and Eliasmith (2018). Chapter 4 on biologically plausible modelling of the RAT is based on several published papers: Kajić and Wennekers (2015), Kajić et al. (2016) and Kajić et al. (2017b). Kajić and Wennekers (2015) was published in Proceeding of the NIPS Workshop on Cognitive Computation, co-located with the 29th Annual Conference on Neural Information Processing Systems. This was joint work with Dr. Thomas Wennekers from University of Plymouth, who was involved in all aspects of that research. Kajić et al. (2016) was published in Proceedings of the 38th Annual Conference of the Cognitive Science Society, and it was a joint work with contributions from Dr. Jan Gosmann, Dr. Terrence C. Stewart, Dr. Thomas Wennekers, and Dr. Chris Eliasmith, all of whom contributed to the journal article published in Frontiers in Psychology (Kajić et al. 2017b). The model and experiments presented in Kajić et al. (2016) and Kajić et al. (2017b) were conceived of by all of the authors, while Dr. Gosmann also contributed to the model implementation, data analysis, and writing of the manuscript. Chapter 5 is based on research in Kajić et al. (2017a), published in Proceedings of the 39th Annual Conference of the Cognitive Science Society. This research was done in collaboration with Dr. Jan Gosmann, Dr. Brent Komer, Mr. Ryan W. Orr, Dr. Terrence C. Stewart, and Dr. Chris Eliasmith. Finally, parts of Chapter 6 are based on research conducted during my internship at DeepMind Montreal in collaboration with Mr. Eser Aygün and Dr. Doina Precup, and published in Kajić et al. (2020).

# Abstract

Linguistic communication is a unique characteristic of intelligent behaviour that distinguishes humans from non-human animals. Natural language is a structured, complex communication system supported by a variety of cognitive functions, realized by hundreds of millions of neurons in the brain. Artificial neural networks typically used in natural language processing (NLP) are often designed to focus on benchmark performance, where one of the main goals is reaching the state-of-the-art performance on a set of language tasks. Although the advances in NLP have been tremendous in the past decade, such networks provide only limited insights into biological mechanisms underlying linguistic processing in the brain.

In this thesis, we propose an integrative approach to the study of computational mechanisms underlying fundamental language processes, spanning biologically plausible neural networks, and learning of basic communicative abilities through environmentally grounded behaviour. In doing so, we argue for the usage-based approach to language, where language is supported by a variety of cognitive functions and learning mechanisms. Thus, we focus on the three following questions: How are basic linguistic units, such as words, represented in the brain? Which neural mechanisms operate on those representations in cognitive tasks? How can aspects of such representations, such as associative similarity and structure, be learned in a usage-based framework?

To answer the first two questions, we build novel, biologically realistic models of neural function that perform different semantic processing tasks: the Remote Associates Test (RAT) and the semantic fluency task. Both tasks have been used in experimental and clinical environments to study organizational principles and retrieval mechanisms from semantic memory. The models we propose realize the mental lexicon and cognitive retrieval processes operating on that lexicon using associative mechanisms in a biologically plausible manner. We argue that such models are the first and only biologically plausible models that propose specific mechanisms as well as reproduce a wide range of human behavioural data on those tasks, further corroborating their plausibility.

To address the last question, we use an interactive, collaborative agent-based reinforcement learning setup in a navigation task where agents learn to communicate to solve the task. We argue that agents in such a setup learn to jointly coordinate their actions, and develop a communication protocol that is often optimal for the performance on the task, while exhibiting some core properties of language, such as representational similarity structure and compositionality, essential for associative mechanisms underlying cognitive representations.

# Acknowledgements

Writing this thesis would have not been possible without the help, support and encouragement I received from a number of different people throughout the years. First, I would like to express my deepest gratitude to my supervisor, Chris Eliasmith. I am thankful for his support, numerous insightful discussions and for giving me the freedom to explore and undertake different research directions. Terry Stewart had an integral role in shaping the direction of my research interests, and I am grateful for his continuous encouragement and presence in all aspects of my graduate career. I have also tremendously benefited from the great company afforded by the current and former members of the CNRG lab—thank you all for the fun discussions about the brains and for letting me frequently pick your brains. I would also like to extend my gratitude to the members of my examining committee: Jesse Hoey, Kate Larson, Olga Vechtomova and Suzanne Stevenson, for their thought-provoking questions and for providing invaluable feedback that helped improve this thesis.

My time at the University of Waterloo would not have been as fulfilling and intellectually rewarding without many other members of its community. I want to thank Jesse Hoey and Robin Cohen for continuously providing feedback on my research. I consider myself lucky to have had the opportunity to work with Paul Thagard, and benefit from his rich knowledge and wisdom on a variety of scientific and non-scientific topics. I am also thankful to Tobias Schröder, for a research collaboration that helped set me on this path. I enjoyed working with WICS and thank Jo Atlee for giving me an opportunity to do so. I am happy to have crossed paths with many wonderful students at the university, and would like to thank Sean, Jasmine, Sajed, Alan and Ryan for their help with the workshops, and to Florian, Bohdan and Barbara for their company. I am grateful for the support I received at the institutional level, in particular, from the David R. Cheriton School of Computer Science and the Faculty of Math. One of the most memorable experiences during my PhD was my participation in the Telluride Neuromorphic Workshop, and I thank the organizers for creating such a great community. In Montreal, I want to thank Eser Aygün and Doina Precup, as well as the rest of the DeepMind team for an amazing research environment. Benoît provided a much needed perspective on "how to finish your PhD" which I greatly appreciate.

Finally, thanks to all my friends scattered around the world, who remained close despite the distance and time zone differences: Ursel, Colleen, Ivana B., Ilaria, Helen, Thomas, Mihaela, and Pippijn. Thanks to Giles who made sure I kept on running, despite the rain, snow and all the wonders of Canadian winters. I thank my family for their support and love, and especially my mom for all her sacrifices and hard work.

# Table of Contents

# List of Tables

# List of Figures

# 1 | Introduction

Linguistic communication is a unique characteristic of intelligent behaviour that distinguishes humans from non-human animals. A sophisticated communication system in the form of a language that is acquired, highly structured, and arbitrarily conventionalized is a quintessential aspect of human cognition. Language is powerful—it enables us to talk about the past, present and future, transmit knowledge across generations, tell jokes, make friends and enemies, plan, organize, collaborate, engage in scientific endeavours, develop and maintain culture, among numerous other things.

The human capacity for language is closely tied to other intelligent behaviours and cognitive abilities (Tomasello 2009; Elman et al. 1997; Clark 1996). Even though the thousands of different languages that exist in the world today are remarkably diverse in terms of their topology and forms (e.g., spoken, written or signed), they all share some important characteristics. First, natural language is known for its *unboundedness*, as there are infinite ways to construct meaningful sentences and expressions from a limited set of units such as symbols or words. Thus, humans possess the ability to understand sentences they have never encountered before, and, at the same time, they are also able to generate new sentences they have never uttered before. This is possible due to the highly structured nature of language. Meaningless units such as individual letters are combined together to form larger, meaningful units such as morphemes and words. Words are further combined into sentences, which are larger meaningful constructions. This property of language, where the meaning of a whole construction depends on the meaning of its parts, is known as *compositionality* (Fodor and Pylyshyn 1988; Frege 1892). Compositionality is considered to be one of the core properties of language (Bohn et al. 2019; Brighton and Kirby 2006), giving it its expressive power.

Second, language is supported by a wide variety of cognitive processes that start to develop in childhood, and continue to be shaped into adulthood (Tomasello 2009). It is argued that the capacity for language evolved from existing more general cognitive and motor mechanisms. For example, from those mechanisms related to sequential processing and motor execution (Kolodny and Edelman 2018), or those involved in combinations of gesture and facial movements (Hewes 1973; Rizzolatti and Arbib 1998). Since a wide variety of brain areas and structures are engaged in the processes of language acquisition, production and understanding (Ullman 2004), language is realized by hundreds of millions of neurons in the brain. It is therefore reasonable to assume that features of natural language are shaped by constraints imposed by cognitive processes and biological systems that support it.

Given the ubiquitous importance of language in a variety of human behaviours, it is understandable that language has been central to artificial intelligence (AI) research since its early days (McCarthy et al. 1955; Newell and Simon 1975; Quillian 1967). As we discuss in Chapter 2, language is recognized as an important trait of intelligent machines, and has been traditionally seen as a plausible means by which machines can realize a variety of tasks, or even learn to improve themselves (McCarthy et al. 1955). While AI researchers have always been focussed on reproducing aspects of intelligent behaviours, early AI researchers also placed a strong emphasis on cognitive processes and understanding of mechanisms giving rise to such behaviours (Newell and Simon 1975).

Today, the capacity of AI to use natural language has been predominantly studied within the field of natural language processing (NLP). Current NLP research is typically focussed on specific language tasks such as machine translation, question answering, dialogue generation, named entity recognition, text summarization, or some combination thereof. Particulars of general language use, acquisition, or underlying cognitive processes that support language are less commonly studied within NLP. Often, models performing NLP tasks are trained on vast amounts of textual data, typically with a goal of achieving state-of-the-art results on benchmark language tasks (Mikolov et al. 2013a; Mikolov et al. 2013b; Peters et al. 2018; Vaswani et al. 2017; Devlin et al. 2018; Lewis et al. 2019; Raffel et al. 2019; Radford et al. 2019; Brown et al. 2020).

The advances in NLP have been tremendous in the past decade, showing that such systems are capable of sophisticated language processing, sometimes even exceeding human performance on selected tasks (Rajpurkar et al. 2018). The success of such models also extends beyond the field of NLP, as distributed semantic representations produced by such models have found a wide array of applications in cognitive science. Such representations have been used to model aspects of cognitive processes related to semantic processing, such as word similarity judgments (Baroni et al. 2014; De Deyne et al. 2016), semantic priming (Mandera et al. 2017), and word associations (Utsumi 2015; Nematzadeh et al. 2017). Moreover, they have also been used to decode meanings of words, sentences and narratives from recordings of neural activity (Huth et al. 2016; Pereira et al. 2018; Anderson et al. 2019). In Chapter 3, we compare semantic networks constructed from two distributed representations commonly used in NLP to those constructed from human word association data, and show that while these representations are comparable in many aspects, they also differ in important ways.

Interpreting the workings of state-of-the-art NLP models, as well as understanding their relationship to linguistic structure, is often challenging (Rogers et al. 2020). While different techniques have been used to uncover aspects of linguistic structure learned by those models (Linzen and Baroni 2020), their inner workings and architecture typically do not resemble cognitive or biological mechanisms underlying linguistic processing. Although aspects of representations learned by such models can be regarded as cognitively plausible (Günther et al. 2019), the contributions of such models to our understanding of linguistic processing at the level of neural networks in the brain have been limited.

In this thesis, we propose an approach to the computational study of language understanding and use that is based on biologically realistic and behaviourally motivated modelling approaches. Biologically realistic models are typically constrained to the computational mechanisms underlying biological systems. Here, we rely on the methods of the Neural Engineering Framework (NEF; Eliasmith and Anderson 2003) and the Semantic Pointer Architecture (SPA; Eliasmith 2013) to build biologically constrained models of semantic processing in the brain. The NEF and SPA have been used to model a diverse set of neural systems and cognitive functions, such as serial working memory (Choo 2010), action selection (Stewart et al. 2012), time cells (Voelker and Eliasmith 2018), affective processing (Kajić et al. 2019), and biologically plausible spatial representations (Komer et al. 2019; Dumont and Eliasmith 2020). As well, the NEF and SPA have been used to integrate such models, resulting in the world's largest functional brain model (Eliasmith et al. 2012; Choo 2018).

We use these methods to address several tasks investigating aspects of language function. The tasks we focus on have been used in psycholinguistics, cognitive science and neuroscience, including clinical and research settings, to study how the brain stores, organizes and retrieves semantic information. In contrast to commonly used tasks in NLP, such tasks are characterized by limited amounts of experimental data, often acquired under different experimental conditions. While in some tasks responses can be characterized as correct or incorrect, in others, they are generated according to task rules and thus no precise accuracy metric can be derived. Instead, the performance on tasks is evaluated by comparing model responses with the human data, typically by using a variety of analysis methods devised to describe properties of the human data. The effort in this modelling approach is placed on interpreting the inner workings of the model in the context of relevant theories such as those in psycholinguistics, cognitive science or neuroscience. As such, these models can be used to propose specific mechanisms that might be realized by the brain, as well as to make predictions for experimental conditions for which human data may not be available. This thesis therefore argues for AI methods that simulate aspects of intelligent behaviour by constraining models of language comprehension and use in a way that is consistent with experimental observations of aspects of language function in the brain and behaviour.

Over the course of the thesis, we use this brain and behaviour perspective to propose an integrative approach for studying aspects of language function across different levels of explanation, consistent with the usage-based view of language that emphasizes the role of a variety of cognitive functions in language. Using the example of two semantic memory tasks, Remote Associates Test (RAT; Mednick 1962) in Chapter 4, and semantic fluency task (Bousfield and Sedgewick 1944; Troyer et al. 1997; Hills et al. 2012a) in Chapter 5, we examine biologically plausible models of word associations and related search processes in the brain. We propose models that implement the mental lexicon storing knowledge about words, relationships between them, as well as semantic search processes operating on such representations. We demonstrate that our models provide a good match with human data on those tasks, and examine how changes to the model parameters affect task performance. Then, in Chapter 6, we address the question of how aspects of such representations, including basic linguistic structure, can be learned in a usage-based, collaborative multi-agent reinforcement learning framework.

Thus, the goal of this thesis is twofold. First, it is to propose biologically constrained computational mechanisms underlying aspects of language function. Proposed mechanisms are constrained at the neural level by computations realized with groups of inter-connected neurons and biologically motivated parameters, and, at the cognitive level, by cognitive mechanisms proposed by theories of semantic processing in psycholinguistics and cognitive science. Such mechanisms are also general, meaning that the same or similar mechanisms are used in other models of higher cognitive functions implemented using the same computational frameworks. This consistency is important in a biological sense, since language is processed in different brain regions, some of which also process non-linguistic information. Second, the goal is to study how aspects of core linguistic features can be learned in a usage-based computational framework. Language is typically studied as an end product and, given that established human communities have little incentive to emerge novel languages (Galantucci 2005), understanding factors that shape linguistic structure can be challenging. We contribute to the large body of simulation-based approaches to language in AI by showing that artificial agents in a reinforcement learning setup offer a useful testbed for experimenting with conditions that give rise to interesting linguistic properties.

Since language in humans and artificial agents is a topic of research interest in many different domains, it is important to stress some of the aspects of language function that are not addressed in this thesis. First, we do not aim to model the anatomy of language in the brain. As mentioned previously, language processing is distributed across different brain areas involved in various language-related tasks, such as perceptual analyses of the linguistic input, speech, syntax, semantics, reading, comprehension, to name a few. Instead, we focus on specific aspects of semantic processing related to the mental lexicon and related processes. Second, we do not address the process of language acquisition, which, while fundamental to the human capacity for language, would likely require a comprehensive, functional model of the brain of a developing child. The aspects of linguistic emergence that we address with agent-based simulations in Chapter 6 can be considered a step in the direction of understanding learning processes, some of which may play an important role in language acquisition. Finally, because our research aims to understand the relationship between the brain and linguistic behaviours, we do not focus on benchmark tasks in NLP. Rather, we strive to understand how cognitive and biological constraints affect language comprehension and use, and hope that such efforts contribute to the creation of AI systems with communicative abilities that consider human linguistic capacity.

# 2 | Background

Understanding and reproducing human linguistic abilities has been central to different research domains that investigate aspects of this unique trait. On the one hand, researchers want to understand the nature of language, its characteristics, as well as principles and mechanisms that guide language learning and production. On the other hand, there are ongoing efforts in artificial intelligence to develop systems that are able to understand human language, often without the need to establish a relationship between such artificial systems and biological systems underlying natural language. For those interested in human language as a natural phenomenon, several distinct but overlapping methods are used. The study of language form, meaning and context is the primary goal in linguistics, while language acquisition and development is typically studied in the context of psycholinguistics. Taking a more mechanistic perspective, cognitive neuroscience is concerned with the relationship between the brain and the linguistic ability.

In contrast to seeking answers to questions on the human capacity for language, artificial intelligence, and natural language processing (NLP) more specifically, investigate how to replicate aspects of human linguistic performance without much reference to the linguistic or cognitive aspects studied by linguists, psychologists and neuroscientists. In this thesis, we argue for an approach to the study of language that spans explanations across scientific domains, by integrating neuroscientific observations at the level of brain functions, up to the linguistic behaviour consistent with aspects of human communication. In this chapter, we outline benefits of such an approach and its significance for language research in artificial intelligence.

We provide a survey of theoretical contributions and findings in language research, starting from those in linguistics and philosophy, and continuing with psycholinguistics and neuroscience. The final section on computational approaches to language highlights various methods used for language modelling, some of which are mentioned for their historical value, while others are included as powerful state-of-the-art modelling methods on specific language tasks at the time of writing. Finally, the section concludes with a detailed explanation of methods that are used in the remainder of this thesis to explore the relationship between biologically realistic neural networks and fundamental language processes, along with a comparison of these methods with others used in natural language modelling.

## 2.1 Theoretical Approaches to the Study of Language

One of the basic learning principles in humans and other animals is conditioning, where certain behaviours are acquired as a result of reinforcement of such behaviours through some feedback signal. In particular, one of the influential proposals of how humans acquire a wide variety of intelligent behaviours, including language, was that of operant conditioning pioneered by Skinner (1957). Operant conditioning is a form of associative learning where desired behaviours are reinforced with positive feedback signals. The gist of this idea can be illustrated with the following example: if we want to teach a child to use the words "Yes, please", then we wait for the child to utter the words as we ask them whether they would want their meal or a toy. Skinner postulated that such reinforced learning underlies the development of human language capacity.

While such reinforcing mechanisms explain many behaviours, and may account for some linguistic behaviours, the theory was generally considered inadequate as an explanation of language learning. In particular, the *nativist* approach to language advocated by Noam Chomsky (Chomsky 1965; Chomsky 2007) postulated that children cannot learn the complexity of linguistic structure just from associative mechanisms and sensory experience. He argued for an innate language faculty that somehow supports syntax learning. However, evidence largely originating from developmental and neuroscientific studies, that accumulated since such first proposals, supports an alternative view of language function. According to such a view, although language is unique to humans, it relies on mechanisms that are important for a wide variety of cognitive skills, including those necessary for social interactions. Such view of language, where language is situated in the context of cognition as well as social interactions and the environment is known as the *usage-based* view of language. After all, humans are social species and the normal developmental trajectory requires children's learning to be supported by their caregivers. While the usage-based view of language does not entirely rule out the possibility that some aspects of language function are innate, it rejects Chomsky's ideas on innateness by demonstrating that language function is acquired and developed jointly with a range of more general socio-cognitive skills, effectively bypassing the need for innateness.

### 2.1.1 Language as an Innate Ability

The nativist approach to language assumes an innate language faculty underlying linguistic competence, with a genetically determined component for universal grammar (UG). Thus, according to the nativist theory, the function of human language structure is separated from language acquisition, which is treated as a "parameter setting" (Chomsky 2007). This theory, pioneered by Chomsky in 1960s, postulates the innateness of UG based on the idea of language universals (i.e., features of language that are common among languages of the world) and the poverty of the stimulus argument. Poverty of stimulus describes children's linguistic learning experience, where children are able to acquire language despite finite exposure to linguistic data during their development. Therefore, proponents of nativism argue that an innate propensity for grammar learning must exist that allows children to become competent language users with so little data and no formal instruction. This innate wiring, hypothesized to have some form of neural correlates, Chomsky labelled as *Language Acquisition Device* (LAD) that "sets limits on the attainable languages" (Chomsky 2007). Then, according to Chomsky, experience determines the features of a specific language. He argues for the distinction between *competence*, which is the theoretical, idealized ability to generate grammatically correct linguistic output, and from *performance*, which characterizes actual, ordinary language use with all of its imperfections.

However, these views on UG and dedicated LAD were challenged on several grounds. The Chomskian theory of what constitutes language universals has seen multiple revisions, where in each revision some genetically determined grammar universals were excluded. This typically occurred as more data became available on languages beyond the Indo-European language group as well as data on sign languages. For instance, Evans and Levinson (2009) challenge the existence of universals such as major lexical categories (such as adjectives and adverbs) as well as major phrasal categories (such as a noun phrase or a verb phrase), providing examples of less known languages that do not conform to such syntactic patterns. Currently, the theory postulates only one universal "tool" present among all languages by stating that recursion is "the only uniquely human component of the faculty of language" (Hauser et al. 2002). However, Evans and Levinson (2009) point out that "many languages show distinct limits on recursion in this sense, or even lack it altogether." It is worth noting that the debate on language universals remains ongoing, with proponents on both sides of the "nature versus nurture" spectrum. This does not mean that all researchers reject the existence of such universals; instead, they question the need for such universals to be genetically determined, and some, consider them as tendencies and statistical regularities observed in the languages of the world, rather than fixed rules.

The poverty of the stimulus rationale is rejected by many researchers in developmental psychology, linguistics and cognitive science, who argue that the case for the nativist approach is poor in light of evidence from developmental and neuroscientific observations. In particular, they state that children have at their disposal powerful learning mechanisms, such as categorization and statistical learning, integrated with other cognitive and social-cognitive skills, that allow them to acquire complex linguistic expressions and constructions (Tomasello 2009; Evans and Levinson 2009). This view of language emphasizes the contextualized and malleable nature of language development, both from the phylogenetic and ontogenetic perspective.

Thus, empirical evidence for the poverty of the stimulus argument and universal grammar supported by the genetically defined LAD, has been weak, and different lines of evidence have been provided in favour of developmental explanations.[1] Consequently, these ideas were rejected by many researchers in favour of usage-based view of language that explains human capacity for language in the context of learning of more general cognitive and social domains. According to such views, there is no need for a "pre-programmed" faculty for language structure, as children acquire it jointly with other skills. Furthermore, proponents of this view do not distinguish between the competence and performance, as they find no empirical evidence to substantiate such a distinction.

### 2.1.2 Usage-based Approaches

In contrast to nativist views of language that aim to explain linguistic ability in terms of an innate predisposition for universal grammar, usage-based approaches highlight the role of context, environment, socio-cultural factors and cognition in language use and development. In usage-based views of language, linguistic skill is closely tied to other cognitive skills, and the acquisition of language skills is a stage of ontogeny related to the development of other skills. Historically, language use and its relevance for meaning, or semantics, has been studied within pragmatics, the theory of language use within philosophy. Rather than focusing on formal, idealized view of language as a set of fixed rules, pragmatics studies ways in which the context contributes to the meaning.

[1] For a detailed discussion on this topic see Elman et al. (1997, Chatper 7, p. 371–391)

One of the most notable proponents of the view that meaning is derived from *language use* was the philosopher Ludwig Wittgenstein (1953), who emphasized the contextual versatility of language in his later works, perhaps best summarized by the following metaphor:

> "Language is a labyrinth of paths. You approach from one side and know your way about; you approach the same place from another side and no longer know your way about."
>
> Ludwig Wittgenstein (1953, §203., p. 88e)

Wittgenstein claimed that a word's meaning is determined by its use in a language. A similar view had been earlier championed by Frege (1884), who postulated that a word has a meaning only in the context of a sentence. Wittgenstein explicated usage-based nature of language through *language games*, different models of discourse that emphasize aspects of language use. He uses numerous examples of such games to argue that the study of word meaning or sentence meaning without characterizing their contexts is futile.

Language as a coordination problem was first proposed by Lewis (1969), where a solution to such a problem is *convention*. According to Lewis, conventions are behavioural regularities in action that serve interests of those involved. They thus persist in the society because they are solutions to a recurrent coordination problem. He points to a diverse nature of such coordination problems. For example, a situation where two individuals greet each other is a form of coordination since they perform actions (e.g., bowing or shaking hands) in accordance with conventions of their culture. To Lewis (1969), language is also a coordination problem as linguistic units are given conventional meanings in their use. Conventions are tied to a community and culture, and are acquired by being a part of that community.

Clark (1996) argues for the study of language use as a form of joint, coordinated action relying on individual as well as social processes. He points out that words and sentences are types of signals, and when abstracted away from the context of usage, describing them is a mere description of conventions for their use in speech communities. While such conventions provide important aspects of context as established by the community, they do not provide a comprehensive view of meaning resulting from language use. Instead, utterances used in conversations, discourse, books, and ordinary language more generally contribute to providing the context beyond conventions. For example, Grice (1975) distinguishes between *natural* meaning and *non-natural* meaning. Natural meaning arises from natural events, and, roughly speaking, can be understood as a statement of a fact or an observation. As an example of natural meaning, Grice (1975) uses the sentence “Those spots mean (meant) measles”, where measles is a naturally occurring event manifested through a patient’s skin reaction. In contrast, if a doctor waves his hand through a window to indicate to parents that their child has measles, this is a signal that has been agreed upon by interlocutors in a specific situation and is thus considered non-natural—it only carries meaning for those involved. Ordinary language use often relies on such non-natural meanings, and Grice (1975) further elaborates a type of non-natural meaning that is *speaker’s meaning*. The study of speaker’s meaning is a basis for “intention-based semantics” where the meaning of an utternace, according to Grice, has two components: that of what is being said, and that of what is being implicated. For example, if I am considering visiting a newly opened restaurant and I ask a friend of mine who has eaten there to tell me about the experience, and they respond with “It was alright, the service was friendly,” I might assume mediocre food quality. Therefore, by avoiding to mention food, my friend might be implying that they did not enjoy it. This kind of statement is what Grice called a *conversational implicature*, characterized by typically being non-conventional and requiring the listener to infer the meaning.

Further, Clark (1996) emphasises one other important aspect of language use, that of *common ground* or *common knowledge*. An important component thereof, he argues, is *self-awareness*, or the awareness of information that each individual participating in a communicative exchange has. Although Clark himself does not refer to it in such a way, this ability is related to *theory of mind* (Premack and Woodruff 1978), the ability of individuals to attribute mental states to others and understand them in relation to their own.

Consistent with usage-based views of language is an approach to linguistic theorizing called cognitive linguistics. Cognitive linguistics views language as a non-autonomous cognitive faculty, where knowledge of language emerges from language use (Croft and Cruse 2004). This approach has been particularly favored by some artificial intelligence researchers who have investigated how artificial agents can learn syntax, grammars and lexicons (Cangelosi and Parisi 2012; Steels 2004). Often, to study the emergence of linguistic competence, they turn to cognitive science and developmental psychology to simulate constraints and environmental pressures that language users or learners (e.g., infants and young children) face during the language acquisition processes. Cognitive linguists aim to understand linguistic competence in terms of different cognitive functions and their underlying neural mechanisms (Gallese and Lakoff 2005), thus attempting to link different levels of explanation, ranging from cognitive processing in the brain, to behaviour and culture. In the next section, we provide an overview of cognitive, developmental and neuroscientific bases of language that are important for the understanding of environmental constraints and conditions faced by language learners.

## 2.2 Language in the Brain and Behaviour

### 2.2.1 How do Humans Learn Language?

Usage-based linguistics extends pragmatics to the broader context of language use that includes language acquisition and development. In contrast to generative linguistics that portrays linguistic competence as an innate ability, usage-based language theories consider the joint development of language and other cognitive and socio-cognitive skills. Such views are supported by neuroscientific evidence, showing that language is enabled by general-purpose brain mechanisms both in children acquiring their native language and adults learning an additional language (Hamrick et al. 2018; Ullman 2004). Moreover, the theory is aligned with evolutionary views on language postulating that human capacity for language evolved by "'hijacking' of existing cognitive mechanisms related to sequential processing and motor execution" (Kolodny and Edelman 2018).

According to Tomasello (2009), two important general purpose skills are intention-reading and categorization (also referred to as pattern-finding). Intention-reading relies on the development of theory of mind, which allows humans to recognize and understand the mental states of others, as well as comparison of such states to theirs. In the context of language, having a theory of mind underlies the development of a joint attentional frame necessary for communication, which is the ability to share attention with others and refer to specific objects, events, and the environment. In infants, this ability typically develops between 9 and 12 months of age. It is assumed that such skills underlie abstraction processes such as analogy and more complex communicative processes. Pattern-finding skills are important for detecting similarities in conceptual categories, sensory-motor schemas, or behavioural sequences. Such skills emerge early in human development, some of them even prelinguistically, and are generally believed to be evolutionarily old and shared in some form by all primates (Tomasello 2009).

According to the usage-based view, intention-reading and categorization underlie the process of *grammaticalization*—the development of structured linguistic constructions. In children, this process occurs gradually, first starting with the contextualized word learning, and then proceeding with more complex, grammatical patterns of word usage. Between 12 and 16 months of age, the process is slow and prone to errors. Usually, during this period children learn two to three novel words per week (Fenson et al. 1994). Word learning at this stage is facilitated by interactions with a parent or a caregiver, who directs the attention of a child to an action, object or an event, and provides an associated label. This kind of learning is known as learning by ostension, as explicit naming and pointing play a crucial role when conveying the meaning of a word.

The second year of life is characterized by a more rapid learning phase in which children learn eight or nine words per week (Bloom 1976). During this period, the accelerated word learning capacity leads to a vocabulary expansion, a phenomenon also known as the *vocabulary burst*. It is assumed that the emergence of grammar and combinatorial speech plays an important role in the acceleration of the learning rate (Anisfeld et al. 1998). At this stage, however, most words are learned from sentential context introduced through conversations, and, later in life, through reading (Gleitman 1990; Goodman et al. 1998). This represents a shift in utilized learning mechanisms, as up to that stage most of word learning happened by ostension. The context in which a word occurs provides important syntactic and semantic cues that help to convey the meaning of the word. As learning mechanisms become more developed, the word learning process becomes less dependent on external supervision or guidance. The rate of learning slows down towards the end of the preschool period, when children learn one novel word per week (Fenson et al. 1994). Therefore, grammaticalization and resulting language competence are a result of psychological and socio-communicative processes, highly dependent on learning mechanisms that develop through interaction with the environment and others.

### 2.2.2 How does the Brain Support Language?

In contrast to some other functions such as vision, memory or motor control, there are only limited insights to be gained from studying animal communication and animal brain function in explaining human capacity for language. In addition, many aspects of language function, such as language production and perception, depend on many different brain areas such as motor cortex and subcortical structures involved in motor movement and timing, presenting challenges in comprehensively mapping the anatomy of language.

Over the past few decades, the advances in cognitive neuroscience and psycholinguistics accelerated our understanding of brain networks that support and realize language. Language processing systems in the brain have have been identified using neuroimaging studies, as well through observations of individuals with language deficits. In the following section, we provide a brief overview of what is known about the anatomy of the brain and neural networks supporting the language function.

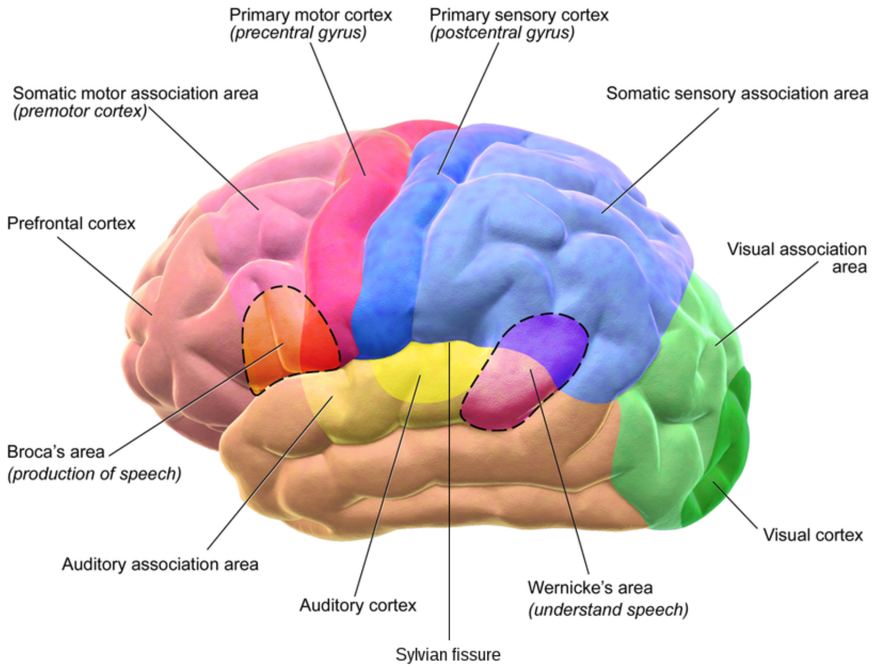


Figure 2.1: An illustration of the lateral view of the left brain hemisphere. Broca's area, Wernicke's area, auditory cortex and auditory association areas are involved in language processing. Image credit.

Neural networks involved in language production and comprehension are primarily located in the left hemishphere of the brain, surrounding the Sylvian fissure. A few such networks and regions, such as the Broca's area, Wernicke's area and surrounding auditory processing areas are highlighted in Figure 2.1. Broca's area is implicated in speech production, grammar and syntax, and patients with lesions in that area have difficulties with spontaneous speech, or with processing of grammatically non-trivial sentences. Such patients still retain some ability to comprehend spoken or written language, in contrast to patients with impaired Wernicke's area, who have difficulties understanding language. Their language might appear fluent compared to patients with damages to Broca's area, but what they say is nonsensical. While the left hemisphere does most of language processing, the right hemisphere is also known to be involved in some aspects, in particular in processing the rhythm of language (Ross and Mesulam 1979), or processing of semantically distant meaning such as metaphors (Schmidt and Seger 2009).

However, this characterization of the neural underpinnings of language in the brain does not answer some critical questions regarding language. For instance, how does the brain represent 50,000 words, which is approximately the size of the vocabulary of a native English speaker?[2] Moreover, how does it organize and derive meaning from a variety of linguistic inputs, such as spoken, written or signed language? To store information about the words and concepts, the brain is thought to use what is called the *mental lexicon*, a store of learned information about words and their meaning (Gazzaniga et al. 2014). The mental lexicon is a part of the semantic memory system, a form of a long-term memory that stores learned objective knowledge such as facts, and the general knowledge about the world (Tulving 1983).

Priming experiments and computational studies indicate that the mental lexicon is organized in a principled way so as to facilitate the access to words during language comprehension and production (Collins and Loftus 1975; Gazzaniga et al. 2014). One such principle is the organization based on the frequency of the words, and it postulates that words that are used more frequently in language are accessed more quickly compared to words with lower usage frequency. Another important organizational principle is semantic (or featural) similarity, stating that words with similar meanings need to be organized together in the brain to facilitate access to related items. Such principles implicate the importance of structured representation of meaning in the brain that are realized with networks of millions of interconnected neurons.

The topic of what computational principles and mechanisms underlie such structured representation still remains intensively studied. It has been proposed that concept organization follows categorical groupings, such as modality specific groupings (Warrington 1975); or semantic groupings, such as concrete versus abstract words (Breedin et al. 1994; Warrington 1981), animate versus inanimate entities (Warrington 1975; Caramazza and Shelton 1998), and so on. Support for such categorical views of memory organization is provided by neurophysiology studies investigating high-level concept organization, such as object recognition, in non-human primates. Groups of neurons in the inferotemporal (IT) cortex, the final area to process visual inputs in the ventral stream, show responses to abstract stimuli such as faces or hands (Quiroga 2012). The IT cortex has numerous projections to the medial temporal lobe (MTL), which contains structures such as the hippocampus, the amygdala and the entorhinal cortex. Those structures have been implicated in the formation and storage of new declarative memories, as shown in the case of the patient H.M. (Scoville and Milner 1957) who had parts of those brain areas surgically removed as epilepsy treatment. Following the treatment, he was unable to form new

[2] See Brysbaert et al. (2016) and Goulden et al. (1990) for a detailed discussion on the variability and challenges involved in estimating the vocabulary size.

semantic knowledge, although some memory functions such as short-term memories and stores for words were not affected. The ability of individual neurons within MTL to selectively respond to high-level stimuli, such as faces, objects or places, supports their purported function in semantic memory (Quiroga 2012).

Recent advances in neuroimaging allow recording a vast amount of brain activity data, and in combination with statistical and machine learning techniques that handle processing of such amounts of data, enable a better understanding of the relationship between the representation of semantic concepts and neural activity. Huth et al. (2016) comprehensively mapped semantic selectivity in the brain by identifying areas that reliably represent various linguistic categories. To record the neural data, they used fMRI imaging while human subjects were listening to hours of narrative stories. Using multivoxel statistical pattern analysis they isolated semantic dimensions and categories that are consistent among neural recordings of individuals. This consistency is discussed as relevant insofar as it might suggest the plausible role of anatomy or cortical cytoarchitecture in constraining of semantic representations. They identified 12 distinct categories, such as tactile, visual, numeric, locational and temporal, to name a few. This is consistent with other studies (see Binder et al. (2016) for an extensive review) confirming that semantic representations are intrinsically related to basic functions such as perception, and action, but also to other types of social and cognitive experiences.

Such studies provide evidence in support of the distributed representation of conceptual knowledge (Rissman and Wagner 2012; Tyler and Moss 2001), where the activity of many neurons jointly contributes to a representation of a concept. Single neurons can still exhibit preference for certain input stimuli, as demonstrated by single-neuron recordings in MTL, where neurons respond selectively to higher-level semantic concepts (Quiroga 2012). However, as stated by Binder et al. (2016), knowing that such cells exist provides ”no account of the means by which external or internal stimuli lead to their activation”, nor does it offer any mechanistic account of how concepts are organized in the brain.

While the existing body of work paves a path to our understanding of *where* in the brain the semantic information is processed, it provides only limited insights as to *how* networks of neurons engage to process such information. Specifically, how are biological properties of neurons leveraged in representing and organizing information to produce different linguistic behaviours remains an open challenge. One effective approach to address such a challenge is with computational models of brain function that are constrained with what is known about biological systems. In particular, more realistic brain models are seen as a good candidate to address the “need to identify the relevant features of brains, and then find a level of modeling which continues to make contact with the anatomy and physiology, but which can also make contact with behaviour” (Elman et al. 1997, Chapter 7, p. 395). It should be noted that while many computational models do not impose biological constraints, they still provide valuable information and hypotheses on how words are represented and manipulated in a distributed fashion. In the next section, we provide an overview of different computational methods concerned with different aspects of language, together with an outline of their significance and contributions to our understanding of the computational mechanisms of language.

## 2.3 AI/Computational Approaches to Language

Russell and Norvig (2002) quote many different, yet related definitions of artificial intelligence (AI), pointing out that many of them are centered around human intelligence and rationality. In general, such definitions are concerned with concepts such as reasoning, thought processes and behaviour in machines and artificial agents. Each of these are examples of complex functions, and according to one specific definition AI, can be seen more generally as the "simulated cognitive processes [...], artificial in the sense that they are artifacts, human creations—and intelligent in that the computers perform complex functions" (Gazzaniga et al. 2014). Thus, in AI, computational methods are used to simulate or reproduce aspects of human behaviour and/or processes underlying such behaviours. This tight bond between human cognition and behaviour was particularly present in the founding days of AI research, as exemplified by the research proposal for the historical conference on artificial intelligence held at Dartmouth College in 1956:

> "An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems reserved for humans, and improve themselves."
>
> McCarthy et al. (1955)

Thus, the understanding of human language for the purpose of simulating it in machines was one of the central tenets of AI from its beginnings. Computational methods to do so are by no means limited to computer science, as this interest was, and continues to be shared among researchers in different fields. Computational models used to understand and simulate these aspects of cognition and behaviour come in a wide variety of flavours, due to the multifaceted nature of natural language as well as different research questions. In this section, we provide an overview of different approaches to computational modelling used to simulate neural, cognitive and behavioural aspects of language processing.

### 2.3.1 Semantic Networks

More than 50 years ago, Quillian (1967) proposed that word networks are of essential importance as models of long-term memory for artificial intelligence, suggesting that "further advances in reproducing human performance with a computer critically depend on giving such programs memories which can effectively provide them with a 'knowledge of the world."' Semantic networks have come a long way since then as powerful, yet simple computational models for the representation of meaning. A semantic network is typically implemented as a graph, with the set of nodes representing concepts, such as words, and a set of edges, representing a semantic relationship between those concepts. Some of the computational appeals of such networks are the relative ease of construction, as well as the variety of graph algorithms that can be used to study network properties and simulate different semantic processes.

Using semantic networks to model the mental lexicon, Collins and Loftus (1975) show that, in combination with a proposed spreading activation algorithm, such models account for a variety of behavioural effects observed in various experiments pertaining to semantics, including sentence verification, categorization, and word production, among others. Such models have also been used to support the argument that conceptual knowledge is structured in a hierarchical fashion, as semantic networks implementing such hierarchy were used to reproduce experimental reaction times in different word priming tasks.

Although semantic networks are one of the earliest computational proposals for the structure of semantic knowledge, they continue to be an important model for the study of human language (Steyvers and Tenenbaum 2005; Miller 1995; Cancho and Solé 2001; Dorogovtsev and Mendes 2001). Semantic networks capturing the scale of a human mental lexicon have been used to analyze the structure and the organization of the human semantic memory (Steyvers and Tenenbaum 2005; Morais et al. 2013; De Deyne et al. 2016). One of the most popular lexical databases used as a semantic network in natural language processing, computational linguistics, and in modelling of semantic knowledge more generally is WordNet (Miller 1995). Words in WordNet are grouped in so called *synsets* that contain synonymous nouns, verbs, adjectives or adverbs. Over 110,000 synsets are connected by conceptual-semantic and lexical relations such as a *is-a* relationship (e.g., *A chair is a piece of furniture*) or *has-a* relationship (e.g., *A chair has four legs.*). WordNet has been used in different natural language processing tasks such as word-sense disambiguation, text classification, text summarization and information retrieval.

### 2.3.2 Probabilistic Language Modelling

Probabilistic approaches based on techniques of Bayesian statistics have been used to model various aspects of cognition and linguistic function that deal with the uncertain, incomplete and inherently noisy nature of sensory information. The focus of such models is on computational principles of inference and learning under probabilistically constrained conditions. Specifically, in the context of language, Bayesian frameworks formalize problems pertaining to language understanding and use as inference from limited amounts of data. Central to that formulation is the Bayes' theorem

$$P(h|d) = \frac{P(d|h)P(h)}{P(d)}, \tag{2.1}$$

where $h$ represents a hypothesis (e.g., such as a set of model parameters) and $d$ represents data (e.g., observations provided to the model). Thus, prior knowledge is modelled as a distribution over hypotheses, with $P(h)$ expressing the degree of belief that a specific hypothesis $h$ is correct. The likelihood term $P(d|h)$ expresses how likely is the data $d$ to be observed, assuming the correctness of $h$. While this formulation is quite simple, it is at the heart of more sophisticated methods such as hierarchical Bayesian modelling and probabilistic graphical models.

Such methods have been effectively used to model different features of linguistic ability, such as word learning (Xu and Tenenbaum 2007; Fazly et al. 2010), valid syntactic expressions and structure acquisition (Chater and Manning 2006; Alishahi and Stevenson 2008), and different semantic and syntactic classes (Griffiths et al. 2005; Barak et al. 2014). Typically, such models do not intend to capture mechanisms or processes underlying some cognitive ability; rather, they focus on computations for explaining the data given a set of assumptions and constraints. As such, assumptions on priors that need to be incorporated constitute an important modelling decision.

### 2.3.3 Vector-based Word Representations

While semantic networks continue to be a popular tool for the modelling and the analysis of semantic processes, they have been limited in their capacity to capture a multitude of semantic relations that exist between words in natural language. If we assume that word meanings are derived from their usage, we can characterize features associated with those words to understand their relationship with other words. For example, the word *cat* might be associated with other words such as *fur*, *tail*, *pet*, *four-legged*, *animal*, to name a few. Therefore, we would expect that *cat* shares many features with some other words such as *lion*, but fewer features with others, such as *ambulance* or *banana*. A simple representation that captures a set of features is a $n$-dimensional vector, where each dimension can either be $0$ or $1$ to denote whether that feature is present or not. Using the features: *fur*, *tail*, *pet*, *four-legged*, *animal*; the word *cat* can be represented with a vector $[1, 1, 1, 1, 1]$, while *lion* would be represented as $[1, 1, 0, 1, 1]$. To conclude that *cat* and *lion* are similar words, we can compute the similarity between them as the dot product between the two vectors, and observe that it is higher than the dot product between either *cat* or *lion* and other words which would have more zeros in their feature vector. While this toy example has been simplified for clarity, it suffices to demonstrate a few challenges with such a binary, vector-based representation. One of the challenges is that of non-binary attributes: while some features can be described with *yes/no* values, others might be better described with continuous values (e.g., a cat breed that doesn't have much fur might be better described with a value such as $0.4$ or $0.1$). Furthermore, what features and how many of them should be used? What continuous feature value best describes a word? Finally, the annotation of such features can be an arduous manual task which can easily render the annotation of large corpora of text infeasible.

Distributional semantic models (DSMs) are a class of semantic models that addresses those concerns. Instead of manually annotating dimensions of vectors, such vectors are learned or computed from large linguistic corpora based on word occurrences in the text. Such vector-based word representations capture statistical regularities observed in natural, written language. At the heart of such methods is the decomposition of a high-dimensional matrix $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ into smaller matrices, that can be used to reconstruct the original matrix with a varying degree of approximation. The matrix $\boldsymbol{A}$ is called co-occurrence matrix, and it is created by counting word occurrences in a set of $m$ documents (also referred to as texts, or contents). Thus, each row in $\boldsymbol{A}$ denotes one of the $n$ words, and each column one of the $m$ documents, with a single $\boldsymbol{A}_{ij}$ entry counting how many times the word in $i$-th row occurs in the $j$-th document. A common matrix factorization method used for dimensionality reduction is the singular value decomposition (SVD). SVD decomposes the matrix $\boldsymbol{A}$ as

$$\boldsymbol{A} = \boldsymbol{U}\, \boldsymbol{\Sigma}\, \boldsymbol{V}^\top, \tag{2.2}$$

where the columns of $\boldsymbol{U} \in \mathbb{R}^{n \times n}$ are orthonormal eigenvectors of $\boldsymbol{A}\boldsymbol{A}^\top$ and thus $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}$, and the columns of $\boldsymbol{V}^\top \in \mathbb{R}^{m \times m}$ are orthonormal eigenvectors of $\boldsymbol{A}^\top \boldsymbol{A}$ with $\boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I}$. $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times m}$ is a diagonal matrix with non-zero square roots of eigenvalues of $\boldsymbol{U}$ or $\boldsymbol{V}$, ordered by the magnitude. In this decomposition, the rows of $\boldsymbol{U}$ represent words, and the columns of $\boldsymbol{U}$ are linearly independent components (vector dimensions or features, as discussed above). In practice, a reduced form of the $\boldsymbol{U}$ matrix is used, where only the first $k$ columns, corresponding to the dimensions with the $k$ highest singular values are kept. The choice of $k$ is equivalent to the number of features discussed in the example above, and it is selected empirically with respect to the performance on the task where such representations are used.[3] The similarity between two words can be computed by taking a dot product or cosine similarity measure between their vector representations in $\tilde{\boldsymbol{U}} \in \mathbb{R}^{n \times k}$.

DSMs using this approach have been known as *count-based* models, with Latent Semantic Analysis (LSA; Deerwester et al. 1990) being among the first ones to use SVD to model word meanings in different natural language tasks. While LSA is widely used in NLP and information retrieval, it was introduced to model aspects of human knowledge induction. DSMs continue to be used to model aspects of semantic memory (Steyvers et al. 2004a), and their relationship to semantic networks has been extensively studied (Steyvers and Tenenbaum 2005; Utsumi 2015).

In NLP, one of the most widely used count-based models is GloVe (Pennington et al. 2014), where vectors are computed by minimizing the loss based on the logarithm of the reconstructed word-word co-occurrence matrix. The loss function is

$$\mathcal{L}(\theta) = \sum_{i=1,j=1}^{V} f(\boldsymbol{X}_{ij})(\boldsymbol{w}_i{}^{\top}\tilde{\boldsymbol{w}}_{\boldsymbol{j}} + \boldsymbol{b}_{\boldsymbol{i}} + \tilde{\boldsymbol{b}}_{\boldsymbol{j}} - \log \boldsymbol{X}_{ij})^2, \tag{2.3}$$

where $X_{ij}$ is the count of how many times the word $i$ appears in the context of the word $j$ in the corpus, $\boldsymbol{w}_{\boldsymbol{i}}$ and $\tilde{\boldsymbol{w}}_{\boldsymbol{j}}$ are learned word vectors, $\boldsymbol{b}_{\boldsymbol{i}}$ and $\tilde{\boldsymbol{b}}_{\boldsymbol{j}}$ are bias parameters and $f$ is a weighting function that weights co-occurrences.

Another popular and related class of semantic models are *predictive* models that use localized word co-occurrences to learn word vectors. Word2vec (Mikolov et al. 2013a; Mikolov et al. 2013b) is a popular predictive model that computes vectors from a large corpus of text by maximizing the probability of the target word, which can either be a single word or a set of context words. It consists of two different neural network models: Continuous Bag-of-Words (CBOW) and Skip-gram (SG), both of which are shown in Figure 2.2. To learn vector representations, the CBOW model learns to maximize the probability of a missing word occurring in a given context. For example, to train CBOW on the sentence *"The cat chases the dog"*, we iterate over each word in the sentence and use that word as the target (output), while the remaining words would be used as the context (input). For this specific sentence, a few such input/output pairs would be "cat

[3] In practice, values between 300 and 1,024 are used.

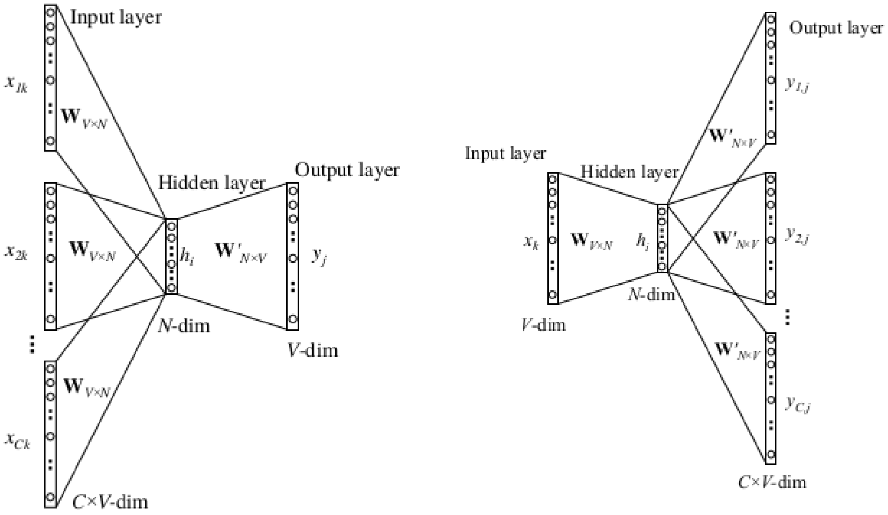


Figure 2.2: Word2vec models for learning of vector representations of words from text (Mikolov et al. 2013b). **Left**: Continuous Bag-of-Words semantic model. **Right**: Skip-gram semantic model. Figures adapted from Rong (2014).

chases the dog“/“The“, “The chases the dog“/“cat“ etc. Each context word, as well as the target, is a one-hot vector encoding. The essence of this algorithm is the weight matrix $\boldsymbol{W} \in \mathbb{R}^{n\times d}$ connecting the input layer to the linear hidden layer with $d$ units. Since each input is one-hot encoded (with a single $1$ and $n-1$ zeros), only those rows of that matrix will be updated during training that correspond to the words in the input. For a single word, there will be one row that is selected with one-hot encoding and those $d$ weights are the elements of the word embedding that are extracted after the training over the whole corpus is completed. In practice, due to the large vocabulary and corpus sizes, different techniques, such as noise-contrastive estimation (Gutmann and Hyvärinen 2010; Mnih and Teh 2012) and negative sampling (Mikolov et al. 2013a), are used to reduce the number of necessary computations.

