Forget About It: Entity-Level Working Memory Models for Referring Expression Generation in Robot Cognitive Architectures

Rafael Sousa Silva (rsousasilva@mines.edu)[1]

Michelle Lieng (mylieng@mines.edu)[1]

Tom Williams (twilliams@mines.edu)[1]

[1] MIRRORLab, Colorado School of Mines, Golden, CO, USA

Abstract

Working Memory (WM) plays a key role in natural language understanding and generation. To enable a human-like breadth and flexibility of language understanding and generation capabilities, cognitive systems for language-capable robots should feature a human-like WM system in a similarly central role. However, it is still quite unclear how robotic WM should be designed, as a variety of models of human WM have been proposed in cognitive psychology. Moreover, human reliance on WM during language production is sometimes to help the speaker rather than to help hearers. Thus, it is unclear whether different robotic WM systems might harm certain dimensions of interaction for the sake of the robot speaker's ostensible ease of cognitive processing. In this paper we demonstrate how different models of human WM can be implemented into robot cognitive architectures. Our results suggest that these models can be effective in terms of accuracy, perceived naturalness, and perceived human-likeness.

Keywords: robot cognitive architectures; cognitive systems; working memory; referring expression generation; forgetting

Introduction

Working Memory (WM) is a central component of human cognition, providing temporary storage and manipulation of information necessary for core cognitive tasks (Baddeley, 1992). The dynamics of WM steer key cognitive processes, such as reasoning (Kyllonen & Christal, 1990; Süß et al., 2002), comprehension (Halford et al., 1998), and learning (Baddeley, 2010). WM also plays a key role in language processing, including natural language understanding (Rönnberg et al., 2010), generation (Gundel et al., 1993), and acquisition (Baddeley et al., 1998; Denhovska et al., 2016). For example, WM may steer Referring Expression Generation (REG) (Bannon, 2019; Gatt et al., 2011; Goudbeek & Krahmer, 2011), during which humans select the set of properties and relations they will use to describe entities (e.g., objects and locations) to other people (Van Deemter, 2016). Yet, despite the centrality of WM in human cognition, it has rarely played a significant role in robotic approaches to language understanding and generation (including REG, which we specifically consider in this work) even within the cognitive systems tradition.

Cognitive Robotics seeks to create genuinely capable and interactive robots through cognitive architectures whose modules, data structures, processes, and architectural design are inspired by key theories from cognitive psychology (Kurup & Lebiere, 2012; Laird et al., 2012, 2017). Research in this tradition includes much work on robot communication, including language acquisition (Attamimi et al., 2016; Miyazawa et al., 2019; Scheutz et al., 2019), reference resolution (Williams et al., 2016), referring form selection (Moulin-Frier et al., 2017), and REG (Liu et al., 2022; Williams et al., 2020; Williams & Scheutz, 2017).

In cognitive architectures like ACT-R and SOAR, WM is typically modeled through buffers of "activated entities" that serve as a short-term cache, but which do not explicitly aim to model what is known about the dynamics of this cache (Baxter & Browne, 2010). Moreover, these approaches do not typically model the connections between WM and natural language generation. As such, it is unclear whether REG models that leverage robotic WM systems with human-like dynamics will be able to produce human-like, natural, and easy to understand referring expressions.

To address these gaps, research must explore how competing models of WM dynamics can be implemented in robot cognitive architectures, and how language produced according to those dynamics will be perceived. Substantial research focus has been given to the limited storage of WM (Ma et al., 2014), and the ways information leaves WM through forgetting. Two popular theories model forgetting in terms of either *decay* (systematic removal of unrehearsed information from WM buffers over time) and *interference* (limited storage capacity (Brown, 1958; Waugh & Norman, 1965)).

In this work, we ask whether human-like models of WM forgetting would make robot REG natural and human-like – or negatively and unnecessarily harm performance. To do so, we (1) demonstrate how an *entity-based* model of WM can be integrated into a cognitive architecture, (2) demonstrate how different WM forgetting models can be implemented within that approach, and (3) experimentally validate this approach through a user study with an autonomous cognitive architecture.