The SG model is in some sense complementary to CBOW, as the target word is presented as the input and the context is predicted at the output of the network. The loss that is being minimized in the SG model is

$$\mathcal{L}(\theta) = -\log\, \sigma(\tilde{\boldsymbol{w}}_{\boldsymbol{j}}^{\top} \boldsymbol{w_i}) - \sum_{l=1}^{k} \mathbb{E}_{c\sim P(\tilde{w})}\big[\log\, \sigma(-\tilde{\boldsymbol{w}}_c^{\top} \boldsymbol{w_i})\big]. \tag{2.4}$$

The first term ensures that the target vector $\boldsymbol{w_i}$ is similar to the context word $\boldsymbol{\tilde{w}_j}$ by maximizing the dot product between them, while the sum in the second term is negative sampling, a technique that samples $k$ context words from the distribution $P(\tilde{w})$ and minimizes the negative inner product between those context words and the target (i.e., makes their vectors more dissimilar). Due to the way the models are set up (e.g., linear activation function in the hidden layer), linear regularities are preserved in the training and give rise to interesting semantic properties when algebraic operations are applied to such vectors (Mikolov et al. 2013a). For example, to solve for $X$ in the analogy problem *"Paris is to France as Berlin is to X?"* one can add together learned vectors for *Paris*, *France* and subtract the vector for *Berlin*, and the resulting vector will be very similar to the vector for *Germany*. The ability to preserve those semantic relationships extracted from co-occurrence data has been one of the main reasons behind the success of such models in different NLP applications, and language research more generally.

### 2.3.4 Language Modelling Methods in NLP

One issue with embedding models such as GloVe or word2vec is that they derive only one embedding for a single word. As such, they cannot easily account for homonyms,[4] different word senses, or the fact that some words have different syntactic functions. Such nuances, essential to language understanding, have been captured by many different and more complex neural language models developed over the last decade that derive context-dependent embeddings. Such embeddings are typically pre-trained in an unsupervised fashion on the language modelling task (e.g., next word prediction) and then fine tuned in downstream tasks such as named entity recognition, question answering, text summarization, machine translation and others. The models in this category can be divided into two classes depending on their architecture: recurrent neural networks (RNN) and a special class of large feedforward networks known as transformers.

Chronologically, and in the context of language modelling, RNNs first demonstrated impressive performance on sequence to sequence modelling tasks, such as machine translation, where the sentence in a source language is encoded into a single vector representation, and then decoded in the target language (Sutskever et al. 2014; Bahdanau et al. 2014). Some challenges associated with such models such as vanishing or exploding gradients (Bengio et al. 1994) were successfully addressed by different interventions

[4] Homonyms are either homographs, words with the same spelling but different meanings such as the word *row*, or homophones, words with the different spelling and same pronunciation such as the words *here* and *hear*.

during training, such as gradient clipping (Pascanu et al. 2013) and by using of more complex neural network units such as LSTMs (Hochreiter and Schmidhuber 1997) or GRUs (Chung et al. 2014). However, one of the challenges with such models was their inherent sequential nature that made their training difficult to parallelize. Consequently they could not effectively leverage optimized hardware, such as GPUs, that optimize abundant matrix computations in training of neural networks.

By adopting some of the important concepts pioneered by RNNs such as encoder-decoder configuration, attention mechanisms, layered and bi-directional structure, and casting them in large, feedforward networks, transformer-based models addressed the major issue with parallelization. The first transformer-based model was presented in Vaswani et al. (2017), where the authors show that such an architecture outperforms existing models on a machine translation benchmark. Years since then have seen a proliferation of such transformer-based models, including ELMo (Peters et al. 2018), BERT (Devlin et al. 2018), BART (Lewis et al. 2019), T5 (Raffel et al. 2019), GPT-2 (Radford et al. 2019) and GPT-3 (Brown et al. 2020).

At the time of writing, such models continue to set state-of-the-art results on many NLP tasks, and even outperform humans on some of them.[5] This immense success of transformers in NLP has been deemed as the "ImageNet moment of NLP" (Ruder 2018) referring to the success of convolutional neural networks in computer vision (Krizhevsky et al. 2012). However, this success comes at financial and environmental costs due to immense requirements on computational resources. Such models often require large datasets, and, more importantly, hardware powerhouses that for the training of a single transformer model on GPUs emit as much $CO_2$ as 17 American citizens during their entire lifetimes (Strubell et al. 2019). Another issue with this line of research is the lack of accessibility to the required hardware, which is not always available commercially to all researchers, or is prohibitively expensive.

### 2.3.5 Implications for Language Research in NLP and AI

Given the strong performance record of neural language models in NLP, it is perhaps appropriate to wonder about the relationship between such models and human language abilities. After all, language is a distinguishing feature of our species and the fact that these models outperform humans on some tasks is remarkable. While the past few

[5] The leaderboard at `https://rajpurkar.github.io/SQuAD-explorer/` shows ranked performance for over 50 NLP models on the reading comprehension dataset SQuAD (Rajpurkar et al. 2018), where the best performing NLP model as of April 27, 2020 on average outperforms humans by 4 percentage points.

years have seen benchmark tasks grow both in numbers and diversity, demonstrating that these models are capable of sophisticated language processing, they still lack the ability to exhibit a gamut of linguistic skills seen in humans. For example, although such models excel at specific problems, they still cannot engage in spontaneous, meaningful discourse, something that a mature infant can do when interacting with other people and the environment.

Nonetheless, the current state of research in NLP prompts many interesting questions relevant for linguists, cognitive scientists, as well as researchers in natural and artificial intelligence more generally. How do computations carried out by such models compare to word manipulations in natural language? How do those neural networks compare to neural networks in the brain that underlie human capacity for language?

Since NLP models are not intended to comprehensively model language processing in the brain or behaviour, their connection to various aspects of linguistic processing is not immediately apparent. Often, given the size of such models, it can be a major challenge to interpret their workings, understand why they perform as well as they do, and characterize them in the context of the relationship to the structure of human language. Recent efforts in interpretability have examined syntactic abilities of such models by devising novel methods to study the resulting word embeddings, and their relationship to linguistic structure.[6] By designing computational probes to uncover such relationships, they show that contextualized word embeddings are indeed capable of capturing morphological (Belinkov et al. 2017), and syntactic tree structure of English language (Hewitt and Manning 2019). In some way, this is to be expected, knowing that these large models are trained on data that is inherently structured, and that a model that generalizes well should exploit that structure. Nonetheless, these findings are important as they help in settling the debate on whether connectionist models can represent and learn symbolic structure (Fodor and Pylyshyn 1988; Jackendoff 2002) by demonstrating that deep neural models are able to capture the essence of symbolic representations, even if it is not always obvious how they do so.

While uncovering such structure is an important step for understanding the principles captured by these models, the relationship between those principles and the model architecture remains elusive for the majority of models used in NLP. It is only in the post-hoc analysis of trained models and by probing their behaviours, as if those were some naturally occurring phenomena, that we begin to understand what these models learn and why. This approach contrasts with that of neural network modelling for the

[6] See Linzen and Baroni (2020) for an overview of studies that investigate syntactic abilities of deep learning models and the relationship to aspects of linguistic theories.

purpose of explaining neural or cognitive functions, where models are motivated by cognitive or neural theories. Such models are fit, or designed, in such a way so as to approximate much smaller amounts of experimentally controlled human data, while being congruent with theories postulating the processes that generated such data. Often, such models are said to be *cognitively* or *neurally* plausible, as they aim to be consistent with relevant theories while explaining the observed behavioural data. Consistency with cognitive theories, or adhering to biological details of neural networks is rarely a goal for most widely used language models in NLP. However, some NLP models display a greater degree of consistency with cognitive theories than others (Günther et al. 2019), even though this is typically found in the analysis of their performance after they have been built and trained.

Rather than relying on vast amounts of training data and large architectures in order to set state-of-the-art results on a specific set of tasks, biologically or cognitively realistic models are computational theories of how we currently think the brain or cognition works. Such modelling efforts often consist of aggregating human data from different experimental conditions that are relevant for the study of a specific neural or cognitive function, or behaviour. Such data can be performative (e.g., task accuracy), behavioural (e.g., response times) or physiological (e.g., EEG or fMRI signal), or it may involve any other experimentally relevant measure recorded while human subjects were performing the task.

Then, the model (e.g., a neural network) is designed to address the process that generated such data, while elaborating the relationship between the neural network architecture and its outputs. For example, a network might be used to model human performance on a free association task, a task used to study the organization and properties of human semantic networks. In the task, participants are asked to list as many words as possible that are related to a given cue word, under some task constraints. A network performing this task might, for example, contain a component that controls the speed of production of words, and altering parameters of that component would account for individual differences among human subjects, such as those that produce more words and those that produce fewer words within the given time limit.

There are several benefits of such models that are motivated by cognitive or neural theories. First, they can be used to make predictions for which human data may not be available, but the manipulation or the adjustment of the model components can simulate such conditions. Second, such models can be constrained to make them hypotheses of computational mechanisms underlying some function or behaviour. Such constrained models implement hypothesized local neural or cognitive mechanisms in an attempt to explain processes that produce experimentally observable responses from the controlled input stimuli. As well, their predictions can be tested in biological systems and used to guide decision making on potential treatments or interventions in case of neural dysfunction. This is not possible with models commonly used in NLP as they are not designed to be neuroanatomically or neurophysiologically accurate. Model constraints can vary in quality and quantity, and finding the appropriate level, as well as the kind of constraints to impose on the models is an important modelling decision. For example, one type of constraint might refer to the plausibility of the model in terms of how accurately its components map to their neural or cognitive counterparts in the brain or behaviour. Another type of constraint might refer to computational resources, such as the number of neurons used to simulate the function of a certain brain region. The following section introduces such modelling efforts in a greater detail, focusing on methods in AI that are used to simulate aspects of intelligent behaviour, and are used in this research to build neural and behavioural models of language comprehension and use.

### 2.3.6 Neurally Plausible Modelling Methods

Researchers building neural and cognitive models in order to study and characterize aspects of intelligent behaviour typically adopt one of a few different modelling approaches, such as the symbolic, connectionist, probabilistic, or dynamicist approach, or a combination thereof. Some have argued about the difficulty, or even impossibility of combining such different approaches (Fodor and Pylyshyn 1988; Jackendoff 2002). Eliasmith (2013) shows that these are not mutually exclusive. In order to explain a sophisticated ability realized in the brain and manifested in the behaviour, such as language, we consider the possibility that it might be even necessary to combine them.

The symbolic approach is of a particular historical relevance due to its roots in early AI research in 1950s, and as such is often described as “good old fashioned AI”. For proponents of this approach, computation is viewed as a formal manipulation of atomic symbols, thus emphasizing the role of first-order logic as a method for modelling cognition and understanding of human cognition (Newell and Simon 1975). Nowadays, ACT-R (Anderson et al. 1997) is one of the most prominent examples of a cognitive architecture relying on such symbolic and sub-symbolic processing principles, and it is used to model human response patterns in a variety of cognitive and behavioural tasks. Logic formalism is particularly appealing in modelling of language due to the inherent symbolic nature of language, and it lies at the heart of formal languages widely studied in computer science, mathematics and linguistics. It is then perhaps unsurprising that Chomsky, who pioneered numerous contributions in the theory of formal languages (Chomsky 1957) frames natural languages as formal languages through his proposal of Universal Grammar as discussed in Section 2.1. One important question arising from symbolic approaches is that of *symbol grounding*, which asks how symbols become connected to their perceptual experiences and actions. More informally—where do symbols come from? This question remains a topic of ongoing research, and the attempts to address it emphasize the relevance of interaction and the environment in process of grounding symbols to perceptual experiences.

The connectionist approach also emerged in 1950s, although it became widely adopted only a few decades later for reasons outlined below. It emerged as means to deal with the continuity of inputs by learning from statistical regularities in the data. Instead of specifying rules by hand, as is often done with the symbolic systems, such networks are able to learn rules from the data, which led to their description as *pattern recognition* algorithms. The perceptron model was one of the first proposals of such a rudimentary neural network inspired by functional aspects of biological neurons (Rosenblatt 1958). Initially, such neural networks gained only limited attention due to their inability to be scaled up to functional multi-layer architectures. Further extensions and modifications that addressed this issue, such as the multi-layer perceptron (nowadays referred to as feedforward neural networks) and a spectrum of learning algorithms, such as backpropagation (Rumelhart et al. 1986), contributed to the popularity of neural networks in modelling of cognition and human learning. Their computational power as *universal function approximators* made them particularly popular in the early era of language modelling that opposed the view of language as an innate faculty (Elman et al. 1997).

In Section 2.3.4 we have discussed a few different types of neural networks used in natural language processing. Such networks typically rely on neuron units that are not much different from the neural unit of the perceptron. In contrast, biologically realistic neural networks aim to capture a broader spectrum of functional neural behaviours, such as spiking activity resulting from voltage changes in neural membrane potentials. The next section on neurally plausible modelling introduces methods that marry ideas of classic connectionist neural networks with the symbolic approach, while respecting many constraints imposed by biological systems. Those methods are used in this thesis to build biologically realistic models of language processes in the brain.

For the purpose of this thesis, we will distinguish between the *neural level* and the *cognitive level* of a model description. Typically those levels will refer to the same system, yet they allow us to describe the system using a different degree of functional granularity. This is a pragmatic division that helps us focus on different aspects of the system, and is not meant to be conflated with the levels of analysis proposed by Marr (1982). At the neural level, we study neurobiological computations and mechanisms that are constrained by parameters and functions known to be performed by biological neurons. At this level, the modelling efforts are focused on populations of connected neurons, also referred to as *neural ensembles*. For example, an ensemble of neurons might represent a concept such as word that is encoded by the activity of a group of neurons. The constraints imposed on such models, such as neural firing rates or voltage potentials at the membrane, aim to be consistent with the neuroscientific evidence. At the cognitive level, we focus on larger networks of such ensembles where networks perform some cognitively realistic function, such as a task-constrained semantic search in the mental lexicon. The orchestrated activity of such networks is to some extent constrained by processes and mechanisms described in psycholinguistics and cognitive psychology. We denote that our use of the expression *cognitive level* might be rather unconventional, as cognitive models, as used in cognitive sciences, are typically agnostic to the details of a neural implementation. Our focus is on cognitive functions that are realizable in biologically plausible neural networks as we aim to explain aspects of language function across the brain and behaviour, and thus require cognitive models to satisfy constraints imposed by neural computations.

### The Neural Engineering Framework

The Neural Engineering Framework (NEF; Eliasmith and Anderson 2003) is a computational theory and a set of methods for the construction of biologically constrained neural networks. It specifies how a wide variety of computations underlying cognitive processes and behaviour can be realized by simulating the activity of biological neurons. The NEF is used to model a diverse set of neural systems such as those controlling eye position, directed arm movements, and lamprey locomotion (Eliasmith and Anderson 2003), and in conjunction with Semantic Pointer Architecture (SPA; Eliasmith 2013) is used to model aspects of higher cognitive functions. In this section we introduce core ideas behind NEF methods that are important for understanding the models of semantic processes presented in Chapter 4 and Chapter 5.

We first describe how a group of neurons encodes a vector-valued stimulus $\boldsymbol{x}$ in the NEF. While such vectors can represent different input stimuli (or internal mental representations thereof) such as an image or a sound, in the context of most models presented in this work such vectors will be used to represent words. Biological neurons show preference for certain stimuli, that is, they respond more strongly to some inputs than to others. For example, neurons in the striate cortex show selective responses to vertical bars of different orientations (Hubel and Wiesel 1968) and neurons known as place cells in the hippocampus selectively exhibit specific firing patterns when an animal is present in a particular location in an environment (Moser et al. 2008). This stimulus preference can be expressed by assigning a preferred direction vector $\boldsymbol{e}_i$ to each neuron $i$.

The inner product $\boldsymbol{e}_i^\top \boldsymbol{x}$ expresses how strongly a neuron will respond to a given stimulus; it increases as the stimulus vector aligns with the preferred direction. This value can be thought of as being proportional to the amount of current flowing into a neuron, expressed as

$$a_i(t) = a_i\big(\boldsymbol{x}(t)\big) = G_i\Big[\alpha_i \boldsymbol{e}_i^\top \boldsymbol{x}(t) + J_i^{\text{bias}}\Big]\,, \tag{2.5}$$

which gives the neuron activity $a_i(t)$ at time $t$ for a time-dependent stimulus $\boldsymbol{x}(t)$. Here we convert the inner product into an input current to a neuron by means of a gain factor $\alpha_i$ and a bias current $J_i^{\text{bias}}$, used to capture observed neural responses also known as neural tuning curves. The spiking activity $a_i$ of a neuron is given by applying a neuron non-linearity $G_i$ to the input current.

While a wide variety of neuron non-linearities can be used with the NEF, here we use the leaky integrate-and-fire (LIF) neuron model, which captures important properties related to neuronal excitability observed in biological neurons (Koch 2004, Chapter 14). The incoming currents are accumulated as membrane voltage until a firing threshold is reached. At that point, the neuron emits a spike and the membrane voltage is reset to its resting value for a refractory period during which the neuron is unable to produce spikes. Without incoming currents, the membrane voltage will slowly decay to a resting potential due to leak currents. The left panel of Figure 2.3 shows an example of how individual neurons in a set of seven LIF neurons respond to inputs in the range $x \in [0, 1]$. In this specific example, all neurons were assigned preferred directions of $1$, as indicated by the increasing firing rate with increase of $x$. This captures the effect where stronger environmental stimuli (larger values of $x$) elicit stronger neural responses.

Given the firing activity in a group of neurons, how do we reconstruct the represented value $\boldsymbol{x}$? With LIF neurons, $a_i(t)$ is a spike train, i.e., $a_i(t)$ is 0 at all times $t$ that no spike occurred and peaks at the spike times. In biological neurons, each spike causes a post-synaptic current, which in the NEF is typically modeled as an exponential filter[7] of the form

$$h(t) = \frac{1}{\tau}\exp(-t/\tau). \tag{2.6}$$

This function can be combined with a linear decoding to provide a weighted linear filter that estimates the original vector $\boldsymbol{x}$. That is:

$$\hat{\boldsymbol{x}}(t) = \sum_i a_i(t) * \big[\boldsymbol{d}_i h(t)\big]\,, \tag{2.7}$$

[7] Other types of linear filters are also possible, see Voelker and Eliasmith (2018) for more details.

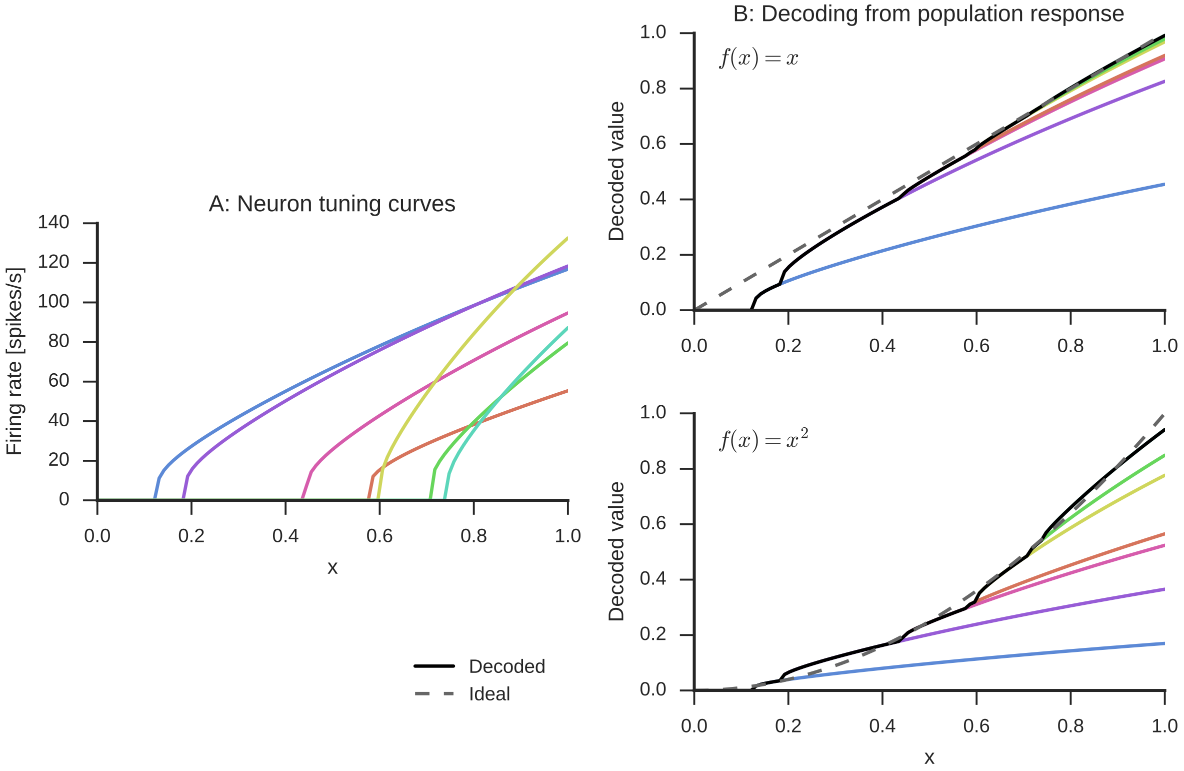


Figure 2.3: **Left:** Randomly generated tuning curves for seven neurons. **Right:** Linear combination of these to decode the represented value $x$ (top right) or decode a function, here $x^2$ (bottom right). The dashed gray line is the ideal output and the black solid line the decoded value from all seven neurons. Reproduced from Kajić et al. (2017b, Figure 1).

where $*$ indicates convolution and the weights $\boldsymbol{d}_i$ are obtained by a global least-squares optimization of the error

$$E = \sum_k \|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k\|^2 \tag{2.8}$$

of the reconstructed stimulus across $k$ sample points and all neurons in the group. The decoding process for the simple identity function $f(x) = x$ is shown in the top right panel of Figure 2.3. The decoding weights scale the tuning curves (left panel) and the represented value is estimated with a sum over the scaled tuning curves.

However, in most cases, we want to decode transformations of the input $\boldsymbol{x}$, rather than the input itself, and transmit this information between groups of neurons. For example, in the bottom right panel of Figure 2.3 seven tuning curves are shown that are weighted by decoders found for the function $f(x) = x^2, x > 0$. In more general case, decoders are found for any function $f(x)$ by minimizing the error function using least-squares optimization

$$E^f = \sum_k \left\| f_k(\boldsymbol{x}) - \hat{f}_k(\boldsymbol{x}) \right\|^2 . \tag{2.9}$$

The synaptic connection weights that perform this transformation $f(x)$ can be computed from the decoding weights $\boldsymbol{d}_i$ of the pre-synaptic neurons that reconstruct an estimate of the represented value $\hat{f}_k(\boldsymbol{x})$. In addition, the input current to a post-synaptic neuron $j$ depends on its preferred direction $\boldsymbol{e}_j$ and gain $\alpha_j$. Because the quantities $\boldsymbol{d}_i$, $\boldsymbol{e}_j$, and $\alpha_j$ do not change over time[8], they can be multiplied together to provide standard neural network connection weights as follows

$$W_{ji} = \alpha_j \boldsymbol{e}_j^\top \boldsymbol{d}_i \tag{2.10}$$

where $W_{ji}$ comprise the synaptic weight matrix $\boldsymbol{W}$. This is the optimal synaptic connection weight matrix for transmitting information from one neural group to another (Eliasmith and Anderson 2003).

To summarize, the NEF allows us to state how a time-varying, vector-valued stimulus is encoded in spiking neural populations, how the value represented in a neural population can be decoded, and how to connect neural populations to compute functions using those represented values. All connection weights are determined in an offline optimization without the need for an online learning process, although online algorithms can also

[8] This assumes no synaptic weight changes, e.g. through learning, are happening. These could also be handled in a biologically realistic manner by the NEF (Bekolay et al. 2013), but are out of the scope of this work.

be used. The tendency to rely on offline optimization is one feature that distinguishes NEF from online optimization methods commonly used in machine learning.

### The Semantic Pointer Architecture

The Semantic Pointer Architecture (SPA; Eliasmith 2013) builds on top of the NEF to provide methods for realizing higher-level cognitive functions, such as memory, semantics, syntax, decision making and learning. While the NEF provides computational building blocks that are combined together to achieve desired computations in biological neurons, the SPA combines such blocks into larger networks, or modules, that perform more complex functions. The function, organization and structure of such SPA modules is to a large extent motivated by evidence from neuroscience, psychology, or both, as means to preserve neural and psychological plausibility.

The SPA is used to model aspects of higher cognition such as serial working memory (Choo 2010), action selection (Stewart et al. 2012), lexical knowledge (Crawford et al. 2015), affective processing (Kajić et al. 2019), spatial representation (Komer et al. 2019; Dumont and Eliasmith 2020) and is the basis for the construction of Spaun, the first detailed brain model capable of human-like performance on a variety of cognitive and sensorimotor tasks (Eliasmith et al. 2012).

At the heart of SPA methods is the manipulation of vectors in high-dimensional spaces, which, in the context of the SPA are referred to as *semantic pointers*. Vectors used as semantic pointers are, on some occasions, Holographic Reduced Representations (HRRs; Plate 1995), meaning that they satisfy certain mathematical constraints. For example, one of the constraints is that the elements of such vectors are drawn from a normal distribution with a zero mean and variance $1/n$, where $n$ is the dimensionality of the vector. Such constraints, together with corresponding mathematical operations such as the circular convolution, its inverse, and the addition operation, allow aggregating multiple semantic pointers into a new semantic pointer with the same dimensionality as its constituents. Circular convolution is used as a *binding* method operating on two vectors of the same dimensionality to generate a third vector.[9] It is commonly used to bind *roles* to *fillers* in order to create structured representations. For example, in order to represent the sentence "We love oranges" with semantic pointers, we can define three role vectors named: *subject, verb, object*, as well as three filler vectors *we, love, oranges*. The sentence is then represented as a single semantic pointer by combining the vectors in

[9] Other semantic pointer binding methods also exist, for example Vector-Derived Transformation Binding as proposed in Gosmann and Eliasmith (2019).

the following way: $subject \circledast we + verb \circledast love + object \circledast oranges$, where $\circledast$ denotes the circular convolution. By means of operations defined on HRRs it is possible to retrieve individual vectors from the sentence represented in this way, akin to answering the question “Who loves oranges?” or “What do we love?”.

Semantic pointers are therefore compressed representations, achieved by combining initial vectors in a structured way depending on the task. By combining vectors using convolutions and additions to create new vectors, the resulting vectors will be more similar to each other if they share similar constituents, which can be also seen as features. Thus, the meaning of the resulting vector can be interpreted as being derived from the meaning of its parts (hence the adjective *semantic*). Since the resulting vector can be used to uncover its constituents, it is analogous to the concept of a “pointer” in computer science, a memory address that refers to some (typically much larger) content, instead of being the content itself. In contrast to HRRs, all relevant operations are implemented in neurons using the NEF methods.

As mentioned previously, the SPA has been used to model various aspects of biological cognition, and for the models in this thesis, two SPA modules are of particular relevance: associative memory and the basal ganglia. Associative memory is a long-term, declarative memory system supporting associative learning, or learning of relationships between unrelated items or events. While it is not localized to a specific structure in the brain, the MTL area discussed in Section 2.2.2 is implicated in associative learning. More details on different associative learning mechanisms are provided in the next section on reinforcement learning, while here we briefly introduce the associative memory module in the SPA. Associative memory in the SPA is used to encode and recall semantic pointers. It can be used in different ways, for example to recall the “clean” version of a noisy semantic pointer (Stewart et al. 2009). But, it can also be used to recall a different semantic pointer that is associated with the input semantic pointer. Such input-output mappings can be learned in a biologically plausible manner (Voelker et al. 2014), or can be directly computed from the data by estimating encoders if all of the inputs and outputs are known, as discussed in the NEF section. In the SPA, the associative memory module can be used in conjunction with a winner-take-all (WTA) mechanism, so that if there are multiple vectors associated with the input, only one with the strongest association will appear at the output. Associative memories are one of the major building blocks in different models of cognitive functions in the SPA since they provide a reconstructive mechanism to deal with the noisy representation resulting from spiking neurons and the lossy circular convolution operator.

The other SPA module relevant for this work is the module simulating aspects of the function of the basal ganglia. The basal ganglia are subcortical structures in the brain with a function implicated in different cognitive processes and behaviours, such as the action selection process, decision making, goal-oriented behaviour, motor learning, and speech production, to name a few. The basal ganglia take inputs from a variety of brain regions, most notably from the cortex that represents different brain states. Such representations are then used to direct the behaviour via other cortical and subcortical structures. The SPA basal ganglia module (Stewart et al. 2012) implements computational mechanisms proposed by Gurney et al. (2001b) and Gurney et al. (2001a) in a spiking dynamical system. As such, it respects the anatomical and functional division of different nuclei in the basal ganglia, while operating on semantic pointers as time-varying state representations. In contrast to the associative memory, it provides a more powerful and stable WTA mechanism used in action selection.

### 2.3.7 Reinforcement Learning

Reinforcement learning (RL) is a framework for behavioural learning within AI in which artificial agents learn to achieve goals by interacting with the environment. Typically, such agents learn various aspects of human or animal behaviours, such as game playing (Silver et al. 2017; Mnih et al. 2015) or motor control (Kohl and Stone 2004; Levine et al. 2016; Lillicrap et al. 2015). Remarkable methodological advances in the field over the past decade demonstrated that agents trained in a such way can outperform humans in highly complex games, as well as develop tactical playing by "inventing" novel moves and strategies (Silver et al. 2017).

While such agents are typically not endowed with language capabilities, primarily due to the nature of commonly used non-linguistic tasks, the past few years have seen a greater integration of language in RL frameworks. Such integration is particularly prominent in multi-agent setups where agent cooperation is required to achieve a mutual goal (Mordatch and Abbeel 2018; Lazaridou et al. 2018; Kottur et al. 2017). There are a few different directions of interest for studying language in RL. For example, RL can be used to study parsing of linguistic information, which can be essential to the task performance such as in instruction following (Hermann et al. 2017; Bahdanau et al. 2018), or RL can be used to introduce structure in agent behaviours by means of task descriptions (Andreas et al. 2017; Narasimhan et al. 2018). The domain of *emergent communication* in RL investigates the emergence of artificial languages in RL settings, where agents need to communicate in order to achieve a task. In such setups, agents are often segregated into listeners and speakers and need to coordinate their linguistic and non-linguistic actions in an environment for the purpose of achieving a common goal (Lazaridou et al. 2018; Li and Bowling 2019). Such setups offer an experimental testbed for realizing various *language games* akin to those discussed by Wittgenstein (1953) in order to study characteristics of emergent languages, their relationship to natural language, and the influence of communication on agent behaviours.

Such emergent views of language are a form of *active* language learning, since agents interact with the environment and influence the learning process by controlling behaviours that are beneficial to learning. This grounded interaction stands in contrast to static, text-based methods presented in Section 2.3.3, where language representations are learned by processing written language from large text corpora. In Chapter 6 we show that a simple grounded interaction scenario can be used to learn communicative signals that exhibit important linguistic features, some of which are reminiscent of some features of vector-based representations. We also argue that this approach to language learning is more psychologically plausible compared to learning without any interaction, and discuss our experiments in the context of natural language characteristics. In the remainder of this section we introduce some fundamental concepts and methods in RL relevant for that research.

### Reinforcement Learning Fundamentals

Informally, RL is concerned with methods for finding optimal behaviours in a setup consisting of an agent and the environment the agent is situated in. An agent is a decision-making algorithm that bases its choices or *actions* on the *state* of the environment, with the goal of finding actions that lead to maximizing rewards in a given task. The environment consists of a set of states and it specifies transitions between the states. After an agent performs an action in the environment, it enters a new state and observes a *reward*, usually represented with a scalar value. Each action-selection event is referred to as a *step* and a sequence of steps is known as an *episode* or a *trial*. An episode can terminate after a fixed number of steps, or it can be controlled in some other way by the environment.

An agent's decision-making is guided by a *policy* determining which actions to take in any given state. A policy often incorporates some form of trial-and-error behaviour, balancing the decision making between *exploitation*, where actions are taken with respect to their known utility, and *exploration*, where occasionally a non-optimal action is selected. In contrast to some other learning methods commonly used in machine learning, such as supervised learning, in a RL framework the environment does not tell the learner what to do. In other words, the environment does not provide the agent with the "target", instead, it is the task of a learner to discover what actions maximize the cumulative reward, referred to as the *return*.

Formally, much of RL is concerned with methods for solving problems expressed as a finite Markov decision processes (MDP) by finding the optimal policy $\pi$. An MDP is defined as 4-tuple $(S, A, P, R)$, where $S$ is the set of states in the environment, $A$ is the set of actions, $P$ is the probability transition function $P\colon S \times A \times S \to [0,1]$ defining the environment dynamics, and $R$ is the reward function $R\colon S \times A \times S \to \mathbb{R}$, mapping choices on to rewards. A policy $\pi$ is then a function $\pi : S \times A \to [0,1]$ that specifies the probability of selecting an action in a state.[10] An action selected at time step $t$ is denoted as $a_t$, while the state an agent is in at time $t$ is $s_t$, $\forall t \in \{1, ..., T\}$ where $T$ is the length of an episode.

A *value function* $V(s)$ or an *action-value function* $Q(s,a)$, $s \in S, a \in A$ is often used to determine an optimal policy. Such functions express the utility of each state or state-action pair, which is then used when estimating how useful an action is for achieving the goal. The value function for a state $s$ when following the policy $\pi$ for deciding on subsequent actions is defined as

$$V_\pi(s) \coloneqq \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)\Big[r + \gamma V_\pi(s')\Big]. \tag{2.11}$$

In Equation 2.11, $\pi(a|s)$ is the probability of selecting action $a$ in a state $s$, and the probability of each such action is multiplied with the probability of ending in the next state $s'$ and receiving the reward $r$, weighted by the sum of the reward $r$ and expected future reward $\gamma\, V_\pi(s')$. This equation is known as the Bellman equation in dynamic programming, that expresses value of the current state in terms of values of possible future states. Here, it averages over probabilities of all possible actions $a \in A$, state transitions $s' \in S$ and rewards $r \in R$. The discount factor $\gamma$ is a parameter in the range $[0,1]$ used to model the relevance of rewards at different points in time; values closer to $0$ bias the agent to collect more immediate rewards, while those closer to $1$ consider more distant rewards.

An optimal policy $\pi^*$ is better than or equal to all other policies $\pi$, if and only if $V_{\pi^*}(s) \geq V_\pi(s), \forall s \in S$ (Sutton and Barto 2018). An optimal policy $\pi^*$ will always select an action that maximizes the expected return for that state, and such a policy will assign a non-zero probability to all such actions. Value functions computed with an optimal policy are denoted as $V^*(s_t)$ and they can be found by solving the recursive formulation of the set of equations, one for each state. Similarly, the action-value function,

$$Q(s,a) \coloneqq \sum_{s',r} p(s', r|s, a)\Big[r' + \gamma \sum_{a'} \pi(a'|s')Q(s', a')\Big], \tag{2.12}$$

[10] A policy can also be deterministic, in which case it maps to specific actions instead of probabilities.

can also be used to find an optimal policy by finding actions that maximize action-values.

An approach to RL where the agent directly learns the model of the environment defined by the probability transition function $P$ and the reward function $R$ is known as *model-based* RL. In many cases, such knowledge might not be available, or it might be prohibitively expensive to compute. Another cost arises from the need to sweep through the set of states at each time point to update $V(s)$ or $Q(s, a)$, which means that finding the optimal policy is polynomial in the number of states and actions (Sutton and Barto 2018). For these reasons, many problems are more easily solved with *model-free* methods, such as the TD learning discussed later.

### Psychological and Neural Basis of Reinforcement Learning

Selective reinforcement of the relationship between an action and the response elicited by such an action as a method of behavioural learning has its roots in psychological studies investigating animal learning, where it is known as *classical* or *operant* conditioning. Research in classical conditioning has been pioneered by Ivan Pavlov, who studied behavioural conditioning patterns in dogs (Pavlov 1927).

While dogs spontaneously produce saliva when presented with food, Pavlov's experiments have shown that they can learn to salivate in response to other stimuli, which, under normal conditions would not influence saliva production (e.g., the sound of a metronome). Food is regarded as the unconditioned stimulus (US), as the animal will reflexively respond to such stimulus. After multiple trials of co-presenting the US with the sound, regarded as the conditioned stimulus (CS), an animal starts to exhibit the response when presented with the CS only, in the absence of the US. Thus, in classical conditioning, innate reflexes are associated with novel stimuli to elicit a response.

One of the most prominent models in this domain is the Rescorla-Wagner rule (Rescorla and Wagner 1972), formalizing the associative relationship between different components of a compound CS with the response. For example, a compound CS might consist of two stimuli, such as a sound and a light. For a single stimulus $x \in X$, where $X$ is a set of stimuli forming the compound CS, the association strength of that stimulus in trial $t$ is defined as $p_x^t$. For simplicity, and without loss of generality, we can treat the association strength as a probability of conditioned response given the conditioned stimulus $x$. According to the Rescorla-Wagner rule, the associative strength in the next trial, $t+1$, is

$$p_x^{t+1} = p_x^t + \Delta p_x^t \tag{2.13}$$

$$\Delta p_x^t = \alpha_x \beta(\lambda - p_T^t), \tag{2.14}$$

where $p_T^t$ is the total associative strength defined as $p_T^t = \sum_{x \in X} p_x^t$, $\lambda$ is the maximum association strength, $\alpha_x$ and $\beta$ are parameters. In Chapter 14 of their book, Sutton and Barto (2018) cast the difference $\lambda - p_x^T$ as the prediction error, after considering $\lambda$ as the magnitude of a reward signal. This prediction error, seen as difference between expected and the experienced reward, is the main driver of behavioural learning and an important concept in the study of learning both in natural and artificial systems.

The significance of the Rescorla-Wagner rule lies in its ability to account for various aspects of experimentally observed behavioural learning patterns, such as blocking and extinction. Blocking is observed when a new, additional stimulus to a CS is added, and yet the association strength of that stimulus remains low due to other stimuli reaching the maximum $\lambda$. Thus the new stimulus is being blocked by the existing CS due its inability to evoke a response. The phenomenon of extinction occurs when the associative strength of a CS is reduced sufficiently as to not prompt the response anymore, as a result of removing the US during training.

In *operant* or *instrumental* conditioning, animal behaviour affects the learning process as it is able perform different actions in response to stimuli. Such choice might be a simple action such as pressing a lever, a key, or navigating to a specific location in an environment. Each such action can be associated with a positive, negative or neutral reinforcement signal, and the existence of negative reinforcement incentivizes the animal to avoid performing actions that lead to it. In contrast, an animal might learn to perform an action in order to receive positive reinforcement for a given state of the environment. Operant conditioning in animals has been first characterized by Thorndike (1898), and made well known by Skinner (1938), who studied behaviours of rats and pigeons, with positive reinforcement signals being food pellets or droplets of liquid, and negative signals being electric shocks or other forms of interventions inducing physical discomfort. In RL, operant conditioning is modelled with *multi-armed bandit* models, where an agent has to learn a policy in an environment that consists of a single state and several possible actions in that state.

Despite their relative computational simplicity, models based on Pavlovian conditioning and the Rescorla-Wagner rule play an important role in psychology, as well as in neuroscience and artificial intelligence, among other fields.[11] For example, midbrain dopamine neurons have been shown to signal changes or errors in the predictions of future rewarding events (Schultz et al. 1997), implicating an important role of such signals to higher order cognitive functions (Hazy et al. 2010). In computer science, the prediction

[11] See Neftci and Averbeck (2019) for an extensive review of the contributions of RL methods in AI to neuroscience and vice versa.

error is a basis for a variety of error-driven learning algorithms including the delta rule, omnipresent in machine learning, and temporal difference (TD) learning (Sutton and Barto 1987), one of the most fundamental methods in reinforcement learning.

### Temporal-Difference (TD) Learning

Temporal-Difference (TD) learning (Sutton and Barto 1987) is a bootstrapping technique for online learning in reinforcement learning. While different TD learning algorithms exist, all of them have two main things in common. First, and in contrast to Monte Carlo methods, TD algorithms are applied at every time step to update estimated values in value functions (or action-value functions), without having to wait for an episode to terminate. Second, in order to update the value, the immediate reward is used as well as the estimate of the future rewards. As noted before, the gist of the idea of TD learning is in using the prediction error to improve the estimates of value or action-value functions.

In its most simple form, the difference used to update the value function with TD learning is

$$\Delta V(s_t) = \eta\big[\overbrace{\underbrace{r_t + \gamma\, V(s_{t+1})}_{TD\, target} - V(s_t)}^{TD\, error}\big]. \tag{2.15}$$

The expression in brackets, $r_t + \gamma\, V(s_{t+1}) - V(s_t)$, is known as the TD error, consisting of the TD target $r_t + \gamma\, V(s_{t+1})$ and the estimate of the value function for the current state $V(s_t)$. Informally, with the insights obtained by previously experiencing the environment, the agent has a better estimate of how valuable a state is for achieving the goal. This is because the current estimate is adjusted by the difference between "How much did I think I would get?" (corresponding to $V(s_t)$) and "How much did I really get?" (corresponding to the TD target).

An important model-free, off-policy TD learning method is the Q-learning algorithm (Watkins and Dayan 1992). It estimates action-value functions, using a formalism similar to Equation 2.15:

$$\Delta Q(s_t, a_t) = \eta\big[r_t + \gamma\, \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\big] \tag{2.16}$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \Delta Q(s_t, a_t). \tag{2.17}$$

Q-learning is a model-free method as it does not to attempt to estimate state-transition function, and it is off-policy as the choice of next action when estimating the Q-value is greedy, instead of the current behavioural policy the agent is using. One of the appeals of the Q-learning algorithm is that irrespective of the policy, it provides optimal policy convergence guarantees with minimal requirements (Sutton and Barto 2018). For an agent, such Q-values can be stored and update using a tabular representation. However, this becomes computationally infeasible with large number of states and actions, in which case it is implemented with function approximation, often via a neural network.

# 3 Semantic Networks

Choosing how to represent semantic knowledge poses an important decision in many models of cognitive phenomena, as well as in various natural language applications. In cognitive modelling, such representations are used to model and study conceptual knowledge such as semantic memory and accompanying cognitive processes. In machine learning, such models are often used to convert one-hot representation of words into dense vectors that can be conveniently represented with different neural network architectures and are used in various applications such as machine translation, machine comprehension and question answering, among others.

However, not all such word representations reflect properties observed in human inter-concept semantic relations. One of the most popular datasets used to model human semantic memory and related processes is the University of South Florida Free Association Norms (Nelson et al. 2004, USF Norms). Because it is a psychologically plausible representation of human semantic relations, the USF Norms have been successfully used to reproduce human-level performance on tasks such as verbal semantic search (Abbott et al. 2015; Kajić et al. 2017a), recognition memory and recall (Steyvers et al. 2004a) and many others.

A semantic network is implicitly captured by datasets like the USF Norms. Such a network can be explicitly represented by allowing each node to be a word, with weighted links representing the strength of association between words. A few examples of words and their associations from the USF Norms are shown in Table 3.1. These associations are gathered from human free recall experiments. The properties of semantic networks built from the USF Norms have been described and compared to semantic networks built from various distributional semantic models (Steyvers and Tenenbaum 2005). It has been shown that semantic networks built from the USF Norms exhibit *small-world*

| Word | Degree | Associations (strength) |
|---|---|---|
| food | 340 | eat (0.41), drink (0.05), hunger (0.04) |
| money | 320 | cash (0.21), spend (0.10), green (0.06) |
| water | 294 | drink (0.17), cool (0.08), wet (0.07) |
| good | 281 | bad (0.76), great (0.04), evil (0.02) |
| school | 263 | work (0.13), college (0.11), bus (0.07) |
| house | 238 | home (0.58), apartment (0.06), family (0.03) |
| bad | 215 | good (0.75), boy (0.05), dog (0.02) |
| work | 203 | hard (0.14), play (0.12), job (0.10) |
| love | 199 | hate (0.46), kiss (0.06), like (0.04) |
| car | 198 | auto (0.13), drive (0.12), truck (0.11) |

Table 3.1: Top ten words with the highest number of word associations in the USF Norms database.

characteristics. That means, although a single node in such a network connects to only a few other nodes, the path between any two nodes in the networks is rather short, as quantified by the number of links between the two nodes. Aggregated, as well as individual USF networks have been shown to have degree distributions that are well approximated with a power-law (Steyvers and Tenenbaum 2005), or truncated power-law distributions (Morais et al. 2013). In practical terms, this means that there are just a few densely connected nodes that have links to many other words in the semantic network, while the majority of nodes is sparsely connected.

In this chapter, we investigate the presence of such network characteristics in semantic networks constructed from two widely used semantic models, word2vec and GloVe introduced in Section 2.3.3. Here, we evaluate the plausibility of those models by comparing them to the networks created from the human association norms.

## 3.1 Word Association Norms

The University of South Florida Free Association Norms (Nelson et al. 2004) were collected in a series of free association experiments conducted with more than 6,000 participants over the course of a few decades. In the experiments, participants were presented with a cue word and asked to write down the first word they thought of as a response to the cue. In this way, a distribution of responses for each cue word was created by norming the frequency of a response with the number of participants performing the task. Each such cue-target pair is referred to as a word association pair. There are 5,018 words in the database, amounting to 63,619 non-zero weighted word association pairs. Nouns account for 76% of entries in the database, adjectives for 13%, verbs for 7% and the remaining entries are other parts of speech.

The top 10 words with most associations, as well as corresponding three strongest word associates are listed in Table 3.1. The degree of a node is the number of outgoing and ingoing links, that is, it counts both the number of words that were given in response to that word as a cue, as well as the number of words where that word was given as a response to some other cue. Looking at the list of such words, it is not surprising that words with the most connections are those that are related to many daily activities or behaviours. Arguably, some of these words and their associations reflect the experiences of the experiment demographics—undergraduate students—which might explain why the word *school* ranks highly in fifth place. While some words have dominant association strengths, such as that between *good* and *bad*, the association hierarchy is more flat for others (e.g., *car* and the associates *auto, drive, truck*). The histogram showing the distribution of degrees for all words in the database is shown in the left panel of Figure 3.1, where we can see see that only a few words have more than 200 different associations, while on average, each word has 25 associates.

Even though the USF Norms are relatively small compared to the size of the vocabulary of an average English speaker, and are certainly not representative of the human population at large, they continue to be a valuable resource in the study of the organization of human semantic memory, as well as in modelling various aspects of linguistic behaviours. In particular, they are considered to capture aspects of word meanings acquired through perception and interaction with the environment, and as such reflect relationships between words as used in ordinary language. This is in contrast to relationships occurring solely in written text, where some associations might be implicit in their usage. For example, the second and third strongest associate of the word *banana* in the norms are *apple* and *yellow*. In word2vec, the vector for word *apple* is only the 23rd most similar word to *banana*, as measured by the cosine similarity, while *yellow* does not even appear in the list of top 100 words most similar to *banana*. Thus, the sort of relationships that are “common knowledge”, such as that bananas are typically yellow might be often omitted from typical discourse used to train vector-based models studied here. Instead, additional processing is needed to infer such relationships from those models.

This type of associative asymmetry, together with other geometrical constraints imposed by vector-space models, are argued to underlie the difficulty DSMs have in capturing human association data (Hayashi 2016; Nematzadeh et al. 2017). The norms display a

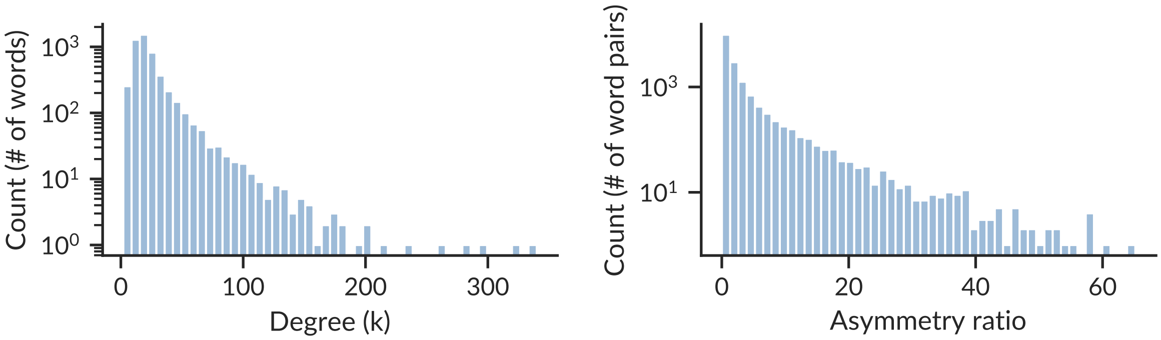


Figure 3.1: The distribution of degrees and asymmetry ratios in the USF Norms.

high degree of asymmetry: 73.64% of word association pairs are *not* reciprocal, meaning that if a word pair $(w_1, w_2)$ has a non-zero weight, the word pair $(w_2, w_1)$ has a zero association strength (i.e., is not present in the database). Some examples of such non-reciprocal word pairs are *(check, bounce)*, *(ivy, school)* and *(wine, glass)*. Second, the asymmetry is also reflected in the magnitude of association strengths between the word pairs. Most people think of the word *stop* when prompted with the word *halt*, yet, the reverse is not true. As per Nematzadeh et al. (2017), it is possible to quantify the extent of asymmetry between two words by taking the ratio

$$asym(w_1, w_2) = \frac{p(w_1, w_2)}{p(w_2, w_1)}, \tag{3.1}$$

where $p(w_1, w_2)$ stands for the normed association strength between the two words. The right panel in Figure 3.1 shows the distribution of asymmetry ratios on a log-linear scale for all reciprocal word association pairs (26.36% of the total number of word association pairs). While the majority of reciprocal word pairs have a symmetric relationship ($median = 1$, $average = 2.57$), about 30% of them have the asymmetric relationship where the strength of association in one direction is at least twice as large in the other direction.

## 3.2 Cognitive Plausibility of DSMs

Alongside human word association norms, distributional semantic models (DSMs) such as word2vec (Mikolov et al. 2013a; Mikolov et al. 2013b) and GloVe (Pennington et al. 2014) have been widely used as models of human semantic memory, even though they were originally developed in the context of natural language processing applications. For example, they achieve strong performance on a variety of tasks such as word similarity judgements (Baroni et al. 2014; De Deyne et al. 2016), synonym detection (Bullinaria and Levy 2012), and semantic relatedness (Mandera et al. 2017). Different arguments have been put forward to argue for *plausibility* of such models as models of human semantic memory. Plausibility refers to *cognitive plausibility* or *psychological plausibility*,[1] although this definition is still context dependent and might be used differently among different researchers. Generally, it refers to the consistency of such models with theories of semantic cognitive processes related to language learning, understanding and use. In the following, we discuss a few common characterizations of cognitive plausibility of such models, and propose an additional characterization of plausibility based on the comparison of DSMs to word association norms.

Mandera et al. (2017) argue for psychological plausibility of word2vec and other predictive models by stating that "the models are trained using a similar technique as the Rescorla-Wagner learning rule", where the "similar technique" referred to are the backpropagation and stochastic gradient descent (SGD) algorithms. The Rescorla-Wagner rule was discussed in Section 2.3.7 on reinforcement learning, and it explains important aspects of learning behaviours in humans and animals. However, calling predictive DSMs psychologically plausible because the Rescorla-Wagner rule is mathematically equivalent to the delta rule at the heart of backpropagation and SGD is not a strong argument, when considering other models. Any supervised machine learning problem where the goal is to find the target or learn a label relies on an error-driven learning rule. As such, the delta rule and its generalization, backpropagation, are one of the most widely used learning rules in all of machine learning, and, according to this criterion of plausibility, almost any contemporary NLP algorithm would be deemed psychologically plausible.

Another important condition underlying the success of DSMs is the vast amount of data required for learning of such high quality vectors. Word2vec vectors are typically trained on billions of words (Mikolov et al. 2013b), often in a few passes. In contrast, language acquisition in children is characterized by comparably smaller vocabularies. Another important difference is the nature of learning, where instead of providing large amounts of labelled data, children learn by interacting with the environment and relying on feedback signals, rather than always being provided with the "correct" answer. Thus, in this respect, such models are not representative of human word learning. A review of plausibility of DSMs as cognitive theories of semantic memory can be found in Günther et al. (2019), where the authors argue that such models capture interesting features

[1] Although cognitive plausibility can be seen as a more specific aspect of psychological plausibility, for the present purpose we will use those two terms interchangeably.

beyond pure co-occurrence data, such as polysemy, compositionality, and some aspects of grounded or embodied cognition theories.

Here, we propose an approach that can be used to characterize plausibility of vector-based models. In particular, we focus on the ability of such models to express the multitude of relationships found among words in the associations norms. Similarly to DSMs, word norms have also been used in a wide variety of linguistic tasks such as semantic search (Abbott et al. 2015; Kajić et al. 2017a), recognition memory and recall (Steyvers et al. 2004a), or predicting semantic similarity (De Deyne et al. 2016). As such, even though they are an order of magnitude smaller in terms of the vocabulary size, they condense semantic relationships important for various linguistic tasks, sometimes even performing better than DSMs.

Instead of using DSMs to directly model distributions of word associations for each word, we adopt a network analysis approach based on comparisons of semantic network properties. The structure of semantic networks constructed from word associations has been extensively characterized, and such structure has been shown to support efficient search in large-scale networks (Steyvers and Tenenbaum 2005; Utsumi 2015). Here, we argue that another useful view on psychological plausibility can be provided by studying such large-scale networks. To this extent, we assess the plausibility of word2vec and GloVe semantic models. We examine the structure of semantic networks constructed from the models, and compare them to that of the human word association norms (Nelson et al. 2004). This study examines which properties of semantic networks are comparable to those of human association networks, and points out differences where such models might be an inadequate representation of human semantic knowledge.

## 3.3 Constructing Semantic Networks

To construct the semantic networks, we use pre-trained GloVe and word2vec vectors that are made available online for public use by their authors. GloVe vectors were trained using the Common Crawl dataset containing approximately 840 billion word tokens. Word2vec vectors were trained using the Skip-gram model on the Google News dataset containing about 100 billion words. All vectors used here contain 300 dimensions. The USF Norms (Nelson et al. 2004) are used to model the human association network. Here, the size of vocabularies of the word2vec and GloVe datasets is reduced to that of the USF Norms to enable comparable analyses.

We generate undirected and directed semantic networks from word2vec and GloVe semantic models, and use a series of graph-theoretic analyses to compare them to corresponding networks constructed from the USF Norms. A node in a network represents a word, and an edge between two nodes represents some relationship between two words. We will refer to all networks constructed from word2vec and GloVe models as synthetic semantic networks, to differentiate them from the empirically derived USF network.

### 3.3.1 Undirected Networks

Given the word association matrix $\boldsymbol{A}$, where $\boldsymbol{A}_{ij}$ represents the association strength between the word $w_i$ and the word $w_j$ in the USF Norms database, the undirected graph or network *USF undirected* is defined as $G = \{E_{USF}, N_{USF}\}$. The edges $E_{USF}$ are defined as

$$E_{USF} = \big\{\{i,j\} \mid \boldsymbol{A}_{ij} + \boldsymbol{A}_{ji} > 0\big\}, \tag{3.2}$$

and the set of nodes as

$$N_{USF} = \big\{i \mid w_i \in W_{USF}\big\}, \tag{3.3}$$

where $W_{USF}$ is the set of all words, and $i$ and $j$ are unique numeric indices assigned to each word. Thus, an edge is placed between two nodes $i$ and $j$, if $w_j$ was produced in the experiment as a response to the cue word $w_i$, or vise versa. We note that this definition of the graph disregards the association strengths between the two words, it only considers the existence of the association in either direction.

For each DSM we construct a corresponding $G_{DSM} = \{E_{DSM}, N_{DSM}\}$, where the set of nodes $N_{DSM} = N_{USF}$, since the set of words in all models is the same. The set of edges for undirected networks constructed from DSMs is

$$E_{DSM} = \big\{\{i,j\} \mid \boldsymbol{S}_{ij} > \tau\big\}, \tag{3.4}$$

where $\boldsymbol{S}$ is a similarity matrix and $\tau$ is a similarity threshold, both unique to each DSM and the specific method used to construct the network. We use two different methods to construct such undirected networks. The first method computes the cosine angle between all vectors and stores those similarities in the matrix $\boldsymbol{S}$. Thus, $\boldsymbol{S}_{ij}$ is the semantic similarity between the word $w_i$ and the word $w_j$. This method creates two undirected networks which we simply refer to as *glove* and *word2vec*. The second method applies Luce's Choice Axiom (Luce 1959):

$$\boldsymbol{P}(i,j) = \frac{\boldsymbol{S}_{ij}}{\sum_{k=1}^{\tau} \boldsymbol{S}_{ik}} \tag{3.5}$$

to convert similarity values in $\boldsymbol{S}$ to probabilities $\boldsymbol{P}$. The threshold values $\tau$ have been determined separately for each DSM and each method by sweeping over a range of values. A single $\tau$ was selected such that it generates a network with the average node degree $\langle k \rangle$ closest to that of the *USF undirected* network. This method approximates the process of free associations, and according to Jones et al. (2018) it is the most representative way of comparing DSMs to free association data. Two networks generated by this method are referred to as *glove-luce* and *word2vec-luce*. To summarize, the five undirected networks generated by the above methods are *USF undirected, glove, word2vec, glove-luce* and *word2vec-luce.*

### 3.3.2 Directed Networks

The directed association network *USF directed* is defined as $G^d = \{E^d_{USF}, N^d_{USF}\}$, where the set of edges is

$$E^d_{USF} = \big\{\{i, j\} \mid \boldsymbol{A}_{ij} > 0\big\}, \tag{3.6}$$

and the set of nodes is the same as in the undirected networks. Thus, the major difference compared to undirected networks is that this network distinguishes between the directionality of the association.

Three different methods are used to construct directed networks from DSMs: the $k$-nn method (Steyvers and Tenenbaum 2005), the $cs$-method (Utsumi 2015) and Luce's Choice Axiom. The $k$-nn method produces a network with the same out-degree distribution as the out-degree distribution of the *USF directed* network by connecting each node in the network to the $k$ other nodes. Those $k$ nodes are the most similar words to the word represented by that node, $k$ corresponds to the number of outgoing links for that word in *USF directed*. The two directed networks constructed with these methods are *glove-knn* and *word2vec-knn*.

In the $cs$-method, the following formula is used to determine the smallest neighbourhood size $k$ for each word $w_i$:

$$\frac{\sum_{j \in V_i^N} \boldsymbol{S_{ij}}}{\sum_{j' \neq i} \boldsymbol{S_{ij'}}} > R, \tag{3.7}$$

where $V_i^N$ is the set of indices of $k$ most similar words in the neighbourhood of the word $w_i$ with $k = |V_i^N|$. The threshold $R$ is determined in such a way that the resulting average node degree $\langle k \rangle$ of the network is the same as that in the directed association network. For GloVe, we used $R = 38$ and for word2vec $R = 76$, and we refer to the resulting networks as *glove-cs* and *word2vec-cs*.

Luce's Choice Axiom is also applied as for the undirected networks, and it yields networks that have $\langle k \rangle$ as close as possible to that of the *USF directed* network. Appropriate thresholds $\tau$ are found by exploring a range of values for each DSM. This method yields two more networks: *glove-luce-dir* and *word2vec-luce-dir*. Thus, in total there are six directed networks created from DSMs: *glove-knn, word2vec-knn, glove-cs, word2vec-cs, glove-luce-dir* and *word2vec-luce-dir*.

## 3.4 Network Analyses

### 3.4.1 Network Statistics

Human word association networks exhibit a few interesting structural principles such as the scale-free property and small-world characteristics (Morais et al. 2013; Steyvers et al. 2004a; Utsumi 2015). The scale-free property is an important feature of many complex systems whereby some nodes, called hubs, have a large number of connections to other nodes, while others have only a few connections (Barabási and Bonabeau 2003). This property is discussed in a greater detail in the next section on degree distributions. Small-world networks are characterized by sparse connectivity, small average shortest path lengths (ASPLs) relative to their size, and high average clustering coefficients. In the context of semantic networks, such small-world structure is important for the efficient search and retrieval of items from memory. The following analyses are inspired by similar research (Morais et al. 2013; Steyvers et al. 2004a) that investigated properties of other semantic networks.

For a network with $n$ nodes, the average shortest path length $L$ is computed as

$$L = \frac{1}{n(n-1)} \sum_{i,j \in G} d(i,j), \tag{3.8}$$

where $i$ and $j$ are two different nodes, and $d(i,j)$ is the shortest distance between the two nodes measured as the number of edges between them. The diameter $D$ of a network is the longest shortest distance between any two nodes in the network.

Another measure typically considered in the context of small-world networks is the average clustering coefficient $C$. The average clustering coefficients of such networks are higher than the clustering coefficients of random networks[2] of the same size that have the

[2] We use random networks generated with the commonly used Erdős-Rényi model (Erdös and Rényi 1959).

|  | $L$ | $\langle k \rangle$ | $C$ | $C_k$ | $D$ | $m$ | $L_{rnd}$ | $C_{rnd}$ | $s$ |
|---|---|---|---|---|---|---|---|---|---|
| USF undirected | 3.04 | 22.0 | 0.186 | 100.00 | 5 | 55,236 | 3.03 | 0.004 | 0.44 |
| glove | 4.61 | 22.1 | 0.373 | 98.88 | 12 | 51,244 | 2.99 | 0.005 | 0.48 |
| glove-luce | 3.72 | 23.0 | 0.260 | 100.00 | 7 | 57,300 | 2.99 | 0.005 | 0.46 |
| word2vec | 4.24 | 21.3 | 0.325 | 99.84 | 12 | 52,317 | 3.04 | 0.004 | 0.44 |
| word2vec-luce | 3.66 | 22.8 | 0.248 | 100.00 | 6 | 56,809 | 2.99 | 0.005 | 0.46 |
| USF directed | 4.26 | 12.7 | 0.187 | 96.51 | 10 | 63,619 | 3.62 | 0.005 | 0.25 |
| glove-cs | 5.06 | 12.3 | 0.266 | 97.21 | 12 | 61,470 | 3.65 | 0.005 | 0.25 |
| glove-knn | 5.03 | 12.7 | 0.259 | 97.91 | 13 | 63,262 | 3.62 | 0.005 | 0.25 |
| glove-luce-dir | 5.38 | 12.3 | 0.272 | 97.27 | 21 | 61,374 | 3.63 | 0.005 | 0.25 |
| word2vec-cs | 4.81 | 12.5 | 0.237 | 99.28 | 11 | 62,328 | 3.64 | 0.005 | 0.25 |
| word2vec-knn | 4.77 | 12.7 | 0.232 | 99.32 | 12 | 63,165 | 3.64 | 0.005 | 0.26 |
| word2vec-luce-dir | 4.92 | 12.3 | 0.260 | 98.91 | 13 | 60,984 | 3.67 | 0.005 | 0.25 |

Table 3.2: Graph-theoretic statistics of networks derived from the USF Norms, word2vec and GloVe vectors. The results for undirected networks are shown in the upper panel of the table, and of directed networks in the lower panel. Abbreviations: $L$ = the average shortest path length, $k$ = the average node degree, $C$ = the average clustering coefficient, $C_k$ = connectivity, the number of nodes in the largest connected component (expressed in %), $D$ = the network diameter, $m$ = the number of edges, $L_{rnd}$ = the average shortest path of a random network, $C_{rnd}$ = the average clustering coefficient of a random network, $s$ = network sparsity (expressed in %).

same probability of a connection between any two nodes. It is computed as the average of individual clustering coefficients $c_i$ of all nodes as

$$C = \frac{1}{n} \sum_{i \in G} c_i = \frac{1}{n} \sum_{i \in G} \frac{2t_i}{k_i(k_i - 1)}, \tag{3.9}$$

where $t_i$ is the number of triangles in the neighborhood of the node $i$. A triangle is a connectivity pattern where a node $i$ is connected to two other nodes $h$ and $j$, and at the same time the nodes $h$ and $j$ are also connected. The denominator in the equation above is the total number of all possible connections in a neighborhood of a node with the degree $k_i$: $k_i(k_i - 1)/2$.

To test whether networks constructed from word2vec and GloVe models exhibit small-world characteristics, we compare them to those of the USF Norms according to different measures of network topology. The sparsity $s$ of a network is computed by dividing the average node degree $\langle k \rangle$ with the total number of edges in the network. Other measures such as the clustering coefficient $C$, the average shortest path length $L$ and the diameter $D$ are computed in the largest connected component of each network.

The results are shown in Table 3.2. We were able to replicate results for the two USF networks reported previously (Steyvers et al. 2004a; Utsumi 2015). Due to the methods used to construct the networks, all synthetic networks have sparsity that is comparable to the sparsity of the human association network. Their average shortest path lengths are consistently higher, but still comparable to those of the human association networks. However, in undirected networks, the diameter of some synthetic networks is more than twice the length as that of the USF network, meaning that the distance between the two farthest words is longer in the synthetic networks than it is in the association network. Interestingly, this is not the case for undirected networks created with the process model, where we observe comparable diameters. Although synthetic networks on average have fewer edges and fewer nodes, they are more modular compared to the corresponding USF networks, as indicated by higher clustering coefficients. This effect is slightly more pronounced for undirected versions of synthetic networks that were not created with a method using Luce's choice axiom. Based on these observations, we conclude that synthetic networks exhibit small-world characteristics reminiscent of word association networks.

### 3.4.2 Degree Distributions

Degree distributions are studied to understand the structure of the network, as well as the connectivity patterns of nodes in the network. Such distributions for the USF Norms are well characterized, and we examine the extent to which networks built from GloVe and word2vec display such characteristics. To obtain the distribution of degrees in a network, we count the number of nodes with $k$ degrees, where $k$ ranges from one to $k_{max}$. The $k_{max}$ value denotes the highest node degree and it is different for each network as a result of different methods used to create the networks. Following the approach in related literature (Steyvers et al. 2004a; Utsumi 2015; Morais et al. 2013), we focus on the distribution of in-degrees in all analyses. According to De Deyne and Storms (2008), using the out-degree distribution introduces a bias since that distribution depends on the task characteristics, such as the number of associations per cue word. The degree distribution of the directed association network is known to follow a truncated power-law distribution $P(k) \sim e^{-\lambda k} k^{-\alpha}$ , or, in some cases, a pure power law $P(k) \sim k^{-\alpha}$ (Utsumi 2015; Morais et al. 2013). The power law predicts that most nodes in the network have a few connections, while a small number of nodes, regarded as *hubs*, have a rich local neighborhood, which is a typical scale-free property.

To fit degree distributions to different models and to evaluate the plausibility of fits we follow the methods of Clauset et al. (2009). In line with those methods, we first test the plausibility of a power-law behaviour using the goodness-of-fit test, followed by the loglikelihood-ratio (LR) test that examines whether other heavy-tailed distributions provide a better fit. To fit and evaluate different models, we use the Python powerlaw package (Alstott et al. 2014).

First, we fit the empirical degree distribution to a power-law model using the maximum likelihood estimation (MLE) for the parameter $\alpha$ in the exponent. The fit is performed for values of $k > k_{min}$, where $k_{min}$ was determined such that the Kolmogorov-Smirnov (KS) distance between the empirical distribution and the model distribution for values greater than $k_{min}$ is minimized. The plausibility of a power-law behaviour is then evaluated by sampling thousands of distributions using the model with the $\alpha$ parameter. For each sampled distribution the KS distance is measured between the distribution and the model. Then, the KS distance is also measured between the model and the empirical distribution. The resulting $p$-value is a fraction of sampled distributions that have a greater KS distance than the empirical distribution. Large $p$-values denote that sampled distributions are more distant than the empirical distribution, in which case the model is regarded as a plausible fit to the empirical data.[3]

The LR-test is a comparative test that evaluates which of the two distributions is more likely to generate samples from the empirical data based on maximum likelihood functions of each distribution. The resulting $R$ value is positive if the first distribution is more likely, and negative if the second distribution is more likely. If $R = 0$ both distributions are equally likely. The alternative heavy-tailed distributions we tested are: truncated power law, (discretized) lognormal and exponential. The distributions used in the tests are shown for a toy example of $k \in [1...10]$ in Figure 3.2 on different scales to highlight the difference in their tails that are less apparent using the linear scale.

All test results are summarized in Table 3.3. According to the performed goodness-of-fit test for the power-law distribution, it is an unlikely description of degree distributions for any network, since all of the power-law models generating data with the fitted parameter produced distributions that are closer to the model distribution than the empirical distribution, as evidenced by $p = 0.00$ in all cases.

[3] This usage of $p$-value as a fraction has the opposite interpretation of the one used for statistical significance testing, where $p$ values larger than some significance threshold suggest that null hypothesis cannot be rejected in favor of the alternative hypothesis.

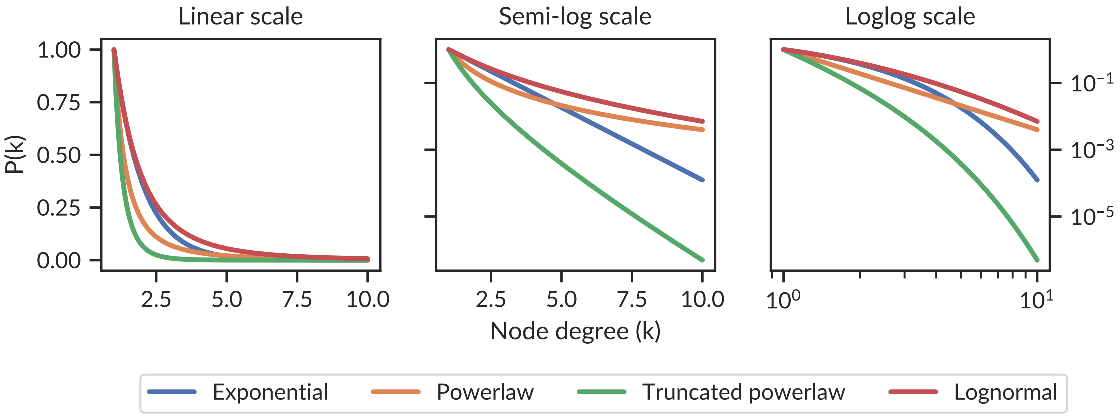


Figure 3.2: Plausible degree distributions for word networks shown using different scales.

Table 3.3: Goodness-of-fit test for the power law distribution and loglikelihood ratio tests evaluating plausibility of the power-law versus other heavy-tailed distributions. The results of undirected networks are presented in the upper panel. Abbreviations: $KS =$ Kolmogorov-Smirnov statistic, $LR =$ loglikelihood ratio.

| | Power Law | | Power Law vs. Truncated Power Law | | Power Law vs. Lognormal | | Power Law vs. Exponential | |
|---|---|---|---|---|---|---|---|---|
| | KS | $p$ | LR | $p$ | LR | $p$ | LR | $p$ |
| USF undirected | 0.014 | 0.00 | -1.80 | 0.02 | -0.91 | 0.37 | 7.46 | 0.00 |
| glove | 0.035 | 0.00 | -7.29 | 0.00 | -5.03 | 0.00 | 7.82 | 0.00 |
| glove-luce | 0.026 | 0.00 | -1.20 | 0.08 | -1.08 | 0.28 | -0.30 | 0.76 |
| word2vec | 0.064 | 0.00 | -7.72 | 0.00 | -5.31 | 0.00 | -5.89 | 0.00 |
| word2vec-luce | 0.031 | 0.00 | -0.97 | 0.05 | -1.03 | 0.30 | -0.26 | 0.80 |
| USF directed | 0.055 | 0.00 | -2.54 | 0.00 | -2.36 | 0.02 | 0.27 | 0.79 |
| glove-cs | 0.016 | 0.00 | -1.14 | 0.17 | -0.67 | 0.50 | 3.52 | 0.00 |
| glove-knn | 0.020 | 0.00 | -1.05 | 0.16 | -0.82 | 0.41 | 1.08 | 0.28 |
| glove-luce-dir | 0.047 | 0.00 | -3.99 | 0.00 | -2.79 | 0.01 | -3.47 | 0.00 |
| word2vec-cs | 0.032 | 0.00 | -0.57 | 0.40 | -0.51 | 0.61 | 0.29 | 0.77 |
| word2vec-knn | 0.028 | 0.00 | -0.13 | 0.82 | -0.12 | 0.90 | 0.69 | 0.49 |
| word2vec-luce-dir | 0.027 | 0.00 | -0.21 | 0.81 | -0.13 | 0.89 | 0.73 | 0.47 |

As per the loglikelihood-ratio test, the truncated power-law is a better description of degree distributions for two out of four undirected networks ($p < 0.05$), namely those that were not generated with the process model. However, alternative heavy-tailed distributions such as the lognormal could not be ruled out for *glove* and *word2vec*. The tests for other undirected networks provided no insights into the plausibility of other distributions.

For directed networks, we find that the truncated power law, rather than a pure power law, is a better description for the distribution of degrees for the direct USF network, which is consistent with previous research (Morais et al. 2013; Utsumi 2015). However, our results indicate that the lognormal distribution is also a plausible model for the USF network, as it is not possible to distinguish between the lognormal and truncated power law distributions ($R = -0.43, p = 0.67$, not shown in the table). We also observed high $p$-values ($p \geq 0.15$) for all log-likelihood ratio tests performed with degree distributions of all synthetic directed networks, yielding inconclusive results. The only exception is *glove-luce-dir*, where the test could not distinguish between the truncated power-law ($\lambda = -3.99,\ p < 0.01$), lognormal ($\lambda = -2.79,\ p < 0.05$) or exponential distribution ($\lambda = -3.47,\ p < 0.01$),

Given that many of the test results were inconclusive, we further plot model fits to empirical degree distributions on a semi-log scale in Figure 3.3 to qualitatively investigate the relationships between fitted models and the empirical data. Degree distributions are expressed as complementary cumulative distribution functions (CCDFs), and fits for the power law, truncated power law, lognormal, and exponential models with best MLE parameters are shown. While the distribution of degrees of USF networks is bounded from above by the power-law distribution and from below by the exponential distribution, this is only somewhat the case for the *glove* networks and less so for the *word2vec* networks, indicating differences between degree distributions of human and synthetic word networks. We also note that among directed networks, synthetic networks have fewer nodes with high degrees compared to the word association network. For example, none of the synthetic networks has a node with $k > 150$, and the highest degrees in some networks ($k_{max}$) are an order of magnitude lower than those of the association networks. Thus, it is likely that the lack of data points in this regime containing highly connected nodes contributes to the poor fits and inconclusive LR test outcomes, since it is precisely those heavy tails that distinguish such distributions.

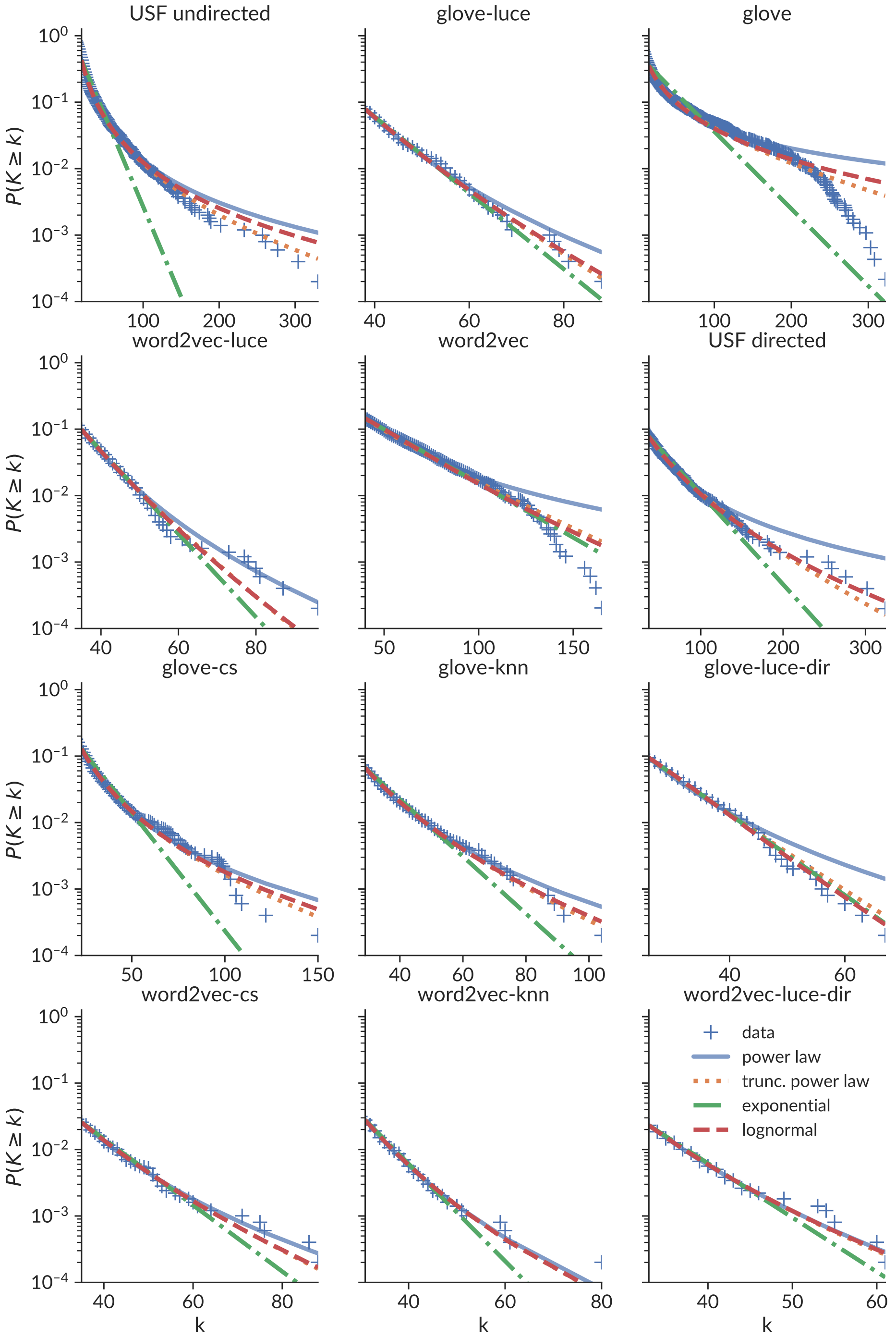


Figure 3.3: Complementary cumulative distributions and model fits for the *USF Norms*, *glove* and *word2vec* networks.

### 3.4.3 Hierarchical Topology

Human association networks have hierarchical organization, resulting from high modularity as measured by the dense connectivity between groups of nodes (Utsumi 2015). Such groups are also known as clusters, and they are highly interconnected so that they form only a few connections to nodes that are not part of the group. The presence of such clusters indicates that there are features shared among nodes in the network (e.g., semantic relatedness).

While the average clustering coefficients are reported in Table 3.2, to investigate the presence of hierarchical structure, we consider the relationship between every node degree $k$ and the local clustering coefficients $c_i$, computed using Equation 3.9. In networks that exhibit hierarchical organization, the local clustering coefficient is dependent on the node degree and has been observed to follow a scaling law of the form $C(k) \sim k^{-\gamma}$ (Ravasz and Barabási 2003). While many hierarchical networks have been observed to have $\gamma = 1$, hierarchical structure has also been observed in networks with $\gamma < 1$. Directed USF network has been shown to follow that scaling law with $\gamma = 0.75$ (Utsumi 2015).

To explore the relationship between the local clustering coefficients and node degrees in synthetic word networks we implement the methods from Utsumi (2015). First, we compute local clustering coefficients $c_i$ for all nodes in the largest connected component in each network. We then compute the average clustering coefficient for each neighborhood size $k$ and connect those values to form a line. Finally, we use linear regression in the logarithmic space to determine the slope of the regression line and the correlation coefficient.

The results are shown in Figure 3.4. For undirected versions of synthetic networks, we find small positive slopes and weak to moderate correlation for all networks. This indicates that the clustering tendency increases for nodes with high degrees, which is the pattern opposite of that observed with association norms. However, we found strong negative correlations between the local clustering coefficient and the node degree for some directed networks. Specifically, for *glove-cs*, *glove-knn* and *word2vec-knn*. The network with the highest scaling exponent is *glove-knn* with $\gamma = 0.49$, which is still lower than the corresponding coefficient for associations $\gamma = 0.75$. Thus, while the clustering pattern is not as prominent as that of human associations, it is present in some of the directed networks.

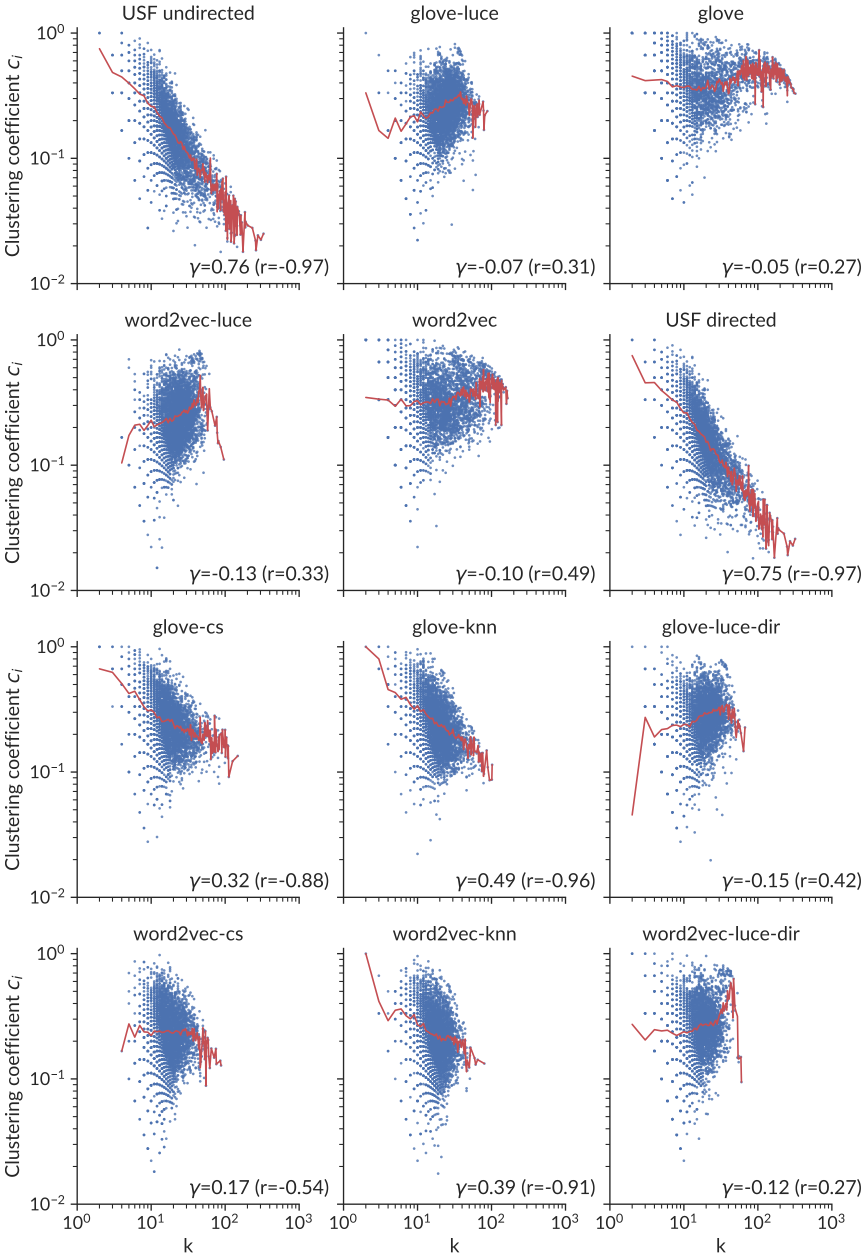


Figure 3.4: Local clustering coefficients $c_i$ as a function of node degree $k$. Dots represent local clustering coefficients of each individual node. The line connects $c_i$ averages of nodes with the same degree $k$. $\gamma$ = scaling law exponent, $r$ = the correlation coefficient between $c_i$ and $k$.

## 3.5 Discussion

In this chapter, we applied a set of network-theoretic analyses to word2vec and GloVe, two widely used distributional semantic models. To this end we constructed semantic networks from such models using different methods proposed in related research (Utsumi 2015; Steyvers and Tenenbaum 2005), and compared such networks to those of human word associations (Nelson et al. 2004). We argued that such analyses can be used to complement existing definitions of psychological plausibility of such models (Günther et al. 2019; Mandera et al. 2017).

We found that all networks exhibit the small-world property, characterized by short path lengths and high clustering coefficients. These results are consistent with previous studies that demonstrated small-world structure in other kinds of DSMs (Steyvers and Tenenbaum 2005; Utsumi 2015). Degree distribution analyses based on a goodness-of-fit test revealed that the power law is not a plausible model in any of the networks. This finding may not be surprising considering that some semantic networks created from word associations have distributions that can be well described with alternative heavy-tailed distributions, such as the truncated power law (Morais et al. 2013; Utsumi 2015). The degree distribution of synthetic undirected networks tested here is also explained best by the truncated power-law model.

Truncated power-law behaviour could not be reliably inferred for any of the directed *word2vec* and *glove* networks. Analyzing the tails of distributions in Figure 3.3 provides some insight as to why it is difficult to obtain a clear fit in those cases. Directed networks have only a few nodes with a high number of connections. For example, there are only four nodes with $k > 80$ for *glove-knn*, less than ten nodes with $k > 55$ for both *word2vec* networks, and for networks generated with Luce's choice axiom there are five or less nodes with $k > 50$ in both cases. This is in stark contrast to human word networks, where there are more than 260 words with $k \geq 60$. In addition, inspecting those "hub" nodes for each synthetic network reveals that those are different kinds of words, compared to those listed in Table 3.1. More specifically, words with the most connections to other words in association networks reflect activities and behaviours typical of daily life, such as those related to eating, socializing and working. For synthetic networks we found that those words are more specific. For example, for some *glove* networks the words with most connections were *salad, sauce* and *butter*, and many of the top 10 words were often ingredients or food-related words. However, the word *food* appeared only at the 232nd place in the list of words with highest degrees for that network. We speculate that this trend, where more specific food words have more connections to other words than general words, may be due to the training data used. GloVe is trained on data crawled from the Web and as such might be biased towards usage of such food words in the written context (e.g., food blogs and recipes). Thus, given that those heavy-tails look different for synthetic networks and word associations, we postulate that tests we conducted to understand which model explains such distributions did not have enough data to reliably discriminate between different distributions.

We finally analyzed the level of hierarchical structure in synthetic networks and compared it to that of word associations. Overall, we found that synthetic networks on average have much higher clustering coefficients. Some directed networks, more specifically those constructed with the $k$-nn method, exhibit a moderate level of hierarchical organization that is reminiscent of clustering observed in the human association network. We also qualitatively observe highly-connected nodes in the directed *glove-cs* and *glove-knn* networks with fluctuating clustering coefficients, a trend especially prominent in word association networks in Figure 3.4. Although such hubs are semantically different for synthetic networks and word associations, as established with the analysis of words contained in the distribution tails, they share an interesting property—some hubs with higher degrees are "embedded" in clusters, thus connecting to all words in their neighbourhoods (e.g., words such as *hurt*, *music* and *eat*). Other hubs are acting as intermediaries in the network by connecting different groups of the word network, where words in those groups might not have as many connections (e.g., *money*, *water* and *car*).

Overall, these results indicate that different semantic networks constructed from word2vec and GloVe models are capable of capturing some aspects of human association networks. For example, such similarities were evident in general network statistics such as the shortest paths and clustering tendencies. The *glove-knn* network exhibits clustering properties that are most similar to those of the word association norms. However, for the majority of synthetic datasets there are clear differences with the empirical networks. These differences were most apparent for words represented in the long tails in degree distributions. We note that these differences should be considered when using DSMs for research in human semantic memory as, depending on the specific semantic task, some of DSMs might provide a poor match for cognitive representations or processes. For example, while some such models might be plausible when using them in tasks that require listing of specific word categories (e.g., semantic fluency task) due to their hierarchical structure, they might only weakly capture the relevance of semantic relationships between words commonly used in ordinary language. To conclude, while DSMs offer invaluable and convenient resources for researchers interested in modelling human cognition, it is important to understand their structure, limitations, and the implications thereof for modelling of psychologically plausible semantic processes.

# 4 | Associative Mechanisms in the Brain: The Remote Associates Test (RAT)

In the previous chapter, we compared semantic networks created from different sources of semantic data, and showed that networks derived from distributed semantic models, such as word2vec (Mikolov et al. 2013a; Mikolov et al. 2013b) and GloVe (Pennington et al. 2014), differ from human word associations (Nelson et al. 2004) in important ways. Given the significance of associative word relationships in language acquisition as well as in a variety of linguistic behaviours more generally, in this chapter we investigate a biologically plausible model of word associations in the brain. To this end, we use different sources of semantic data to model associative relationships in a biologically constrained neural network and evaluate them on a specific semantic memory task known as the Remote Associates Test (RAT).

The RAT was proposed in the context of research on creative thinking and insight problem solving (Mednick 1962), and has been used to study the organization and retrieval processes in semantic memory (Davelaar 2015; Kenett et al. 2014; Smith et al. 2013; Gupta et al. 2012). The test consists of word triplets, and the task is to find a fourth word that relates to all three words in a triplet. While cognitive strategies used in the RAT have been exhaustively characterized, little is known about the neurocomputational basis underlying semantic search in the RAT. In this chapter, we present a set of analyses and models that propose biologically constrained computational mechanisms that solve the RAT in a human-like way. That is, the model achieves similar accuracy on the test, and it produces responses that exhibit semantic similarity characteristics similar to those of human responses.

Section 4.1 explains the task in a greater detail and discusses the related work in this domain with a focus on existing computational models. In Section 4.2, we evaluate different sources of association and word co-occurrence data and examine their performance on the test assuming the insight test condition. These preliminary analyses do not attempt to characterize neural or cognitive mechanisms underlying RAT solving. Instead, they are used as a stepping stone for investigation of association data used in the models presented in later sections. Aspects of that section are adapted from Kajić et al. (2016). Based on those evaluations, in Section 4.3, we present a cognitive neural network model that simulates the semantic search process in the RAT. We show that the parameters of the model can be adjusted in such a way as to interpret the model performance as differentiating between RAT problems of varying difficulty. The section is adapted from Kajić and Wennekers (2015). Finally, an extension of the model implemented with the methods of the NEF (Eliasmith and Anderson 2003) is presented in Section 4.4, based on Kajić et al. (2017b). We argue that the resulting model is biologically plausible, as it incorporates neurally and cognitively constrained mechanisms of the semantic search process in the RAT, while providing a robust match with the human data.

## 4.1 Background and Related Work

Creating word associations is an important skill for the development of many cognitive abilities. For example, language acquisition is highly dependent on the ability to create associations (Elman et al. 1997; Rogers and McClelland 2004), as they are a central means of expanding both vocabulary and syntax (Brown and Berko 1960; Hills 2013). As well, associations allow infants to learn about previously unseen objects or concepts in terms of semantic similarities and semantic distinctions (Mandler and McDonough 1993). Empirically derived word associations have been successfully used to model a variety of semantic memory tasks (Steyvers et al. 2004a; Abbott et al. 2015; De Deyne et al. 2016; Kajić et al. 2017a), substantiating the importance of associative relationships for different linguistic behaviours. Because associations play such a crucial role in language and human cognition more generally, it is important to understand how the brain might represent, store, and deploy them.

In order to do so, we investigate a biologically and cognitively motivated representation of word associations and accompanying processes that operate on such representations on the example of the Remote Associates Test (RAT). The RAT was developed in the 1960s by Mednick (1962) to measure an individual's ability to think of words that are remotely associated with a given word or a set of words. Since then, it has been widely used in creativity research, postulating that such remote associations require thinking of word relationships that are novel or unusual, two characteristics of creative ideas (Boden 2003). The RAT consists of a list of problem items where each item contains three cue words, and the task is to find a solution word that is related to all three cues. The test is administered with a time limit that can range anywhere from just a few seconds up to a few minutes. It is argued that an individual's creative ability correlates with their performance on the test, as highly creative individuals are better at finding solutions to those RAT problems that require them to relate familiar words in a novel way. In other words, such individuals are portrayed as being better at "thinking outside the box."

An example of a RAT problem is the following triplet of cue words: *fish*, *mine, rush*. The task is to find a word, also referred to as the *target*, that is related to all three cues. In this case, thinking about common associations of each of the cue words, such as *water*, *coal*, or *hour*, meaningfully matches with individual cue words, but none of them is a good match with all three cues. Instead, *gold*, a less frequent associate of each of the words, is the correct solution as it can be meaningfully combined with all three of them. The associative relationship between the cues and the solution in the RAT can vary: it can be a compound word such that each cue and the solution form a new word (e.g., *firefly*); it can be semantically related (e.g., *water* and *ice*); or it can form an expression (e.g., *mind games*). Mednick (1962) proposed that creative individuals are more likely to think of unstereotypical words that are solutions in the RAT. He attributed this to their flat associative hierarchy, in which the probability of coming up with an association is not very different for typical and atypical associations. In contrast, individuals scoring lower on the RAT would produce stereotypical associates with higher probability than atypical associates, which he described as a steep associative hierarchy.

Performance on the test is expressed as the number of correctly solved items within a fixed time limit. Longer time limits (e.g., one or two minutes) correlate with higher solution rates (Bowden and Jung-Beeman 2003a), and it has been argued that when people are given more time to solve a problem they tend to use *analytical* solving skills that rely on a deliberate search process. In doing so, people think of words and test them individually against the cue words, a conscious act characterized by incremental awareness of a solution (Smith and Kounios 1996). In contrast, shorter solving times are thought of as invoking sudden and involuntary *insight* solutions characterized by the "Aha!" moment of reinterpreting a stimulus, situation or an event (Kounious and Beeman 2014).

The RAT has also been used in semantic memory research to study and characterize responses people give when attempting to solve a problem. Although each individual cue word elicits associations, the solution is found in the search space constrained by all three cues. Analyses of responses people give when solving a RAT problem have shown that such a search process exhibits particular characteristics that differentiate it from other memory search processes (Raaijmakers and Shiffrin 1981; Hills et al. 2012b). Specifically, the RAT search process retrieves words that are strongly related to one of the three problem cues, shows occasional switching between the cues (Davelaar 2015; Smith et al. 2013), and it involves a local search strategy (Smith et al. 2013; Smith and Vul 2015).

Performance on the RAT has been characterized by numerous experimental, theoretical, and computational studies (Gupta et al. 2012; Kenett et al. 2014; Klein and Badia 2015; Olteteanu and Falomir 2015). The proposal by Mednick (1962) on flat associative hierarchies of high-scoring individuals has been supported experimentally by showing that individuals who score higher on the RAT tend to avoid high-frequency answers on both incorrect and correct trials (Gupta et al. 2012; Kenett et al. 2014). The properties of individual subjects' semantic networks correlates with their performance on the RAT (Kenett et al. 2014; Monaghan et al. 2014). Specifically, individuals who score high on a battery of creativity tests have semantic networks with small-world properties (Kenett et al. 2014). The connectivity in such networks is sparse, as they are characterized by short average path lengths between words, and strong local clustering. Even though every node in the network is only sparsely connected, it takes just a few associations to reach any other node in the network. This kind of topology facilitates solution-finding in the RAT because quick, efficient searches can cover much of the semantic network. High scores on the test have been achieved with corpus-based approaches relying on large co-occurrence statistics (Toivonen et al. 2013; Klein and Badia 2015).

However, fewer studies have investigated models which exhaustively match performance on the RAT with the human performance (Bourgin et al. 2014; Gupta et al. 2012). More specifically, instead of comparing the accuracy on the test, a more detailed evaluation of the model with respect to human performance can be achieved by considering the difficulty of RAT problems, and the responses people give when solving a problem. Thus, a model that matches the human data more robustly should be able to account for differences in difficulty among different test items. We address that gap in this chapter, by presenting a biologically plausible model of the RAT solving that robustly matches human performance on the test, while incorporating aspects of neural and cognitive computational constraints.

## 4.2 Evaluating Different Semantic Spaces

Following our conclusions in Chapter 3 on the importance of understanding the datasets for cognitive modelling, we first examine the relationship between different association and co-occurrence datasets and their performance on the RAT. In particular, we examine associative relationships in different semantic datasets in the context of human performance on the RAT. We use different methods to derive representations from such semantic spaces, and evaluate those representations to understand how they affect the test performance. These evaluations are not intended to address aspects of cognitive or biological mechanisms underlying RAT solving. Rather, they are used as a way of investigating how well remote associations, which are important for good performance on the test, are being captured by different semantic datasets. Insights obtained from these investigations are used to inform the design of cognitively and biologically plausible models presented later in this chapter.

Two different datasets are used to construct different word representations that are subsequently evaluated on the RAT. The first dataset contains Free Association Norms (FAN; Nelson et al. 2004). We collect the FAN association strengths in an asymmetric association matrix, $FAN_{\text{asym}}$, with rows representing cues, columns representing targets, and individual entries representing normed frequency responses used in the free association experiment as described in Section 3.1. As previously discussed, one distinguishing feature of this dataset is the association asymmetry. For example, given the cue *left*, $94\%$ of subjects respond with *right*, however, given the cue *right*, $41\%$ of subjects respond with *left* and $39\%$ of subjects respond with *wrong*. In addition to the asymmetric matrix, the symmetric matrix, $FAN_{\text{sym}}$, is defined as

$$FAN_{\text{sym}} = \frac{1}{2}\left(FAN_{\text{asym}} + FAN_{\text{asym}}^{\top}\right). \tag{4.1}$$

The second dataset we derive associative information from is the Google Books Ngram Viewer dataset (version 2 from July 2012; Michel et al. 2011). An n-gram is a sequence of $n$ words, and this dataset provides occurrence frequencies of n-grams across over 5 million books published up to 2008. The set of words used in this study is restricted to the words in the association norms, and n-gram frequencies used are those from 2008. For every combination of two words $w_1$ and $w_2$, the corresponding entry in the matrix $NG_{\text{asym}}$ was set to the sum of occurrences of the 2-gram $(w_1, w_2)$ and the 1-gram $w_1 w_2$ in the corpus. An 2-gram is a tuple of two words, such as *(peep, hole)*, while 1-gram is a single word such as *peephole*. Each row of the matrix was then normalized to sum to one. The symmetric matrix, $NG_{\text{sym}}$, is computed in a way similar to the computation of the $FAN_{\text{sym}}$matrix with Equation 4.1. In the rest of the analysis, we use $NG_{\text{sym}}$, which includes the backward strength between word co-occurrences. This is necessary to solve the problems where only the second part of the compound word is given as one of the three cues (e.g., *board* for *blackboard*). Even though the NG matrices give co-occurrence

Table 4.1: Target positions for 117 RAT problems. Reproduced from Kajić et al. (2016, Table 1).

| | Within top | | | | |
|---|---|---|---|---|---|
| Association matrix | 1 | 2 | 3 | 5 | 10 |
| $FAN_{\mathrm{asym}}$ | 5 | 12 | 16 | 31 | 49 |
| $FAN_{\mathrm{sym}}$ | 4 | 5 | 6 | 14 | 36 |
| $NG_{\mathrm{sym}}$ | 11 | 15 | 16 | 22 | 35 |

counts, we will use the terms *association matrix* and *association strength* as they are used in the same manner as the FAN association matrix.

Given such matrices, there are two common ways of representing word associations. First, we can directly use the association matrix, where we represent a word as a localist, one-hot encoded vector. Then, to perform the association we multiply the word by the association matrix. This will "extract" the row of the matrix corresponding to that word and the resulting vector will represent the word associates. Alternatively, we can embed the associates in a vector space using one of the methods described in Section 2.3.3. That is, instead of representing the full association matrix, we compress that matrix into a lower-dimensional space. In certain cases this approach can adjust the similarity space between the words to uncover latent structure among the associations, such as with LSA (Deerwester et al. 1990). Like LSA, we use singular value decomposition (SVD) to take the 5,018-dimensional localist word representation, and compress it into an $D$-dimensional distributed representation, where $D$ is varied between 128 and 4,096.

To determine which representational approach gives the best performance on the task, we use the problem set from Bowden and Jung-Beeman (2003a). Out of 144 RAT problems, we used the 117 problems for which the cues and the target exist in the vocabulary of free association norms. We take the sum of the vectors representing each cue word, and multiply it by the association matrix. The resulting vector is compared to the vectors for all of the possible response words. In the ideal case, the correct solution word would be the most similar to this output value. However, we also determined if the solution word was in the top 2, 3, 5, and 10 most similar words with recounting,[1] as reported in Table 4.1.

The results indicate that the solution appears as the top-ranked word more often for

[1] Each subsequent top count entry includes words from the previous entry. For example, all words in the top 1 entry are also counted in the top 2 entry.

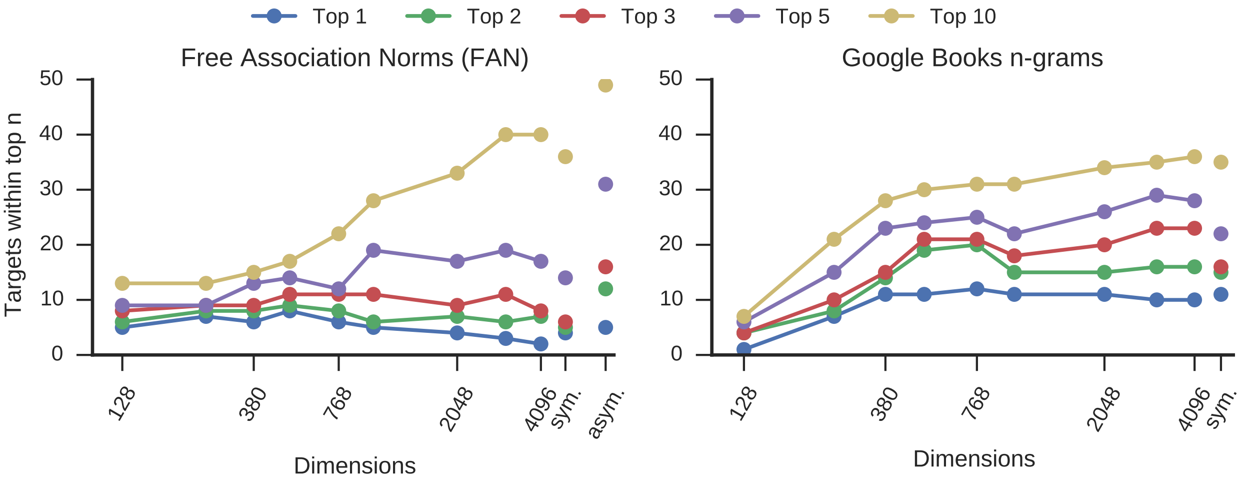


Figure 4.1: The number of correct solutions within the top $n$ most similar words over 117 problems. The isolated points at the end of the x-axis in both graphs represent the original symmetric and (for FAN) asymmetric matrices. All other data points are computed by applying the SVD to the matrices. **Left:** The results with representations based on free association norms. **Right:** The results based on Google n-gram data. Reproduced from Kajić et al. (2016, Figure 1).

Google n-grams ($NG_{\text{sym}}$) than for association norms (11 solutions in the first position versus 5 and 4 solutions for symmetric and asymmetric matrices). However, when relaxing the condition and allowing the solution word to appear among the top 5 or 10 words, we see the $FAN_{\text{asym}}$ association matrix outperforming $NG_{\text{sym}}$. Figure 4.1 also includes the results from applying SVD to the association matrices. Contrary to the expectation, SVD does not improve the performance on the RAT, except in a few circumstances. First, there is no dimension of vectors derived form SVD that produces representations performing better than the asymmetric matrix in the case of FAN. However, SVD-derived representations with dimensionality 512 or higher for $NG_{\text{sym}}$appear to perform better than either $FAN_{\text{asym}}$or the $NG_{\text{sym}}$matrix in most cases (with some exceptions, for example, for solutions within top 10 and top 5 $FAN_{\text{asym}}$achieves the best score overall). Thus, in the majority of the cases, the statistical n-gram data performs better than the free association norms. However, these analyses only indicate increased performance on the task, not whether the n-gram approach performs similarly to people on this task. To address this question, we compare the two approaches with human performance.