Related Work

Working Memory in Psychology

Although the first mentions of Working Memory come out of early literature on artificial intelligence (Newell & Simon, 1956), the most influential model of WM in psychology was introduced by Baddeley (1983). This model is comprised of a multi-modal short-term memory structure in which a cen-

tral controller coordinates information and strategies from three subsystems: the visuospatial sketchpad, the phonological loop, and the episodic buffer (Baddeley, 2000).

Since Baddeley's seminal work, new theories have diverged along foundational criteria (Cowan, 2017), including whether WM should be divided into multiple stores, WM's relationship with LTM, and the grounding of WM capacity in representational quantity vs quality (Ma et al., 2014). Regardless of these differences, WM researchers agree that at its core, WM is a set of mechanisms and processes for maintaining and operating on the available mental representations most relevant for an ongoing cognitive task (Oberauer, 2019).

Memory researchers also agree on the highly influential role that WM plays across disparate cognitive processes, like reasoning ability (Kyllonen & Christal, 1990; Süß et al., 2002), concept formation (Halford et al., 1998), and attentional control (Gilchrist & Cowan, 2011; Kane & Engle, 2003; Kiyonaga & Egner, 2013; Oberauer, 2013). WM also informs language processing. Since the late 1800s, psychologists have recognized the influence of memory on human language production (EbbingHaus, 1885), and recent evidence suggests that WM itself may arise from our need to communicate (Schwering & MacDonald, 2020). As such, many language-related tasks naturally rely on WM, such as vocabulary acquisition (Baddeley et al., 1998) and language comprehension (Daneman & Merikle, 1996). In language *production*, WM plays a key role in maintaining multiple possible structures that could be used (Myachykov et al., 2013), sequencing of syntactic constituents (Ivanova & Ferreira, 2019) and pre-articulatory monitoring (Pickering & Garrod, 2014). These tasks are critical to language capability, regardless of whether the speaker is a human or a robot. But little attention has been given to the dynamics of WM in robot cognitive architectures.

Computational Models of WM

While a number of processes govern WM dynamics, in this work, we focus on *forgetting*. Two key models of forgetting in WM have been proposed: the theory of decay (Brown, 1958) posits that memory items leave WM after a certain amount of time if not rehearsed or reinforced, while the theory of interference (Waugh & Norman, 1965) posits that WM buffers have limited capacity, and items that are not maintained get replaced by newer entries. Both models are supported by strong empirical evidence (e.g. Jonides et al. (2008); Muter (1980); Oberauer & Lewandowsky (2014); Reiter & Dale (1997)), and have been computationally modeled in various ways, such as through the interference-based Serial Order in a Box-Complex Span (SOB-CS) model (Oberauer et al., 2012) and the decay-based Time-Based Resource-Sharing (TBRS) model (Barrouillet et al., 2004; Oberauer & Lewandowsky, 2011).

While these models were not developed or evaluated as parts of general, integrated systems, complex cognitive architectures like ACT-R and SOAR have modeled WM in ways that allow for influence over general cognitive processes. These approaches typically reduce WM to limited-size buffers that can hold a fixed number of arbitrary chunks or representations (Giorgi et al., 2021; Lindes & Laird, 2017; Martín et al., 2020; Rodgers et al., 2013) or as a literal phonological loop for rehearsing inner speech (Chella & Pipitone, 2020). However, these models typically focus more on the storage of entities as a whole rather than the storage of *properties* of those entities, whose importance is emphasized by recent WM research (Ma et al., 2014).

In contrast, some HRI researchers have begun augmenting existing robot architectural components with short-term buffers that maintain properties of recently activated entities (Williams, Thielstrom, et al., 2018). Yet there has been little formal exploration of this approach, and how different interference and decay strategies might be implemented and parameterized. Nor has there been significant investigation of how these approaches might guide natural language generation. In short, it is still unclear how these WM systems can be implemented, and whether robots whose WM systems are governed by forgetting dynamics can produce human-like, natural, and effective language.

Given the scarcity of robotic computational models of WM forgetting dynamics, we investigate how decay and interference can function within a cognitive architecture in order to answer two key research questions:

RQ1: How can working memory modules based on the decay and interference models of forgetting be integrated into a robot cognitive architecture?