Table 4.2: Model fits and best fitting parameters. Reproduced from Kajić et al. (2016, Table 2).

| Association matrix | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ | $\beta$ | $r^2$ |
|---|---|---|---|---|---|
| $FAN_{\text{asym}}$ | 1.0 | 2.06 | 1.20 | 1.13 | 0.58 |
| $FAN_{\text{sym}}$ | 1.0 | 2.50 | 1.63 | 2.86 | 0.58 |
| $NG_{\text{sym}}$ | 1.0 | 13.45 | 1.25 | 3.55 | 0.30 |
| $NG^{768}_{\text{sym}}$ | 1.0 | 11.88 | 1.00 | 8.23 | 0.22 |

### 4.2.1 Matching Human Performance

Instead of examining the best performance on the RAT, we now explore which method yields the results most similar to human results. That is, we investigate which method is better in solving problems that humans find easy, and worse in solving problems humans find hard. To do so, we predict a probability of producing the correct solution within a 2 second time limit. We use the same set of problems as in the previous section, and match to the percentage of people solving each problem from Bowden and Jung-Beeman (2003a). Let $s(w, v)$ be the associative strength from word $w$ to $v$. Given the three cues $c_k$ with $k = 1, 2, 3$ each word $w_i$ in the vocabulary is activated according to

$$a(w_i) = \sum_{k=1}^{3} \alpha_k \cdot s(c_k, w_i), \tag{4.2}$$

where $\alpha_k$ are free parameters intended to model the effect that subjects might differently prioritize the problem cues. We set $a(c_k) = 0$ for the cues to prevent them from appearing high in the results. Moreover, we fix $\alpha_1 = 1.0$, as a scaling of all $\alpha_k$ with a constant will produce the same predictions. Given that $w_{\text{s}}$ is the solution word, we calculate the predicted probability for producing the correct answer as

$$P = \beta \cdot \frac{a(w_{\text{s}})}{\sum_i a(w_i)} \tag{4.3}$$

with $\beta$ being another free parameter. Note, that we are not calculating the probability of each individual word being given as answer, but the probability of producing the correct versus the wrong response. Because of that, $\beta$ is not fixed to one, but should be chosen such that $P \leq 1$. We did curve fits to the data from Bowden and Jung-Beeman (2003a) by minimizing the root mean square error between the proportion of participants solving the problem within the time limit and our predicted solving probability. For the curve

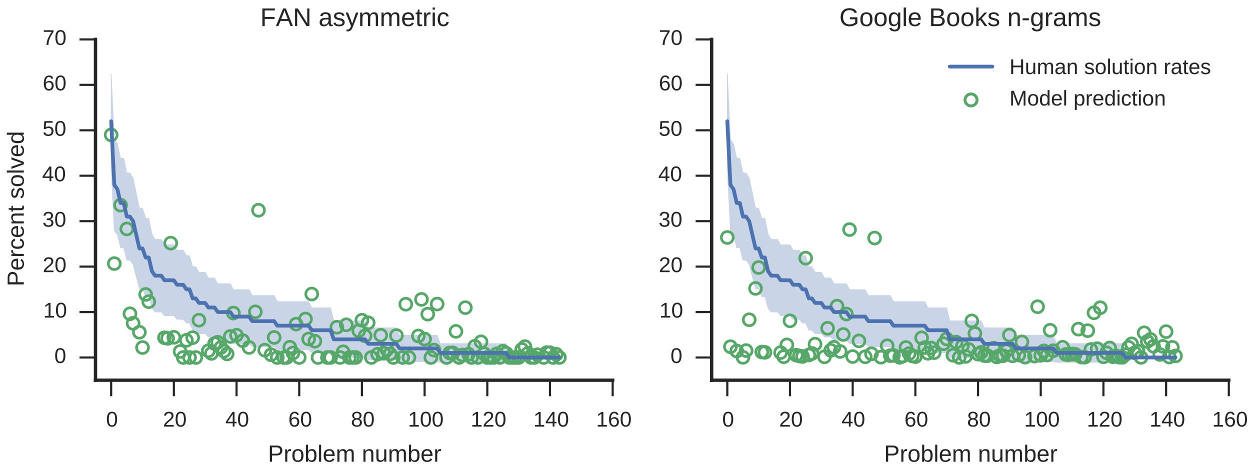


Figure 4.2: Model fits to human data (blue line with shaded $95\,\%$ confidence interval). Human data are percentages of participants solving a RAT problem in $2\,\mathrm{s}$ and they have been sorted in descending order, so that problems solved by more participants have lower problem indices. Every green circle represents the probability of producing the correct solution to a RAT problem predicted by the model. Reproduced from Kajić et al. (2016, Figure 2).

fits we used the association strengths from the original $FAN_{\text{asym}}$, $FAN_{\text{sym}}$, and $NG_{\text{sym}}$ matrices. In addition, we used the 768-dimensional $NG^{768}_{\text{sym}}$ matrix, which showed good performance in Figure 4.1.

The fitted parameter values are given in Table 4.2. Representations derived from free norms yield a better fit on this data set ($r^2 = 0.58$) compared to the n-gram data ($r^2 = 0.30$). There was no difference between the asymmetric and symmetric FAN matrices. Interestingly, the second cue is consistently assigned a higher weight. We speculate that this is caused by this cue appearing in the center of the screen with the other cues above and below. For n-gram fits, the parameters $\alpha_2$ and $\beta$ are large in comparison to those of free norms, but because of the low $r^2$-value, these values cannot be seen as meaningful. For visual inspection, we plotted the model fits using free association norms and Google n-grams in Figure 4.2. Further error analysis revealed that the Google n-grams underestimate the solution probabilities of easy items (more than 32% solved by humans) while at the same time predicting a non-zero probability for items unsolved by humans. While none of the models provides a perfect match with the human data, judging by the relatively high number of data points outside of confidence intervals in each plot in the figure, we note that the FAN dataset provides a better match with the easier problems. We attribute this to the fact that easier RAT problems are characterized by stronger associative strengths between the cue words and the target, which are well captured with the FAN. All solutions in the set of RAT problems used here are based on compound words, which, on the other hand, might explain why the n-gram dataset shows a better match with the harder RAT problems.

Taken together, our results indicate that the insight process of RAT solving is likely more reliant on association data as produced in the unconstrained free association task, rather than the co-occurrence information. These kinds of associations are likely to be based on additional semantic information that is either not available to, or is less present in, purely statistical approaches.

### 4.2.2 Discussion

We performed a computational analysis of two different sources of word association data to describe how well they predict the human performance on the RAT. While statistical language data like n-grams achieve the highest solution rates on this task, consistent with some of the previous work (Klein and Badia 2015; Toivonen et al. 2013), a more accurate prediction of the human performance on the RAT is obtained with the free association norms.

First, we discovered structural differences between n-grams and association norms by applying dimension reduction with SVD. Previous studies have shown that the dimensionality reduction on association norms can be used to accurately predict human responses on certain episodic memory tasks, such as recognition memory and cued word recall (Steyvers et al. 2004b). Interestingly, our analysis provides evidence that dimensionality reduction does not improve performance on the RAT. Moreover, it impairs the performance when the solution word is among the ten most similar words to the cue words. This indicates that for some RAT problems the important associations are contained in links that are not present in a low-dimensional representation. This is reminiscent of the finding that direct association strengths are the best predictor of intrusion rates in free recall (Steyvers et al. 2004b). The dimensionality reduction on n-grams revealed a considerable amount of redundant information: the original 5,018 dimensional word vectors can be reduced to at least 768 dimensions without large differences in the results. Moreover, SVD can even lead to improvement when looking at targets that appear within the three most similar words to the cues. Second, the modelling analysis showed that n-gram data, yielding the best scores on the RAT, is a worse predictor of human performance. The FAN data model was a better fit to human solution probabilities in the RAT. As expected, the model was not able to solve any problems for which there was no association between the cues and the target in the association norms, indicating that free norms might not be the only source of associative information. However, for the other problems we have demonstrated that free associations play an important role in the insight process. While both free associations norms and n-grams contain associations relevant for solving of RAT problems, the FAN dataset showed a closer match to the human solution accuracies. This was the case despite it being derived from a much smaller and more limited vocabulary. For example, approximately 6.5 million association pairs exist in n-grams but not in the FAN. Conversely, only about 26,000 associations exist in the FAN that do not exist in n-grams. In some aspects, the better fit with the FAN might not be too surprising given that the semantic search process underlying free associations likely shares some of the mechanisms with the process of solving for remote associations. Nonetheless, it indicates that the sheer size of the underlying resources used to generate a dataset might not be the primary concern when modelling aspects of human performance.

In the next section, we take a first step towards a cognitively plausible model of RAT solving. To this end, and consistent with this evaluation, we present a neural network model based on free associations that simulates aspects of cognitive processes involved in the semantic search process.

## 4.3 A Cognitive Model

Here, we present a neural network model that realizes the semantic search process in the RAT. The model resembles "classic" connectionist architectures for simulation of cognitive processes related to language function that gained popularity in 1980s (McClelland et al. 1987), and that remain in use in cognitive modelling due to their simplicity and effectiveness. This model implements two cognitively plausible mechanisms involved in the search process: a competition process based on the winner-take-all (WTA) principle, and the spreading activation algorithm (Collins and Loftus 1975). A high-level overview of the algorithm implemented by the network is shown in Algorithm 1. Some aspects of the algorithm have been simplified for clarity, however, those simplifications are still consistent with the behaviour of the network presented later.

The algorithm maintains an array of word activations ($activations$), and selects the word $j$ with the highest activation at each time step as the current word to be explored. Initially, all activations are set to zero, except for the three cue words—their activations are externally driven by the input $I$. Following the theory of spreading activation, activations

**Algorithm 1** Semantic search for a solution in a RAT problem based on spreading activation and the WTA mechanism. Adapted from Kajić and Wennekers (2015).

```
Require: cues, target, $\vartheta_s$, $max\_nodes$,
Ensure: visited
$visited = [\,]$
$activations = [\,]$

for $i \leftarrow 1, N$ do
    $activations_i \leftarrow 0$
    if $i$ in cues then
        $activations_i \leftarrow 1$

$j \leftarrow -1$
while len($visited$) $< max\_nodes + 1$ do
    for all neighbours $i$ of the current node $j$ do
        if $w_{ij} > \vartheta_s$ and $j \neq -1$ then
            $activations_i \leftarrow activations_i + w_{ij} \cdot activations_j$
    $visited.append(j)$
    if $j\ == \ target$ then
        break
    $j \leftarrow WTA(activations, visited)$
```

of all words that are associated with the word $j$ are elevated by the value dependent on the connection strength with the word $j$ as well as the level of activation of the word $j$. Such connection strengths are derived from the free association norms (Nelson et al. 2004), found to yield human-like performance on the RAT in the previous section. The search process terminates if the currently explored word is the target (i.e., the solution to the RAT problem), or, if the number of visited nodes exceeds $max\_nodes$. Otherwise, it proceeds by picking the next word with the highest activation value as the current word. The WTA function in the algorithm sorts the words based on their activations, and returns the word with the highest activity level that has not been visited before. In this way, the words that have been explored (i.e., the previous winners) will not be selected again as potential targets. This approach makes certain assumptions that simplify the search process. For example, it does not determine whether the current winner is a solution to the RAT problem. Instead, it models the process of ideation of possible words, assuming that when an individual thinks of a word, they will be able to determine if it is a correct

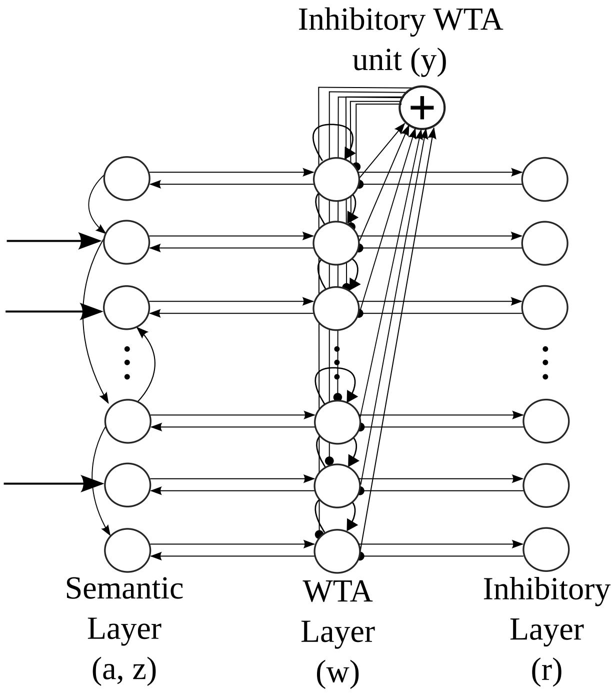


Figure 4.3: Neural network model of the semantic search in the RAT. The word with the highest activity level in the semantic layer is selected by the WTA layer as a response to a RAT problem. If the selected word does not match the solution, the inhibitory layer suppresses the activity of the selected word allowing the WTA to select the next winner. Adapted from Kajić and Wennekers (2015, Figure 1).

solution by testing it against cue words using cognitive mechanisms not modelled here. Also, this search does not account for cue switching observed in people, who are known to attend to a primary word cue (Smith et al. 2013).

To explore the influence of different word pair association strengths on the RAT performance, the spreading of activity is implemented only to the nodes adjacent to the node $j$ whose connection strength to it is greater than the spreading threshold $\vartheta_s$. Weak connection strengths are a characteristic of rare and uncommon word associations, which, according to the theory of Mednick (1962), are more likely to be generated by highly creative individuals. Increasing that threshold corresponds to the removal of such association pairs, resulting in a search path that either has more edges between the two nodes, or does not have them at all.

Next, we describe the neural network implementing Algorithm 1 to simulate the semantic search process over time. The network consists of three layers with distinct functions: the semantic layer, winner-take-all (WTA) layer and inhibitory layer. The model is shown in Figure 4.3. The semantic layer implements a semantic network, where each node in the network represents a single word, and connections among words represent their associative strength. The activation of units in this layer is akin to the array $activations$ in the algorithm above. The WTA layer selects a winning unit, simulating a solution guess to a RAT problem. The spread of activity from the winning unit proceeds via feedback connections from the WTA layer to the semantic layer. If the selected word is not a solution to the RAT problem, the inhibitory layer suppresses the activity of the winning unit in the WTA layer. This allows the next unit with the highest activity level to win in the next step. The activity of units in all layers is updated in parallel in each time step.

The time-dependent activity of a single unit in the semantic layer is defined by

$$a_i(t+1) = a_i(t) + \rho_a\left(\sum_{j=1}^{N} \omega_{ij} z_j(t) a_j(t) + I_i(t)\right) \tag{4.4}$$

$$z_i(t) = \Theta(w_i(t) - \vartheta_w) \tag{4.5}$$

$$\left\|\boldsymbol{z}(t)\right\|_1 = 1, \tag{4.6}$$

where $a_i(t)$ describes a non-decaying activity of a unit $i$ at time $t$ with $i \in \{1, \ldots, N\}$. $\omega_{ij} \in \boldsymbol{W}$ is the connection strength from the unit $j$ to the unit $i$, as derived from free associations (Nelson et al. 2004). The constant $\rho_a$ controls activity levels, and is inversely proportional to the stimulus length $t_n$. The binary vector $\boldsymbol{z}(t)$ always has exactly one or zero active elements, with the active element representing the currently visited node (variable $j$ in Algorithm 1) in the semantic layer. The winner is determined based on $\boldsymbol{w}(t)$, the activation values of the units in the WTA layer, and it is a word considered as a solution to the RAT problem. Consistent with the algorithm, only the activity of nodes whose edges incident to the winning node are greater than the spreading threshold $\vartheta_s$, or the activity of nodes receiving external input, will be elevated. Weights smaller than the threshold $\vartheta_s$ in the connection matrix $\boldsymbol{W}$ are set to zero. If the current node only has connections to other nodes that are less than the threshold, the activations in the network for that node will not change. Instead, nodes with activations higher than zero continue to compete in the WTA process. If a unit in the WTA layer crosses the threshold $\vartheta_w$, the Heaviside step function will toggle the corresponding bit in $\boldsymbol{z}(t)$. Only at the beginning of a simulation, the external input $\boldsymbol{I}(t)$ is used to sequentially increase the activity of cue nodes in the semantic layer.This in turn will elevate the activity of the units in the WTA layer, yielding a winner whose feedback connection will help spread the activity from the corresponding unit in the semantic layer.

The first three winners are thus going to be the three problem cues. $I_i(t)$ is clamped to $1$ for the duration $t_n$, separately for each cue word $i \in \{cue_1, cue_2, cue_3\}$, in such a way that at each time point only one cue is active. In Algorithm 1 this has been simplified by directly setting the activity of cue nodes to one, and then randomly picking the order of cues in the WTA. Those cue nodes are guaranteed to be picked first, as they receive the highest input among all nodes. In the network, this process occurs sequentially with the predetermined order as described above, but the resulting level of node activities does not depend on the cue order of presentation, thus yielding the same behaviour as the algorithm. The activity of the units in the WTA layer is determined by the following two equations

$$\begin{aligned} w_i(t+1) &= w_i(t) + \rho_w \big[ c_1 z_i(t) + c_2 \widetilde{a}_i(t) - c_3 y(t) - c_4 r_i(t) + c_5 \eta_i(t) \big] && (4.7) \\ y(t+1) &= \big\| \boldsymbol{z}(t) \big\|_1 . && (4.8) \end{aligned}$$

Units in this layer receive one-to-one feedforward input $\widetilde{a}_i$ from the units in the semantic layer. The first WTA unit to cross the threshold $\vartheta_w$ will be the one receiving the strongest normalized input $\widetilde{a}_i(t) = \frac{a_i(t)}{a_k(t)}$, where $k = \arg\max_j a_j(t)$.[2] $y(t)$ is a single inhibitory unit that projects back to all units in the WTA layer, shown as the unit with the **+** sign in Figure 4.3. In this way, the activity of all the units, except that of a single winning one, will be suppressed and kept below the threshold $\vartheta_w$. The winning neuron toggles the corresponding bit in the $\boldsymbol{z}(t)$ as per Equation 4.5 and elevates the activities of its neighbours in the semantic layer according to Equation 4.4. Due to the self-excitation, the activity of a winning unit in the WTA layer continues to increase until it becomes suppressed by a unit in the third layer. The noise term $\eta_i(t)$ is randomly sampled from a uniform distribution in the interval $[-0.5, 0.5]$. It is added to allow the WTA to select a winning unit when two or more units receive the same input $\widetilde{a}(t)$ from the semantic layer.

Finally, after a unit $z_i(t)$ has been active for a certain amount of time $t_n$, it will activate the corresponding inhibitory unit in the third layer according to

$$r_i(t+1) = r_i(t) + \Theta\Bigg( \sum_{j=t-t_n}^{t} z_i(t_j) - t_n \Bigg). \tag{4.9}$$

The feedback connections from this layer to the WTA layer will inhibit the activity of the winning unit. As input integration in $\boldsymbol{r}(t)$ is non-decaying, the inhibition will be

[2] The normalization helps to bound parameters that stabilize the WTA competition mechanism.

Table 4.3: Hyperparameter values for the neural network RAT model.

| Name | Values | Description |
|---|---|---|
| $\rho_a$ | 0.02 | Integration constant for the semantic layer |
| $\vartheta_w$ | 1 | Threshold for the units in the WTA layer |
| $\rho_w$ | 0.995 | Integration constant for the WTA layer |
| $t_n$ | 50 | Stimulus and WTA unit activity duration |
| $c_1$ | 1 | Excitatory connection |
| $c_2$ | 1 | Normalization constant |
| $c_3$ | 1 | Inhibitory unit constant |
| $c_4$ | 50 | Inhibitory layer constant |
| $c_5$ | 0.1 | Noise amplitude |

permanent, preventing the winning unit from winning again. This permanent inhibition is comparable to the aspect of WTA's behaviour in Algorithm 1, where WTA excludes previously visited nodes from winning again. A new unit will be selected in the semantic layer based on the highest level of activity $\widetilde{a}_i(t)$. Parameter values used in the model simulations are listed in Table 4.3. The values in this table are either selected, or analytically derived, to bound the values of activation functions and stabilize the WTA mechanism.

Figure 4.4 shows unit activities in the network over a course of time while solving the RAT problem: *river, note, account*. First three winners in the network are the cues *river, note* and *account*, followed by the first response, *money*. That response is inhibited and the next response, *bank* is the solution to the RAT problem. For visual clarity, only a small fraction of activated units in the semantic layer is shown.

### 4.3.1 Simulation Results

The model performance is evaluated on the same 117 RAT problems as in the previous section. The problems are divided into the three difficulty categories (easy, medium and hard) based on the percentage of participants solving an item in the 15 seconds condition (Bowden and Jung-Beeman 2003b). The percentage of participants solving a problem varies between 0% (e.g., no one solved the problem) and 96% (e.g., almost everyone did). The categories are divided in three parts according to 32 percentage point increments: easy problems are solved by 64% to 96% participants (17 problems), medium problems by 32% to 64% (43 problems) and hard problems by 0% to 32% participants (58 problems).

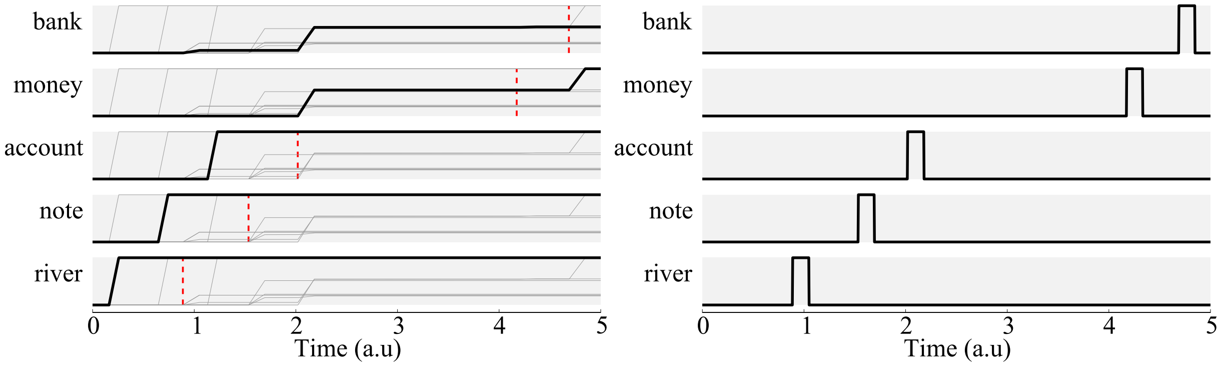


Figure 4.4: A model simulation of the search process for the RAT problem *river, note, account*. **Left:** Normalized unit activities in the semantic layer. Red dashed lines represent a moment when a winning unit has been selected, and determines the onset of spreading activity to its neighbouring units. **Right:** Activities of the winning units. The problem solution *bank* is found as a second response, after *money*. The first three winners are the given problem cues. Reproduced from Kajić and Wennekers (2015, Figure 2).

The performance on all three categories is tested by varying two model parameters: the number of responses, and the spreading threshold $\vartheta_s$ in the semantic layer. The number of responses is the number of words selected by the WTA. If the correct response was not among the predetermined number of responses, the problem is annotated as unsolved. As a first approximation, the number of responses can be related to the time allowed to produce the answer. The basic assumption is that with more time a participant will be able to think of more words. By increasing the threshold $\vartheta_s$ we are discarding all word pair associations with the association strength lower than that threshold. Lower association strengths correspond to word association pairs produced by fewer individuals in the free association task (Nelson et al. 2004), and this parameter can be thought of as controlling the ability to think of such rare associations.

Network simulation results for 117 problems and the three difficulty categories are shown in Figure 4.5. As expected, the performance on the RAT increases with the number of responses. Compared to the medium and hard items, all easy items are solved within ten or fewer responses. Approximately 20% of all easy problems are solved after a single response, implying that the solution to easy RAT items is the strongest association between one of the problem cues and the solution. For the medium and hard problems, there is a continuous increase in performance with the increased number of response

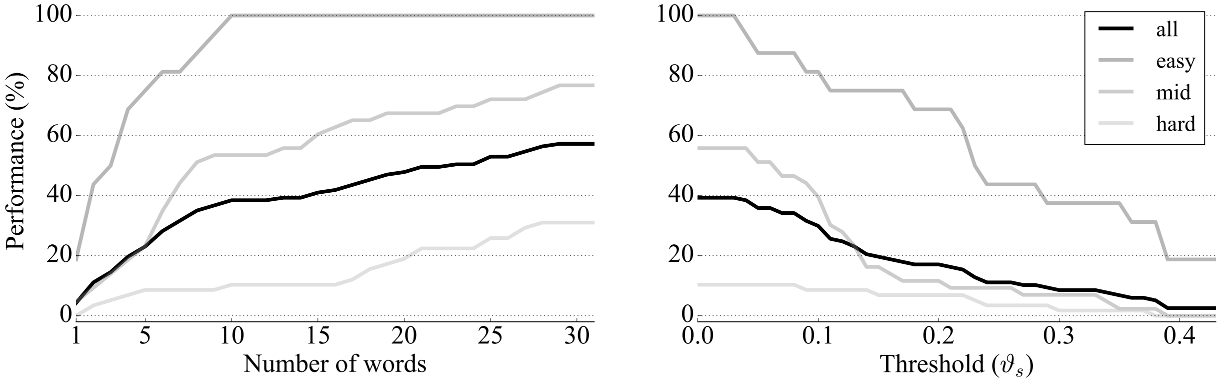


Figure 4.5: **Left**: Performance on the easy, medium and difficult RAT items depending on the number of responses in the search process. **Right**: Performance depending on the spreading threshold. Increasing the threshold removes word pair associations with association strengths weaker than the threshold. Reproduced from Kajić and Wennekers (2015, Figure 3).

words. For the purposes of current analysis, we have restricted the range of responses to an interval could be interpreted in the context of known data. Participants instructed to report every word they consider a solution when trying to solve a RAT problem on average produce eight words within two minutes (Smith et al. 2013), although there are large variations in the number of responses. As we are not explicitly modelling this process, we take this number as a reference, and assume a greater number in the model since it does not differentiate between reported and unreported words. The average percentage of solved items for 117 problems given a 15 second time limit for humans is 29% (Bowden and Jung-Beeman 2003a), and the model achieves comparable performance of 28% when given a word limit of six responses.

The right plot in Figure 4.5 shows how the removal of associations in the semantic layer affects the performance on RAT items of different difficulties. 98% of association strengths in free association norms (Nelson et al. 2004) are of value $0.4$ or less, which we use as the upper bound for the experimental range for $\vartheta_s$. As expected, removing word associations negatively impacts the performance on the test for all problem difficulties, however, easy and difficult problems are differently affected. Relative to the performance on the RAT without removal of associations ($\vartheta_s = 0$), the drop in performance by 50 percentage points occurs at lower threshold values for problems of medium difficulty compared to easy problems. For easy items, a 50 percentage point decrease in performance occurs when all word pairs of association strength $\vartheta_s = 0.23$ (94th percentile) or less are removed, while for the items of medium difficulty this already occurs for the threshold value of $\vartheta_s = 0.12$ (87th percentile). This effect is also observed for smaller drops in performance, when comparing the performance on the RAT problems of medium and hard difficulty with easy RAT problems, and when using a different number of words (here only shown for 15 words). This observation indicates that infrequent word associations are more important for difficult RAT problems, as consistent with the theory of associative hierarchies by (Mednick 1962).

### 4.3.2 Discussion

The neural network model of the semantic search process in the RAT, as presented in this section, proposes a basic set of underlying cognitively plausible mechanisms. The model is consistent with the theory of human semantic processing that postulates a process of spreading activation in semantic memory (Collins and Loftus 1975). The semantic layer in the model implements a localist representation of lexical knowledge as a semantic network as well as the spreading of activity among units in the layer. In the context of associative hierarchies used to characterize creative thinking (Mednick 1962) it is argued that such processes occur at the subconscious level of semantic processing (Yaniv and Meyer 1987; Collins and Loftus 1975). The WTA mechanism and the inhibitory layer responsible for selecting a single word in the vocabulary are reminiscent of attentional mechanisms mediating cognitive control attributed to the function of the frontal brain areas (Fink et al. 2007; Kounious and Beeman 2014). The focus of attention is directed towards a single word which is selected among several competing alternatives. Anterior cingulate cortex (ACC) has been shown to play an important role in attentional switches and conflict resolution in case of competing alternatives (Botvinick et al. 2004; Kerns et al. 2004). We validated the model by analyzing its performance in the context of RAT problems of varying difficulty and have shown that it is able to distinguish between easy and difficult RAT problems in the normative RAT data set by Bowden and Jung-Beeman (2003b).

While the proposed model is based on theories of semantic processing and cognitively plausible mechanisms, aspects of its architecture are unlikely to be plausible in biological terms. For example, every word in the model is represented with a single unit, which is inconsistent with the theory of distributed representation in the brain. Even if such a unit is interpreted as a population of neurons representing a concept, it is unlikely that individual units in such populations contribute exclusively to the representation of a single concept. Thus, the current model most closely resembles ideas within the localist representational framework (Bowers 2009). In such a framework, it is argued that a single neuron or a small group of neurons represents word meaning. This approach is often considered problematic in that it implies the existence of so-called "grandmother cells" (Gross 2002). The underlying idea captured by this type of cells is that there are neurons dedicated to specifically and uniquely representing a single concept (such as the concept of a "grandmother", for example). Some support for this type of representation can be seen in studies recording from single-cells which show high degrees of specificity in their response to external stimuli (Hubel and Wiesel 1968; Moser et al. 2008; Quiroga 2012). While such studies show that there are neurons exhibiting specificity, it has been difficult to establish that they do so uniquely. In addition, and as discussed later in greater detail, response specificity has also been observed with distributed representations, thus overall providing little support for the theory of a localist representation.

Another requirement in the model is the mapping of units in the semantic layer to corresponding units in the subsequent WTA and inhibitory layers. Instead of such dedicated "winning" and "inhibitory" units for each word in the semantic layer, it is possible that an alternative mechanism exists that relaxes the requirements on such a one-to-one mapping. Compared to the current architecture, it is likely that such alternative mechanisms would have better scaling properties. Finally, the activity of units in this network is modelled with a non-decaying function, which is different from the activity of biological neurons. Biological neurons instead have dynamic membranes that actively respond to stimulation by emitting spikes. In the next section, we address some of these issues and explore a biologically plausible extension of this model by implementing a model using the Neural Engineering Framework (Eliasmith and Anderson 2003).

## 4.4 A Biologically Constrained Model

In Section 2.2.2 we discussed various neuroimaging studies that provide detailed characterizations of brain regions involved in aspects of semantic processing, such as the organization, retrieval and manipulation of semantic knowledge. A comprehensive meta-analysis of such studies in Binder et al. (2009) further corroborates the view that semantic processing is distributed across cortical regions. It also points out that such regions are expanded in the human brain relative to the nonhuman primate brain, thus emphasizing the distributed nature of language function in the brain. However, although such studies provide a network-level perspective on the organization and the function of semantic areas, they offer limited insights into how words and word associations are represented by either individual neurons, or small groups of neurons. More specifically, it is unclear what computational mechanisms operate at the level of individual neurons that contribute to the representation of words and associations. Understanding of such mechanisms operating at lower levels is a daunting task due to the difficulty of locating, accessing, and recording from relevant brain regions. More specifically, direct single-unit recordings are technically complex, invasive and are only performed in controlled clinical settings, typically with patients undergoing treatment for severe forms of epilepsy.

Given such challenges associated with the experimental approach to the study of semantic processing, we argue that, nevertheless, neurocomputational models can provide useful insights for understanding the underlying neural mechanisms. In particular, biologically detailed neural models provide a way to investigate plausible word representations in the brain. In the previous section, we discussed a model of the semantic search process in the RAT that uses individual units to represent words and corresponding word selection mechanisms. In contrast to the localist representation used in that model, distributed representations in neural networks assume that a concept is represented by a population of neurons (McClelland et al. 1987; Rogers and McClelland 2004), where each individual neuron participates in the representation of multiple concepts. It has been argued that the kind of data used to support localist representations is often exhibited by distributed models (Stewart and Eliasmith 2011; Eliasmith 2013, p. 98–99, 369–370), demonstrating the relationship between the two. Importantly, and as described later in this section, the method for the distributed representation of concepts used to build a biologically realistic RAT model here suggests that each neuron within a distributed representation has a preferred state. This means that some neurons might be highly specific, while others will have broad responses in the biologically informed distributed representation used here (Stewart et al. 2011b). Despite arguments and evidence that distributed representations are used in many parts of the brain, there is no agreed upon approach to characterizing the representation of cognitive or linguistic structures using such representations. In particular, it is an open question as to how such representations support word associations and how they might be employed in tasks requiring associative processing.

To bolster the plausibility of the proposed neural network, we use the leaky integrate-and-fire (LIF) neuron model that exhibits behaviours more similar to those of biological neurons. We select this neuron model due to its favorable trade-off between computational efficiency, analytical tractability, and its ability to capture some of the basic features of neuronal dynamics observed in biological systems (Izhikevich 2007). In particular, synaptic dynamics and noise from fluctuations introduced by spiking impose constraints that a theoretical approach used to simulate neural systems needs to account for. The LIF neuron model is a spiking neuron model as it imitates the spiking behaviour observed in biological neurons. It is important to note that there is no explicit “translation” of computations from localist units used in the previous model to to the LIF units here. While for some functions this can be done more easily (e.g., representing a value in a population of neurons), more complex mechanisms such as the WTA require a different approach due to noise introduced by neurons' spiking activity.

In biological neurons, electrically charged ions are exchanged across the cell membrane and an influx of positive ions into the cell can cause the neuron to trigger an action potential (also known as a spike). A spike can be registered by another, receiving neuron, if it has a synaptic connection with the neuron emitting a spike. In our modelling approach, spiking neurons are also connected by synapses so that the arrival of a spike at the side of a receiving neuron causes the post-synaptic current. In our model, the relevant neuron and synapse model parameters such as the membrane and synaptic time constants, and the shape of the post-synaptic currents conform to empirically measured value ranges and properties. These constraints are ensuring that the modeled system approximates the biological system and provides an account of the internal mechanisms underlying the investigated behaviour.

The model presented in this section aims to provide neural account of the semantic search process in the RAT while reproducing aspects of linguistic behaviours observed with human subjects solving the task. To construct the model, we use the Neural Engineering Framework (NEF; Eliasmith and Anderson 2003) discussed in Section 2.3.6. The NEF allows us to derive the required neural network to implement the necessary representations and transformations for performing the semantic search process in the RAT. We describe the specific model in Section 4.4.2 and evaluation methods in Section 4.4.3. These are followed with a discussion of quantitative and qualitative results.

### 4.4.1 Representing Words and Associations with the NEF

To model the word search process in the RAT, words and associations among them need to be represented in connections or neuron activations. We adopt a representation where the activity of several neurons contributes to a representation of multiple words. In the NEF and the SPA, this is achieved by using vectors, or *semantic pointers* discussed in Section 2.3.6, to represent words. With the random distribution of preferred direction vectors $\boldsymbol{e}_i$, each neuron will be involved in the representation of multiple words and the representation is distributed across the neurons.

Specifically, we generate random unit-vectors with the constraint that no pair of such vectors exceeds a similarity of 0.1 as measured by the dot product. To fulfill the similarity constraint, a sufficient vector dimensionality has to be used. For the $N = 5{,}018$ words used in the model, we set the dimensionality to $D = 2{,}048$ with 50 neurons per dimension. The dimensionality was determined empirically by testing different values and selecting the one that showed the most accurate and stable performance. This is still considerably below the number of words, because the number of almost orthogonal vectors that can be fit into a vector space grows exponentially with the number of dimensions (Wyner 1967).

Such vector-based word representations have been successfully used to implement a variety of cognitive tasks such as the Tower of Hanoi task (Stewart and Eliasmith 2011), inferential word categorization (Blouw et al. 2015) and Raven's Advanced Progressive Matrices (Rasmussen and Eliasmith 2014). These representations have been shown in simulation to be robust to neural damage (Stewart et al. 2011b), and are consistent with the type of distributed representation found throughout sensory and motor cortex (Georgopoulos et al. 1986).

We next turn to the methods used to compute the connection matrix between groups of neurons representing associations. These methods refer to algebraic operations and we do not consider them to be a part of the model. Instead, we use the methods to derive the matrix $\tilde{\boldsymbol{A}}$, which is implemented in connection weights among groups of neurons. The matrix $\tilde{\boldsymbol{A}}$ is used to describe associations between words, and it transforms a word vector $\boldsymbol{w}$ to a linear combination of its associates. This matrix can be derived from an association matrix $\boldsymbol{A}$ where $A_{ij}$ gives the associative strength from word $i$ to word $j$. To do so, we need to define the $N \times D$ matrix $\boldsymbol{V}$ that collects all the word vectors, i.e., row $i$ of $\boldsymbol{V}$ is the vector representing word $i$. Then, we can state $\tilde{\boldsymbol{A}} = \boldsymbol{V}^\top \boldsymbol{A}^\top \boldsymbol{V}$. Applied to a vector $\boldsymbol{w}$, this will first correlate $\boldsymbol{w}$ with all the word vectors ($\boldsymbol{V}\boldsymbol{w}$) to yield an $N$-dimensional vector indicating the similarity with each word; then $\boldsymbol{A}^\top$ is used to retrieve the corresponding associations before $\boldsymbol{V}^\top$ projects those associations back into a $D$-dimensional vector. As all of this collapses into a single linear transformation matrix $\tilde{\boldsymbol{A}}$, the retrieval of associations can be easily implemented with the NEF in the connection weights between two groups of neurons, computing the function $\boldsymbol{y} = \tilde{\boldsymbol{A}}\boldsymbol{x}$.

To set the values in the transformation matrix $\tilde{\boldsymbol{A}}$ on the connections between the cue selection network and the response network (further elaborated below) we collect free association norms (FAN) in the matrix $\boldsymbol{A}$. The FAN dataset has been identified as a reliable source of semantic information on the RAT, as shown and discussed in previous sections. Our preliminary analyses indicated that a binary version of that matrix was appropriate for the good performance on the task, so we set all non-zero values in the matrix to one. In other words, for this version of the RAT task, just capturing the existence of the association between any two words appears to be sufficient. To model individual differences in associative networks and adjust solution probabilities to match the human data, we randomly remove between 60% and 80% of associations in the matrix by setting them to zero. This range has been determined empirically. In Section 4.3 we studied the effect of adjusting the threshold on task performance. Here, that approach is not suited to study individual differences given that there are only two possible values in the association matrix. Instead, by randomly removing connections we are modelling individual differences as differences in the set of word associations in individual lexicons.

Our model assumes that the connection weights $\tilde{\boldsymbol{A}}$ are given, and here we do not aim to explain the underlying developmental process that yields the data captured by $\boldsymbol{A}$. While such a developmental process is an important aspect of language acquisition, and ontogeny in general, it is out of scope of this model. However, we re-visit this question in Chapter 6, where we propose a mechanism by which learning of associative spatial relationships occurs through cooperative behaviours in emergent communication.

### 4.4.2 Model Description

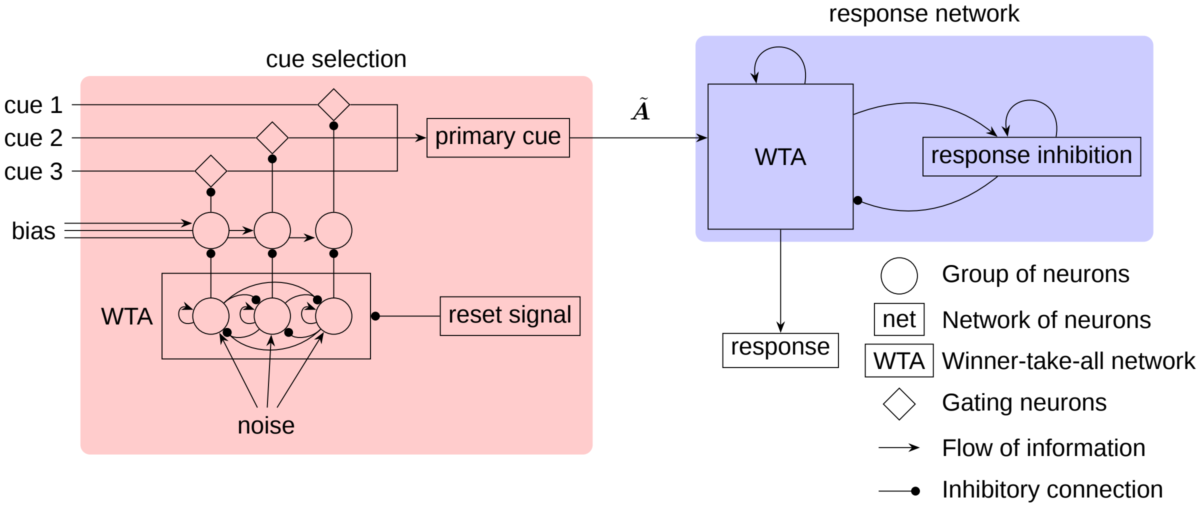


Figure 4.6: The architecture of the RAT model. All neural groups, gating neurons, and networks consist of spiking neurons. The cues, noise, and bias are provided as external input to the model. The $\tilde{\boldsymbol{A}}$ label indicates the transformation, implemented on the connection to produce the associates of the primary cue. Reproduced from Kajić et al. (2017b, Figure 2).

The model is implemented using the Nengo neural network simulator (Bekolay et al. 2014) and it is depicted in Figure 4.6. Two major model components simulating different cognitive processes involved in solving of RAT problems are the *cue selection network* and the *response network*. The *cue selection network* randomly selects one of the three input cues as the primary cue, and repeats this process at certain time intervals to realize *cue switching* typical of the search process in the RAT (Smith et al. 2013; Davelaar 2015). The *response network* realizes the response search process by selecting a word associated with the primary cue and previous responses.

We first describe the cue selection network. While all three cue words are provided as input to the model, only one of them at a time will be regarded as the *primary cue* to generate associations. This is consistent with the way humans generate responses, where they are found to primarily attend to one cue at a time (Smith et al. 2013). In the model, each cue vector is fed through a group of gating neurons, determining whether the input cue vector will be forwarded and represented as the primary cue at the output of the network. By default, all gating neurons are inhibited so none of them are forwarding their corresponding cues. They are inhibited by a group of neurons biased to represent a constant value of one (illustrated with "bias" in Fig. 4.6). Thus, in order to let one out of three cues appear as the primary cue, the inhibition has to be turned off for one of the gating groups. This will be done by a white noise-driven WTA mechanism that randomly inhibits one bias group, that in turn stops the projected inhibition to its corresponding gating group. With one out of three gates open, one cue will be regarded as the primary cue. This process will repeat after a certain time limit defined in the *reset signal* group.

In the response network a single associated word is selected by a clean-up associative memory (Stewart et al. 2011a) with an added winner-take-all (WTA) mechanism. In the associative memory, the input vector is correlated with the individual word vectors, and each correlation value is represented by a group of neurons. In this way, the preferred direction vectors are not randomly distributed as in other parts of the model, but the preferred direction of every neuron and every group of neurons corresponds to one of the words. Furthermore, these groups of neurons threshold the represented value at 0.1. Each group is connected with every other group with lateral inhibitory connections, and to itself with a self-excitatory connection. This allows the group with the strongest input to remain active, while inhibiting all other groups. Another feedback connection is used to capture the evidence on the locality of search (Smith et al. 2013). This connection implements a transformation $\tilde{\boldsymbol{A}}$, so that the associates of the current response are fed as additional input to the WTA network. In Figure 4.6, all of these recurrent connections are denoted by a single feedback connection on the WTA in the response network.

The response inhibition plays a crucial role in allowing the next word to appear in the search process. It is realized as a neural group acting as a leaky integrator. Without external input, the represented vector will slowly decay to the zero vector. A recurrent

Table 4.4: Free parameters of the RAT model. Reproduced from Kajić et al. (2017b, Table 1).

| Parameter | Value | Description |
|---|---|---|
| $d$ | 2,048 | Number of dimensions per word vector |
| $th$ | 0.6 to 0.8 | Percentage of randomly removed associations in the association matrix (varies with simulation) |
| assoc_th | 0.05 | WTA cut-off input threshold |
| cue_strength | 0.1 | Input strength of individual cues to WTA network |
| primary_cue_strength | 0.7 | Input strength of primary cue to WTA network |
| wta_feedback_strength | 0.5 | Input strength of associates of current response to WTA network |
| noise_std | 0.01 | Standard deviation of the zero-centered Gaussian noise in the cue selection network |
| integrator_feedback | 0.95 | Strength of recurrent connection on response inhibition |

connection feeding the output of the neural group back to itself prevents this otherwise very quick decay. External input to the integrator will slowly shift the represented value towards the input vector. This input is provided from the WTA network, while at the same time the response inhibition is used to inhibit the WTA network. Thus, the active word in the WTA network will be subject to increasing inhibition until finally a new word is selected. This switch will typically happen before the vector represented by the response inhibition shifted completely to the input vector. Because of that, the response inhibition will represent an additive mixture of the sequence of the last few words and prevents those from reappearing in the search process in short succession. The list of free model parameters and their values that produce the described model behaviour is provided in Table 4.4. The values have been determined manually by observing which ranges produce stable word-selection behaviour.

Not all potential responses produced by the response network qualify as a valid response to a RAT problem. Some words might be the result of an implicit priming effect, where a previous response primed a word that is not related to any of the cues. Also, it is reasonable to assume that participants in the experiment have typed only a subset of words that they thought of. To account for these effects, we implement a filtering procedure that regards only certain words as responses to a RAT problem. We examine the effect of four different semantic sources on the filtering procedure: FAN, Google

Ngrams, binary FAN (bFAN) and binary Ngram (bNgram). In Section 4.2 we showed that both FAN and Ngrams contain associative links relevant for different RAT problems, these datasets differ in important ways that may or may not influence the filtering process. The filtering is implemented as follows: for every generated word, a similarity measure to the problem cues is calculated and, if it is below a threshold, the word is dismissed. The similarity is the sum of association strengths between every cue and the word.

### 4.4.3 Model Evaluation

The model is evaluated using a set of 25 RAT problems from Smith et al. (2013) by comparing the model responses to the human responses from the same study. For each of the 25 problems, we ran 56 simulations with different random number seeds to ensure the independence of the results from the initial conditions, such as the choice of neurons and word vectors. For the analysis of responses, we adapt a set of analysis tools from Smith et al. (2013), which was originally developed to analyze human responses and characterize memory search in the RAT. Thus, the same analysis tools are used for human responses and model responses. While the experimental details about the data collection and detailed descriptions of analysis methods are available in the original study, we present a brief overview of the data and a description of the adapted methods.

The data set contains responses from 56 participants, which were given 2 minutes to solve each RAT problem. Every participant was given 25 problems and was instructed to type every word that came to their mind as they were solving the problem. Participants indicated when they thought they had provided the correct solution word with a key press. Thus, every trial consists of a sequence of responses from one participant to one RAT problem, ideally ending with the correct solution. Here, the analysis of responses has been performed over 1,396 human trials and 1,400 model trials. For each RAT problem, we thus ran 56 simulations, corresponding to the number of human participants. In 169 trials, human participants marked an incorrect response as correct, and we excluded those from qualitative analyses, as they could have skewed analyses comparing how participants approached the final answer on incorrect and correct trials.

For every trial we did a series of pre-processing steps, as per Smith et al. (2013). Word pairs with words not available in the Free Norms or words identical to one of the cues were excluded from the analysis. Responses repeated twice in a row were merged into a single response. Then, we assigned a $300$-dimensional word vector to every word, including problem cues, the solution, and human responses. Those vectors were based on the Word Association Space (WAS; Steyvers et al. 2004b), constructed by reducing the dimensionality of an association matrix. This matrix was the WAS $\boldsymbol{S}^{(2)}$ measure based on the Free Association Norms, which includes not only direct association strengths between two words $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$, but also links across one intermediary word, i.e., associations from $\boldsymbol{w}_i$ to $\boldsymbol{w}_k$ to $\boldsymbol{w}_j$. The similarity between words was measured as the cosine angle between the assigned word vectors. To conclude the pre-processing, every response was assigned the word vector with the highest similarity to the primary cue vector.

Metrics were calculated on the pre-processed data to evaluate the model. First, we determined the *average response similarity* for within and across cluster response pairs of adjacent responses. Clusters were defined on the primary cue of the responses; adjacent responses with the same primary cue are considered to be part of the same cluster. This was done to test for bunching of responses around cues by comparing the similarity between word pairs in each cluster. The assumption is validated with a *permutation test for average response similarity* by assigning cues from another trial and checking for conservation of similarity trends. The average response similarity within clusters is also computed in a cleaned data set, where all missing entries were dropped, which yielded new response pairs. Second, the *probability of switching primary cues* is computed as the number of response pairs with the different cues divided by the total number of response pairs. This value needs to be compared against a baseline probability based on the frequency each cue was selected under an independence assumption. This baseline calculation is required because certain cues might be selected more or less often than pure chance would predict. Third, the *similarity between adjacent and non-adjacent responses* within a cluster is computed to test for the direct influence of the previous response on the next one. The same is done for the responses with different primary cues, which occur right at the cluster breaks. Fourth, we tested whether the similarity to the final response increases as participants approach the final answer (either correct or incorrect).

### 4.4.4 Quantitative Results

The model solved on average 43% of the problems, showing a moderate correlation (Pearson correlation coefficient $r = 0.49, p < 0.05$) with humans who on average solved 42% problems. The left panel of Figure 4.7 shows the accuracy on the 25 problems averaged, respectively, over all model simulations and over all human subjects. By applying the two-sided exact binomial test we find that for 14 out of 25 problems there is a statistically significant difference ($p < 0.05$) between the human and model responses.[3]

[3] It should be noted that if the number of problems is increased sufficiently, then there will always be a statistically significant difference for all conditions. For this reason, we take this test as a means of

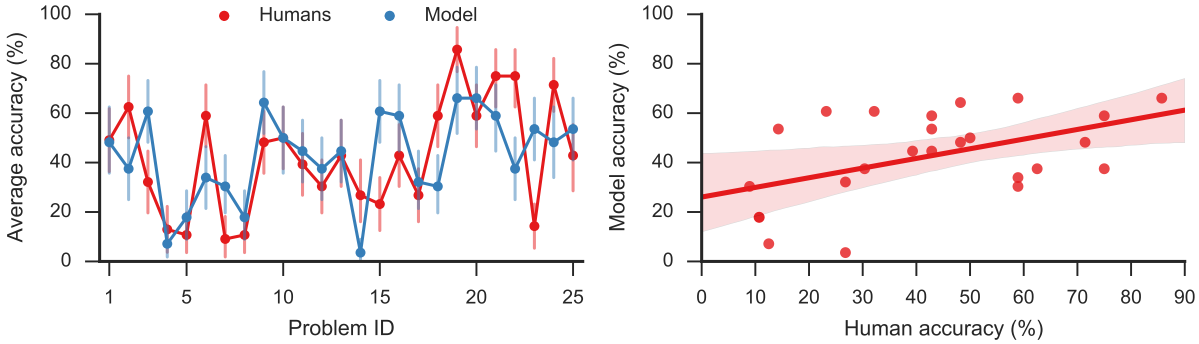


Figure 4.7: **Left**: Average accuracy on 25 RAT problems for model responses and human responses. Error bars denote 95% bootstrap confidence intervals. **Right**: Linear regression (Pearson correlation coefficient $r = 0.49, p < 0.05$) with 95% bootstrap confidence intervals. Reproduced from Kajić et al. (2017b, Figure 3).

These results are expected given that there are some problems that are easier for humans, and others that are easier for the model. On two problems—*dust, cereal, fish*; and *speak, money, street*—the model accuracy was more than 35 percentage points greater than the human accuracy on the same problems. On the other hand, there was one problem, *safety, cushion, point* where the human score was more than 35 percentage points higher than the model. However, based on the results in columns labelled "Humans" and "Raw" in Table 4.5, we see that while the accuracy of this model is comparable to that of humans, the model on average produces longer response sequences than the human response sequences ($40.20$ versus $7.78$).

To deal with this discrepancy, we consider that there is some filter applied between the output of the model and the actual reported responses, as discussed at the end of Section 4.4.2. In other words, we assume that the subjects do not actually write down all the words that come to mind while performing the task. This means that only a subset of all words produced by the model will be regarded as a set of responses to a RAT problem. In particular, a word that has a connection strength to all three cues below a threshold will be discarded. Thresholds have been determined as the lowest connection strength between the sets of three cues and solution for all problems. In this way, filtering will ensure that all solution words pass the filter. As a result, the accuracy and the correlation with the human accuracies are independent of the filtering method. Table 4.5 summarizes

identifying problems where model responses deviate the most from human responses, rather than as a measure of the quality of the model.

Table 4.5: Quantitative analysis of raw and filtered model responses. Association matrices used for the filtering are FAN: Free Association Norms, bFAN: binary FAN, Ngram, bNgram: binary Ngram. Values significant at $p < 0.05$ are marked with *, significant at $p < 0.001$ with ***. Reproduced from Kajić et al. (2017b, Table 2).

|  |  |  | Filtering method |  |  |  |
|---|---|---|---|---|---|---|
| Analysis | Humans | Raw | FAN | bFAN | Ngram | bNgram |
| Filtering threshold |  |  | 0.006 | 1 | 0.006 | 3 |
| Shortest response sequence | 1 | 2 | 1 | 1 | 1 | 1 |
| Longest response sequence | 49 | 46 | 39 | 40 | 27 | 33 |
| Mean response sequence length | 7.78 | 40.20 | 16.99 | 17.47 | 8.33 | 8.44 |
| – Correlation with human data ($r$) |  | -0.30* | 0.54*** | 0.51*** | 0.95*** | 0.93*** |

the statistics for the raw data and various filtering methods. We compared the average number of responses per trial, the shortest and longest response sequence, and the match between distributions of response sequence lengths.

Overall, the Ngram matrix and the binary Ngram matrix yield distributions that provide the best match to human data ($r = 0.95$ and $r = 0.93$, respectively). The threshold for the binary Ngram matrix has been set to 3, so that a word will pass the filter if it is an associate of all three problem cues. Reducing the threshold to two decreases the correlation with the distribution to $r = 0.36$ ($p < 0.05$) and increases the average number of responses per problem to $17.73$. Figure 4.8 displays the distributions for all filters plotted against the distribution of human responses. The Ngram-derived matrices are more aggressive in filtering the responses, compared to the FAN-derived matrices. The former preserve approximately 20% of words produced by the model, while the latter did so for approximately 40% of the responses. Although the Ngram matrix and the binary Ngram matrix yield comparably good matches with response distributions, in the following analyses we use the binary Ngram matrix as it provided a slightly better match for some of the qualitative analyses.

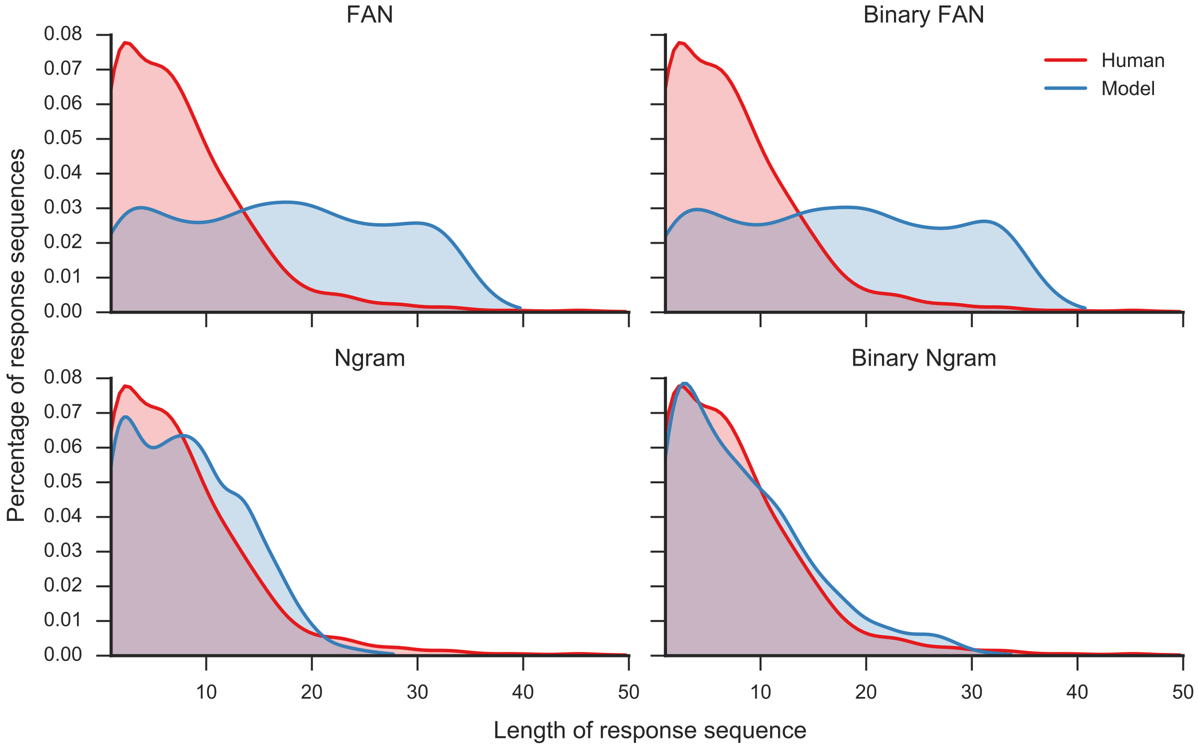


Figure 4.8: Distribution of number of responses per trial for human responses and model responses plotted as Kernel Density Estimates (Gaussian kernel, bandwidth determined with Scott's rule (Scott 1979)). Four different model distributions were produced with four different filters (see text for details): Free Association Norms (FAN), binary FAN, Ngram, and binary Ngram. Reproduced from Kajić et al. (2017b, Figure 4).

### 4.4.5 Qualitative Results

In the following analysis, we study semantic characteristics of filtered response patterns. We focus on the responses produced by the model that uses the bNgram matrix as the source of association data. The analysis compares the similarity between two groups of response pairs, where groups refer to primary cue assignment of response pairs (same cue versus different cue) and their proximity in a sequence of responses (adjacent versus non-adjacent word pairs). Such analyses performed with human responses (Smith et al. 2013; Davelaar 2015) showed that responses humans give tend to bunch around one of the three problem cues, and that different cues can be selected while the search for the solution unfolds. Also, responses show sequential dependence, where the next response is dependent on the previous one. We use the set of analysis methods described in Section 4.4.3 to explore whether model responses exhibit such similarity patterns.

Results are summarized in Table 4.6. To test for bunching of responses around problem cues, we explore the similarity of response pairs with a common primary cue. The similarity is greater for word pairs with the same cue compared to word pairs with different cues (0.141 vs. 0.054; two-sided t-test $t(9{,}915) = 20.4$). This trend is preserved when we use the permutation test, which randomly assigns cues from a different trial (0.142 vs. 0.054; $t(4{,}729) = 13.7$). Evidence for sequential dependence of word responses has been found by comparing similarities for word pairs within the same cluster; adjacent word pairs within the same cluster are more similar than pairs that are further apart (0.141 vs. 0.076; $t(13{,}652) = 17.8$). Additional evidence for sequential search arises from greater similarity between adjacent word pairs with different primary cues compared to non-adjacent word pairs with different primary cues (0.054 vs. 0.011; $t(12{,}819) = 22.4$).

We found that when the model produced a response, it produced another response with the same primary cue in 54.4% of cases. As done in the previous studies (Smith et al. 2013; Bourgin et al. 2014), we also analyzed the change in similarity between the final response (either correct or incorrect) and each one of the ten words prior to the final response. We identified a positive slope in similarity rates as responses were approaching the final answer.

### 4.4.6 Analyses of Neural Responses

In the previous analyses, we demonstrated that our model provides a robust match with the human data by comparing model responses with human responses. In this section, we investigate characteristics of neural responses of individual neurons, as well as groups

Table 4.6: Performance on the RAT and similarity patterns in the response search. Stated 95% confidence intervals are computed on the difference of reported mean values. Values significant at $p < 0.05$ are marked with *, significant at $p < 0.001$ with ***. Reproduced from Kajić et al. (2017b, Table 3).

| Analysis | Humans | Model |
|---|---|---|
| Average problem accuracy | 42% | 43% |
| – Correlation with human data ($r$) | | 0.49* |
| Shortest response sequence | 1 | 1 |
| Longest response sequence | 49 | 33 |
| Average number of responses per trial | 7.78 | 8.44 |
| – Correlation with human data ($r$) | | 0.93*** |
| **Average response similarity** | | |
| – Within vs. across cue clusters | 0.189 vs. 0.041 | 0.141 vs. 0.054 |
| | CI: [0.134, 0.162] | CI: [0.079, 0.095] |
| – Permutation test | 0.182 vs. 0.040 | 0.142 vs. 0.054 |
| | CI: [0.124, 0.160] | CI: [0.077, 0.100] |
| – Within vs. across cue clusters (cleaned responses) | 0.180 vs. 0.039 | 0.141 vs. 0.054 |
| | CI: [0.128, 0.154] | CI: [0.079, 0.095] |
| Baseline vs. actual percentage of response pairs with the same primary cue (two-sided exact binomial test) | 33.3% vs. 37.1%*** | 34.2% vs. 54.4%*** |
| **Average similarity between adjacent and non-adjacent responses** | | |
| – With different primary cues (across cluster) | 0.041 vs. 0.016 | 0.054 vs. 0.011 |
| | CI: [0.063, 0.098] | CI: [0.038, 0.047] |
| – With same primary cues (within cluster) | 0.189 vs. 0.108 | 0.141 vs. 0.076 |
| | CI: [0.063, 0.098] | CI: [0.057, 0.072] |

of neurons, performing computations that underlie the word search process in the RAT. Most observations in this section can be regarded as qualitative comparisons to spiking patterns observed in cortical neurons.

Figure 4.9 shows an excerpt of the spiking activity in three different parts of the model

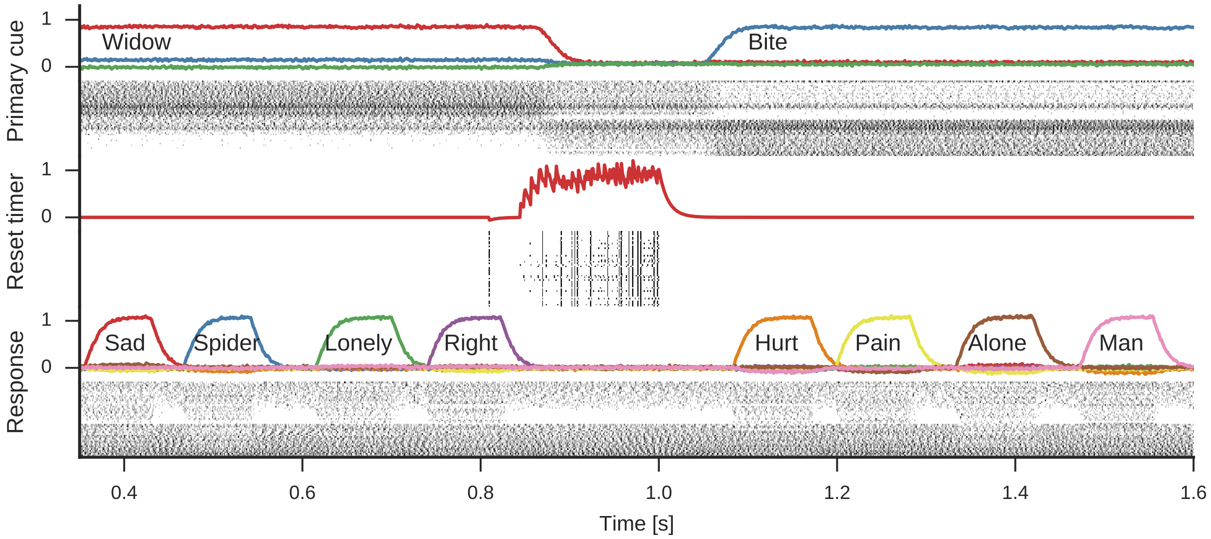


Figure 4.9: Spikes and decoded values for three neural groups in the model. Data shown are an excerpt from a longer single simulation run. From the top to the bottom, the activity of neurons is shown in the following networks: the *primary cue*, the cue selection *reset signal*, and the *response*. Line plots for the *primary cue* and *response* show the similarity of the vector decoded from the activity of neurons with the ideal vectors (annotated for clarity). The reset signal line plot shows the decoded scalar value. Reproduced from Kajić et al. (2017b, Figure 5).

for the simulation the RAT problem with the cues *widow, bite, monkey*. Only a subset of all neural activities in the networks *primary cue*, *reset signal* and *response* is shown (see Figure 4.6 for a reminder on these networks within the model). The primary cue starts as *widow*, but changes to *bite* about halfway through. This change is induced by the rising reset signal inhibiting the cue selection and causing a reselection of the primary cue. During the active period of either cue, the response neurons sequentially represent different words associated with the cue. Note, while four associations are shown for either cue, the number of responses generated during each active phase of a primary cue differs.

The spike raster plots in Figure 4.9 and firing rate estimates in Figure 4.10 reveal interesting neuron tuning properties. We observe neurons that appear to be selective to cue words: some neurons only fire for *widow* (Fig. 4.10A), while others only fire for *bite* (Fig. 4.10B) in the shown time span. However, it is important to note that we did not test the response of these neurons to all possible cues and there are likely other words

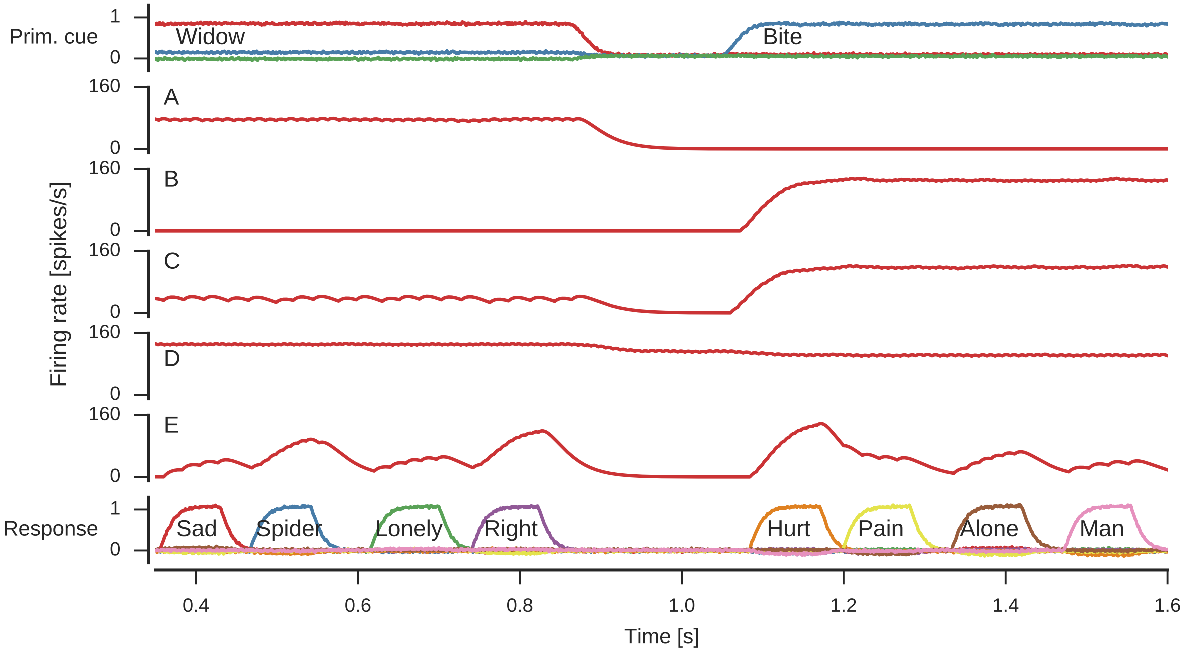


Figure 4.10: Firing rates of individual neurons. Spike trains where filtered with $h(t) = [\alpha^2 t \exp(-\alpha t)]_+$ to obtain firing rate estimates. (**A**) Neuron responding to *widow*. (**B**) Neuron responding to *bite*. (**C**) Neuron responding to both *widow* and *bite* to a varying degree. (**D**) Neuron responding to both *widow* and *bite* with a more subtle difference. (**E**) Neuron responding to varying degrees whenever a response is produced. Reproduced from Kajić et al. (2017b, Figure 6).

that also elicit such responses. Notwithstanding, such selective and explicit response behaviour is consistent with observations from single-neuron recordings in the medial temporal cortex in humans (Quiroga 2012; Földiák 2009). We also observe neurons that fire for both cues, but with different firing rates. This word-dependant change in firing rate is more prominent for some neurons, for example, in Fig. 4.10C we see a neuron that is active for both words, though noticeably more so for *bite*. The change is more subtle for a neuron whose activity is depicted in Fig. 4.10D. The response population also includes neurons that are primarily active when a word is being represented, but not otherwise (Fig. 4.10E). From a single neuron perspective, of particular interest is the reset signal. Here, the neurons produce a clear bursting pattern during the onset of the reset signal. Such behaviour is often thought to need an explanation in terms of complex neuron models that intrinsically burst (Izhikevich 2007), which is not a characteristic of

LIF neurons. Such bursting behaviour occurs because of the recurrent network dynamics producing the reset signal.

We have shown that the spiking RAT model presented in this section, constrained by biological properties such as membrane and synaptic time constants, and the range of neural firing rates, provides a robust match to human data, as demonstrated by the exhaustive comparison between model and human responses. Thus, we argue, the model connects different levels of explanation; at the neural level it proposes specific computational mechanisms such as the WTA and the gating behaviour realized through inhibition, at the cognitive level we observe two processes: cue selection and response generation, both of which are consistent with empirical evidence on semantic search in the RAT (Smith et al. 2013; Davelaar 2015). Finally, the model replicates response patterns and achieves accuracy comparable to those of human subjects, showing that it is able to solve the task and thus replicate important aspects of linguistic behaviours.

## 4.5 General Discussion

In this chapter, we investigated the neurocomputational basis of the semantic search process in the Remote Associates Test (RAT), a task used in research on creativity and insight problem solving. The theory of associative hierarchies by Mednick (1962) proposes that highly creative individuals perform better on this task, as they are able to think of unusual or less frequent word associations that are solutions to harder RAT problems. Such arguments have also been supported empirically, as semantic networks of individuals scoring higher on the test have been found to have small-world network characteristics, which are of importance for efficient word search strategies (Kenett et al. 2014). However, it is unclear how such networks as well as processes operating on them are supported by biological neural networks. To address this gap, in this chapter we investigated two specific aspects of RAT solving: first, what sort of semantic data is most suitable for finding solutions to RAT problems; second, what are different neurally and cognitively plausible mechanisms that might realize semantic search underlying RAT solving.

To address the first aspect, in Section 4.2 we analyzed the performance on the normative RAT dataset from Bowden and Jung-Beeman (2003a) using two different sources of semantic data: Free Association Norms (Nelson et al. 2004) and the Google Ngram dataset (Michel et al. 2011). We argued that the methods used to evaluate the datasets are comparable to the cognitive strategy relying on insight, rather than explicit problem solving. While we found that the Ngram dataset produces the most correct solutions, word association norms provided a closer match to human data. Somewhat surprisingly, it was also found that latent structure representation as derived with SVD did not improve performance on the task.

Using word association norms, we then presented a cognitive model in Section 4.3 that implements the semantic network and search processes involved in the RAT based on a precisely tuned WTA mechanism and a layer of inhibitory neurons. The model implements cognitively plausible mechanisms such as spreading activation and the WTA, while distinguishing between the RAT problems of different difficulty. However, aspects of such a model are unrealistic in biological terms, such as the localist representation and poor scaling properties. To deal with such limitations, we extended that model using the methods of Neural Engineering Framework (Eliasmith and Anderson 2003) in Section 4.4. In particular, we designed its components to search for a solution in a way that is consistent with what is known about how humans search for a solution to RAT problems (Smith et al. 2013; Davelaar 2015).

The resulting model is constrained by biological properties such as membrane and synaptic time constants, and the range of neural firing rates, while exhibiting a moderate correlation with human performance on the test. We argue that the model bridges descriptions at different levels of explanation, starting from the low-level neural mechanisms (e.g., spiking activity) to cognitive processes (e.g., memory search) and behaviour (e.g., linguistic task solving). We corroborated this claim with a detailed analysis of model responses that replicate similarity patterns observed in human responses (Smith et al. 2013; Davelaar 2015). The model has reproduced the pattern of RAT item difficulty correlating with human accuracies on the 25 problems from Smith et al. (2013). Individual differences in associative networks known to influence the performance on the test (Kenett et al. 2014) were modeled by randomly dropping a fraction of associations from the association matrix.

Aside from the test accuracy, we also demonstrated that quantitative and qualitative properties of model responses show strong correlations with human data, in particular with aspects pertaining to response sequence lengths, bunching of responses around single cues, cue switching, and sequential search. Those same measures have been used to characterize human responses (Smith et al. 2013), as well as some model responses (Bourgin et al. 2014). While the probabilistic approach used in Bourgin et al. (2014) were also successful in reproducing those effects, they made no reference to underlying cognitive or neural processes. Interestingly, we found that the best match with the human data was observed when the Ngram dataset was used in conjunction with a response filtering method. These observations are somewhat at odds with conclusions obtained in Section 4.2. We hypothesize that this could be due to several reasons. First, with the biologically plausible model, we focused on a different cognitive strategy that involves a deliberate search process, in contrast to the more rapid insight-based process discussed in Section 4.2. Second, with the model presented in this section we matched human data more rigorously on a different dataset that consists of 25 RAT problems, where the solutions appear to be mostly compound words, or word phrases involving the problem cues. These features appear to be better captured with co-occurrence statistics present in the Ngram dataset. Thus, these observations lead us to speculate that humans use both the associative and co-occurrence semantic information when searching for a specific word, as they do in RAT problems. We argue that this spiking model offers a first unified account of the RAT search process in terms of both psychological and biological mechanisms. Consistent with evidence on single neuron responses, we have shown that individual neurons in the model are capable of displaying both specificity, which is commonly attributed to the theory of the localist representation in the brain, as well as broad tuning that is commonly attributed to the theory of distributed representation.

We also point out that there are several aspects of the model that can be improved and investigated further in light of additional human data. First, the primary cue switching is induced in quite regular intervals in the model. While we cannot exclude the possibility that this is the case in the actual cognitive process, we expect the actual process to be more complex, and possibly depend on the RAT problem at hand. It would be interesting to explore how changing this part of the model can improve the match to human data, especially regarding the percentage of response pairs with the same primary cue. Second, the filtering of potential responses could be further investigated by exploring a more sophisticated process that discards less of the human and model responses, providing a closer match with the plausible cognitive mechanism. Third, current analysis methods filter out repeated responses, but these might give additional information on the search process and considering their occurrence patterns would allow us to refine the response inhibition network. Finally, the current model does not explain how humans learn word associations, or how the process of learning relates to changes in connection weights that store the relevant information. Since the acquisition of linguistic structure happens early in childhood and continues to develop throughout adulthood, a full account of word representation in the brain would also need to address learning at multiple time-scales, as well as mechanisms which enable such learning. In Chapter 6 we revisit the question of how aspects of such associations are learned in an interactive setup with agents solving a collaborative task while situated in an environment.

# 5 | Associative Mechanisms in the Brain: Semantic Fluency

In this chapter, we explore the neurocomputational basis of a semantic memory search paradigm known as the semantic fluency task. The search process in the task unfolds from a single cue word, and, compared to the search process in the Remote Associates Test (RAT), discussed in the previous chapter, it is less constrained as it does not have a single correct solution. Instead, all words that are generated according to the task rule (e.g., "list all animals you can think of") are considered as valid responses. Although related, the semantic fluency paradigm is also different from the free association task discussed in Section 3.1, as in the free association task participants are asked to produce only one response that comes to their mind, or is meaningfully related to the given cue word.

The semantic fluency task has been used to investigate how people retrieve related items from memory (Bousfield and Sedgewick 1944). The responses they produce conform to a characteristic semantic pattern, where the words tend to group in *clusters* (Troyer et al. 1997; Hills et al. 2012a). All words in such a cluster belong to one semantic subcategory. In this chapter, we present and expand on the biologically constrained model of the semantic fluency task first proposed in Kajić et al. (2017a). We show that some model components and mechanisms are the same as those in the RAT model in Chapter 4. Again, we use different sources of semantic data to model associative relationships between words in the model, and evaluate which ones produce responses that reflect characteristics of human responses on the task. Finally, we discuss the plausibility of the model and the relationship between its components and their neurally plausible counterparts.

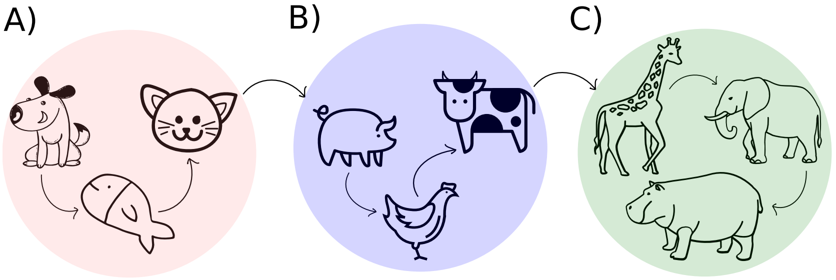


Figure 5.1: A toy example of possible word responses in the semantic fluency task with the cue word “animal”. Each colored area denotes a different animal category or cluster: A is for pets, B is for farm animals and C is for African animals. Arrows denote the sequence of responses.

## 5.1 The Semantic Fluency Task

Different verbal fluency tasks, also known as *word generation* tasks, have been used to study how humans search memory when retrieving related items according to task instructions (Thurstone 1938; Bousfield and Sedgewick 1944). In a typical fluency experiment, an individual is asked to list as many words as possible, usually within a time limit of 60 seconds. Two versions of the task are common: In the *phonemic* fluency task, participants are asked to list all the words starting with a given letter, such as *f* or *s*. In the *semantic* fluency task, they are asked to list all words that relate to a given cue word such as *animal* or *grocery items*. In experimental settings investigating memory processes, the tasks are used to study characteristics of the organization and retrieval processes from semantic memory (Troyer et al. 1997; Abbott et al. 2015). In clinical evaluations of patients with impaired memory function, the task is used to investigate and characterize effects of different brain lesions on word retrieval (Randolph et al. 1993; Gourovitch et al. 2000; Benke et al. 2003).

The responses in the task tend to be grouped into clusters corresponding to subcategories (Troyer et al. 1997), such as *pets* or *farm animals*. For example, responses might start with the animals most familiar to an individual, such as *dog*, *fish*, *cat*, and then continue with a list of farm animals such as *pig*, *chicken* and *cow*. Such responses can be categorized in groups also referred to as *clusters*, as shown in Figure 5.1. Measures typically reported to characterize the search process are the number of words generated, cluster switching patterns, response times, word similarity trends, and so on.

Functional brain imaging studies have identified contributions of frontal and temporal lobes in the task. More specifically, frontal lobes have been implicated in the strategic search process and category switching, while temporal lobes are involved in word clustering patterns, verbal memory and temporal storage (Troyer et al. 1998). It has also been shown that age-associated diseases, such as Alzheimer's and Parkinson's, cause reduced performance on the task (Randolph et al. 1993). It is assumed that the loss of neurons and synapses occurring in the Alzheimer's disease negatively impacts cognitive processes involved in the production of word clusters and animal exemplars within a cluster (Binetti et al. 1995). The impairment of the basal ganglia function in Parkinson's disease affects the word retrieval process, so that affected individuals on average produce fewer words in the task compared to healthy subjects (Raskin et al. 1992).

To explain the clustering trend observed in the responses, Hills et al. (2012a) suggest that individuals generate responses according to the optimal foraging policy (Charnov 1976). Animals use such a strategy when searching for food in natural environments: after resources in one area have been depleted, animals continue their search for food in a new spatial patch. In the context of the semantic fluency task, an individual listing animals in a specific subcategory would stop listing animals from that category after being unable to generate new items at a certain rate. Search behaviour suggestive of optimal foraging has been reproduced with several different representations and algorithms, including a random walk on a semantic network constructed from free association norms (Abbott et al. 2015). Jones et al. (2015) attribute the simplicity of this particular algorithm to the association norms being a direct result of an experimental design that is very similar to the semantic fluency task. They argue that the fundamental memory retrieval processes and representations are obscured by the data underlying the model and the behaviours that are being explained. However, association data from sources other than association norms, such as data learned from natural language, have also been used to successfully reproduce human response patterns with random walks (Nematzadeh et al. 2016). Thus, such alternative accounts argue that it is the structure of the network, rather than the processing mechanisms that lead to responses that conform to the optimal foraging patterns.

The modelling work presented here does not directly address the debate on the structure versus mechanisms, as we assume that a structured semantic representation is already available to the model, independently of how it is acquired. This differs from the approach in Nematzadeh et al. (2016), where such representations are learned from linguistic input and where assumptions on the origin of structure are made explicit. Instead, assuming that the model is provided with a structured semantic representation standing for semantic memory, we investigate how such representation can be realized in a distributed way and how mechanisms operating on such representations can produce observed responses.

The existing models typically do not attempt to address neural or cognitive mechanisms underlying the task performance. As such, these models remain largely agnostic to two components of cognitive processes suggested by Troyer et al. (1997): clustering, "the production of words within semantic or phonemic categories" and switching, "the ability to shift efficiently to a new subcategory". In what follows, we present a model that we argue realizes semantic fluency search in a biologically constrained way, while being consistent with proposed cognitive processes in the literature.

## 5.2 Biologically Constrained Model

We propose a network of simulated spiking neurons that is able to perform the semantic fluency task in a manner that we argue is similar to human performance on the task. While providing a good match with behavioural data, the model also proposes specific neural mechanisms that may be involved in semantic processes. The components of the model are discussed in terms of functionally and neurologically plausible counterparts found in the human brain. The model performs the search based on associative weights encoded within connections between neurons, resembling aspects of a random walk, while conforming to constraints of neural computations. The noise resulting from spiking neurons and the diversity in neuron parameter values lead to the response variability.

As with the biologically constrained RAT model presented in Chapter 4, we use the NEF methods (Eliasmith and Anderson 2003) to implement the model in the Nengo simulation environment (Bekolay et al. 2014). Given that both models implement a semantic search process, there is a considerable overlap between model components and mechanisms. As consistent with the theory of a distributed representation in the brain (Huth et al. 2016; Rissman and Wagner 2012), we employ semantic pointers to represent words.

### 5.2.1 Word Representations and Association Data

Since we focus on matching the human data from the semantic version of the fluency task with a cue word *animal*, the vocabulary used in the model is restricted to animal words only. Such words are represented by $256$-dimensional unit vectors. The vectors are generated randomly such that the similarity between any two vectors is generally less than $0.1$, a default value in the SPA framework. This ensures almost orthogonal vectors,

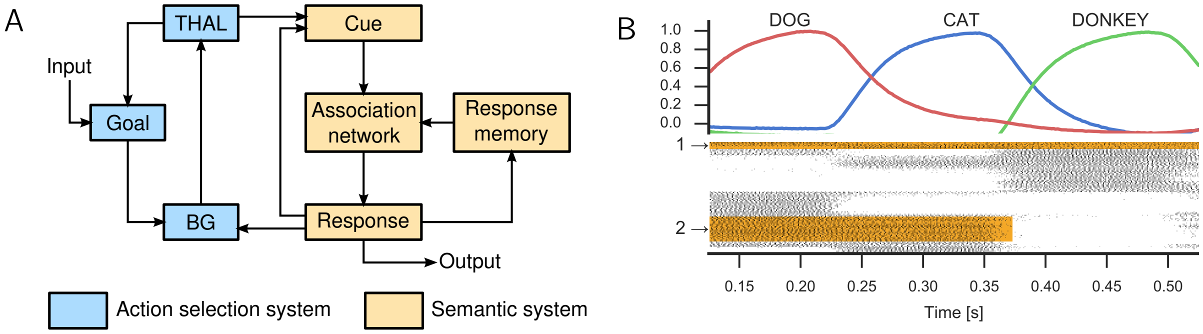


Figure 5.2: **A**: Architecture of the neural network model performing the semantic fluency task. Each box represents a population of spiking neurons. **B**: Neuronal spiking activity in the model recorded from the population *cue*. Some neurons are actively spiking when representing words *dog*, *cat* and *donkey* (highlighted area 1), while others only spike when representing words *dog* and *cat* (highlighted area 2). The similarity between these spikes and the ideal spike pattern for each word is shown above. Reproduced from Kajić et al. (2017a, Figure 1).

with some overlap in the representation, meaning the same neurons will be involved in the representation of different words. An example of this representational overlap can be seen in the spike raster plot in the lower panel of Figure 5.2B, where some neurons fire independently of the word they are representing, such as those whose activity is highlighted with the number "1". Other neurons are more selective and fire only when representing a subset of all possible words, such as those labelled with the number "2" in the same plot. In the upper panel of Fig. 5.2B, the similarity of the decoded spiking activity is shown in terms of the words found in the overall vocabulary. Since the activity of neurons is noisy, the decoded vectors can be more or less similar to the pattern as defined by a canonical semantic pointer.

Associative relationships between words are represented as linear transformations implemented in the connections between two groups of neurons, as with the RAT model in the previous chapter. Word vectors are collected row-wise into a single matrix $\boldsymbol{V}$ and associations between pairs of words are encoded into a matrix $\boldsymbol{A}$ such that $A_{ij}$ is the association strength from word $i$ to word $j$. We then express a new matrix $\tilde{\boldsymbol{A}} = \boldsymbol{V}^\top \boldsymbol{A}^\top \boldsymbol{V}$ to implement a transformation that multiplies the vector represented by the first group of neurons by the matrix $\tilde{\boldsymbol{A}}$ and transmits the result to the second group. This operation results in a weighted linear combination of vectors that represents words associated with the word represented in the first group of neurons.

Five different datasets are used to construct five association matrices $\boldsymbol{A}$: Free Association Norms (FAN; Nelson et al. 2004), Google Ngrams (Michel et al. 2011), word2vec (Mikolov et al. 2013a; Mikolov et al. 2013b), GloVe (Pennington et al. 2014), and BEAGLE (Jones and Mewhort 2007). The FAN and Ngrams datasets have been introduced in Section 4.2 in the context of the RAT model. Here, we use the asymmetric version of both matrices, and for Ngrams we only use co-occurrences of bi-grams. The word2vec and GloVe datasets were discussed in Sections 2.3.3 and in Chapter 3. We build association matrices for those two models by calculating the cosine angle between two words, where the first word is represented by a row and the second word is represented by the column. The BEAGLE dataset (Jones and Mewhort 2007) is constructed from the semantic space model trained on a 400M-word Wikipedia corpus, yielding unique vector representations for each word. Cosine similarity is also used to compute the similarity between pairs of word vectors, analogously to GloVe and word2vec. Here, we use pre-computed similarities between pairs of animal word-vectors as in Hills et al. (2012a). We restrict the vocabulary size associated with each dataset to 157 words, corresponding to the number of unique animal words extracted from human responses.

### 5.2.2 Model Description

The architecture of the model performing the semantic fluency task is depicted in Figure 5.2A. Functionally, the model can be divided into two components: the *semantic system*, and the *action selection system*, approximately corresponding to two cognitive components of *clustering* and *switching* as suggested by Troyer et al. (1997). In terms of their biological correlates, the semantic system can be thought of as representing areas of the medial temporal cortex, and the action selection system as representing the basal ganglia and the thalamus. The action selection system maintains two possible phases: initializing the task and performing the task.

The initialization phase is active only at the beginning of a simulation, where external input is used to drive the *goal*[1] population of neurons to represent the vector **start**. The second phase consists of performing the task itself, and occurs once a cue is provided. After the task has been initialized, the action selection system (consisting of the *basal ganglia* BG and *thalamus* THAL populations) switches to the process of generating word responses within the semantic system. The recurrent action selection system maintains

[1] We use *italics* to refer to the name of a population of neurons or the vector that is represented by that population, which is to be inferred from the context. The **bold** font is used to refer to labels assigned to vectors representing a word. For example, *cue* · **animal** refers to the dot product between the vector represented by the population of neurons labeled “cue” and the vector corresponding to the word “animal”.

Table 5.1: Utility calculations for different goals and the corresponding actions. Reproduced from Kajić et al. (2017a, Table 1).

| | Goal | Utility calculation | Action |
|---|---|---|---|
| 1. | Start | $goal \cdot \mathbf{start}$ | Set *cue* to **animal**, set *goal* to **think** |
| 2. | Think | $goal \cdot \mathbf{think} + \text{response\_magnitude} - 1$ | Copy *response* to *cue*, add *response* to *responses*, set *goal* to **think** |
| 3. | Default | $0.4$ | Set *cue* to **animal**, set *goal* to **think** |

word generation by simultaneously evaluating utilities of actions and selecting the action with the highest utility value. Table 5.1 shows the mapping between utility calculations and corresponding actions by the action selection system. Since the external input initially sets the *goal* to **start**, the action selection system will select the first action due to its high utility value. This action will feed the vector **animal** as input to the population *cue*, and set the representation in the *goal* population to **think**. This action can be interpreted as the instruction "list all animals you can think of".

Next, the semantic system begins to generate associations of the word **animal** within the *association network*. The connection between *cue* and the *association network* implements the transformation $\tilde{\boldsymbol{A}}$, as described in the previous section. The *association network* then represents a vector which is a linear combination of word-vectors associated with **animal**. For example, there might be a representation corresponding to the vector: $0.5$·**cat** + $0.4$·**dog** + $0.1$·**fish**. Coefficients represent association strengths between each individual word and the word **animal**, as derived from the association matrix $\boldsymbol{A}$. A WTA mechanism within the network selects the vector with the largest coefficient, and projects it to the *response* population. In this example, the *response* population would now represent the vector **cat**.

When a response has been generated, the action selection system selects the second action (see Table 5.1) due to its high utility value. This action projects the word represented in *response* (e.g., **cat**) to *cue*, simultaneously adding it to the representations stored in *response memory*. The *goal* continues to be **think**. This process within the semantic system continues, with the action selection system selecting the second action most of the time. To prevent the same responses from re-appearing immediately, *response memory* is implemented as a neural integrator population. It projects inhibitory connections to the *association network* in order to suppress representations of words previously generated as

Table 5.2: List of model parameters. Reproduced from Kajić et al. (2017a, Table 2)

| Name | Value (unit) | Explanation |
|---|---|---|
| $d$ | 256 | Dimensionality of word vectors |
| $assoc\_th$ | 0.3 (or 0.25) | Default WTA input threshold (Ngram, BEAGLE threshold) |
| $c_{\mathrm{cs}}$ | 3 | *Cue* to *association network* connection strength |
| $c_{\mathrm{fs}}$ | 0.2 | *Cue* feedback connection strength |
| $c_{\mathrm{inh}}$ | $-5$ | *Response memory* to *association network* inhibitory connection strength |
| $\tau_{\mathrm{syn}}$ | 100 ms | Synaptic time constant between *association network* and *response* |
| $\tau_{\mathrm{syn}}$ | 5 ms | Synaptic time constant (default) |
| $max\_rate$ | 200 Hz to 400 Hz | Range for maximal neural firing rates (default) |

responses.

The last action with a fixed utility value of 0.4 is selected if utilities of all previous actions have evaluated to a lower value. This occurs when the system is unable to come up with a new response (e.g., the WTA mechanism takes too long to decide between two words). While rare, when this situation occurs, the action selection system sets *cue* to represent the input **animal** and the *goal* is set to **think**, which can also be interpreted as the "switching" pattern observed in human responses.

Table 5.2 lists the major parameters in the model. Most of them have been left at their default values as defined in Nengo software (Bekolay et al. 2014). Some parameter values (e.g., maximal firing rates) are selected randomly. Each time the model is run, a new set of such parameters are chosen. Such diversity in parameter settings is a first approximation of differences in cognitive processing that may occur across cortical regions of different individuals.

## 5.3 Results

We ran 141 model simulations for each of the five association matrices (Beagle, Ngram, FAN, word2vec and GloVe) and compared the results with human data from Hills et al. (2012a). The number of simulations corresponds to the number of participants in the experiment. For each comparison, we performed the same analysis with model responses

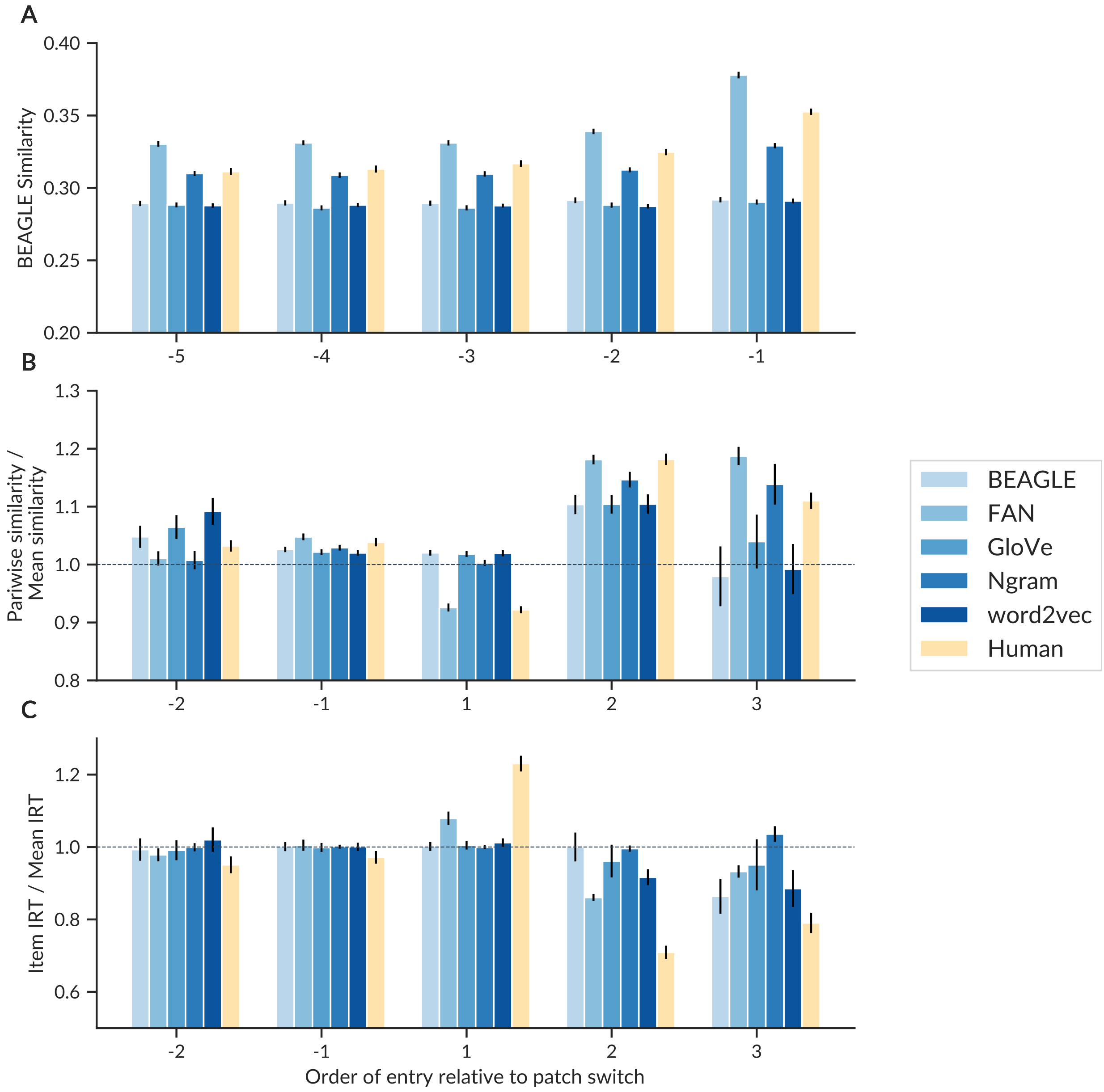
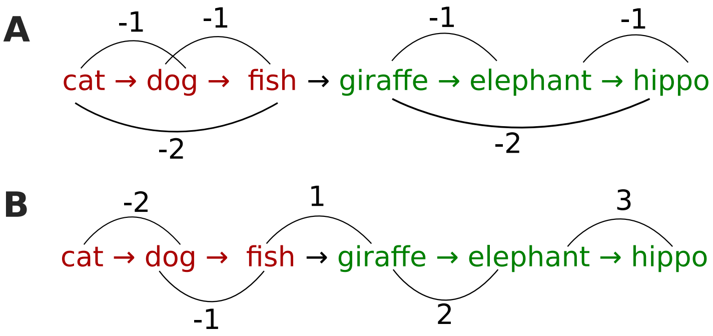


Figure 5.3: A comparison between model responses (blue) and human responses (yellow, reproduced from Hills et al. (2012a). **A**: Pairwise similarity between a word and the words preceding it within the same categorical cluster. **B**: Pairwise similarity between subsequent words. **C**: Inter-item response times (IRT) between subsequent words. Standard errors of the mean are shown with error bars in all plots.



A
-1
-1
-1
-1
cat → dog → fish → giraffe → elephant → hippo
-2
-2
B
-2
1
3
cat → dog → fish → giraffe → elephant → hippo
-1
2

Figure 5.4: A toy example of six responses in two colour-coded clusters in the semantic fluency task: *pets* in red, and *African animals* in green. Numbers refer to relative item indices: **A** for within cluster similarity in Fig. 5.3A, and **B** for word similarity across clusters as well as timing data in Fig. 5.3B and C.

that were conducted with the human data in the original study. The simulations were run until the average number of responses produced by the model matched the average number of responses produced by participants within three minutes.

For each simulation, we recorded the sequence of the model's responses (as decoded vector representations in the *response* population), and inter-item response times (IRT) as times between the onset of the current response and the previous response. The onset of the current response is defined as the moment when the vector represented in the *response* population matches any of the pre-defined animal word vectors in the vocabulary. A match is defined a dot-product of at least $0.8$, a threshold that has been found empirically to produce the desired behaviour. We only consider relative timings, that is, the time differences between responses. A mapping to the absolute timing, that is, a mapping to exact times as reported in the human experiments would require a model of other cognitive and sensory-motor processes that are beyond the scope of this model. Model responses were pre-processed in the same way as the human data, using the set of scripts provided publicly by the authors of the original study. Each response produced by the model is assigned an animal category, and the clusters are identified as sequences of responses within the same category. An animal that could be assigned to two clusters is assigned to both.[2]

The first analysis compares the pairwise similarity of a word in a cluster (also referred to as a *patch*) and each of the five words preceding it. The results are shown in

[2] See Troyer et al. (1997) for more detailed description of the categorization procedure.

Figure 5.3A. The graph presentation in the figure follows that of the original study to aid the comparison, however, for clarity, in Figure 5.4 we included a toy example that illustrates the meaning of indices on the x-axis in Figure 5.3. The similarity shown on the y-axis is computed as a dot product between two BEAGLE vectors corresponding to the two words in a word pair. The reproduced results for human data in Figure 5.3A confirm that the word most similar to the recent word in the patch is the one preceding it, observed by the gradual increase in the average word similarity as moving from the index $-5$ to $-1$. It has been argued that this similarity trend supports the theory of locality in a memory structure (Hills et al. 2012a). For the model, this trend is observed with the Ngram and the FAN association matrices, with Ngrams providing a closer match. The effect is not observed with the BEAGLE, GloVe or word2vec association matrices, for which the pairwise word similarity remains flat across different word positions in the cluster.

The second analysis compares the pairwise similarity of subsequent items, relative to the position of an item in the cluster, with results shown in Figure 5.3B. With human responses, the lowest pairwise similarity occurs at the *cluster transition points*, indicated by "1" on the x-axis in the figure as well as in Figure 5.4B. That point shows the average similarity between the *first* word in a cluster and the *last* word in the previous cluster. For example, in Figure 5.4B that is the similarity between the words *fish* and *giraffe*. For humans, the mean similarity $\mu$ at the cluster switch is $\mu = 0.92$ with the standard deviation $\sigma_\mu = 0.01$. The model using the FAN data yields comparable results with $\mu = 0.93$ and $\sigma_\mu = 0.01$. This effect was weakly observed with the Ngram ($\mu = 1.00$, $\sigma_\mu = 0.01$) and BEAGLE datasets ($\mu = 1.01$, $\sigma_\mu = 0.01$), and less so with GloVe and word2vec (both $\mu = 1.02$, $\sigma_\mu = 0.01$), as the word similarity at the transition point remains above a subject's average.

The third analysis is concerned with the position of a word item within a cluster and the speed of generating that word. The ratio between the average IRT for an item and the participant's mean IRT over the entire task is shown in Figure 5.3C. Human participants take the most time to produce the first word in a new cluster (reported $t(140) = 13.14, p < .001$) and the least time to produce the second word in a new cluster (reported $t(140) = 11.92, p < .001$). This effect is the hallmark prediction of the optimal foraging strategy, suggesting that cluster switches occur when the current IRT increases over the mean IRT value. Figure 5.3C shows that the model using the FAN association matrix exhibits the same effect. It takes significantly more time to generate the first words in a new cluster ($t(140) = 4.78, p < .001$), compared with the second words in the cluster ($t(140) = 4.78, p < .001$). Interestingly, the effect is also significant, although much weaker for the word2vec dataset, for both the first item ($t(140) = 3.08$, $p < .005$) and the second

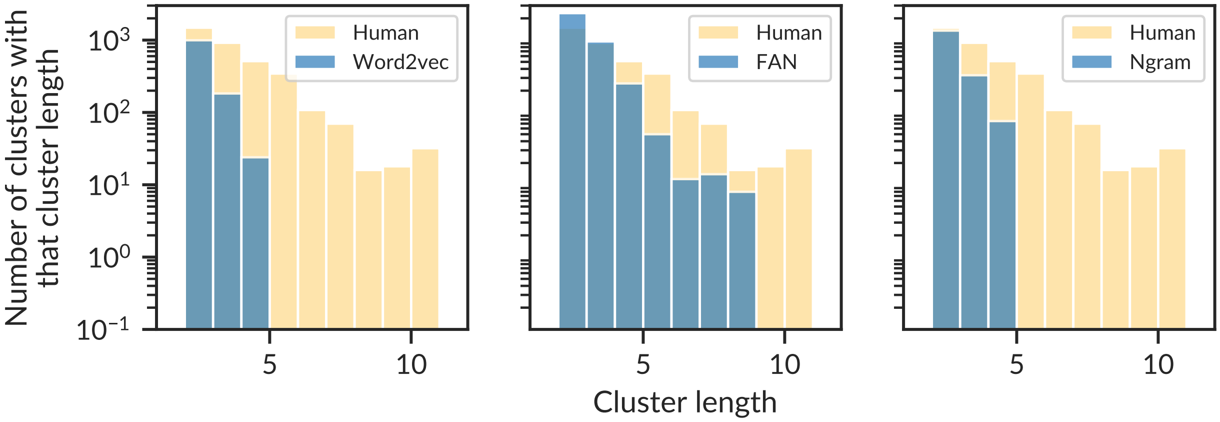


Figure 5.5: A comparison of distribution lengths between the model responses and responses generated by the models using the word2vec, FAN and Ngram matrices.

item ($t(140) = 3.50$, $p < .005$).

Lastly, we expanded the set of analyses with an additional comparison of average cluster length distributions. These lengths are calculated for each sequence of responses by counting the number of words in a cluster. Those computed from human responses are then compared with those generated by the models, using the matrices that showed a good match with the human data, namely the word2vec, FAN and Ngram datasets. The distribution of cluster lengths over the number of clusters with corresponding lengths are plotted as histograms in Figure 5.5. In all cases, the models on average produce clusters with fewer words: while for human responses the average cluster length is $\mu = 2.61$, it is $\mu = 2.03$ for FAN, $\mu = 1.28$ for word2vec and $\mu = 1.43$ for Ngram. Also, the maximum cluster length for human data is 11, while for models that number is smaller for the model: 8 for FAN, and 4 for Ngram and word2vec. These results indicate that, compared to human responses, model responses are characterized by more frequent cluster switches. In summary, we conclude that across different analyses conducted here, the FAN matrix provides the closest match with the human data.