RQ2: Will a robot cognitive architecture using these models produce effective natural language, in terms of human accuracy, ease of the listener's cognitive processing, perceived naturalness, and perceived human-likeness?

In the next section, we will explain the architectural approach used to answer these research question.

Architectural Approach

Our computational models were developed on the Distributed, Integrated, Affect, Reflection, and Cognition (DIARC) architecture, which incorporates key theories from cognitive psychology and linguistics to enable language-capable robots (Scheutz et al., 2019, 2013). DIARC is implemented in the Agent Development Environment (ADE) (Scheutz, 2006) middleware, a secure and fault-tolerant robotic framework that allows architectural components to operate in parallel and communicate asynchronously.

DIARC maintains a decentralized long-term memory store through a set of components that function as distributed heterogeneous knowledge bases (DHKBs) (Williams & Scheutz, 2016). Each DHKB holds information about a set of *entities* that may be referenced by a robot in dialogue. Entities within a DHKB fit a distinct type of world *object*, such as people, locations, and observable objects. A consultant framework (Williams, 2017) provides DIARC components

with domain-independent access to DHKBs. That is, consultants provide information about entities without needing to share how the domain-specific processes are handled within a DHKB. Each DHKB is managed by a consultant.

Building on Williams, Thielstrom, et al. (2018), we take an *Entity-Based Resource Management* approach, in which each DHKB maintains a WM buffer storing a set of *features* for each entity. That is, rather than encoding a small set of *entities*, these buffers encode a small set of *properties* for *every* entity known of within the DHKB, along with a memory trace back to a full representation of that entity stored elsewhere in the DHKB (cp. (Nozari & Novick, 2017)). Figure 1 illustrates how these buffers are structured under an Entity-Based Resource Management approach.

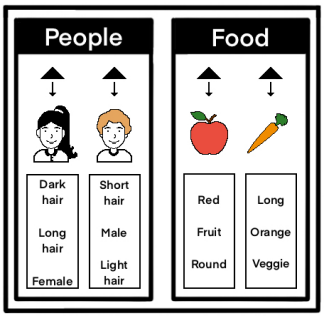

Figure 1: People and Food represent two DIARC consultants, each containing two entities. Each entity is equipped with a resource manager (triangles) that determines how properties are placed and removed from WM buffers, which are queues containing properties that describe the given entity. Entities adapted from "User Avatar Icons" by Users Insights (usersinsights.com/user-avatar-icons), used under CC BY (creativecommons.org/licenses/by/3.0). This figure is licensed under CC BY by Sousa Silva et al.

Decay and interference WM models of forgetting are implemented in these buffers through the WM Manager (Williams et al., 2020), which automatically detects and connects to DHKBs when they register with the architecture, and is responsible for adding or removing properties from entity buffers according to the selected WM dynamics model.

Most WM decay models are based on activation and the fact that it takes a certain amount of time for this activation to decay. In addition, reuse of information bumps activation, restoring the amount of time for items to decay. In this model, for the sake of computational efficiency, we only model the total amount of decay time rather than activation as a proxy. The *Decay* model removes elements from each DHKB's WM buffers at set intervals according to parameter δ. That is, the WM Manager removes the least-recently-added property from each WM Buffer every δ seconds.

Most WM interference models assume a limited capacity. Thus, the *Interference* model removes elements from each DHKB's WM buffers based on their set storage capacities according to parameter α. That is, the WM Manager removes the least-recently-added property from each WM Buffer that contains more than α properties.

System Description

DIARC is a flexible, modular architecture, in which components (1) are used in different configurations based on task requirements, and (2) can be dynamically started and stopped within and between tasks. As such, to understand how our WM implementation influences REG, it is necessary to understand the full set of architectural components that must be run in parallel for these capabilities to interact. Depending on the architectural configuration, DIARC modules might be instantiated as distinct components, or subsumed by a narrower set of components. In our configuration (Fig. 2), we attempt to minimize the number of components used, to most effectively validate the Working Memory Manager alone.