## 5.4 Discussion

In this chapter, we proposed a biologically constrained neural network model of cognitive processes involved in the semantic fluency task. We evaluated the model on five different semantic datasets: Free Association Norms, Google Ngrams, word2vec, GloVe and

BEAGLE, by embedding association strengths extracted from the datasets in connections between neurons within a large recurrent network. We explored which of the datasets provides the closest match with the human data from Hills et al. (2012a). In doing so, our focus was on identifying plausible, causal neural mechanisms for performing the semantic fluency task.

The word similarity trends and inter-response timing patterns of responses produced by the model are consistent with predictions made by optimal foraging theory as proposed by Hills et al. (2012a). More specifically, the model is more likely to switch to a new animal category when when the average similarity of subsequent responses drops below, or gets close to, the overall mean similarity. While this effect was observed with all five association matrices, it was most pronounced with the FAN matrix. Also, the Ngram matrix as well as the FAN matrix produced responses that support the hypothesis on the locality of search in the memory as described by the increased similarity of subsequent responses within a cluster.

The analysis of timing effects allowed us to clearly distinguish between the five matrices in terms of their matches with the human data. The model using FAN exhibited the same timing effects as observed with human responses, which is that of increased time needed to produce the first word in a new cluster, and decreased time to produce the second word. This timing effect was weakly observed with the word2vec dataset, although FAN provided a closer match. None of the effects observed with other association matrices were significant. Finally, we qualitatively analyzed how closely the distribution of cluster length responses produced by the model matches that of human responses. On average, model responses show more frequent cluster switching compared to the human data. Out of analyzed datasets, the effect was most prominent for word2vec and Ngram, while the FAN produced cluster lengths that were on average longer and thus closer to human data.

The ability of association norms (FAN) to model aspects of the search process in the semantic fluency task is likely due to the similarity between cognitive processes involved in the fluency task and in the free association task, as pointed out by Jones et al. (2015). However, this result could also be seen as support for the plausibility of the proposed neural mechanisms, as they are able to generate behaviours in accordance with these underlying associations. We expect that a better understanding of cognitive processes involved in free associations could aid understanding of the processes underlying semantic fluency. Our model may prove useful in exploring a variety of possible ways that such associations are neurally realized, as the direct embedding in connection weights as done here is only one possibility.

When building biologically constrained neural models, timing is a highly constrained property of a model—the timing of responses is sensitive to both neural time constants and our characterization of the concept representation. As such, timing is an inherent property of the model, and only some ranges of parameters influencing the timing of responses in the model, such as synaptic time constants, will yield the desired behaviour. This is one of the major differences compared to models based on semantic networks, where the timing function needs to be specified by the modeller. For example, Abbott et al. (2015) assign a discrete timestamp to each response based on whether it is unique in the sequence and depending on the number of previous, non-unique responses. In our model, we needed to explicitly control for exclusion of repeated responses, which we did with the *response memory*, a network that inhibits recent responses from appearing again, simulating the function of working memory in the fluency task.

We also identified that a longer synaptic time constant ($\tau_{syn} = 100\,\mathrm{ms}$) was needed between the *association* network and the *response* populations to stabilize the representation. This observation can be interpreted as a prediction that this network will be rich with NMDA receptors in the biological system, as such receptors are known to have longer time constants in the range of $50\,\mathrm{ms}$ to $100\,\mathrm{ms}$ (Sah et al. 1991). In comparison, the more common AMPA receptors have time constants in the range of $5\,\mathrm{ms}$ to $10\,\mathrm{ms}$ (Destexhe et al. 1994). NMDA receptors are found in hippocampus, and their function has been implicated in cellular learning and memory mechanisms in brain areas such as the medial temporal lobe, neocortex and others (Gazzaniga et al. 2014).

Our characterization of the neural concept representation also has an effect on the timing responses. Specifically, we have observed that the dimensionality of employed vector representations needed to be sufficiently large to achieve experimentally observed timing effects. While we find that $d = 256$ suffices for this purpose, a systematic search of dimensionality effects on the performance is needed to see how it affects the behaviour. We have tested this model with lower values (e.g., $d = 64$) and it produced results in support of local search strategy, yet it failed to provide a good match with the timing data. In related literature, it was suggested that $d \approx 500$ is necessary for representing human-scale conceptual structures (Eliasmith 2013), which is consistent with our observations.

# 6 | Behavioural Basis for the Emergence of Linguistic Structure

In previous chapters, we investigated the neurocomputational basis of cognitive processes involved in different semantic memory search paradigms. Our experiments have demonstrated that the most accurate match with human data on those tasks was achieved with biologically realistic models using free association norms as a source of semantic information (Nelson et al. 2004). Such results affirm the importance of associative relationships in a variety of linguistic behaviours. While we have shown how such associations might be realized in the brain, we have not yet addressed the mechanisms underlying *learning* of associations, or any aspects thereof. In Section 2.2, we discussed stages of language acquisition, including the process of word learning and grammaticalization, all of which are inextricably linked to the process of learning of inter-word relationships.

In this chapter, we focus on one specific behavioural aspect of learning of associative relationships. We investigate representations learned in a cooperative communication paradigm using reinforcement learning (RL). The agents are required to communicate useful information in order to jointly complete the task. However, they need to agree on a communication protocol in order to do so. We study the emergent communication system, the mapping between learned signals and meanings, and discuss the properties of the system in the context of some fundamental linguistic features. Parts of this chapter have been adapted from Kajić et al. (2020).

## 6.1 Language as a Collaborative Effort

In Section 2.1.2, we discussed the notion of language as a form of joint, coordinated action, as proposed by Clark (1996). According to such a pragmatic view of language, the context providing meaning extends beyond that of linguistic conventions—instead, it depends on many different factors. For example, such factors at the level of linguistic adaptation are aspects of the social, the physical and the technological environment (Lupyan and Dale 2016). At the level of individual communicative acts, the relevant factors are intentions of the people involved (Grice 1975). This versatile and adaptive nature of language is also evident in studies showing that humans possess a striking ability to adapt their communication in light of changes to such factors: for example, by coordinating their actions to rapidly develop novel communication systems that exhibit core features of natural language (Bohn et al. 2019; Galantucci 2005).

The study of emergent communication systems and emergent languages is generally challenging as "established human communities have very little need to originate novel communication systems" (Galantucci 2005). A few exception are sign languages developed by deaf children born to hearing parents (Goldin-Meadow and Feldman 1977; Goldin-Meadow 2005), and "contact" languages such as pidgin and creole that develop out of a communicative need between communities that do not share a common language (Siegel 2008). Thus, the majority of communication systems used are studied as the end product, with no opportunity for experimental manipulation to examine the influence of different factors on the resulting system. Consequently, the emergence of linguistic structure is often studied in experimental setups using artificial languages and communication systems (Kirby et al. 2008; Little et al. 2017; Galantucci 2005), as well as using computer simulations with artificial agents (Cangelosi and Parisi 2012; Brighton and Kirby 2006; Steels 2004). While such agents are limited in their capacity to capture the complexity of human cognition and behaviour, they offer a valuable testbed for exploring the influence of various factors on the emergence of linguistic structure.

In recent years, emergent artificial languages and their relationship to natural language have been increasingly studied using interactive multi-agent setups in RL. In such setups, agents endowed with a communication capability learn to reason about novel signals while being situated in environments (Li and Bowling 2019; Lazaridou et al. 2018; Mordatch and Abbeel 2018; Kottur et al. 2017). In contrast to "passive" approaches, such as learning of word embeddings from large corpora of text, as discussed in Section 2.3.3, these setups offer an experimental platform for "active" learning through interactions with the environment. Such interactive learning experience is more likely to resemble, or capture, environmental conditions and pressures faced by humans and AI systems acquiring or learning languages.

Emergent communication is often studied in the context of cooperative games, such as the signaling game of Lewis (1969), where two agents, a sender and a receiver, act jointly to achieve a common goal (Li and Bowling 2019; Lazaridou et al. 2018). In such a game, the sender sees an artifact (e.g., an image) and uses a communication channel shared with a receiver to transmit a message (e.g., a symbol or a sequence of symbols). The receiver then conditions its decision on the message to select the target object among distractors. Both agents are rewarded if the receiver selects the target. Senders and receivers are thus able to perform actions and observe the outcome to adjust their behaviour, in line with the view of language as form of joint action coordination (Clark 1996).

Communication protocols in such games exhibit some level of structure reminiscent of natural language. In particular, compositional structure is found with end-to-end training on disentangled input data (Lazaridou et al. 2018) or by introducing environmental pressures during training (Li and Bowling 2019). While referential games provide a useful paradigm for investigating conditions that give rise to communication protocols with natural language-like properties (Kottur et al. 2017), the resulting communication policies are often difficult to interpret (Lowe et al. 2019). Moreover, the agents' actions in such setups often do not affect the environment, as the sender and receiver are each restricted to one choice of action.

Here, we extend such a referential game paradigm as a navigation task, while investigating relationships in the meaning space of emergent signals. We introduce a control component where an agent needs to reach a specific, to it unknown, location in the environment by relying on signals received from another agent that knows the location. This extension of the task is akin to a situation in which a tourist asks a stranger on the street for route directions to a particular location in a city they are visiting for the first time. Such directions are likely to contain the description along the lines of "Go up this street, take the 2nd street to the right, then 1st to the left". We show that the communication protocol learned by our agents exhibits features that can be interpreted in a similar way, while being optimal for this task, as demonstrated by agents' performance on the task. In addition, we demonstrate that when a population of agents coordinates their joint description of a route, even without knowing each other's descriptions, the protocol exhibits structure, where individual messages encode different spatial aspects of the environment such as *left*, *up*, or *upper left room*.

One difference between our work and related research is that we make minimal to no assumptions about the properties of the communication protocol, or the reward agents receive. For example, in this line of research, it is common to use RNNs to encode messages, but this explicitly biases protocols to have sequential structure. By using populations of independent senders, we show that structure also emerges even without assumptions on the order of symbols. Furthermore, we quantify the optimality of the protocol and show that, in many cases, and depending on the dimensionality of the signal space, the agents learn an optimal communication protocol. Such optimal signals exhibit similarity structure, insofar that signals used to encode spatially proximal locations are more similar than those encoding spatially distant locations.

## 6.2 Methods

The task we propose in this section situates agents in an environment, and augments their communicative actions with a set of behavioural, non-communicative actions. More specifically, one or multiple *senders* communicate signals to the *receiver* that acts upon the signals using non-communicative actions to reach the goal state. Thus, the two agent types are assigned a mutually exclusive set of actions, and, in order to complete the task successfully, they need to learn to coordinate their actions. This inductive bias enforces collaboration between the agents, as the optimal performance on this task requires action coordination. Another assumption refers to their communicative ability, which, in this case, is restricted by the model used to produce the signals (for the sender) and the model used to interpret those signals (for the receiver).

### 6.2.1 The Navigation Task

We consider a cooperative navigation task, where the sender sees the goal location in a gridworld environment, and in response to seeing the location, it sends a message to the receiver. The receiver sees the message, and has to navigate to that goal location to collect the reward. The receiver, however, does not know where the goal location is. When the receiver reaches the goal location, both agents are rewarded. Thus, the receiver relies on the information received by the sender, and it is in their mutual interest to "agree" on useful signals so that the receiver can reach the goal. In our experiments, we study characteristics of the communication protocol as the number of signals varies while the number of possible goal locations stays the same. We refer to the number of signals as the *communication channel capacity*, and discuss it in Section 6.3.1. Communication channel capacity and goal locations can be interpreted as the *signal space* and *meaning*

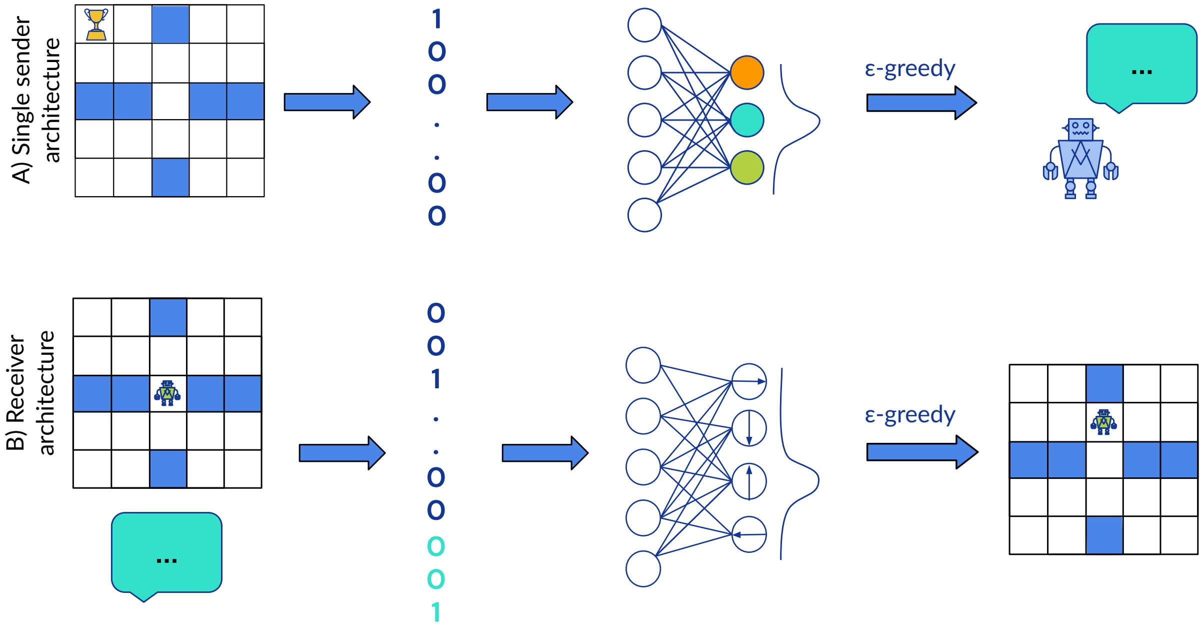


Figure 6.1: Both the sender and the receiver see the gridworld environment, yet only the sender sees the goal location. It selects a message action (a single symbol) based on the one-hot encoding of the goal location. The receiver selects a navigation action based on the multi-hot input vector that encodes its own location and the message. Reproduced from Kajić et al. (2020, Figure 1).

*space*, respectively, in the context of related literature describing language as a mapping between the signal space and the meaning space (Brighton and Kirby 2006).

In a given task setup, there is exactly one receiver, and at most five senders. At the beginning of each episode ($t = 0$), the goal location in the environment is determined by placing a reward at a random, non-occupied location. The sender observes the goal location and it emits a message $m$ from a vocabulary $V$, with $|V| = N$. All messages are represented as single discrete symbols, and in this case we use natural numbers as symbols. The message is stored by the environment and provided as part of the observation available to the receiver. The message observation persists at each subsequent time step $t > 0$. The sender performs no more actions until the end of the episode. For the sender, we interchangeably use the terms "message" and "message action".

After $t = 0$, the receiver observes its own location and the message, and performs a navigation action. This process repeats until either the goal state is reached or the episode terminates with the probability $p_{term}$. Because the episode can be terminated randomly at each step for the receiver, the environment is non-deterministic. The average episode length $T$, given the probability $p_{term}$, can be derived by considering all possible path lengths $l$ by

$$T = \sum_{l} l \cdot p_{term} \cdot (1 - p_{term})^{(l-1)}. \tag{6.1}$$

The path length $l$ is the number of steps the receiver took, excluding the possibility that the episode terminates before the fist step. If the episode terminates because the receiver reached the goal state, both the receiver and the sender are rewarded with the reward $r = 1$, otherwise $r = 0$.

The receiver always starts at the centre of the map. If the setup involves a population of senders, the interaction remains as described, except that the message emitted at $t = 0$ consists of an array of symbols $[m_1, m_2, ..., m_M]$, where $m_i$ is the message emitted by the $i$-th sender and $M$ is the number of senders. A single sender does not have any information about message actions selected by other senders, and selects its own message action independently of actions by others.

### 6.2.2 Experimental Setup

**The Sender Agent**

A sender is modelled as a contextual $N$-armed bandit that selects one message action $m$ out of $N$ possible messages. It selects the message based on the static context vector $c \in \mathbb{R}^d$, where $d$ is the size of the gridworld ($d = height \times weight$). The dimensionality of all gridworlds used in the experiments here is 5×5, thus resulting in 25 locations. The uppermost left location is denoted as $(0, 0)$, and the lowermost right location as $(4, 4)$. The context is a one-hot vector that encodes the goal location by having a single "1" entry corresponding to the location of the reward position, where the location has been mapped from $(x, y)$ coordinates to a single integer index $\{1, ..., d\}$. In setups with multiple senders, each one of them has its own vocabulary that it uses to select messages. Thus, the items in an individual sender's vocabulary are unrelated to items in another sender's vocabulary. In the experiments, we varied $N$ to study the effects of compression on task performance. In particular, we examine the characteristics of the communication protocol when there are more or less messages compared to the possible reward locations in the environment.

The $i$-th sender's action-value function $Q(\cdot)$ is implemented as a single layer feed-forward neural network parameterized by $\theta_{s_i}$. The loss used to adjust neural network parameters during training for a single sender is

$$\mathcal{L}_{s_i}(\theta_{s_i}) = \begin{cases} 0, & 0 \leq t < T \\ \big(R_t - Q(c, m_i; \theta_{s_i})\big)^2 & t = T, \end{cases} \tag{6.2}$$

where $R_t$ is the reward received at the end of an episode of length $T$, $c$ is the context vector and $m_i \in V_i$ is a message action. Message actions are selected with the $\epsilon$-greedy policy, where the agent selects a random message action with $\epsilon$ probability, and the action with the highest Q-value with the $1 - \epsilon$ probability. For simplicity, the $\epsilon$ value is the same for all senders and is determined empirically using hyperparameter search as the one yielding the highest average return on the task. Other hyperparameters are listed and discussed in the subsection 6.2.2 on the environment and training. The implementation of a sender agent is schematically shown in Figure 6.1A.

### The Receiver Agent

The receiver is implemented as a Q-learning agent (Watkins and Dayan 1992) with a neural network representing action-values and parameterized by $\theta_r$. After $t = 0$, which is when it is the receiver's turn to act, the environment provides it with an observation $o_t = [p_t, \bar{m}_1, \bar{m}_2, ...]$ at each time step $t$, where $p_t$ is a one-hot encoding of the receiver position, and each $\bar{m}_i$ is a one-hot encoding of the message emitted by sender $i$. $o_t$ is provided as the input to the neural network, and outputs are Q-values for each of the four possible navigation actions (up, down, left and right). An action is selected using an $\epsilon$-greedy policy, and the TD error (Sutton and Barto 1987) is used to compute the learning loss

$$\mathcal{L}_r(\theta_r) = \begin{cases} 0, & t = 0 \\ \Big(R_t + \gamma \max_a Q(o_{t+1}, a; \theta_r) - Q(o_t, a_t; \theta_r)\Big)^2, & 0 < t \leq T. \end{cases} \tag{6.3}$$

The loss is set to zero for the first time step, because only the sender selects an action at this point, therefore not affecting the receiver's network. Finally, the total loss can be expressed as

$$\mathcal{L}_{total} = \sum_i \mathcal{L}_{s_i}(\theta_{s_i}) + \mathcal{L}_r(\theta_r). \tag{6.4}$$

While the total loss is expressed as the sum of the receiver's loss and individual sender's losses, during the backpropagation process only relevant weights for each network will be adjusted. For example, after the receiver executes a navigation action in one step, it

Table 6.1: Hyperparameters used to train different sender-receiver agent configurations. Reproduced from Kajić et al. (2020, Table 1).

| Name | Values | Description |
|---|---|---|
| $M$ | $[1, 2, 3, 4, 5]$ | Number of sender agents |
| $C$ | $[3, 4, 5, 8, 9, 16, 25, 27, 32, 36, 64]$ | Communication channel capacity ($N^M$) |
| $\eta$ | $[5 \cdot 10^{-5}, 1 \cdot 10^{-4}, 5 \cdot 10^{-4}, 1 \cdot 10^{-3}]$ | Learning rate for RMSprop |
| $\epsilon_s$ | $[0.01, 0.05, 0.1, 0.15]$ | Sender's action exploration rate |
| $\epsilon_r$ | $[0.01, 0.05, 0.1, 0.15]$ | Receiver's action exploration rate |
| $\gamma$ | $[0.7, 0.8, 0.9]$ | Receiver's Q-learning discount factor |
| $layout$ | [Pong, Four room, Two room, Flower, Empty room] | Environments (see Fig. 6.2) |

will either get the reward or not—this information will be used to adjust the weights of its neural network as the derivative of the loss function with respect to only those weights will be greater than zero. Sender's weights will be only adjusted at the end of the episode. These computations are implemented as a computational graph in Tensorflow (Abadi et al. 2016) that automatically tracks the propagation of losses and their assignments to corresponding parts of the nodes in the graph. The loss is minimized using the RMSprop optimization algorithm (Hinton 2014).

In order to additionally incentivize the agents to adopt efficient behaviours, such as reaching the goal state using the shortest possible path, the random termination probability is set as $p_{term} = 1 - \gamma$, where $\gamma$ is the discount factor in the Q-learning algorithm. This ensures that each episode terminates, and we examine the effect of different $p_{term}$ values. This can be also achieved in a different way, for example by setting the reward to $-1$ for each step where the receiver is not at the goal state.

### The Environment and Training

Each experiment consisting of a sender-receiver agent setup is trained for 20 million steps in one gridworld environment. Each step is a step within a single episode, and the length of the episode is determined by the termination probability $p_{term}$, so that on average each episode consists of 3, 5 or 10 steps. Shorter episodes impose an additional pressure on agents to act efficiently. Five gridworld environments: *Pong*, *Four room*, *Two room*, *Flower* and *Empty room* used in experiments are shown in the upper panel in Figure 6.2.

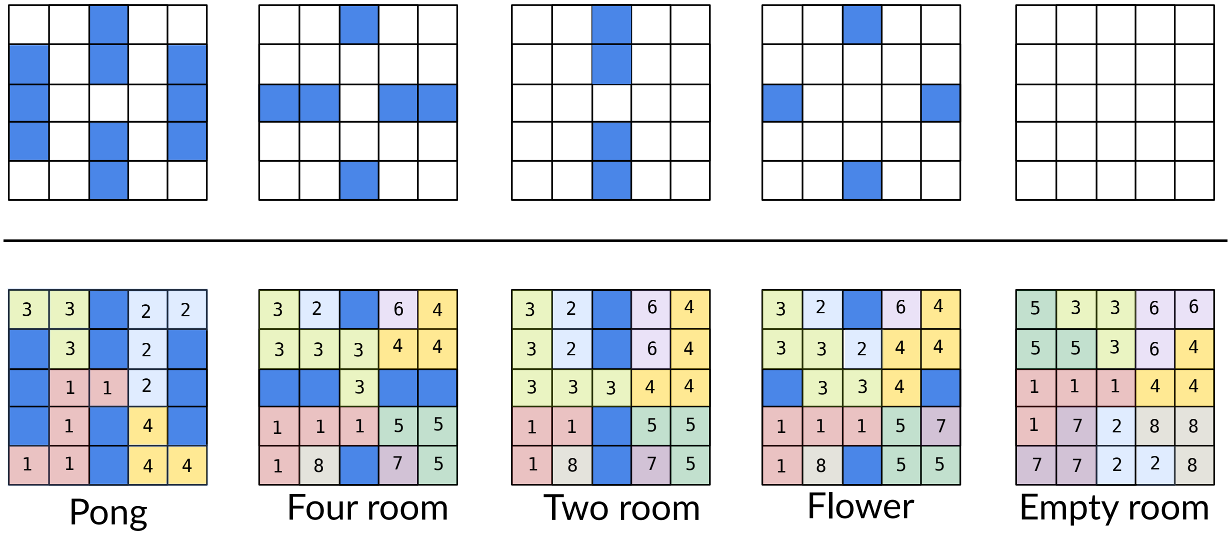


Figure 6.2: **Upper panel**: The five different gridworld environments used in experiments. Blue cells represent walls. Reproduced from Kajić et al. (2020, Figure 2). **Lower panel**: An example of optimal message distributions for different goal locations in all environments. A single number at a location is the message a sender would send if that particular location was the goal state. The locations are colour-coded for redundant presentation to match the message content.

Blue locations in an environment are inaccessible to an agent, representing walls. The centre of each map, which is the starting point for the receiver, is at $(2, 2)$. Table 6.1 lists hyperparameter values, and their descriptions, that were used across experiments. In experiments that contain more than one sender, all senders have the same $\epsilon$ value, and the same vocabulary size $N$. The number of available messages depends on the number of senders, and is selected so as to allow a fair comparison among different configurations in terms of the total number of messages (more details are provided in the following sections). In the analyses, we select the runs with the best performing learning rates and exploration rates, resulting in approximately 180,000 experiments.

## 6.3 Results

We first analyze the agents' performance on the task, expressed as the average training returns. The upper panel in Figure 6.3 shows the returns sampled at regular intervals

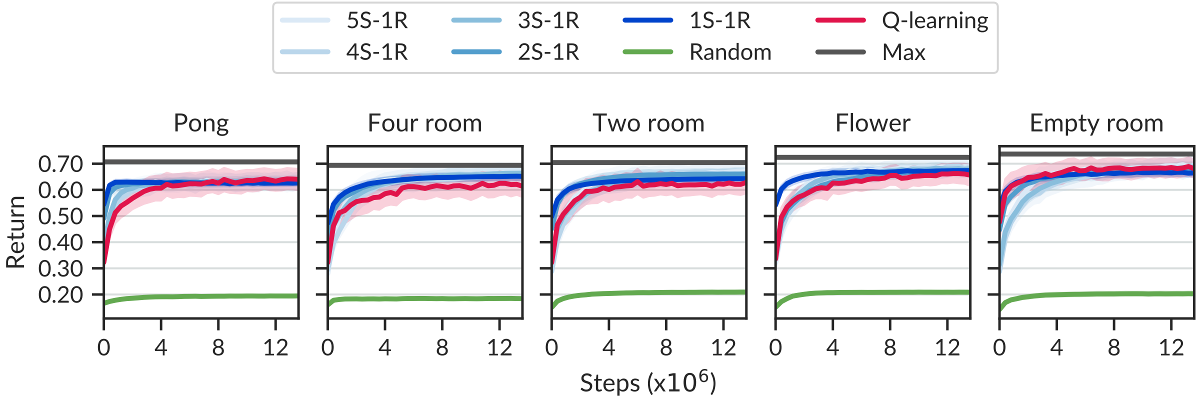


Figure 6.3: Mean training return curves for different sender-receiver agent configurations with baseline comparisons (Q-learning, Random and Theoretical maximum. Reproduced from Kajić et al. (2020, Figure 3).

during training for different sender-receiver configurations, for the first 12 million steps to highlight convergence trends. The highest possible return for each environment is indicated by a gray line, and is calculated as the average of all possible discounted rewards in the given environment, when using the optimal policy (i.e., the shortest path to each location from the centre). The figure also shows comparisons with two additional baselines: a single, non-communicating Q-learning agent that sees the goal, as well as a random baseline, which consists of a sender emitting random messages. In the latter case, the receiver's optimal strategy is to treat the messages as noise, and learn to visit every possible location in the environment in search for the goal.

Based on the training curves, we make two major observations. First, communicating agents are able to successfully solve this task, as shown by the learning curve approaching the theoretical maximum value. The performance of communicating agents is comparable to that of a single Q-learning agent. Second, we observe that different combinations of communicating agents, and in particular the configuration consisting of a single sender and a single receiver, often display faster convergence than the single Q-learning agent. This is apparent in all environments, except in the *Empty room* environment where they are comparable. We note that this could also be due to the lack of extensive tuning of neural network hyperparameters, such as the number of units in the hidden layer or the regularization factors.

From our experiments, it appears that the single Q-learning agent was more affected

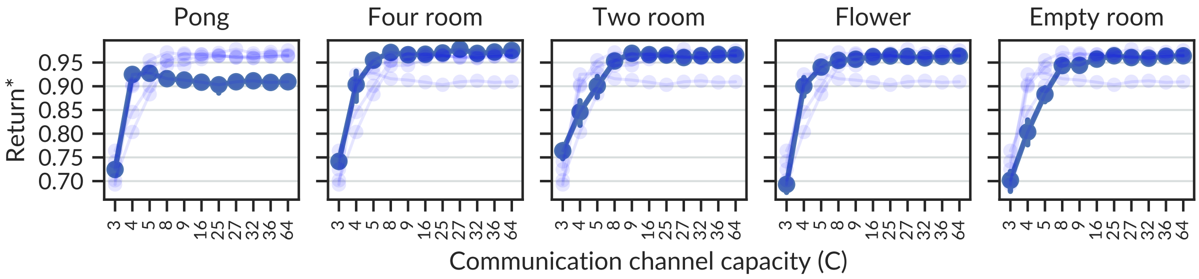


Figure 6.4: Increase in average return depending on the channel capacity $C$, defined as the total number of messages $C = N^M$ in a setup. Each plots shows all environments, yet one is emphasised for the comparison with others. This is a categorical plot and 95% bootstrapped confidence intervals are shown as bars which are not always visible. Abbreviations: Return*=Return normalized by theoretical maximum. Reproduced from Kajić et al. (2020, Figure 3).

by the lack of hyperparameter tuning, but our goal was not to examine the conditions under which the agents perform optimally on this task. Instead, we are interested in the coordination problem when two or more agents need to cooperate to solve the task, given individual restrictions on the action space. The task is purposefully designed to be sufficiently simple to yield an interpretable solution to the coordination problem that allows us to understand the influence of different environmental pressures on communicative success.

Finally, we note two reasons why the training curves do not reach the theoretical maximum. First, these were generated in the training regime, which contains some level of random behaviour as determined by the $\epsilon$-greedy strategy. Second, these curves are averaged over all communication channel capacities, and for some of them the task cannot be solved optimally with the Q-learning algorithm (discussed later).

### 6.3.1 Communication Channel Capacity

The capacity of the communication channel, $C$, is defined as the total number of messages used in an agent setup, and is computed as $C = N^M$. Recall, $N$ was defined as the number of messages available to a sender, and $M$ as the number of senders in a setup. For example, $C = 16$ corresponds to the following setups: 1 sender with 16 messages, 2 senders with 4 messages each, or 4 senders with 2 messages each. We examine the relationship between the channel capacity and the average return normalized by the theoretical maximum.

We know that the agents should perform well on the task if the size of the communication channel is the same as the number of non-occupied locations in the environment. In this case, every message can be used to uniquely identify one location. However, we are particularly interested in understanding the meaning of messages when the size of the channel is small compared to the number of possible goal locations. For example, if there are only 3 or 4 messages available, the sender needs to compress information about goal locations and use a single message to convey information about several goal locations.

Figure 6.4 shows normalized returns for different sizes of the communication channel, averaged over all agent setups. A single point on a curve is an average return computed over all sender-receiver agent configurations that use the total number of messages that is indicated by the corresponding channel capacity $C$ on the $x$-axis. In all environments, we observe two distinct curve regions: a linear increase in the performance with every message added to the communication channel, and a plateau where adding more messages does not improve performance. These regions are most apparent in the *Empty room* and *Two room* environments, where we see an increase in performance up to approximately 8 or 9 messages. For the *Pong* environment, the peak is reached at about 4 messages, and there is a slight drop in performance when 8 or more messages are available. We speculate that the restrictive features of the *Pong* environment make the task more difficult when agents have an abundance of messages available, although it is not entirely clear how.

The point after which we observe little to no improvement is different for each environment, and it approximately corresponds to the smallest number of messages needed to encode shortest paths to all possible goal locations. For example, in the *Pong* environment 4 such paths exist, so when the goal is located anywhere on one of the paths, the agent is guaranteed to reach the goal using the fewest steps possible, if it knows which path to take. For all other environments, there are 8 such paths. The lower panel in Figure 6.2 shows examples of such optimal paths for each environment. Such path configurations may not be unique, as for some environments there are multiple shortest paths to each goal location. In this context, we refer to a *path* or a *route* as a contiguous sequence of the same message label in an environment. Such paths are also colour-coded in Fig. 6.2 for redundancy. Section 6.4.1 on solution optimality provides a detailed explanation on how the smallest number of messages needed for an optimal solution is found. Thus, while having 8 (or 4 for *Pong*) messages requires agents to compress information about goal locations, this compression still allows them to optimally solve the task, and can be seen as a form of *lossless* compression.

Table 6.2: Spearman's rank correlation coefficients $\rho$ between communication channel capacity and the performance in the subcapacity regime.

| $p_{term}$ | .10 | .20 | .30 |
|---|---|---|---|
| Empty room | 0.89 | 0.92 | 0.93 |
| Flower | 0.89 | 0.85 | 0.84 |
| Four room | 0.85 | 0.93 | 0.90 |
| Pong | 0.65 | 0.71 | 0.65 |
| Two room | 0.92 | 0.93 | 0.92 |

All $p < .001$.

### Subcapacity and Supracapacity Regime

The subcapacity regime is defined as 5 or fewer messages in total for *Pong*, and 9 or fewer messages for all other environments. Approximately, these thresholds correspond to the channel capacity regions in which agents need to use lossy compression (subcapacity) or not (supracapacity). In the subcapacity regime, increasing the capacity of the communication channel by adding an additional message strongly correlates with the improvement in performance for all environments, as shown in Table 6.2.

We also investigate the relationship between the environment structure and the normalized performance. While there are several possible interpretations of the meaning of "environment structure", here, we consider structure as a measure of the uniqueness of the optimal receiver's policies. We quantify it as the inverse of the total number of shortest paths to each location in the environment. The number of shortest paths for each location can be calculated iteratively, since only steps "forward" (i.e., starting from the centre towards a corner of the map) are allowed. Then, the number of shortest steps needed to reach one location can be expressed using a recurrence relation, based on the number of steps needed to reach locations that lead to that one.

Following this procedure, we obtain the following values: $\frac{1}{14}$ for *Pong*, $\frac{1}{22}$ for *Four room*, $\frac{1}{32}$ for *Two room*, $\frac{1}{44}$ for *Flower* and $\frac{1}{64}$ for *Empty room*. According to this definition, *Pong* is the environment with most structure, and *Empty room* has the least structure. This measure is also related to the amount of empty space in an environment, as environments with more empty space are less structured. To examine the relationship between environment structure and performance, we first encode layouts as categorical variables by enumerating them, starting from the layout with the least structure, up to the layout with the most structure. Spearman's rank correlation coefficients are computed between

Table 6.3: Spearman's rank correlation coefficients between the environment structure and the average normalized return. Abbreviations: S=Sender, R=Receiver.

| | | | $p_{term}$ | |
|---|---|---|---|---|
| | | .10 | .20 | .30 |
| | Agent Setup | | | |
| Subcapacity (low) | | | | |
| ($C = 3, 4$)[1] | 1S-1R | 0.030 | 0.311** | 0.462*** |
| | 2S-1R | 0.397* | 0.465** | 0.640*** |
| Subcapacity (medium) | | | | |
| ($C = 5, 8, 9$) | 1S-1R | -0.047 | -0.202* | -0.230** |
| | 2S-1R | -0.194 | -0.447*** | -0.387** |
| | 3S-1R | 0.531*** | 0.581*** | 0.610*** |
| Supracapacity | | | | |
| ($C > 9$) | 1S-1R | -0.437*** | -0.656*** | -0.636*** |
| | 2S-1R | -0.626*** | -0.533*** | -0.534*** |
| | 3S-1R | -0.581*** | -0.635*** | -0.654*** |
| | 4S-1R | -0.418** | -0.494*** | -0.782*** |
| | 5S-1R | -0.414** | -0.381** | -0.588*** |

[3] $C = 3$ for Subcapacity (low), $C = 4$ for Subcapacity (medium) and $C > 6$ for Supracapacity for *Pong*, see text for details. $^{*}p < .05$. $^{**}p < .01$. $^{***}p < .001$.

the layout variable (ordinal) and the normalized return (continuous) and reported in Table 6.3.

We find some evidence that agent pairs in the subcapacity regime, specifically in the low subcapacity regime of $C = 3, 4$ and $p_{term} > 0.1$ achieve higher normalized returns in environments with more structure. In this case, the structure is helpful insofar as walls in an environment restrict possible paths for the receiver, making it easier to solve the navigation problem. This effect reverses when more messages are added to the communication channel, so that there is a negative correlation between performance and structure. We speculate that while having more choices (i.e., messages) might be beneficial for layouts with more empty space, the same can hinder learning in environments with fewer empty locations.

### 6.3.2 Emergent Communication Protocol

In this section, we analyze agents' policies in order to qualitatively characterize the learned communication protocol. To this end, we show a few selected examples from different experiments to interpret and highlight aspects of meaning of learned messages. Later, in Section 6.4.1 on optimality of solutions, we quantify effects and observations made in this section, and show that observations which follow from the examples shown here are not "cherry-picked", instead, they are indeed consistent across experiments.

First, we examine what kind of information is conveyed by senders' messages by manipulating goal locations, and observing what effect the manipulation has on the emitted messages. The manipulation consists of placing the goal at all possible locations in an environment. We also analyze the receiver's policy by manipulating messages and observing the resulting navigation trajectories in the environment. Therefore, we will be predominantly concerned with the following two questions:

1. What message does the sender choose for goal *(x, y)*?
2. Where does the receiver go if it receives a message *m* while at location *(x, y)*?

To address the first question, we repeat the process depicted in Figure 6.1A by manually setting goal locations to all possible $(x, y)$ locations. From a trained sender-receiver experiment, we use the sender's network, and probe it with a one-hot encoded vector that represents a single goal location *(x, y)*. We then use greedy action selection ($\epsilon = 0$) to select a message at the output. This process is repeated for all goal locations for an environment, thus obtaining a single message for each non-occupied location in that environment. Then, to obtain the same display as in the lower panel in Figure 6.2, locations with the same message are colour-coded, yielding a single plot in Figure 6.5.

The figure shows message distributions from 25 different experiments, each one corresponding to a different 1S-1R setup in all environments, and for different channel capacity sizes. The centre, corresponding to the location $(2, 2)$, is left blank in all plots, as the starting position of the receiver can never be a goal location. The figure shows that messages in the subcapacity regime, $C = 3, 4, 9$, are typically assigned in such a way to cover a region in space, or a path. The effects of lossy compression are particularly noticeable when $C = 3, 4$—in those cases, a single message is used to encode multiple goal locations. Such regions are also interpretable. For example, when $C = 3$ in the *Pong* environment, a single message encodes *left of the centre*, and in the *Two room* environment, a single message encodes *right of the centre*. $C = 3$ is a particularly interesting case for

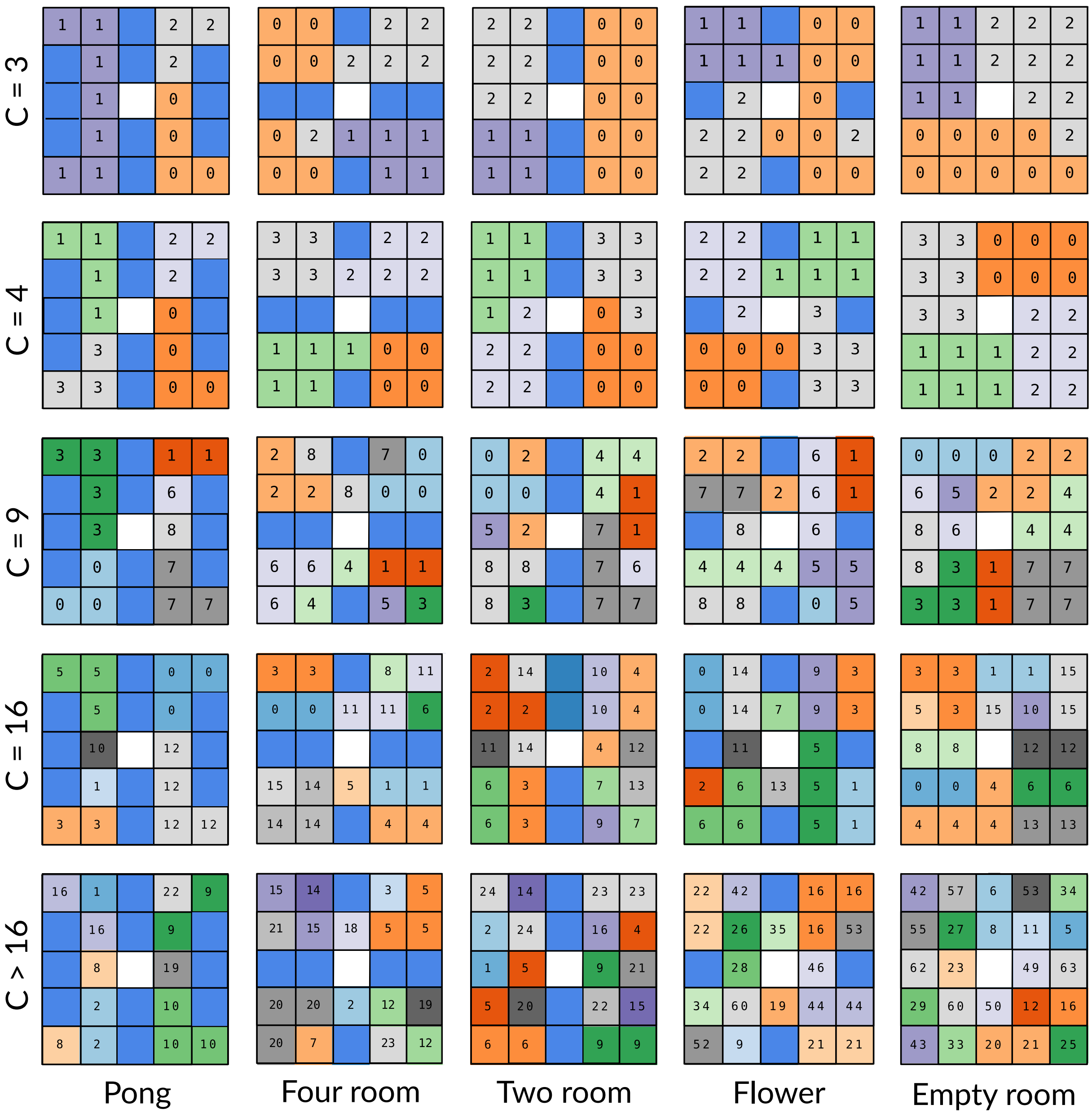


Figure 6.5: 25 examples of different sender policies from different experiments including one sender and one receiver. Channel capacity sizes (C) are shown in rows, and different layouts in columns.

a Q-learning agent in some environments, as the agent bases its next action only on its current location and the message. As such, it cannot learn a strategy where it "goes back" to a previously visited location and select a different action in that state. For example, once the agent enters the left corridor in *Pong* ($C = 3$) upon seeing the message "1", it learns to navigate either up or down in that corridor. It is only due to random action selection that it visits the other part of that corridor. Due to this restriction, agents in this setup can never perform optimally, in a sense that they will never use the shortest possible path to all possible goal locations. Finer spatial interpretations emerge with $C = 4$, where message clusters encode regions that can be described as individual rooms, such as those in *Four room*, or equal and symmetrical areas of space, as those in *Flower* $(C = 4)$ or *Empty room* $(C = 4)$.

As we increase the size of the capacity channel, we observe that messages start encoding shorter paths to all possible goal locations, as seen for $C = 9$ in Figure 6.5. Adding messages shortens the average path length that the receiver takes to reach the goal, which explains why we see such improvements in the subcapacity regime in Figure 6.3. In most cases in Figure 6.5 ($C = 9$), a single message is used to signal goal locations lying on the same shortest path, but we also observe cases where a single message is used to encode a single goal location (e.g., "7" and "3" in *Four room*, or "3", "5" and "6" in *Two room*).

This effect becomes even more prominent as we further increase the number of messages in the communication channel. The paths tend to "disintegrate" into several ones, such that a single message encodes fewer locations, as observed with $C > 16$. In this regime, there is a strong tendency to use a single message to encode a single location. For example, the *Empty room* setup is shown for $C = 64$ in Figure 6.5, where each location is uniquely encoded with a distinct message. There are other interesting path formations, such as discontinuous paths labelled with messages "15" or "12" in *Four room* $(C > 16)$.

Next, we examine receiver policies by answering the second question above: Where does the receiver go if it receives a message *m* while at location *(x, y)*? To answer this question, we repeat the process shown in Figure 6.1B by setting $\epsilon = 0$. For each message and for each location we observe one of the four possible answers corresponding to the movements: up, down, left and right. In the subcapacity regime where information is compressed so that a single message indicates multiple locations, we observe that the receiver adopts a "sweeping" strategy. Figure 6.6 shows the receiver's policy for the sender's policy *Empty room* $(C = 4)$ in Figure 6.5. For each message in that setup, the receiver's greedy actions for each location *(x, y)* are depicted as arrows. That is, a single arrow at a location answers the question we asked above. Iteratively answering the question, starting from the centre of the map, yields a trajectory in the space.

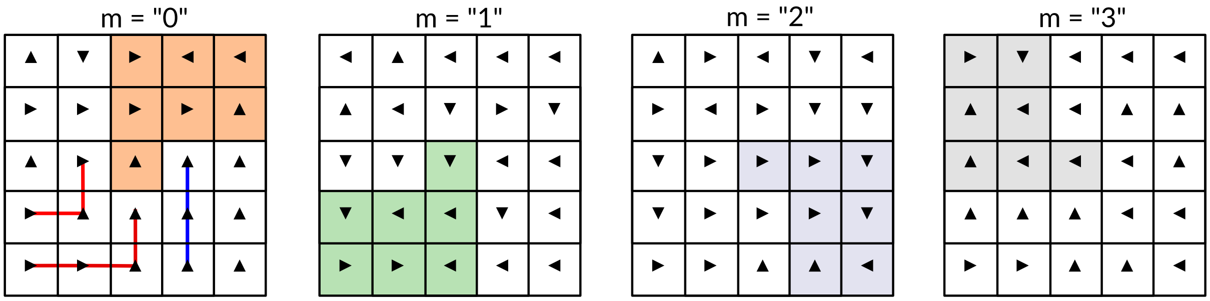


Figure 6.6: Receiver's policy: Navigation trajectories as actions with the highest Q-value for each location and each message $m$ for *Empty room* $(C = 4)$. Red and blue lines show trajectories outside the region encoded by the message that can also lead to the goal location. Reproduced from Kajić et al. (2020, Figure 5).

Highlighted regions in individual subplots in Figure 6.6 are those where the sender would emit the same message, provided that the goal is located at any of the locations in that region. Starting at the centre of the map and following the trajectory in each plot, we notice that such a trajectory visits each highlighted location only once, before looping back to a previously visited location. The path defined by that trajectory is a Hamiltonian path, and therefore an optimal one for this specific case in the subcapacity regime. This is optimal behaviour because each location is visited exactly once using the shortest path *possible* given the constraints (i.e., the communication channel capacity). We emphasize that the shortest path *given these constraints* is not the same as the shortest path without such constraints. For example, let us consider the plot labelled with m="0" in Figure 6.6. If the sender used a unique message for every location in this environment (e.g., $C = 25$), the receiver could reach the location $(0, 2)$ in just two steps, instead of 6 steps (as shown in the plot for $C = 4$).

A few trajectories outside the highlighted regions in Figure 6.6 show effects of $\epsilon$-greedy exploration strategy. Red trajectories originate from locations that do not lie on the Hamiltonian path, but eventually lead the agent to the region encoded by the current message that contains the reward. The blue trajectory leads the agent to the correct region of space where the goal is located, but the agent might still fail to find the goal, if the goal is located at either of the two locations closest to the centre. Due to randomness in exploration, when the receiver finds itself in a random location, it partially learns to recover from the deviation. This effect is somewhat consistent across experiment, although it is not present in all simulations (i.e., not all trajectories outside of highlighted regions will lead to the goal location).

### 6.3.3 Multi-sender Agent Setup

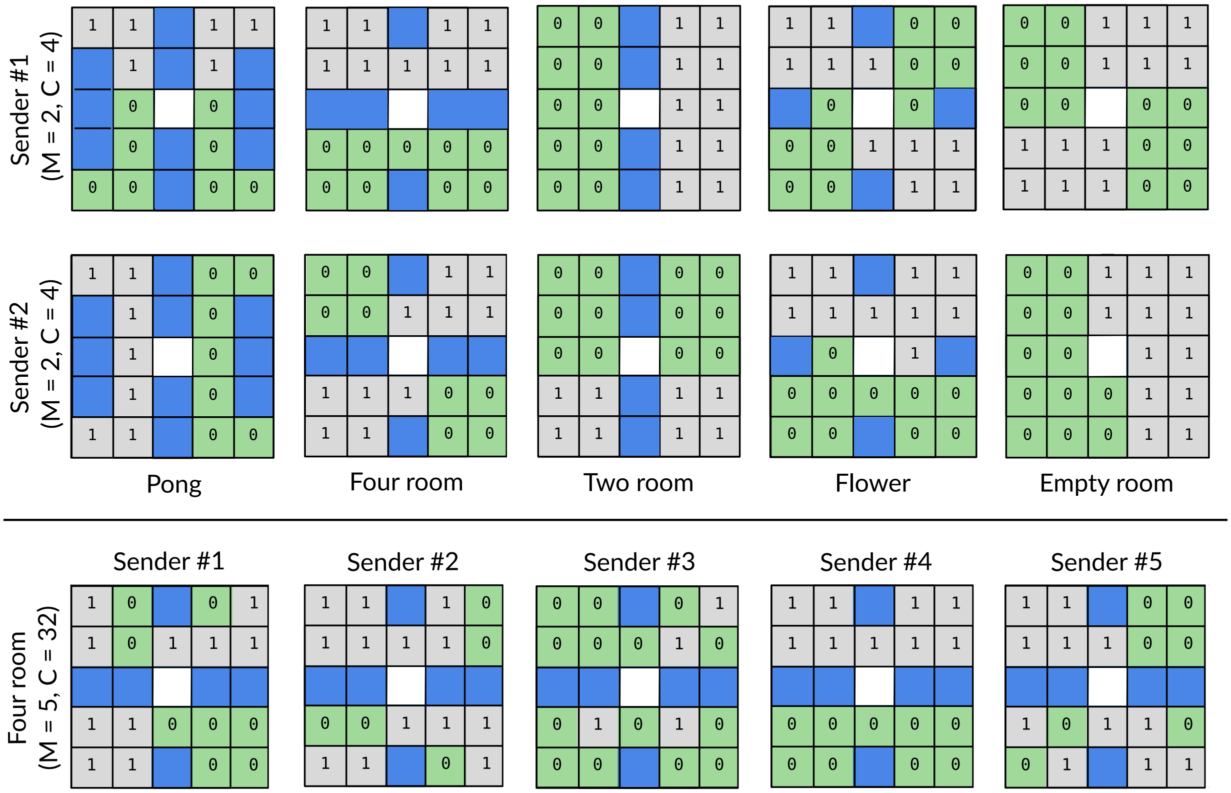


Figure 6.7: **Upper panel**: An example sender policies for five different experiments consisting of two senders and one receiver (2S-1R). **Lower panel**: An example sender policies for a single experiment in the *Four room* environment with five senders and one receiver (5S-1R).

In this section, we take a closer look at learning behaviours in experimental setups with two or more sender agents. In particular, we examine properties of the communication protocol when information about the goal location is encoded using multiple messages. We investigate and characterize the relationship between messages emitted by different senders, and discuss it in the context of task performance, as well as in relation to some core properties of language.

Previously, for setups with a single sender, each $(x, y)$ location in the gridworld was described by a single message $m$. With multiple senders, each emitting one message, a location is now described by an array $[m_1, m_2, ..., m_M]$, where each symbol $m_i$ in the array is the message emitted by the $i$-th sender. It is worth noting that this representation is not quite a sequence (as one might expect when referring to the outputs of an RNN, for example), as there is no inherent ordered relationship in this representation. More precisely, it is a network design decision to "stack" neural networks vertically to represent receiver's input (as depicted in Figure 6.1B), and this design decision ties meaning of each specific message to its position. Another consequence of this design decision is that messages are unrelated, since every sender has its own vocabulary. For example, a message might be $[1, 1]$, denoting that both senders emitted a "1" when the reward is located at some particular position, but since those are independent symbols, there is no inherent relationship between those two symbols. That is, we could have decided to encode the message as $[A, 1]$, $[1, A]$, or $[a, A]$, since each sender's vocabulary contains mutually exclusive items. A message can be meaningfully interpreted as long as we know how each symbol in the message maps to a corresponding sender's vocabulary. Therefore the order matters only insofar that each symbol can be matched to a corresponding sender.

As previously done with sender-receiver setups with one sender, we first discuss senders' policies in multi-sender setups based on their qualitative properties. A few selected sender policies for multi-agent setups are shown in Figure 6.7. The upper panel shows results from five different experiments, one conducted in each environment. A single experiment consists of a (Sender #1, Sender #2) plot, since each sender selects a message for every location in an environment. The communication channel capacity size in each experiment is $C = 4$, as each sender's vocabulary $V$ contains two messages ($V \in \{0, 1\}$). The lower panel shows sender policies in one experiment in the *Four room* environment with five senders and two messages, thus having the communication channel capacity $C = 2^5 = 32$.

Looking at the selected examples shown in the upper figure panel, we observe a highly coordinated, regular, and interpretable pattern of message allocations. For instance, if the first sender allocates messages by partitioning the space according to one axis, the second sender does the same with a different, possibly orthogonal axis. For example, in the *Pong* environment, Sender #1 partitions the space according to the *y*-axis, using the message "0" to denote "down" and "1" to denote "up". Sender #2 then partitions the space according to the *x*-axis, by assigning message "1" to all goal locations to the left of the centre, and "0" to all goal locations to the right of the centre. Similar partitioning patterns are observed in other environments. For example, Sender #2 in *Four room*, as well as Sender #1 in *Flower*, partition the space diagonally. One feature of the partitioning schemes in examples shown is that each sender allocates one message for about the half of all locations, and another message for the other half. Although we enumerate senders for easier identification, we note that there is no training requirement that imposes any kind of order in message assignment process. Therefore, independent senders coordinate the assignment of messages to goal locations, even though they do not directly have access to information about each other's messages. The only signal impacting this coordination process is the joint reward $r$ that is of the same value for all of senders. In later sections, we analyze the optimality of the solutions and show that solutions are highly consistent with each other in terms of their geometry, as well as the behaviours they afford to the receiver.

If we interpret the representation $[m_1, m_2, ..., m_M]$ as a single message $\mathcal{M}$ consisting of individual symbols $m_1, m_2, ...$, then, we argue that the meaning of $\mathcal{M}$ is determined by the meaning of its constituents. More specifically, a location or a region in space that is described by $\mathcal{M}$ derives its meaning from individual spatial descriptions afforded by each individual $m_i$. In linguistics, cognitive sciences and psychology, the type of relationship where the meaning of complex expression is determined by the structure and the meaning of its constituents is known as compositional structure (Fodor and Pylyshyn 1988; Jackendoff 2002). The representations we use here are constrained in such a way that they are unable to capture the complexity of structure in natural language. As such, the aspect of compositional structure observed here is rather limited in comparison to that attributed to natural language. As well, the notion of compositionality in language is often discussed in the context of language *unboundedness*, as given the finite number of words, it is possible to understand an effectively infinite number of word combinations. This property is known as *productivity*, and it gives language its expressive power. In our case, where messages are created by concatenating individual symbols into arrays, productivity is limited as the length of the message is bounded by number of elements in an array, and by the number of possible identities of those elements. This is a consequence of a design decision that could be alleviated in different ways, for example, by binding fillers to their roles and collecting them together, as done in the SPA.

### Quantitative Analysis of Multi-sender Performance

For different multi-sender setups we examine the effects of *message redundancy* and *sender redundancy*. Message redundancy refers to the property where unused messages carry some information that is useful for the receiver needing to reach the goal. An unused message is defined with respect to a location, and it is a message in a sender's vocabulary that is not used to encode that specific location, as it has a lower Q-value than the dominant message. For example, in Figure 6.1A such unused messages would be those represented by the orange and green output nodes for the sender's neural network for the location $(0, 0)$. Thus, if there is any redundancy present, when using such a non-dominant message, there should be an increase in performance compared to a random baseline. To investigate message redundancy, we only consider messages with the second highest Q-value. To evaluate the performance, we re-run experiments in the test setting with the following modification: we replace one message in the sequence emitted by senders with a message with the second highest Q-value. In a test setting, learning is disabled for both agents, as well as the exploration ($\epsilon = 0$). In addition, there is a timeout of 10 steps, which means that if the agent does not reach the reward within that number of steps, the episode terminates with 0 reward. This modification is performed for each message in an array of messages, such that for one sender-receiver setup consisting of $M$ senders, we get $M$ modifications. For each modification, we calculate the average return by simulating 1,000 test episodes of the task. Then, we calculate the performance drop in average return with respect to the non-modified version of the experiment. We also include a random baseline condition, where the message sequence in the communication channel is intercepted and replaced with all random messages.

Similarly, we define *sender redundancy* as the relative impact of individual senders on the task performance and evaluate it by replacing a message in the message array with a random, different message in each episode. By doing this, we are effectively replacing that sender with a random sender, and study how well the receiver performs in such a condition. We examine the performance in a similar way to message redundancy, by letting trained agents perform the task for 1,000 episodes ($\epsilon = 0$), replacing a single message out of $M$ messages in a way just described. This is also done iteratively for each message in a single experiment, and we calculate the average return after completing all episodes.

The results of message redundancy and sender redundancy evaluations, including the random baseline, are shown in Figure 6.8. The results are shown as drops in performance compared to the learned behaviour without any interventions to the communication channel. Overall, we see that setups with more senders are more robust, as evidenced by smaller performance drops. But even in those cases, the performance drops are still substantial, ranging from about 55% (for 5S-1R setups) to about 70% (for 3S-1R setups). There is a weakly significant difference ($p < 0.01$) in performance drops between sender redundancy and message redundancy conditions for the 2S-1R setup. In this case, the performance drop is about 2% smaller for the message redundancy condition, meaning that using the message with the second highest Q-value is better than using a random message. This difference is not significant for 3S-1R setup, and it is non-existent in other cases. These two conditions are exactly the same for 4S-1R and 5S-1R setups:

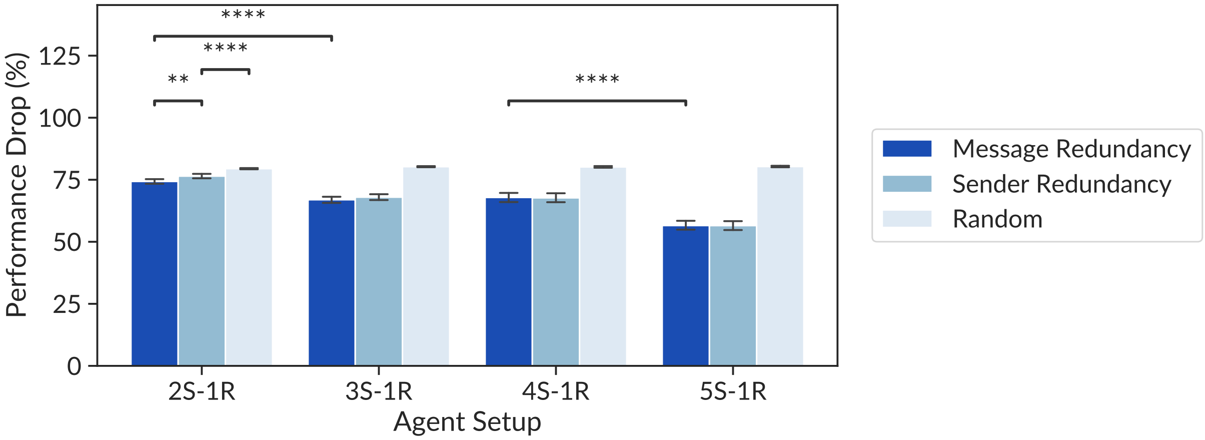


Figure 6.8: Average drop in task performance caused by modifying the message sequence, with a random message (sender redundancy) or the message with the second highest Q-value (message redundancy). 95% bootstrapped confidence intervals are shown. **: $p \leq 0.01$, ****: $p \leq 0.0001$.

sender agents in those two setups have only two messages in their vocabularies, and in both setups replacing the dominant message corresponds to picking the other available message. Both drops are significantly ($p < 0.0001$) smaller than the performance drop observed with a random baseline. We conclude that only 2S-1R setup exhibits weak redundancy in a non-dominant message.

While these results show that interfering with any message in an array of messages has severe effects on the task performance, they reveal little about the relative contribution of individual messages. That is, they do not provide any information on the impact of individual senders on overall task performance. We perform an additional set of experiments to investigate whether messages from all senders contribute equally to the receiver's ability to solve the task. If all messages are equally important, we expect to see an equal drop in task performance when we scramble each message separately, leaving other messages unmodified. To test this hypothesis, we plot the results for all 5S-1R trials, where we sort performances by the amounts of their drop compared to setups with no modifications, in descending order. Thus, for one 5S-1R pair we get five data points, corresponding to five returns, where each return was computed over 1,000 episodes with one randomized message. The randomized message in this particular case is the other, non-dominant message since in this setup each sender has only two items in the vocabulary. We compare those returns to the baseline, which is the performance of trained

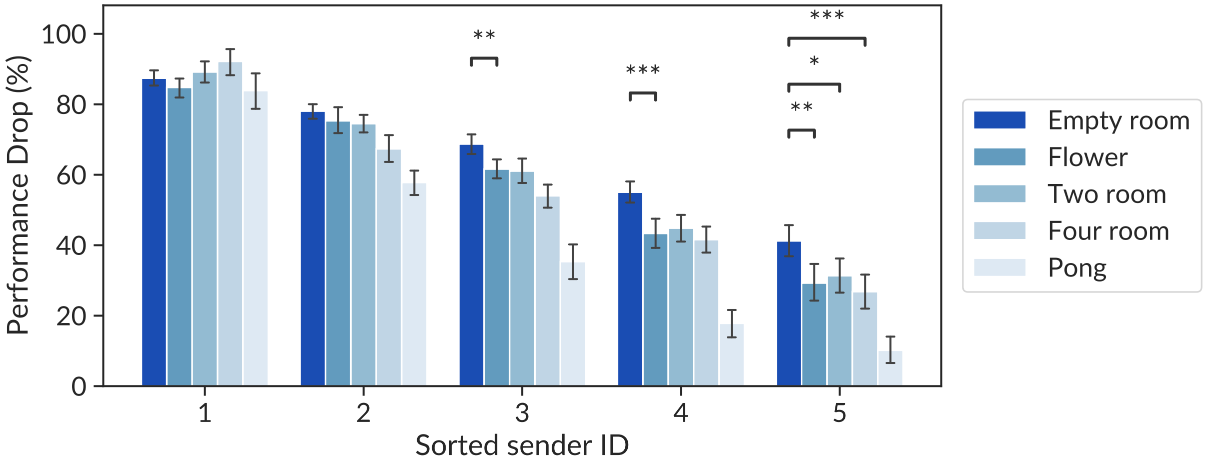


Figure 6.9: Average drop in task performance caused by scrambling each sender's message in 5S-1R setups. 95% bootstrapped confidence intervals are shown. *: $p \leq 0.05$, **: $p \leq 0.01$, ***: $p \leq 0.001$, ****: $p \leq 0.0001$.

agents without any modification to the communication channel, and report performance drops.

The results shown in Figure 6.9 display a gradual decrease in performance drops with each subsequent sender. We can therefore reject the hypothesis that each sender contributes equally, as interfering with a message from some senders will result in higher drops than interfering with messages from other senders. Scrambling the message from a single sender can cause a performance drop anywhere from 10% to 92%. To some extent, this phenomenon appears to be environment specific. For example, the drop in performance is consistently smallest for *Pong* for the second or higher sender ID, and it consistently remains highest for *Empty room*. This gradual decrease in drops observed with "less important" senders can be explained by the saturation of the communication channel. Once the goal location has been uniquely identified with a few messages, adding more messages does not provide any new information.

To better understand the differences in performance drops between different senders, we examined more closely setups in which they occur. One such setup is shown in the lower panel in Figure 6.7. In this setup, the biggest drop in performance (100%) is observed when scrambling the message emitted by Sender #4, which partitions the space according to the y-axis. In this particular case, the message emitted by this sender can be considered as "the most significant bit", since toggling it will direct the receiver to the wrong side of the map, where it will not have a chance to find the reward. The second highest drop (58%) in this setup is found when scrambling the message sent by Sender #1, and the smallest (45%) for Sender #3. The same pattern is observed in other experiments where a single sender uses its messages to partition the space in a way similar to Sender #4. Other assignments of messages such that each message is assigned to the half of all possible locations, and the other message to the other half does not yield a 100% drop in performance. It is unclear what are the benefits of having a single dominant sender in this setup, and we speculate that this phenomenon might emerge due to its ease of learning. However, further analysis of the training data is needed to obtain more conclusive results on learning outcomes.

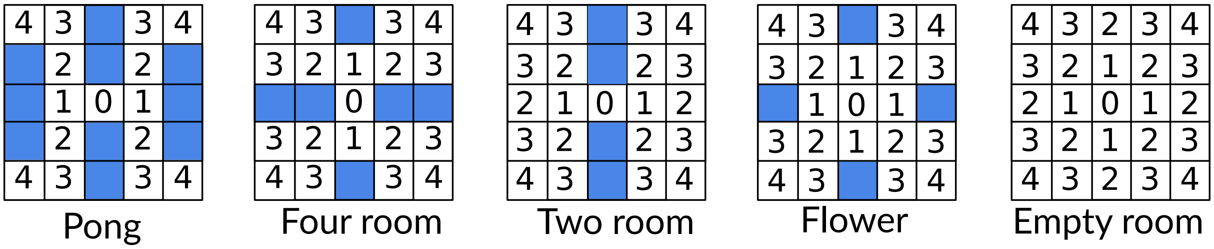
## 6.4 Optimality of the Communication Protocol

In this section, we perform a quantitative analysis of sender's policies that we have qualitatively analyzed in the previous section. While previous analyses were useful in establishing the relationship between messages and features in the environment, we only briefly discussed the notion of *optimality* of policies defining message distributions. In what follows, we will refer to a learned sender's policy as a *solution* to the navigation problem discussed in this chapter. Then, we discuss *optimal solutions* in the context of experimental variables, and propose a cost measure to quantify the distance between a solution and an optimal one. This allows us to directly compare different policies and discuss which ones are "better" in the context of the introduced cost measure.

### 6.4.1 Solution Optimality

For a given number of messages and an environment, we define a problem *solution* as the distribution of messages over all possible goal locations in the environment. As mentioned previously, a solution is a policy learned by the sender, since the distribution of messages corresponds to its actions. For example, each plot in Figure 6.5 is one solution, and so is each pair of (Sender #1, Sender #2) plots in the upper panel of Figure 6.7. An *optimal solution* allows the receiver to find the goal state using the fewest steps possible.[2] In other words, if the receiver is given a message $m$ from an optimal solution, then it is guaranteed to reach the location encoded by $m$ as fast as possible. Equivalently, such a solution maximizes the reward on the task. Intuitively, we can think of such a message as

[2] Assuming the receiver is behaving optimally.



Pong

| | | | | |
|---|---|---|---|---|
| 4 | 3 | 🟦 | 3 | 4 |
| 🟦 | 2 | 🟦 | 2 | 🟦 |
| 🟦 | 1 | 0 | 1 | 🟦 |
| 🟦 | 2 | 🟦 | 2 | 🟦 |
| 4 | 3 | 🟦 | 3 | 4 |

Four room

| | | | | |
|---|---|---|---|---|
| 4 | 3 | 🟦 | 3 | 4 |
| 3 | 2 | 1 | 2 | 3 |
| 🟦 | 🟦 | 0 | 🟦 | 🟦 |
| 3 | 2 | 1 | 2 | 3 |
| 4 | 3 | 🟦 | 3 | 4 |

Two room

| | | | | |
|---|---|---|---|---|
| 4 | 3 | 🟦 | 3 | 4 |
| 3 | 2 | 🟦 | 2 | 3 |
| 2 | 1 | 0 | 1 | 2 |
| 3 | 2 | 🟦 | 2 | 3 |
| 4 | 3 | 🟦 | 3 | 4 |

Flower

| | | | | |
|---|---|---|---|---|
| 4 | 3 | 🟦 | 3 | 4 |
| 3 | 2 | 1 | 2 | 3 |
| 🟦 | 1 | 0 | 1 | 🟦 |
| 3 | 2 | 1 | 2 | 3 |
| 4 | 3 | 🟦 | 3 | 4 |

Empty room

| | | | | |
|---|---|---|---|---|
| 4 | 3 | 2 | 3 | 4 |
| 3 | 2 | 1 | 2 | 3 |
| 2 | 1 | 0 | 1 | 2 |
| 3 | 2 | 1 | 2 | 3 |
| 4 | 3 | 2 | 3 | 4 |

Figure 6.10: Shortest path lengths expressed as the smallest number of steps needed to reach each location in an environment using BFS and starting from the centre of the map.

instructions we would get when we are visiting a city as tourists and asking for directions to reach our point of interest, such as a museum. Assuming that the shortest path to the museum will be the fastest path, optimal directions to the museum will describe that path.

To find the smallest number of steps needed to reach each location in a gridworld environment we use the breadth-first search (BFS) algorithm. To apply the BFS, the gridworld environment is represented as a graph, with nodes in the graph representing 2D locations and edges representing transitions between adjacent locations. Weights of all edges are set to be equal to one. The node representing the location at the centre of the map, i.e., $(2, 2)$, is the designated starting node for the graph traversal. Figure 6.10 shows the smallest numbers of steps needed to reach each location, computed using BFS in each environment.

Knowing such smallest number of steps from the centre to each location provides some clues about the shortest paths between the two locations. For example, on such a path, if $n$ steps are needed to access one location, then $n + 1$ steps will be needed to access the subsequent location on the path, where $n$ is the number of steps computed with BFS. This also means that there are multiple shortest paths possible for some pairs of locations. For example, with the exception of *Pong*, reaching corners of the gridworld in all environments can be achieved via multiple shortest paths. Since an optimal solution implies that every location is visited using the shortest path available, the agent is not allowed to “go back”. That is, it should not visit a location it visited previously on the same path. Therefore, for this set of environments, a path consists of a sequence of steps where each step brings the agent one step closer to the reward location.

Given the smallest number of steps to each location, we now consider what is the smallest number of shortest paths in each environment. Answering this question is analogous to finding the smallest number of messages that allows the receiver to reach every location as quickly as possible. In order to answer that question, we consider the distribution of steps shown in Figure 6.10. In each environment, we enumerate the step counts and consider the highest. For example, in *Empty room*, there are eight locations that can be reached in two steps. This implies that all such locations that can be reached within two steps should lie on eight different paths, since if such two locations would lie on the same path, they could not be visited using at most two steps. In the context of the navigation game agents are solving, this means we need at least *eight* distinct messages to be able to reach each location as quickly as possible. The same holds for *Four room*, *Two room* and *Flower*, while for *Pong* this can be done with *four* messages.

For all environments, there are multiple solutions satisfying the minimum message count constraint as discussed above, and one possible solution is shown for each environment in the lower panel in Figure 6.2. For instance, for *Empty room* we can see that with a small modification we can get another optimal solution, by extending the path labelled with message "4" to reach the upper right corner that is currently labelled with "6". What all such optimal solutions have in common is that they allow the receiving agent to reach each goal location using the fewest steps possible, as consistent with BFS-solutions shown in Figure 6.10.

### 6.4.2 Cost Measure to Quantify Optimality

Given the distribution of step counts for an environment, such as those shown in Figure 6.10, we can compute the cost $S_{total}$, the total number of steps needed to reach each location when starting from the centre. If the number of steps needed to reach the location $(x, y)$, starting from the centre, is $s_{xy}$, then

$$S_{total} = \sum_{i=0}^{5} \sum_{j=0}^{5} s_{ij}, \tag{6.5}$$

where $5$ is both the width and the height of the gridworld environment. $S_{total}$ values for the environments are: $60$ for *Empty room*, $54$ for *Two Room*, $52$ for *Flower*, $50$ for *Four Room*, and $38$ for *Pong*. These numbers correspond to the sums of all step counts shown in Figure 6.10 for each environment. In the following, we will use this measure to discuss the optimality of solutions learned by the agents.

We first discuss how to compute the cost $S_{total}$ for any solution found by agents, some of which contain disconnected paths or might not be optimal. Knowing the $S_{total}$ for any solution will allow us to express the distance between that solution and an optimal one. Before we do so, it is important to highlight and understand the diversity of learned solutions and how that impacts the total cost. For example, many solutions contain *disconnected* paths where the agent has to step over a location encoded with a message different from the one it is currently receiving. A few examples are present in Figure 6.5. For example, paths labelled with "4" and "8" for *Four room* and $C = 9$ are disconnected, and so is each path where a message is used to encode a single location (e.g., message "7" or "3" in the same plot). A disconnected path is an optimal one as long as all the locations on the path are reached in the smallest number of steps, as derived with the BFS.

We also observe solutions where Hamiltonian paths are not possible. This means that an agent has to go over a previously visited location to reach every possible goal location. One of the cases where this is most apparent is the *Pong* environment and three messages ($C = 3$), also depicted in Figure 6.5. Here, one message is assigned to the left half of the environment (e.g., message "1"), and two messages to the right part, splitting it into upper (message "2") and lower (message "0") segments. When finding the shortest path to all locations left of the centre, after visiting either upper or lower branch of the path encoded with the message "1", the path will need to trace back nodes it already visited. A Hamiltonian path is also not possible for the path labelled "0" in *Empty room* when $C = 3$. When a Hamiltonian path is not possible, it means that the agent needs to make additional steps to reach every location in the environment, violating the requirements of an optimal solution.

To take such cases into account when computing $S_{total}$, we frame this problem as the Travelling salesman problem (TSP) for each message partition, where a partition refers to the locations encoded by the same message, and represented as a graph. Each node is a $(x, y)$ location and edges are distances between nodes, computed in a way that is explained below. The TSP formulation of the problem is then the following: given a set of nodes in a (possibly disconnected) partition and the distances between each pair of nodes, what is the shortest possible route[3] that visits each node? In our case, we do not require the route to return to the starting node as the typical formulation of TSP postulates. The distance for a pair of nodes is computed as the $L_1$ distance, which is the sum of the absolute differences of the Cartesian coordinates of those nodes. As such, the distance between the adjacent nodes is one, and for disconnected nodes such distance is always greater than one, implying that reaching such a node will require the route to go over a node belonging to a different partition. For an optimal solution, TSP will find a route that satisfies the shortest number of steps requirement as computed by BFS.

Even though TSP is a NP-hard problem, and it is not known whether a provably effi-

[3] In the context of our problem, we will use the term *path* and *route* interchangeably.

cient (e.g., polynomial-time) algorithm to solve it exists, we can use different approaches to solve it due to the small size of our environments. For example, we can rely on the brute force approach, or use heuristics to find a TSP route in a partition. Since most partitions have 10 nodes or fewer, to find a route in such cases, we test all possible permutations ($10! = 3{,}628{,}800$) of node orderings. A node ordering defines the sequence of node visits in a partition, and thus each node will be associated with a number of steps $s_{xy}$. For each node ordering we compute the associated cost $S_{total}$ and select the one with the lowest cost. While this approach guarantees to find the smallest $S_{total}$ as it searches exhaustively for the solution, if there are more than 10 nodes in a partition, the problem becomes computationally expensive. In such cases, we use the nearest neighbour algorithm for TSP path finding (Lawler 1985). This is a greedy strategy where the next node is selected based on the edge with the lowest value, however, this solution is not guaranteed to be optimal. As an attempt to improve the solution, we further use 2-opt algorithm (Croes 1958) that uses edge swapping to find a better solution given an existing one. This process allows us to compute $S_{total}$ for any solution, including those with disconnected paths and paths that are not Hamiltonian.

### 6.4.3 Best Suboptimal Solution in the Subcapacity Regime

For optimal solutions, the sum of steps $S_{total}$ will be equal to the one computed with BFS, as in those cases the route will be the shortest possible that visits each node once. In some cases where a Hamiltonian path through a partition does not exist, the route will visit some locations more than once, or it will include nodes belonging to another partition, which is manifested in the weight of that edge being greater than one. *Optimal solutions*, as defined above, are only possible if there are at least $8$ messages available in the communication channel ($4$ for *Pong*), as discussed above. If there are more messages available, then shortest paths can be split up in such a way that each message will encode only a segment of a path, which can be as short as a single location. Thus, if there are enough messages, a single message can be assigned to each location. This phenomenon is also observed empirically when agents have a large number of messages, such as for $C > 16$ in Figure 6.5. In contrast, in the subcapacity regime, the agents cannot find the optimal solution, according to the definition provided above. In the absence of a sufficient number of messages, the receiver will form a path that visits each location, but this will not be necessarily done using the shortest number of steps as found with BFS in Figure 6.10. In the subcapacity case, we can instead define the *best suboptimal* solution as one that the agents can find given the insufficient communication capacity.

The *best suboptimal* solution is therefore defined for a specific environment, and a specific number of messages in the subcapacity regime. Informally, we can observe two necessary conditions for such solutions. First, each path should be either of the same, or approximately the same length, so that they cover the area of the roughly same size in an environment. Second, when following such a path, each location should be visited only once. For example, the *Empty room* environment has 24 possible goal locations, and if there are 4 messages available, the agents should allocate each message to a region of space extending over 6 locations (such as the solution in Figure 6.5 for $C = 4$). If there are 3 messages available, the space can be partitioned into three regions with 8 locations each.

When the number of messages does not divide the number of empty locations without a remainder (e.g., 5 messages in *Empty Room*), finding the best suboptimal solution is more involved. For example, there might be several combinations of 5 regions of space (corresponding to five messages) that are approximately the same size (e.g., $[4, 5, 5, 5, 5]$, $[4, 4, 5, 5, 6]$, $[3, 5, 5, 5, 6]$). The sizes of such regions can be found by listing all combinations of path lengths that add to 24, where 24 is the number of empty locations in *Empty Room* (and similarly for other environments). Then, for each combination it is possible to compute the cost $S_{total}$. However, in this process so far we only guarantee path lengths, but not that paths should be Hamiltonian. For all lowest cost configurations we manually try combinations of paths that satisfy the length requirement and select the one with lowest $S_{total}$ that guarantees all paths are Hamiltonian. In this way, we can find the minimum cost for each best suboptimal solution, though we note that this is the best empirical guess at a solution.

### 6.4.4 Optimality Analyses

Figure 6.11 compares the cost $S_{total}$ of experimental solutions to the costs of theoretically optimal (or best suboptimal) solutions found with BFS. The costs of optimal and best suboptimal solutions are shown in straight dashed lines as they are fixed for an environment and the number of messages (shown in different colours). Solid lines are experimental results from all 1S-1R simulations (15 per environment/message setup), sorted in ascending order by the total cost of a solution. Where solid and dashed lines overlap, the agents have found an optimal, or a best suboptimal solution. The fewest optimal solutions are found with setups that have three messages (blue lines), since those show the largest discrepancy between the dashed and solid lines. This effect is smaller for four and five messages, but it is reduced, or almost entirely eliminated, when there are eight or more messages. In other words, it is harder to find the best suboptimal solution than it is to find an optimal solution. In particular, when there are three messages, agents

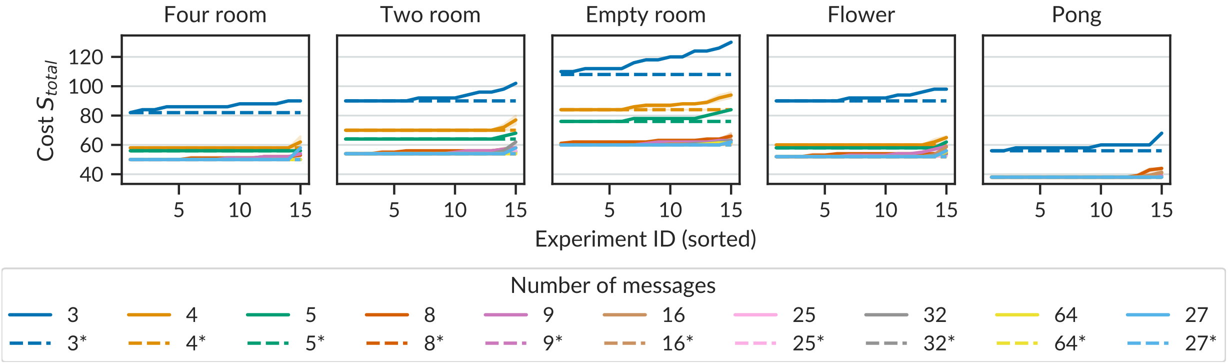


Figure 6.11: A comparison of costs for optimal and experimental solutions for all environments and different message counts. Cost associated with optimal solutions are represented with dashed lines, while solid lines show cost values for individual experimental solutions.

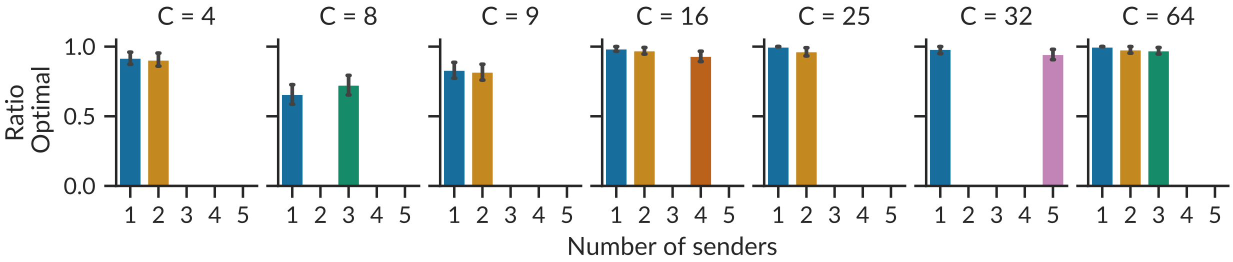


Figure 6.12: Proportion of optimal solutions in different multi-sender setups. Only feasible combinations of the number of senders and messages are shown.

find solutions with non-Hamiltonian paths that for the current implementation of the receiver (a Q-learning agent) can not be deterministically learned.

We also examine whether there is any significant difference in the number of optimal solutions found for multi-sender setups. The results for different capacity channel sizes are shown in Figure 6.12. Bars are only shown for setups that were possible and present in our experiments (for example, it is not possible to have a setup with 4 messages in total when using 3 senders). None of the differences were significant at the level $p < .05$, calculated using the t-test for unequal variances for two independent samples with the Bonferroni correction for multiple comparisons. Therefore, in terms of performance, we do not see significant differences between setups with a single or multiple senders.

To summarize, in this section, we scrutinized senders' policies by assuming an optimal policy on the side of the receiver, and computing a cost associated with each such sender's policy. The cost is expressed as the sum of the total number of steps needed to reach each location in an environment, when starting at the centre of the map. The cost for each sender's policy is calculated based on shortest paths found using the TSP formulation of the problem. Approximately, 78% of all sender policies are optimal or best suboptimal solutions, confirming that agents are able not only to coordinate their actions to solve this task, but they are able to do so optimally.

While we have shown that the majority of all solutions learned by agents are either optimal or best suboptimal, the difference was observed between solutions in the subcapacity regime and those in the supracapacity regime. Specifically, in the subcapacity regime, where the communication channel consists of 9 messages or fewer (5 or fewer for *Pong*), approximately 55% solutions are best suboptimal solutions (i.e., as good as they can be), while in the supracapacity regime 95% of solutions are optimal. These numbers are consistent with prior observations, as this task is easier for agents to perform when they are given an abundance of options to choose from. The space of solutions is thus much more restricted in the subcapacity regime, and even though about the half of them are not best suboptimal (i.e., as good as they can be), many of them are close to it, as shown in Figure 6.11. In the next section, we further investigate the characteristics of solutions, starting from the similarity of solutions in different experiments, up to more general notions of similarity that we argue reflects an inherent structure in the learned communication protocol.

## 6.5 Towards the Emergence of Linguistic Structure

So far, we have shown the difference between optimal (or best suboptimal) solutions and solutions found by agents, exposing different levels of discrepancies between the two. Although these results hint at it, they do not directly provide evidence showing that agents found solutions that are similar, i.e., two solutions can have the same cost, yet the underlying policies might not be identical. These results only confirm that there are solutions that are equally distant from an optimal solution in terms of the TSP-derived cost measure we introduced. The goal of comparing sender policies to each other is to ascertain that there are abstract principles captured by these policies that facilitate behaviours beneficial to performance on this task.

In this section, we further scrutinize senders' policies by investigating types of structure in learned representations. We compare the geometric structure in sender's policies across different experiments, and examine the structure in mappings between messages

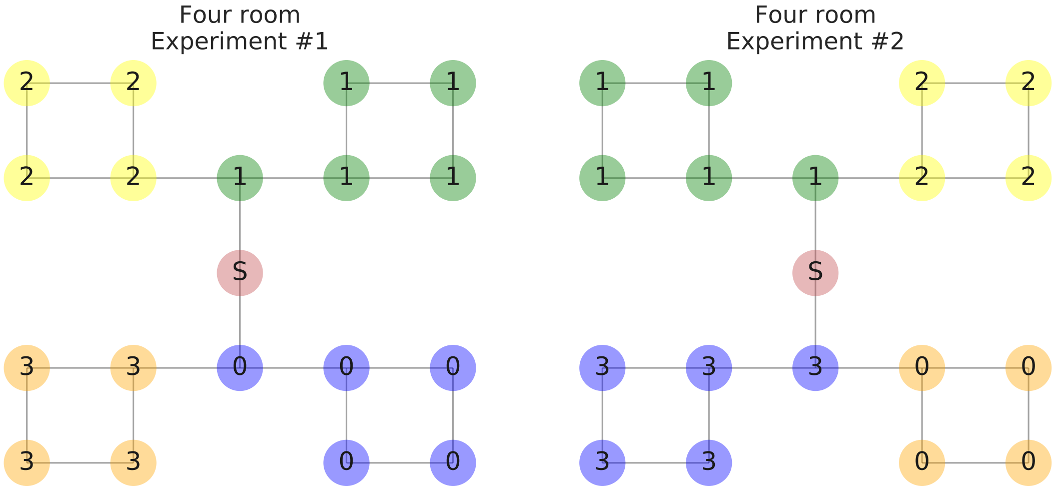


Figure 6.13: Two undirected graphs representing sender's policies from two different 1S-1R experiments. Labels are messages as learned in the experiments, and colours denote subgraph mappings under which the two graphs are isomorphic.

and spatial encodings in multi-sender setups. The purpose of this analysis is to understand to what extent learned representations capture core properties of language, such as structural aspects discussed in Section 6.3.3, or inter-word relationships as modelled in Chapters 4 and 5. In what follows, we define, quantify and investigate two kinds of structural similarities that arise in our experiments: *geometric* similarity refers to the similarity in solution representations across sender policies in different experiments in the subcapacity regime. *Structural* similarity is defined over the mapping of messages to locations they encode.

### 6.5.1 Geometric Similarity

We first discuss the notion of "similarity between two solutions". Intuitively, one could take a look at two different sender policies, such as those shown in Figure 6.5, and visually examine whether they look alike. In doing so, one would focus on the geometric shape of message partitions, where each partition consists of all locations with the same message label, and compare those shapes between solutions, mentally performing different geometric transformations (e.g., rotations, reflections). In doing so, we effectively abstract away specific message labels and instead focus on the structure of the area that is encoded by a message. We can say that if we can map one solution to the other in terms of such geometric transformations, those solutions are the same. In other words, solutions that are similar according to this criterion would partition the space in a similar way. In formal terms, this comparison can be expressed as a graph isomorphism problem which determines whether two graphs are the same. The problem examines the equivalence relationship in graphs, defined as a bijective mapping from a set of nodes of one graph, to the set of nodes of the other graph. While graph isomorphism problem belongs to the NP class, given the small size of our problem, we can exhaustively search by testing all possible combinations of such bijective mappings.

To do so, we first represent both solutions as graphs, where each node in a graph represents one accessible (i.e., non-walled) location in a gridworld, labelled with a corresponding message. For every two adjacent locations in a gridworld, an edge is placed in a graph between the two corresponding nodes. An example of two graphs created from two solutions is shown in Figure 6.13. We fix the first graph, and apply different transformations to the second graph, and for each transformation we count how many nodes overlap between the two graphs. Every transformation is a different mapping of node labels from the first graph to the second graph. Because solutions can be rotated or mirrored versions of each other, we also test label mappings under these conditions. Therefore, for each pair of graphs we do a number of comparisons and return the number of overlapping node labels. In other words, this process examines the size of the largest isomorphic subgraph between the two graphs, where the size of the subgraph corresponds to the number of nodes. The two graphs shown in Figure 6.13 are isomorphic: the graph on the right is the mirrored version of the graph on the left.

The *geometric similarity* is then defined to be the number of matched nodes in such a subgraph normalized by the total number of nodes in a graph. Geometric similarity of the graphs shown in Figure 6.13 is equal to one, as there is a one-to-one mapping between the set of nodes in each graph. We compute this similarity for each pair of solutions for all environments in the subcapacity regime, which results in similarity matrix where each entry is a pairwise geometric similarity between two solutions. The results are shown as similarity matrices in Figure 6.14, with the overall average similarity $\mu$ annotated in each plot.

Only half of the values are shown since the metric (i.e., the number of overlapping nodes) is symmetric, and the average similarity is computed for those values. The highest degree of similarity is observed for $C = 4$ for all environments, and those values are highest for *Four room*, *Flower* and *Pong* environments, where average similarity is at or above $0.98$. Therefore, we can conclude that sender policies have similar structure in those cases. For comparison, the average similarity of randomly generated graphs of the

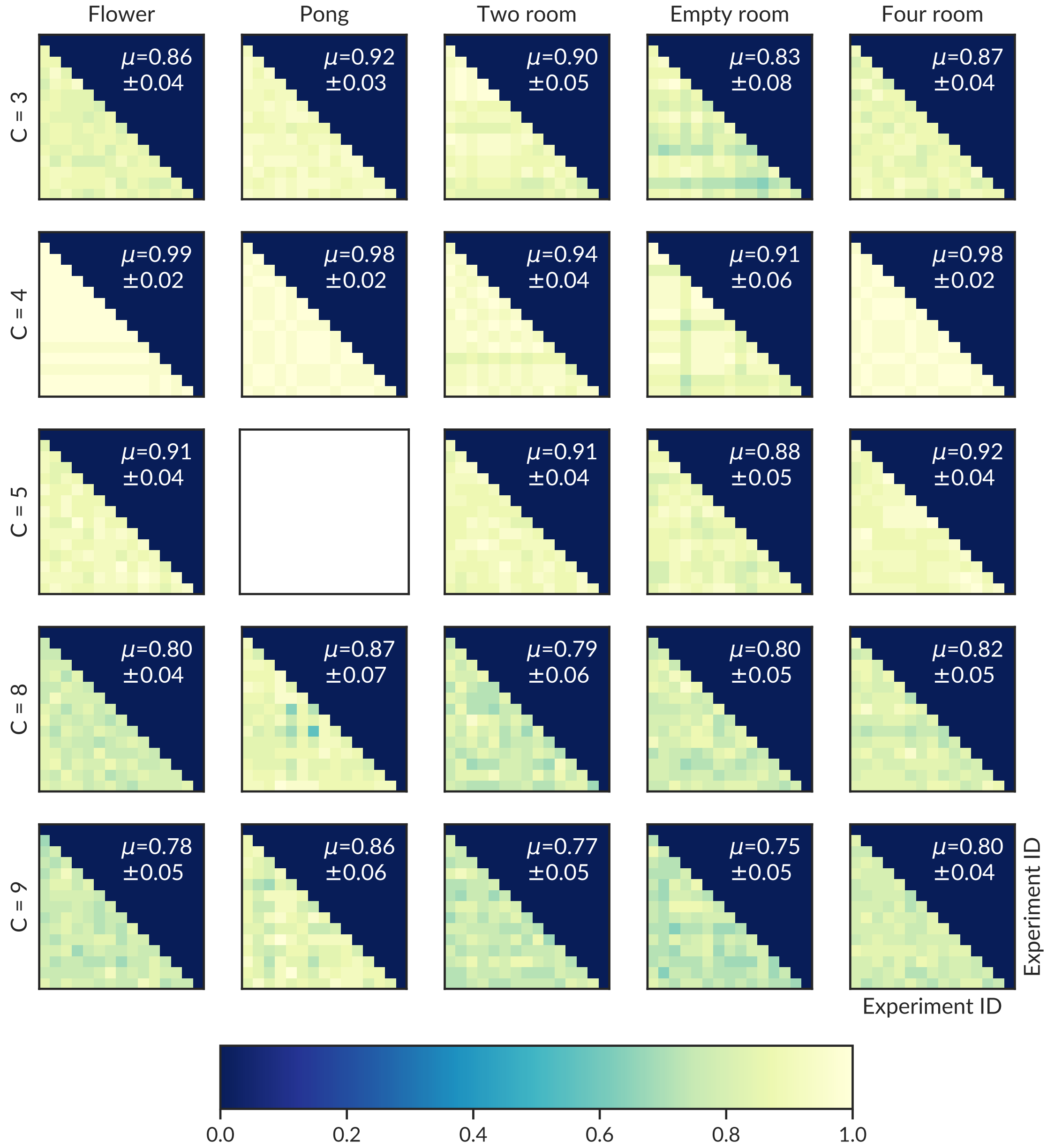


Figure 6.14: Pairwise geometric similarity between message distribution representations in the subcapacity regime. Each plot is annotated with the average similarity value and standard deviation of non-zero elements. There is no data for *Pong* environment with $C = 5$.

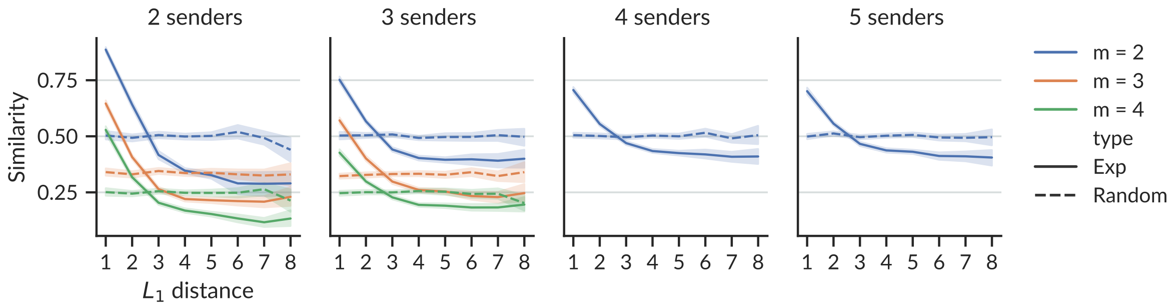


Figure 6.15: Relation between the spatial $L_1$ distance and the similarity of representations.

same size is about $\mu = 0.54 \pm 0.13$. As $C$ increases, the average similarity drops, which is unsurprising, given that the number of possible solutions increases. Anecdotally, we manually inspected a few pairs of solutions that have pairwise similarity lower than the average value $\mu$, which are identified as occasional green "pixels" in plots for *Empty Room* $C = 4$, or *Pong* $C = 8$. Although such solutions are rare, they can still be optimal. We speculate that such solutions have some features that make them more difficult to learn. Overall, we can conclude that solutions in the subcapacity regime are highly consistent with each other, as witnessed by high geometric similarity values among such solutions.

### 6.5.2 Structural Similarity

We now investigate the similarity arising from interpreting a compound message $\mathcal{M} = [m_1, m_2, ..., m_M]$ in multi-sender setups as a single representation encoding one spatial location. In what follows, we examine whether spatially close locations are assigned more similar representations than spatially distant locations. Spatial "closeness" is defined in terms of the $L_1$ distance. We refer to this kind of relation between the distances and encodings as *structural* similarity.

To compute the similarity between different message encodings, we follow a process consisting of a few steps. In the first step, for a single experiment, we map all spatial locations $(x, y)$ to their corresponding encodings $\mathcal{M}$ that represent those locations. For example, a few such mappings constructed from the experiment shown in the lower panel of Figure 6.7 are $(0, 0) \rightarrow 11011$ and $(4, 4) \rightarrow 01001$. This step then yields a dictionary where keys are individual 2D coordinates, and values are specific strings encoding those coordinates. As explained in Section 6.3.3, each digit in an encoding is a categorical value independent of other values, and so applying arithmetic operations to encodings with

Table 6.4: Spearman correlation coefficients for topographic similarity for experimental data $\rho$ and random baseline $\rho_{rnd}$. All $\rho$ values are significant at $p < .001$, while $p > .1$ for all $\rho_{rnd}$.

| Number of senders | $\rho$ | $\rho_{rnd}$ |
|---|---|---|
| 2 | 0.57 | 0.03 |
| 3 | 0.48 | 0.04 |
| 4 | 0.63 | -0.04 |
| 5 | 0.64 | 0.04 |

such values is not meaningful. In the second step, we compute the similarity between such two strings. Commonly used measures of vector similarity such as the Euclidean distance or cosine similarity are not applicable here, and instead, we count the number of overlapping elements between two encodings and normalize the count by the total number of elements. In setups where one or multiple senders have two messages each (such as those shown in Figure 6.7), encodings are binary vectors (with dimensionality $M$, corresponding to the number of senders), and so the count is equivalent to the number of "1"s after applying the negation XOR operator to two vectors.

Average normalized encoding similarities for all distances and all multi-sender experiments are shown in Figure 6.15. Solid lines represent the similarity values computed from the experimental data, while the dashed lines represent a random baseline computed using random strings, where each element is sampled with replacement from possible message values. Curves shown in the figure exhibit two interesting characteristics. First, in all setups, encodings representing locations within two steps from the current location are more similar than what would be expected with random encodings. Second, encodings that are farther apart (3 steps or more) are more dissimilar than what would be expected by chance. This representation, where spatially proximal locations are assigned similar encodings, and spatially distant locations are assigned dissimilar encodings, is robust, since it is mitigating the effect of small perturbations to encodings with respect to task performance. As the number of senders increases, the gap between the random baseline and the experimental data narrows, though it still remains significant. This effect might be due to the larger space of possible encodings, and the fact that there are more messages than locations. In those cases, agents are more likely to agree on the first solution they arrive at without much further fine-tuning of the representation.

This characterization of representations in terms of the relationship between the signal space and the meaning space has been described as the *topographic similarity* in related literature (Brighton and Kirby 2006; Lazaridou et al. 2018). Lazaridou et al. (2018) express it as the negative Spearman correlation $\rho$ between average pairwise vector similarities and Levenshtein distances. A high degree of similarity is taken to reflect the presence of structure in the representation (Brighton and Kirby 2006), where similar signals are used for similar meanings. This is in contrast to unstructured, holistic representations, where there is no relationship between the elements in the signal space and the elements in the meaning space. For example, we find evidence for holistic mapping in agent setups where communication channel capacity $C$, that is, the dimensionality of the signal space, exceeds the dimensionality of the meaning space (i.e., possible goal locations). In such cases, we often find that a single signal is used to encode a single location, and there is no structured relationship between different signals and different locations.

Using the measure of topographic similarity, we examine the presence of structure in the communication protocol in all setups shown in Figure 6.15. We averaged over all experiments with the same number of senders, and calculated their corresponding Spearman $\rho$ correlations. The results are shown in Table 6.4. We observe strong and significant correlations between the distances and representations, concluding that encodings learned by agents are highly structured. In related literature (Brighton et al. 2005; Lazaridou et al. 2018), it has been argued that topographic similarity is associated with the compositional structure due to its purported ability to capture the difference between holistic and compositional representations in language.

## 6.6 Discussion

We have proposed a simple collaborative navigation task as a *language game* (Wittgenstein 1953) to investigate the emergence of basic linguistic features in an interactive setup. While many NLP models rely on large text corpora to learn structured word embeddings, we have shown that basic structure in learned representations can also emerge in a grounded, interactive setup. We argue that such a setup is more likely to reflect environmental pressures and conditions affecting learning experiences of humans, or AI agents interacting with humans. Interaction with the environment by means of trial-and-error-learning is one of the fundamental ways of learning about the associative relationships in the world (Thorndike 1898). In addition, a view of communication as coordinated action taking is consistent with usage-based approaches to language (Clark 1996).

With our experiments we have demonstrated that the communication protocol adopted by agents depends on the topology of the environment, as well as the dimensionality of the signal space, which we referred to as the communication channel capacity. Given sufficient capacity, agents were able to find optimal solutions to the problem, as defined by the shortest paths to goal locations that maximize the reward. In the subcapacity regime, agents compressed the signals, so that a single signal was used to refer to several locations. Therefore, by varying the communication channel capacity, we observed how the efficiency of communication changed from a system that is simple and imprecise, characterized by a small vocabulary where each message encodes a large region in space, to a system that is complex and precise, characterized by a one-to-one mapping of messages to locations. Regier et al. (2015) argue that simplicity and informativeness are two competing communicative principles shaping *efficient communication* observed in natural language. A simple system is the one that minimizes the cognitive load by enabling ease of learning, remembering and use. For example, in our setup simplicity can be quantified as the numbers of messages in the vocabulary. The most simple system would consist of a single message, effectively telling the receiver to "go and explore the environment". Yet, such a simple system is not informative, as the receiver would have to "sweep" the environment in its search for the target location. An informative system enables precision, so that each possible semantic category is assigned a unique label. This increase in precision comes at the increased cost in complexity as more words are needed to refer to a higher number of possible categories. The receiver will know precisely where to go, if there is a single message associated with each possible location. The simplicity of our setup enabled us to find a specific point in each environment where that trade-off between simplicity and precision was optimal, yielding efficient communication.

To study the emergence of structure in this setup, we used a population of independent agents, each emitting a single symbol, to encode a message. We have shown that only the feedback signal in a form of a reward was sufficient for structure to emerge. We argued that we observe a form of structure that is reminiscent of compositionality, one of the core properties of natural language and cognitive systems. For example, a message in a setup can be interpreted as referring to *upper right*, based on one of its constituents referring to *upper* and another one to *right*. Such message representations are topographically similar, so that similar representations encode spatially close locations. The form of structure observed here is too basic to capture the complexity of natural language, due to experimental assumptions that are at odds with the properties of natural language and its use. Individual signals used here are insufficient to resemble the semantic richness of words. As well, the way in which such signals are combined to form messages does not model the syntax of natural language defining the space of valid word sequences, which is one defining feature of compositional systems. This representation is also not productive, which is another property often discussed in the context of compositional representation. Finally, the assumption on the nature of interactions in the model is restricted to that of referential games with a sender and receiver, which is not representative of ordinary language use. Instead, we see this type of representation as a stepping stone for further investigations that might be used to explore more complex communication systems.

Despite the contrived nature of the navigation task used, the emergent protocol exhibits some features that can be meaningfully interpreted in the context of spatial discourse used in route descriptions. Our results are congruent with recent research showing that people establish different linguistic strategies when describing locations of goal states in maze environments (Nölle et al. 2020). Such strategies are dependent on the environmental structure, landmarks and other features, corroborating the hypothesis that aspects of natural language are shaped by environmental affordances. Similarly to language used by individuals describing maze environments, emergent signals in our experiments can be interpreted as describing lines (e.g. "Turn left and go down that way") in stratified environments such as *Pong*, and paths ("Go 1 step up, 1 right, 1 up") in more regular environments such as *Empty room*. However, in our case, these strategies are dependent on the communication channel capacity, as we observe such behaviour only when agents have to compress information about the environment. In the regime where the dimensionality of signal space exceeds that of the meaning space, we observe a tendency for holistic representations, with one-to-one mapping between signals and locations.

Finally, when interpreting message encodings of space as paths, we speculate that there is a relation between our work and the option discovery framework (Sutton et al. 1999; Precup 2001), a body of research in reinforcement learning that studies temporally extended actions. Such actions that can be interpreted in hierarchical terms are postulated to facilitate learning when reused in different tasks (Sutton et al. 1999). We draw this connection as the spatial clustering of environments in our experiments resembles some geometric aspects of eigenvector mappings used for option discovery (Machado et al. 2017). However, we leave the closer examination of this connection to future work.

# 7 | Conclusion

In order to depict the complexity of language, we started this thesis with an overview of its characteristics, and its portrayal in the developmental and neurobiological literature. As well, we have discussed various computational approaches used to model language function, and aspects thereof, in various fields such as natural language processing, cognitive science, and psycholinguistics. While contemporary AI systems show remarkable performance on a variety of language tasks and benchmarks, even outperforming humans in some cases, they are limited in their capacity to explain aspects of human linguistic ability. To some extent, this is expected, considering that the process of language acquisition and realization is remarkably different in machines and humans.

Language processing capacity, as exhibited by many AI systems, is often based on statistical pattern matching learned by processing immense amounts of textual data (Devlin et al. 2018; Radford et al. 2019; Mikolov et al. 2013a; Mikolov et al. 2013b). In contrast, humans acquire language gradually, by interacting with others in physical environments, using different sensory modalities. As well, the human capacity for language develops in unison with a multitude of other socio-cognitive skills during the formative years (Tomasello 2009; Elman et al. 1997). Such skills, language included, are supported by hundreds of millions of spiking neurons in the brain. While certain brain areas and genetic factors play an important role in language function, like many other cognitive functions, the overall linguistic ability, and the development thereof, relies on many different higher-brain functions and general-purpose neurocognitive mechanisms (Hamrick et al. 2018; Kolodny and Edelman 2018; Elman et al. 1997).

Nevertheless, to a large extent, it remains unknown how computations carried out by such large groups of neurons in the brain give rise to the rich variety of linguistic behaviours observed in children and adults. In other words, how the brain and, more specifically, individual neurons and groups of neurons, represent, access and manipulate linguistic information, such as words and sentences, remains an active topic of research. In this thesis, we made steps to bridge the gap between neural computation and observable linguistic behaviours by proposing a set of computational models and simulations that reliably reproduce aspects of linguistic function.

In Chapters 4 and 5, we proposed neural network models that explain how aspects of language function pertaining to semantic processing can be realized in the brain by imposing various neurally and cognitively realistic constraints on such models. We did so using the example of two semantic tasks: the Remote Associates Test (RAT; Mednick 1962), and the semantic fluency task (Bousfield and Sedgewick 1944; Troyer et al. 1997; Hills et al. 2012a), both of which rely on the mental lexicon and accompanying semantic search processes over word associations. While these two tasks have been used in different research domains studying how people search for related items in their memory, the process itself is a fundamental aspect of our memory function, and occurs on a daily basis in a variety of situations. For example, every time we plan a grocery list or pack lunch, we need to think of numerous related items, occasionally adapting the process to account for various constraints such as "make a gluten-free, dairy-free cake without nuts, but with white chocolate" or "substitute lemon juice for vinegar if you have none available". In all those situations, we need to think of several related items, possibly ruling them out as we engage in the activity.

Our implementation of word associations in the models is consistent with the distributed theory of representation in the brain, where a word is represented by the activity of a group of neurons (McClelland et al. 1987; Rogers and McClelland 2004; Stewart et al. 2011b; Eliasmith 2013). Such models propose how specific cognitive or neural mechanisms, such as spreading activation (Collins and Loftus 1975) and the winner-take-all mechanism, can be realized in a spiking neural network to yield high-level aspects of fundamental linguistic behaviours, such as the search process in semantic memory. The proposed models demonstrate a robust match with the experimentally acquired behavioural data on the RAT and the semantic fluency task. In particular, we have demonstrated that the source of semantic information is an important factor influencing the models' ability to reliably reproduce human-like response patterns on these tasks. As well, we showed that the approach we adopt accounts for some of the experimentally observed individual differences. Thus, instead of creating models that outperform people on such tasks, we aimed to answer the question "How can we create models that reproduce human performance on the task with similar computational constraints?" In particular, in the case of the RAT, we have shown that the proposed model is capable of distinguishing between problem items of varying difficulty, such that problems that are easy for humans are also easy for the model (as measured by the task accuracy), and vice versa. We argued that framing the question of language modelling in this way benefits our understanding of cognitive and neural mechanisms underlying aspects of linguistic behaviour.

The models in this thesis demonstrated how biologically constrained computational mechanisms can give rise to aspects of language function. However, we do not claim that this is the only plausible realization. While our models provide a good match with the human data, it may be possible to devise other types of architectures. Nonetheless, we would expect to find some commonality among all such models, such as certain computational mechanisms—for example, a winner-take-all competition mechanism coupled with memory that was important in both of our models. This inherent connection with a biological medium is an important asset distinguishing biologically constrained neural networks from networks typically used in NLP research and applications. While for many NLP applications this connection might not be relevant, we believe that creating machines which exhibit a variety of intelligent behaviours will include capturing some universal computational principles underlying intelligence as exhibited by human cognition and behaviour. To this end, we believe that understanding such principles will be essential for their computational realization in models of language.

To address the question of how aspects of associative relationships fundamental for semantic processes are learned, we proposed a collaborative navigation reinforcement learning setup with communicating agents in Chapter 6. We have shown that in such a framework that highlights the pragmatic aspect of language, agents develop a simple communication protocol that depends on the structure of the environment and reflects some basic features of language, such as topological similarity and compositional structure. While being highly simplified, in order to aid interpretability, the proposed task shows that communicative success was sufficient for a population of agents to reach an agreement on how to optimally use signals to solve the task. This setup is consistent with the usage-based view of language, and in particular with the view of language as a form of coordinated activity (Clark 1996). We argued that this type of communication, grounded in interactions with the environment, is more likely to capture constraints and pressures that affect natural language, and the way humans use it to interact with others.

It is important to acknowledge that different NLP models exhibit aspects of highly sophisticated linguistic behaviour, while, at the same time, bearing little to no similarity with the aspects of neural or cognitive processing involved in language function. Thus, it is reasonable to ask whether we “need” to understand how human brains realize language in order to simulate it, especially given the success of such methods. We would argue that the question of need depends on the goal. While there might be specific domains where human cognition and biological realization are less relevant, such as machine translation or text completion, creating intelligent, reliable and safe machines will require them to demonstrate language use akin to that of a human.

To elaborate, consider the specific example of the GPT-3 language model (Brown et al. 2020), which is likely the most powerful language model to date, as defined by its ability to perform well on a variety of NLP tasks and benchmarks as well as its ability to generate meaningful, coherent and grammatical texts on a wide variety of topics, such as news articles, poetry or even programming code. In fact, the text it is capable of generating can be of such high quality that even humans are unable to distinguish whether it was produced by another human or the model (Brown et al. 2020). Without a doubt, this is one of the most impressive achievements in NLP, and AI more generally, in many years.

From the perspective of describing intelligent behaviour, it is important to recognize that while the linguistic capacity exhibited by GPT-3 appears human-like in many aspects, it does not mean that the model *understands* utterances produced by either itself or others.[1] At the time of writing, the model cannot be accessed publicly, which makes it difficult to systematically evaluate and characterize its behaviours. However, anecdotal examples of interactions with the model are available online by individuals who were given exclusive access, and who probed the model's comprehension in a question-answer form (Lacker 2020; Sabeti 2020). Based on such reports, it can be seen that the model is able to produce correct and meaningful responses on an astonishing number of questions, ranging from factual data such as historical events and geography, to subjective "opinions" such as the best operating system or favourite animals. However, the model often fails when prompted with questions that require physical reasoning or understanding of the relationships between concepts (e.g., it responds with "Yes" to questions such as "Can a bulldozer fit inside a breadbox?", or "Can a human ride a mouse?"). While in some cases it may appear as if the GPT-3 comprehends the meaning of the text, it lacks a cohesive model of the world and cause-effect relationships. In other words, although its responses can appear as if the model has its own agency, it lacks common sense in a form that we humans attribute to other member of our species. In contrast, adults and even young children, understand the world in terms of agents, objects, causal interactions and abstract concepts, such as intuitive physics. GPT-3's language production does not reflect a comprehension of the world, as its knowledge is not grounded in the environment, social interactions and perception. This does not mean that the model will never produce correct responses when probed with questions pertaining to the physical environment—

[1] We refrain from the epistemological discussion on the meaning of "understanding" here, and instead assume it refers to the same, or similar principles that we would probe when evaluating understanding exhibited by young children, as done in developmental literature.

Lacker (2020) and Sabeti (2020) show several successful examples of such reasoning-like ability—instead, the model can *fail catastrophically* in a way in which no healthy individual would when prompted with the most simple questions about the nature of the world. Current interactions with GPT-3 are reminiscent of a verbal exam classroom scenario familiar to many educators, where a teacher is faced with a challenge of determining whether the student truly understood a concept, or whether they just learned it by heart with some serious gaps in their knowledge.

We have argued in this thesis that considering natural intelligence will be an important factor in overcoming such issues that largely distinguish behaviours of AI systems from those of humans. Moreover, we hypothesize that cognitively, biologically and behaviourally constrained models are more likely to reveal principles behind biological computations that underlie human cognition and behaviour. In doing so, to create machines that exhibit human-like intelligent behaviours, not just in the linguistic domain, but across various cognitive domains more generally, we may want to consider methods and approaches that can expand on theories of human cognition and behaviour. This is without a doubt a challenging task, and the difficulty of finding the right approach was discussed almost 70 years ago by Turing (1950), where he concludes:

> “We may hope that machines will eventually compete with men in all purely intellectual fields. But which are the best ones to start with? Even this is a difficult decision. Many people think that a very abstract activity, like the playing of chess, would be best. It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc. Again I do not know what the right answer is, but I think both approaches should be tried.”
>
> Alan M. Turing (1950)

Since this quote was written, developments in AI have come a long way towards showing that machines can indeed not just compete, but also substantially outperform humans in abstract activities such as playing chess, and other board games such as shogi and Go (Silver et al. 2018; Schrittwieser et al. 2019). We hope that with this thesis we made an argument in favour of a tighter connection between behaviours that are natural to human interactions, such as teaching and instruction, using language, and learning in artificial agents endowed with human-like cognitive functions.

# References

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, et al. (2016). “Tensorflow: A system for large-scale machine learning”. In: *12th Symposium on Operating Systems Design and Implementation*, pp. 265–283.

Abbott, J. T., J. L. Austerweil, and T. L. Griffiths (2015). “Random walks on semantic networks can resemble optimal foraging.” In: *Psyc. Rev.* 122.3, p. 558.

Alishahi, A. and S. Stevenson (2008). “A computational model of early argument structure acquisition”. In: *Cognitive Science* 32.5, pp. 789–834.

Alstott, J., E. Bullmore, and D. Plenz (2014). “powerlaw: a Python package for analysis of heavy-tailed distributions”. In: *PloS One* 9.1, e85777.

Anderson, A. J., J. R. Binder, L. Fernandino, C. J. Humphries, L. L. Conant, R. D. Raizada, F. Lin, and E. C. Lalor (2019). “An integrated neural decoder of linguistic and experiential meaning”. In: *Journal of Neuroscience* 39.45, pp. 8969–8987.

Anderson, J. R., M. Matessa, and C. Lebiere (1997). “ACT-R: A theory of higher level cognition and its relation to visual attention”. In: *Human–Computer Interaction* 12.4, pp. 439–462.

Andreas, J., D. Klein, and S. Levine (2017). “Learning with latent language”. In: *arXiv preprint arXiv:1711.00482*.

Anisfeld, M., E. S. Rosenberg, M. J. Hoberman, and D. Gasparini (1998). “Lexical acceleration coincides with the onset of combinatorial speech”. In: *First Language* 18.53, pp. 165–184.

Bahdanau, D., K. Cho, and Y. Bengio (2014). “Neural machine translation by jointly learning to align and translate”. In: *arXiv preprint arXiv:1409.0473*.

Bahdanau, D., F. Hill, J. Leike, E. Hughes, A. Hosseini, P. Kohli, and E. Grefenstette (2018). “Learning to understand goal specifications by modelling reward”. In: *arXiv preprint arXiv:1806.01946*.

Barabási, A.-L. and E. Bonabeau (2003). “Scale-free networks”. In: *Scientific american* 288.5, pp. 60–69.

Barak, L., A. Fazly, and S. Stevenson (2014). “Learning verb classes in an incremental model”. In: *Proceedings of the Fifth Workshop on Cognitive Modeling and Computational Linguistics*, pp. 37–45.

Baroni, M., G. Dinu, and G. Kruszewski (2014). “Don’t count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors”. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Curran Associates, Inc., pp. 238–247.

Bekolay, T., C. Kolbeck, and C. Eliasmith (2013). “Simultaneous unsupervised and supervised learning of cognitive functions in biologically plausible spiking neural networks”. In: *Proceedings of the 35th Annual Conference of the Cognitive Science Society*. Cognitive Science Society, pp. 169–174.

Bekolay, T., J. Bergstra, E. Hunsberger, T. DeWolf, T. C. Stewart, D. Rasmussen, X. Choo, A. R. Voelker, and C. Eliasmith (2014). “Nengo: A Python tool for building large-scale functional brain models”. In: *Frontiers in Neuroinformatics* 7.48.

Belinkov, Y., N. Durrani, F. Dalvi, H. Sajjad, and J. Glass (2017). “What do neural machine translation models learn about morphology?” In: *arXiv preprint arXiv:1704.03471*.

Bengio, Y., P. Simard, and P. Frasconi (1994). “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE transactions on neural networks* 5.2, pp. 157–166.

Benke, T., M. Delazer, L. Bartha, and A. Auer (2003). “Basal ganglia lesions and the theory of fronto-subcortical loops: neuropsychological findings in two patients with left caudate lesions”. In: *Neurocase* 9.1, pp. 70–85.

Binder, J. R., R. H. Desai, W. W. Graves, and L. L. Conant (2009). “Where is the semantic system? A critical review and meta-analysis of 120 functional neuroimaging studies”. In: *Cerebral Cortex* 19.12, pp. 2767–2796.

Binder, J. R., L. L. Conant, C. J. Humphries, L. Fernandino, S. B. Simons, M. Aguilar, and R. H. Desai (2016). “Toward a brain-based componential semantic representation”. In: *Cognitive neuropsychology* 33.3-4, pp. 130–174.

Binetti, G., E. Magni, S. F. Cappa, A. Padovani, A. Bianchetti, and M. Trabucchi (1995). "Semantic memory in Alzheimer's disease: an analysis of category fluency". In: *Journal of Clinical and Experimental Neuropsychology* 17.1, pp. 82–89.

Bloom, L. (1976). *One word at a time: The use of single word utterances before syntax*. Vol. 154. Walter de Gruyter.

Blouw, P., E. Solodkin, P. Thagard, and C. Eliasmith (2015). "Concepts as semantic pointers: A framework and computational model". In: *Cognitive Science* 40 (5), pp. 1128–1162.

Boden, M. A. (2003). *The Creative Mind: Myths and Mechanisms*. 2nd ed. Routledge.

Bohn, M., G. Kachel, and M. Tomasello (2019). "Young children spontaneously recreate core properties of language in a new modality". In: *PNAS* 116.51, pp. 26072–26077.

Botvinick, M. M., J. D. Cohen, and C. S. Carter (2004). "Conflict monitoring and anterior cingulate cortex: an update". In: *Trends in Cognitive Sciences* 8.12, pp. 539 –546.

Bourgin, D. D., J. T. Abbott, T. L. Griffiths, K. A. Smith, and E. Vul (2014). "Empirical Evidence for Markov Chain Monte Carlo in Memory Search". In: *Proceedings of the 36th Annual Meeting of the Cognitive Science Society*, pp. 224–229.

Bousfield, W. and C. Sedgewick (1944). "An analysis of sequences of restricted associative responses". In: *The Journal of General Psychology* 30.2, pp. 149–165.

Bowden, E. M. and M. Jung-Beeman (2003a). "Normative data for 144 compound remote associate problems". In: *Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society* 35.4, pp. 634–639.

— (2003b). "Normative data for 144 compound remote associate problems." In: *Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society, Inc* 35.4, pp. 634–639.

Bowers, J. S. (2009). "On the biological plausibility of grandmother cells: Implications for neural network theories in psychology and neuroscience". In: *Psychological Review* 116.1, pp. 220–251.

Breedin, S. D., E. M. Saffran, and H. B. Coslett (1994). "Reversal of the concreteness effect in a patient with semantic dementia". In: *Cognitive neuropsychology* 11.6, pp. 617–660.

Brighton, H. and S. Kirby (2006). "Understanding linguistic evolution by visualizing the emergence of topographic mappings". In: *Artificial life* 12.2, pp. 229–242.

Brighton, H., K. Smith, and S. Kirby (2005). "Language as an evolutionary system". In: *Physics of Life Reviews* 2.3, pp. 177–226.

Brown, R. and J. Berko (1960). “Word association and the acquisition of grammar”. In: *Child Development* 31.1, pp. 1–14.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, et al. (2020). “Language models are few-shot learners”. In: *arXiv preprint arXiv:2005.14165*.

Brysbaert, M., M. Stevens, P. Mandera, and E. Keuleers (2016). “How many words do we know? Practical estimates of vocabulary size dependent on word definition, the degree of language input and the participant’s age”. In: *Frontiers in psychology* 7, p. 1116.

Bullinaria, J. A. and J. P. Levy (2012). “Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD”. In: *Behavior research methods* 44.3, pp. 890–907.

Cancho, R. F. I. and R. V. Solé (2001). “The small world of human language”. In: *Proceedings of the Royal Society of London. Series B: Biological Sciences* 268.1482, pp. 2261–2265.

Cangelosi, A. and D. Parisi (2012). *Simulating the evolution of language*. Springer Science & Business Media.

Caramazza, A. and J. R. Shelton (1998). “Domain-specific knowledge systems in the brain: The animate-inanimate distinction”. In: *Journal of cognitive neuroscience* 10.1, pp. 1–34.

Charnov, E. L. (1976). “Optimal foraging, the marginal value theorem”. In: *Theoretical population biology* 9.2, pp. 129–136.

Chater, N. and C. D. Manning (2006). “Probabilistic models of language processing and acquisition”. In: *Trends in Cognitive Sciences* 10.7, pp. 335–344.

Chomsky, N. (1957). *Syntactic structures*. The Hague: Mouton.

— (1965). *Aspects of the Theory of Syntax*. MIT press.

— (2007). “Approaching UG from below”. In: *Interfaces + recursion = language? Chomsky’s minimalism and the view from syntax-semantics*, pp. 1–29.

Choo, X. (2010). “The Ordinal Serial Encoding Model: Serial Memory in Spiking Neurons”. Master’s Thesis. Waterloo, ON: University of Waterloo.

— (2018). “Spaun 2.0: Extending the World’s Largest Functional Brain Model”. PhD thesis. University of Waterloo.

Chung, J., C. Gulcehre, K. Cho, and Y. Bengio (2014). “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: *arXiv preprint arXiv:1412.3555*.

Clark, H. H. (1996). *Using Language*. 'Using' Linguistic Books. Cambridge University Press.

Clauset, A., C. R. Shalizi, and M. E. Newman (2009). “Power-law distributions in empirical data”. In: *SIAM review* 51.4, pp. 661–703.

Collins, A. M. and E. F. Loftus (1975). “A spreading-activation theory of semantic processing”. In: *Psychological Review* 82.6, pp. 407 –428.

Crawford, E., M. Gingerich, and C. Eliasmith (2015). “Biologically Plausible, Human-Scale Knowledge Representation”. In: *Cognitive Science*.

Croes, G. A. (1958). “A method for solving traveling-salesman problems”. In: *Operations research* 6.6, pp. 791–812.

Croft, W. and D. A. Cruse (2004). *Cognitive linguistics*. Cambridge University Press.

Davelaar, E. J. (2015). “Semantic Search in the Remote Associates Test”. In: *Topics in Cognitive Science* 7.3, pp. 494–512.

De Deyne, S. and G. Storms (2008). “Word associations: Network and semantic properties”. In: *Behavior research methods* 40.1, pp. 213–231.

De Deyne, S., S. Verheyen, and G. Storms (2016). “Structure and organization of the mental lexicon: A network approach derived from syntactic dependency relations and word associations”. In: *Towards a theoretical framework for analyzing complex linguistic networks*. Springer, pp. 47–79.

Deerwester, S., S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman (1990). “Indexing by Latent Semantic Analysis”. In: *Journal of the American Society for Information Science* 41, pp. 391–407.

Destexhe, A., Z. F. Mainen, and T. J. Sejnowski (1994). “Synthesis of models for excitable membranes, synaptic transmission and neuromodulation using a common kinetic formalism”. In: *Journal of computational neuroscience* 1.3, pp. 195–230.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (2018). “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805*.

Dorogovtsev, S. N. and J. F. F. Mendes (2001). “Language as an evolving word web”. In: *Proceedings of the Royal Society of London. Series B: Biological Sciences* 268.1485, pp. 2603–2606.

Dumont, N. S.-Y. and C. Eliasmith (2020). “Accurate representation for spatial cognition using grid cells”. In: *Proceedings of the 42nd Annual Conference of the Cognitive Science Society*. Toronto, CA: Cognitive Science Society.

Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. New York, NY: Oxford University Press.

Eliasmith, C. and C. H. Anderson (2003). *Neural Engineering: Computation, Representation, and Dynamics in Neurobiological Systems*. Cambridge, MA: MIT Press.

Eliasmith, C., T. C. Stewart, X. Choo, T. Bekolay, T. DeWolf, Y. Tang, and D. Rasmussen (2012). “A large-scale model of the functioning brain”. In: *Science* 338, pp. 1202–1205.

Elman, J. L., E. A. Bates, M. H. Johnson, A. Karmiloff-Smith, D. Parisi, and K. Plunkett (1997). *Rethinking Innateness: A Connectionist Perspective on Development*. Cambridge, MA: MIT Press.

Erdös, P., A. Rényi, et al. (1959). “On random graphs”. In: *Publicationes mathematicae* 6.26, pp. 290–297.

Evans, N. and S. C. Levinson (2009). “The myth of language universals: Language diversity and its importance for cognitive science”. In: *Behavioral and brain sciences* 32.5, pp. 429–448.

Fazly, A., A. Alishahi, and S. Stevenson (2010). “A probabilistic computational model of cross-situational word learning”. In: *Cognitive Science* 34.6, pp. 1017–1063.

Fenson, L., P. S. Dale, J. S. Reznick, E. Bates, D. J. Thal, S. J. Pethick, M. Tomasello, C. B. Mervis, and J. Stiles (1994). “Variability in early communicative development”. In: *Monographs of the society for research in child development*, pp. i–185.

Fink, A., M. Benedek, R. H. Grabner, B. Staudt, and A. C. Neubauer (2007). “Creativity meets neuroscience: Experimental tasks for the neuroscientific study of creative thinking”. In: *Methods* 42.1, pp. 68–76.

Fodor, J. A. and Z. W. Pylyshyn (1988). “Connectionism and cognitive architecture: A critical analysis”. In: *Cognition* 28.1-2, pp. 3–71.

Földiák, P. (2009). “Neural coding: non-local but explicit and conceptual”. In: *Current Biology* 19.19, R904–R906.

Frege, G. (1884). *Die Grundlagen der Arithmetik: eine logisch mathematische Untersuchung über den Begriff der Zahl*. W. Koebner.

Frege, G. (1892). “Über Sinn und Bedeutung”. In: *Zeitschrift für Philosophie und philosophische Kritik* 100, pp. 25–50.

Galantucci, B. (2005). “An experimental study of the emergence of human communication systems”. In: *Cognitive Science* 29.5, pp. 737–767.

Gallese, V. and G. Lakoff (2005). “The brain’s concepts: The role of the sensory-motor system in conceptual knowledge”. In: *Cognitive neuropsychology* 22.3-4, pp. 455–479.

Gazzaniga, M. S., R. B. Ivry, and G. R. Mangun (2014). *Cognitive Neuroscience: The Biology of the Mind*. 4th. W.W. Norton.

Georgopoulos, A. P., A. B. Schwartz, R. E. Kettner, et al. (1986). “Neuronal Population coding of movement direction”. In: *Science* 233.4771, pp. 1416–1419.

Gleitman, L. (1990). “The structural sources of verb meanings”. In: *Language acquisition* 1.1, pp. 3–55.

Goldin-Meadow, S. (2005). *The resilience of language: What gesture creation in deaf children can tell us about how all children learn language*. Psychology Press.

Goldin-Meadow, S. and H. Feldman (1977). “The development of language-like communication without a language model”. In: *Science* 197.4301, pp. 401–403.

Goodman, J. C., L. McDonough, and N. B. Brown (1998). “The role of semantic context and memory in the acquisition of novel nouns”. In: *Child development* 69.5, pp. 1330–1344.

Gosmann, J. and C. Eliasmith (2019). “Vector-Derived Transformation Binding: An Improved Binding Operation for Deep Symbol-Like Processing in Neural Networks”. In: *Neural Computation* 31.5, pp. 849–869.

Goulden, R., P. Nation, and J. Read (1990). “How large can a receptive vocabulary be?” In: *Applied linguistics* 11.4, pp. 341–363.

Gourovitch, M. L., B. S. Kirkby, T. E. Goldberg, D. R. Weinberger, J. M. Gold, G. Esposito, J. D. Van Horn, and K. F. Berman (2000). “A comparison of rCBF patterns during letter and semantic fluency.” In: *Neuropsychology* 14.3, p. 353.

Grice, H. P. (1975). “Logic and conversation”. In: *Speech acts*. Brill, pp. 41–58.

Griffiths, T. L., M. Steyvers, D. M. Blei, and J. B. Tenenbaum (2005). “Integrating topics and syntax”. In: *Advances in neural information processing systems*, pp. 537–544.

Gross, C. G. (2002). “Genealogy of the “grandmother cell””. In: *The Neuroscientist* 8.5, pp. 512–518.

Günther, F., L. Rinaldi, and M. Marelli (2019). “Vector-space models of semantic representation from a cognitive perspective: A discussion of common misconceptions”. In: *Perspectives on Psychological Science* 14.6, pp. 1006–1033.

Gupta, N., Y. Jang, S. C. Mednick, and D. E. Huber (2012). “The Road Not Taken: Creative Solutions Require Avoidance of High-Frequency Responses”. In: *Psychological Science* 23.3, pp. 288–294.

Gurney, K., T. J. Prescott, and P. Redgrave (2001a). “A computational model of action selection in the basal ganglia. I. A new functional anatomy”. In: *Biological cybernetics* 84.6, pp. 401–410.

— (2001b). “A computational model of action selection in the basal ganglia. II. Analysis and simulation of behaviour”. In: *Biological cybernetics* 84.6, pp. 411–423.

Gutmann, M. and A. Hyvärinen (2010). “Noise-contrastive estimation: A new estimation principle for unnormalized statistical models”. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304.

Hamrick, P., J. A. G. Lum, and M. T. Ullman (2018). “Child first language and adult second language are both tied to general-purpose learning systems”. In: *Proceedings of the National Academy of Sciences* 115.7, pp. 1487–1492.

Hauser, M. D., N. Chomsky, and W. T. Fitch (2002). “The faculty of language: what is it, who has it, and how did it evolve?” In: *science* 298.5598, pp. 1569–1579.

Hayashi, Y. (2016). “Predicting the evocation relation between lexicalized concepts”. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1657–1668.

Hazy, T. E., M. J. Frank, and R. C. O’Reilly (2010). “Neural mechanisms of acquired phasic dopamine responses in learning”. In: *Neuroscience & Biobehavioral Reviews* 34.5, pp. 701–720.

Hermann, K. M., F. Hill, S. Green, F. Wang, R. Faulkner, H. Soyer, D. Szepesvari, W. M. Czarnecki, M. Jaderberg, et al. (2017). “Grounded language learning in a simulated 3d world”. In: *arXiv preprint arXiv:1706.06551*.

Hewes, G. W. (1973). “Primate communication and the gestural origin of language”. In: *Current anthropology* 14.1/2, pp. 5–24.

Hewitt, J. and C. D. Manning (2019). “A structural probe for finding syntax in word representations”. In: *Proceedings of the 2019 Conference of the North American Chapter* *of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138.

Hills, T. (2013). "The company that words keep: comparing the statistical structure of child- versus adult-directed language". In: *Journal of Child Language* 40 (03), pp. 586–604.

Hills, T. T., M. N. Jones, and P. M. Todd (2012a). "Optimal foraging in semantic memory." In: *Psyc. Rev.* 119.2, p. 431.

Hills, T. T., P. M. Todd, and M. N. Jones (2012b). "Optimal foraging in semantic memory". In: *Psychological Review* 119.2, pp. 431–440.

Hinton, G. (2014). "Lecture 6e rmsprop: Divide the gradient by a running average of its recent magnitude". University Lecture, University of Toronto.

Hochreiter, S. and J. Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Hubel, D. H. and T. N. Wiesel (1968). "Receptive fields and functional architecture of monkey striate cortex". In: *The Journal of Physiology* 195.1, pp. 215–243.

Huth, A. G., W. A. de Heer, T. L. Griffiths, F. E. Theunissen, and J. L. Gallant (2016). "Natural speech reveals the semantic maps that tile human cerebral cortex". In: *Nature* 532.7600, pp. 453–458.

Izhikevich, E. M. (2007). *Dynamical Systems in Neuroscience*. Cambridge, MA: MIT Press.

Jackendoff, R. (2002). *Foundations of language: Brain, meaning, grammar, evolution*. Oxford University Press, USA.

Jones, M. N. and D. J. K. Mewhort (2007). "Representing word meaning and order information in a composite holographic lexicon". In: *Psychological Review* 114, pp. 1–37.

Jones, M. N., T. T. Hills, and P. M. Todd (2015). "Hidden processes in structural representations: A reply to Abbott, Austerweil, and Griffiths (2015)." In: *Psyc. Rev.*

Jones, M. N., T. M. Gruenenfelder, and G. Recchia (2018). "In defense of spatial models of semantic representation". In: *New Ideas in Psychology* 50, pp. 54–60.

Kajić, I. and C. Eliasmith (2018). *Evaluating the psychological plausibility of word2vec and GloVe distributional semantic models*. Tech. rep. Centre for Theoretical Neuroscience.

Kajić, I. and T. Wennekers (2015). “Neural Network Model of Semantic Processing in the Remote Associates Test”. In: *Proceedings of the NIPS Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches co-located with the 29th Annual Conference on Neural Information Processing Systems (NIPS 2015), Montreal, Canada, December 11-12, 2015.* Pp. 73–81.

Kajić, I., J. Gosmann, T. C. Stewart, T. Wennekers, and C. Eliasmith (2016). “Towards a Cognitively Realistic Representation of Word Associations”. In: *Proceedings 38th Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society, pp. 2183–2188.

Kajić, I., J. Gosmann, B. Komer, R. W. Orr, T. C. Stewart, and C. Eliasmith (2017a). “A Biologically Constrained Model of Semantic Memory Search”. In: *Proceedings of the 39th Annual Conference of the Cognitive Science Society*. London, UK: Cognitive Science Society.

Kajić, I., J. Gosmann, T. C. Stewart, T. Wennekers, and C. Eliasmith (2017b). “A Spiking Neuron Model of Word Associations for the Remote Associates Test”. In: *Frontiers in Psychology* 8, p. 99.

Kajić, I., T. Schröder, T. C. Stewart, and P. Thagard (2019). “The Semantic Pointer Theory of Emotion: Integrating Physiology, Appraisal, and Construction”. In: *Cognitive Systems Research*.

Kajić, I., E. Aygün, and D. Precup (2020). “Learning to cooperate: Emergent communication in multi-agent navigation”. In: *Proceedings of the 42nd Annual Conference of the Cognitive Science Society*. Toronto, CA: Cognitive Science Society, pp. 1993–1999.

Kenett, Y. N., D. Anaki, and M. Faust (2014). “Investigating the structure of semantic networks in low and high creative persons”. In: *Frontiers in Human Neuroscience* 8.407.

Kerns, J. G., J. D. Cohen, A. W. MacDonald, R. Y. Cho, V. A. Stenger, and C. S. Carter (2004). “Anterior Cingulate Conflict Monitoring and Adjustments in Control”. In: *Science* 303.5660, pp. 1023–1026.

Kirby, S., H. Cornish, and K. Smith (2008). “Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language”. In: *Proceedings of the National Academy of Sciences* 105.31, pp. 10681–10686.

Klein, A. and T. Badia (2015). “The usual and the unusual: Solving Remote Associates Test Tasks Using Simple Statistical Natural Language Processing Based on Language Use”. In: *The Journal of Creative Behavior* 49.1, pp. 13–37.

Koch, C. (2004). *Biophysics of Computation: Information Processing in Single Neurons*. Computational Neuroscience Series. Oxford University Press, USA.

Kohl, N. and P. Stone (2004). “Policy gradient reinforcement learning for fast quadrupedal locomotion”. In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*. Vol. 3. IEEE, pp. 2619–2624.

Kolodny, O. and S. Edelman (2018). “The evolution of the capacity for language: the ecological context and adaptive value of a process of cognitive hijacking”. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 373.1743, p. 20170052.

Komer, B., T. C. Stewart, A. R. Voelker, and C. Eliasmith (2019). “A neural representation of continuous space using fractional binding”. In: *41st Annual Meeting of the Cognitive Science Society*. Montreal, QC: Cognitive Science Society.

Kottur, S., J. M. F. Moura, S. Lee, and D. Batra (2017). “Natural Language Does Not Emerge 'Naturally' in Multi-Agent Dialog”. In: *EMNLP*, pp. 2962–2967.

Kounious, J. and M. Beeman (2014). “The Cognitive Neuroscience of Insight”. In: *Annual Review of Psychology* 65, pp. 71–93.

Krizhevsky, A., I. Sutskever, and G. E. Hinton (2012). “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*, pp. 1097–1105.

Lacker, K. (2020). *Giving GPT-3 a Turing Test*. URL: https://lacker.io/ai/2020/07/06/giving-gpt-3-a-turing-test.html (visited on 08/07/2020).

Lawler, E. L. (1985). “The traveling salesman problem: a guided tour of combinatorial optimization”. In: *Wiley-Interscience Series in Discrete Mathematics*.

Lazaridou, A., K. M. Hermann, K. Tuyls, and S. Clark (2018). “Emergence of Linguistic Communication from Referential Games with Symbolic and Pixel Input”. In: *ICLR*.

Levine, S., C. Finn, T. Darrell, and P. Abbeel (2016). “End-to-End Training of Deep Visuomotor Policies”. In: *J. Mach. Learn. Res.* 17.1, 1334–1373.

Lewis, D. (1969). *Convention: A philosophical study*. Harvard University Press.

Lewis, M., Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer (2019). “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension”. In: *arXiv preprint arXiv:1910.13461*.

Li, F. and M. Bowling (2019). “Ease-of-Teaching and Language Structure from Emergent Communication”. In: *NeurIPS*. Curran Associates, Inc., pp. 15825–15835.

Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra (2015). “Continuous control with deep reinforcement learning”. In: *arXiv preprint arXiv:1509.02971*.

Linzen, T. and M. Baroni (2020). “Syntactic Structure from Deep Learning”. In: *arXiv preprint arXiv:2004.10827*.

Little, H., K. Eryılmaz, and B. De Boer (2017). “Signal dimensionality and the emergence of combinatorial structure”. In: *Cognition* 168, pp. 1–15.

Lowe, R., J. Foerster, Y.-L. Boureau, J. Pineau, and Y. Dauphin (2019). “On the Pitfalls of Measuring Emergent Communication”. In: *AAMAS Proceedings*, pp. 693–701.

Luce, R. D. (1959). *Individual choice behavior: A theoretical analysis*. New York: Wiley.

Lupyan, G. and R. Dale (2016). “Why are there different languages? The role of adaptation in linguistic diversity”. In: *Trends in Cognitive Sciences* 20.9, pp. 649–660.

Machado, M. C., M. G. Bellemare, and M. Bowling (2017). “A laplacian framework for option discovery in reinforcement learning”. In: pp. 2295–2304.

Mandera, P., E. Keuleers, and M. Brysbaert (2017). “Explaining human performance in psycholinguistic tasks with models of semantic similarity based on prediction and counting: A review and empirical validation”. In: *Journal of Memory and Language* 92, pp. 57 –78.

Mandler, J. M. and L. McDonough (1993). “Concept Formation in Infancy”. In: *Cognitive Development* 8.3, pp. 291–318.

Marr, D. (1982). “Vision: A computational investigation into the human representation and processing of visual information”. In: *New York: WH Freeman*.

McCarthy, J, M. Minsky, N Rochester, and C. Shannon (1955). “A proposal for the Dartmouth research project on artificial intelligence. Republished in 2006”. In: *AI Magazine* 27.4, pp. 11–14.

McClelland, J. L., D. E. Rumelhart, P. R. Group, et al. (1987). *Parallel Distributed Processing*. Vol. 2. Cambridge, MA: MIT Press.

Mednick, S. A. (1962). “The associative basis of the creative process”. In: *Psychological Review* 69.3, pp. 220–232.

Michel, J.-B., Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, J. P. Pickett, D. Hoiberg, D. Clancy, P. Norvig, et al. (2011). “Quantitative analysis of culture using millions of digitized books”. In: *Science* 331.6014, pp. 176–182.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013a). “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems 26*. Ed. by C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger. Curran Associates, Inc., pp. 3111–3119.

Mikolov, T., K. Chen, G. Corrado, and J. Dean (2013b). “Efficient estimation of word representations in vector space”. In: *arXiv preprint arXiv:1301.3781*.

Miller, G. A. (1995). “WordNet: a lexical database for English”. In: *Communications of the ACM* 38.11, pp. 39–41.

Mnih, A. and Y. W. Teh (2012). “A fast and simple algorithm for training neural probabilistic language models”. In: *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pp. 419–426.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, et al. (2015). “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540, pp. 529–533.

Monaghan, P., T. Ormerod, and U. N. Sio (2014). “Interactive activation networks for modelling problem solving”. In: *Computational Models of Cognitive Processes: Proceedings of the 13th Neural Computation and Psychology Workshop*. Vol. 21, pp. 185–195.

Morais, A. S., H. Olsson, and L. J. Schooler (2013). “Mapping the structure of semantic memory”. In: *Cognitive Science* 37.1, pp. 125–145.

Mordatch, I. and P. Abbeel (2018). “Emergence of grounded compositional language in multi-agent populations”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*.

Moser, E. I., E. Kropff, and M.-B. Moser (2008). “Place cells, grid cells, and the brain’s spatial representation system”. In: *Annual Review of Neuroscience* 31.1, pp. 69–89.

Narasimhan, K., R. Barzilay, and T. Jaakkola (2018). “Grounding language for transfer in deep reinforcement learning”. In: *Journal of Artificial Intelligence Research* 63, pp. 849–874.

Neftci, E. O. and B. B. Averbeck (2019). “Reinforcement learning in artificial and biological systems”. In: *Nature Machine Intelligence* 1.3, pp. 133–143.

Nelson, D. L., C. L. McEvoy, and T. A. Schreiber (2004). “The University of South Florida Free Association, Rhyme, and Word Fragment Norms”. English. In: *Behavior Research Methods, Instruments, & Computers* 36.3, pp. 402–407.

Nematzadeh, A., F. Miscevic, and S. Stevenson (2016). “Simple Search Algorithms on Semantic Networks Learned from Language Use”. In: *Proceedings of the 38th Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society, pp. 1313–1318.

Nematzadeh, A., S. C. Meylan, and T. L. Griffiths (2017). “Evaluating Vector-Space Models of Word Representation, or, The Unreasonable Effectiveness of Counting Words Near Other Words.” In: *CogSci*.

Newell, A. and H. A. Simon (1975). “Computer science as empirical inquiry: Symbols and search”. In: *ACM Turing award lectures*, pp. 113–126.

Nölle, J., R. Fusaroli, G. J. Mills, and K. Tylén (2020). “Language as shaped by the environment: linguistic construal in a collaborative spatial task”. In: *Palgrave Communications* 6.1, pp. 1–10.

Olteteanu, A.-M. and Z. Falomir (2015). “comRAT-C: A computational compound Remote Associates Test solver based on language data and its comparison to human performance”. In: *Pattern Recognition Letters* 67.1, pp. 81–90.

Pascanu, R., T. Mikolov, and Y. Bengio (2013). “On the difficulty of training recurrent neural networks”. In: *International conference on machine learning*, pp. 1310–1318.

Pavlov, P. I. (1927). “Lectures on Conditioned reflexes: an investigation of the physiological activity of the cerebral cortex”. In: *Annals of neurosciences* 17.3. Lecture Series published in 2010, p. 136.

Pennington, J., R. Socher, and C. Manning (2014). “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543.

Pereira, F., B. Lou, B. Pritchett, S. Ritter, S. J. Gershman, N. Kanwisher, M. Botvinick, and E. Fedorenko (2018). “Toward a universal decoder of linguistic meaning from brain activation”. In: *Nature communications* 9.1, pp. 1–13.

Peters, M. E., M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer (2018). “Deep contextualized word representations”. In: *arXiv preprint arXiv:1802.05365*.

Plate, T. A. (1995). “Holographic reduced representations”. In: *IEEE Transactions on Neural networks* 6.3, pp. 623–641.

Precup, D. (2001). “Temporal abstraction in reinforcement learning.” PhD thesis. University of Massachusetts Amherst.

Premack, D. and G. Woodruff (1978). "Does the chimpanzee have a theory of mind?" In: *Behavioral and brain sciences* 1.4, pp. 515–526.

Quillian, M. R. (1967). "Word concepts: A theory and simulation of some basic semantic capabilities". In: *Systems Research and Behavioral Science* 12.5, pp. 410–430.

Quiroga, R. Q. (2012). "Concept cells: the building blocks of declarative memory functions". In: *Nature Rev. Neurosci.* 13.8, pp. 587–597.

Raaijmakers, J. G. and R. M. Shiffrin (1981). "Search of associative memory." In: *Psychological review* 88.2, pp. 93–134.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever (2019). "Language models are unsupervised multitask learners". In: *OpenAI Blog* 1.8, p. 9.

Raffel, C., N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu (2019). "Exploring the limits of transfer learning with a unified text-to-text transformer". In: *arXiv preprint arXiv:1910.10683*.

Rajpurkar, P., R. Jia, and P. Liang (2018). "Know what you don't know: Unanswerable questions for SQuAD". In: *arXiv preprint arXiv:1806.03822*.

Randolph, C., A. R. Braun, T. E. Goldberg, and T. N. Chase (1993). "Semantic fluency in Alzheimer's, Parkinson's, and Huntington's disease: Dissociation of storage and retrieval failures." In: *Neuropsychology* 7.1, p. 82.

Raskin, S. A., M. Sliwinski, and J. C. Borod (1992). "Clustering strategies on tasks of verbal fluency in Parkinson's disease". In: *Neuropsychologia* 30.1, pp. 95–99.

Rasmussen, D. and C. Eliasmith (2014). "A spiking neural model applied to the study of human performance and cognitive decline on Raven's Advanced Progressive Matrices". In: *Intelligence* 42, pp. 53–82.

Ravasz, E. and A.-L. Barabási (2003). "Hierarchical organization in complex networks". In: *Physical Review E* 67.2, p. 026112.

Regier, T., C. Kemp, and P. Kay (2015). "11 Word Meanings across Languages Support Efficient Communication". In: *The handbook of language emergence* 87, p. 237.

Rescorla, R. A., A. R. Wagner, et al. (1972). "A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement". In: *Classical conditioning II: Current research and theory* 2, pp. 64–99.

Rissman, J. and A. D. Wagner (2012). "Distributed representations in memory: insights from functional brain imaging". In: *Annual Rev. of Psyc.* 63, pp. 101–128.

Rizzolatti, G. and M. A. Arbib (1998). "Language within our grasp". In: *Trends in neurosciences* 21.5, pp. 188–194.

Rogers, A., O. Kovaleva, and A. Rumshisky (2020). "A primer in bertology: What we know about how bert works". In: *arXiv preprint arXiv:2002.12327*.

Rogers, T. T. and J. L. McClelland (2004). *Semantic Cognition: A Parallel Distributed Processing Approach*. Cambridge, MA: MIT Press.

Rong, X. (2014). "word2vec parameter learning explained". In: *arXiv preprint arXiv:1411.2738*.

Rosenblatt, F. (1958). "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6, p. 386.

Ross, E. D. and M.-M. Mesulam (1979). "Dominant Language Functions of the Right Hemisphere?: Prosody and Emotional Gesturing". In: *Archives of Neurology* 36.3, pp. 144–148.

Ruder, S. (2018). *NLP's ImageNet moment has arrived*. URL: https://ruder.io/nlp-imagenet/ (visited on 06/02/2020).

Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986). "Learning representations by back-propagating errors". In: *nature* 323.6088, pp. 533–536.

Russell, S. J. and P. Norvig (2002). *Artificial Intelligence: A Modern Approach (2nd Edition)*. Prentice Hall.

Sabeti, A. (2020). *Teaching GPT-3 to Identify Nonsense*. URL: https://arr.am/2020/07/25/gpt-3-uncertainty-prompts/ (visited on 08/07/2020).

Sah, P, S Hestrin, and R. A. Nicoll (1991). "Properties of excitatory postsynaptic currents recorded in vitro from rat hippocampal interneurones." In: *The Journal of Physiology* 430.1, pp. 605–616.

Schmidt, G. L. and C. A. Seger (2009). "Neural correlates of metaphor processing: the roles of figurativeness, familiarity and difficulty". In: *Brain and cognition* 71.3, pp. 375–386.

Schrittwieser, J., I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, et al. (2019). *Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model*.

Schultz, W., P. Dayan, and P. R. Montague (1997). "A Neural Substrate of Prediction and Reward". In: *Science* 275.5306, pp. 1593–1599.

Scott, D. W. (1979). “On optimal and data-based histograms”. In: *Biometrika* 66.3, pp. 605–610.

Scoville, W. B. and B. Milner (1957). “Loss of recent memory after bilateral hippocampal lesions”. In: *Journal of neurology, neurosurgery, and psychiatry* 20.1, p. 11.

Siegel, J. (2008). *The emergence of pidgin and creole languages*. Oxford University Press.

Silver, D., J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, et al. (2017). “Mastering the game of go without human knowledge”. In: *Nature* 550.7676, pp. 354–359.

Silver, D., T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, et al. (2018). “A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play”. In: *Science* 362.6419, pp. 1140–1144.

Skinner, B. F. (1938). *The behavior of organisms: An experimental analysis*. BF Skinner Foundation.

— (1957). *Verbal behavior*. New York: Appleton-Century-Crofts.

Smith, K. A. and E. Vul (2015). “The Role of Sequential Dependence in Creative Semantic Search”. In: *Topics in Cognitive Science* 7.3, pp. 543–546.

Smith, K. A., D. E. Huber, and E Vul (2013). “Multiply-constrained semantic search in the Remote Associates Test”. In: *Cognition* 128, pp. 64–75.

Smith, R. W. and J. Kounios (1996). “Sudden insight: All-or-none processing revealed by speed–accuracy decomposition.” In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 22.6, p. 1443.

Steels, L. (2004). “Constructivist development of grounded construction grammar”. In: *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pp. 9–16.

Stewart, T. C. and C. Eliasmith (2011). “Neural Cognitive Modelling: A Biologically Constrained Spiking Neuron Model of the Tower of Hanoi Task”. In: *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*. Cognitive Science Society, pp. 656–661.

*A biologically realistic cleanup memory: Autoassociation in spiking neurons* (2009).

Stewart, T. C., Y. Tang, and C. Eliasmith (2011a). “A Biologically Realistic Cleanup Memory: Autoassociation in Spiking Neurons”. In: *Cognitive Systems Research* 12, pp. 84–92.

Stewart, T. C., T. Bekolay, and C. Eliasmith (2011b). “Neural Representations of Compositional Structures: Representing and Manipulating Vector Spaces with Spiking Neurons”. In: *Connection Science* 22, pp. 145–153.

— (2012). “Learning to select actions with spiking neurons in the basal ganglia”. In: *Frontiers in Decision Neuroscience* 6.

Steyvers, M. and J. B. Tenenbaum (2005). “The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth”. In: *Cognitive Science* 29.1, pp. 41–78.

Steyvers, M., R. M. Shiffrin, and D. L. Nelson (2004a). “Word association spaces for predicting semantic similarity effects in episodic memory”. In: *Experimental Cognitive Psychology and its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer*. American Psychological Association, pp. 237–249.

— (2004b). “Word association spaces for predicting semantic similarity effects in episodic memory”. In: *Experimental Cognitive Psychology and its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer*. American Psychological Association, pp. 237–249.

Strubell, E., A. Ganesh, and A. McCallum (2019). “Energy and policy considerations for deep learning in NLP”. In: *arXiv preprint arXiv:1906.02243*.

Sutskever, I., O. Vinyals, and Q. V. Le (2014). “Sequence to sequence learning with neural networks”. In: *Advances in neural information processing systems*, pp. 3104–3112.

Sutton, R. S. and A. G. Barto (1987). “A temporal-difference model of classical conditioning”. In: *CogSci Proceedings*. Seattle, WA, pp. 355–378.

— (2018). *Reinforcement learning: An introduction*. MIT press.

Sutton, R. S., D. Precup, and S. Singh (1999). “Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning”. In: *Artificial intelligence* 112.1-2, pp. 181–211.

Thorndike, E. L. (1898). “Animal intelligence: an experimental study of the associative processes in animals.” In: *The Psychological Review: Monograph Supplements* 2.4, p. i.

Thurstone, L. L. (1938). *Primary Mental Abilities*. Double-page reprint series 1. University of Chicago Press.

Toivonen, H., O. Gross, J. M. Toivanen, and A. Valitutti (2013). “On Creative Uses of Word Associations”. In: *Synergies of Soft Computing and Statistics for Intelligent Data Analysis*.

Vol. 190. Advances in Intelligent Systems and Computing. Springer Berlin Heidelberg, pp. 17–24.

Tomasello, M. (2009). *Constructing a language*. Harvard university press.

Troyer, A. K., M. Moscovitch, and G. Winocur (1997). "Clustering and switching as two components of verbal fluency: evidence from younger and older healthy adults." In: *Neuropsychology* 11.1, p. 138.

Troyer, A. K., M. Moscovitch, G. Winocur, M. P. Alexander, and D. Stuss (1998). "Clustering and switching on verbal fluency: The effects of focal frontal-and temporal-lobe lesions". In: *Neuropsychologia* 36.6, pp. 499–504.

Tulving, E. (1983). *Elements of Episodic Memory*. Oxford, UK; New York: Oxford University Press.

Turing, A. M. (1950). "Computing Machinery and Intelligence". In: *Mind* 59.236, pp. 433–460.

Tyler, L. K. and H. E. Moss (2001). "Towards a distributed account of conceptual knowledge". In: *Trends in Cognitive Sciences* 5.6, pp. 244–252.

Ullman, M. T. (2004). "Contributions of memory circuits to language: The declarative/procedural model". In: *Cognition* 92.1-2, pp. 231–270.

Utsumi, A. (2015). "A complex Network approach to distributional semantic models". In: *PloS One* 10.8, e0136277.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin (2017). "Attention is all you need". In: *Advances in neural information processing systems*, pp. 5998–6008.

Voelker, A. R. and C. Eliasmith (2018). "Improving Spiking Dynamical Networks: Accurate Delays, Higher-Order Synapses, and Time Cells". In: *Neural Computation* 30.3, pp. 569–609.

Voelker, A. R., E. Crawford, and C. Eliasmith (2014). "Learning large-scale heteroassociative memories in spiking neurons". In: *Unconventional Computation and Natural Computation*. Ed. by S. K. Oscar H. Ibarra Lila Kari. London, Ontario: Springer International Publishing.

Warrington, E. K. (1975). "The selective impairment of semantic memory". In: *The Quarterly journal of experimental psychology* 27.4, pp. 635–657.

— (1981). "Concrete word dyslexia". In: *British Journal of Psychology* 72.2, pp. 175–196.

Watkins, C. J. and P. Dayan (1992). “Q-learning”. In: *Machine learning* 8.3-4, pp. 279–292.

Wittgenstein, L. (1953). *Philosophical investigations/Philosophische Untersuchungen.* Oxford: Basil Blackwell.

Wyner, A. D. (1967). “Random packings and coverings of the unit n-sphere”. In: *The Bell System Technical Journal* 46.9, pp. 2111–2118.

Xu, F. and J. B. Tenenbaum (2007). “Word learning as Bayesian inference.” In: *Psychological review* 114.2, p. 245.

Yaniv, I. and D. E. Meyer (1987). “Activation and metacognition of inaccessible stored information: potential bases for incubation effects in problem solving.” In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 13.2, p. 187.

# Image Credits

Figure 2.1 adapted from: Blausen.com staff (2014). “Medical gallery of Blausen Medical 2014”. WikiJournal of Medicine 1 (2). DOI:10.15347/wjm/2014.010. ISSN 2002-4436.