This configuration involves at least three key modules: (1) the **WM Manager**, which manages WM buffers as described above; (2) the **Referential Executive (REX) Component**, which oversees all tasks related to Natural Language Reference, and which maintains modules for Reference Resolution and Referring Expression Generation; and (3) **a Task Component**, which serves as a DHKB, and simulates speech recognition, parsing, surface-level natural language generation, and speech synthesis. That is, based on user selections in our Task Component interface (as we will describe later on) the Task Component compiles the Natural Language Packet (NLPACKET) representation that would have been generated if the robot's interactant had been speaking out loud rather than using the Task Component interface. And, based on the robot's decisions, text is visualized within the Task Component rather than spoken out loud by the robot. The ultimate vision of this work is to have a robot replace the interface and fully manage speech communication.

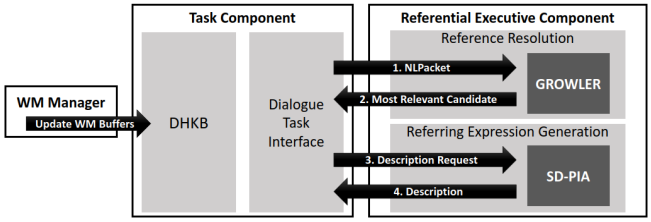

Figure 2: DIARC architectural diagram.

Interaction thus proceeds as follows: First, the user describes an entity to the robot using the Task Component interface, which turns this description into an NLPACKET that is sent to the REX Component (Figure 2, step 1). The REX Component then submits a reference resolution request to its Reference Resolution module.

Reference Resolution uses the GROWLER algorithm (Williams et al., 2018) for Givenness and Relevance-Theoretic Open World Reference Resolution, which builds off the GH-POWER algorithm (Williams et al., 2016; Williams, 2019). GROWLER maintains a set of data

structures informed by the Givenness Hierarchy (Gundel et al., 1993), containing pointers to entities whose full representations are stored in DIARC's DHKBs.

Depending on the type of *referring form* used by an interlocutor (e.g., "it", "this", "that", "this-⟨N'⟩", "that-⟨N'⟩", "the-⟨N'⟩", "a-⟨N'⟩"), GROWLER determines which buffers need to be searched through (e.g., when "it" is used, GROWLER considers representations in the "In Focus" Buffer). For certain referring forms (e.g., "the-⟨N'⟩"), GROWLER will effect a full search of long-term memory (i.e., the DHKBs) as a last resort, using the POWER algorithm (Williams & Scheutz, 2015, 2016) (cp. (Culpepper et al., 2022)). A Consultant Framework (Williams, 2017) mediates the interface between GROWLER and these DHKBs.

While algorithms like GH-POWER return the first candidate referent found to satisfy the semantic constraints imposed by an utterance, GROWLER instead continues until a sufficiently *relevant* candidate referent is found (according to discourse salience and relevance metrics), retaining all suitable (but not necessarily relevant) options found along the way. GROWLER then returns all candidates that are at least half as relevant as the most relevant candidate. While others have used the full set of candidates to generate clarification requests (Jackson & Williams, 2022; Williams et al., 2019), our configuration returns only the most relevant candidate (Figure 2, step 2).

The Task Component then requests the REX Component to generate a description of the inferred entity (Figure 2, step 3). The REX component uses the SD-PIA algorithm (Williams et al., 2018) to generate these descriptions as a set of logical predicates that uniquely describe the target entity. SD-PIA extends the Incremental Algorithm for natural language generation (Dale & Reiter, 1995) in two ways.

First, SD-PIA's knowledge of the world entities and the properties that hold for them is grounded in the DHKBs connected to the Referential Executive. That is, the Referential Executive automatically connects to new architectural components as they come online, allowing SD-PIA to know of new entities and their corresponding properties.

Second, SD-PIA leverages the WM buffers maintained by each DHKB as a "first stop" for properties to use to describe an entity. That is, while other algorithms, such as DIST-PIA (Williams & Scheutz, 2017), will use all available properties to describe an entity, SD-PIA will first attempt to describe the entity only using the properties currently retained for that entity in WM, resorting to other properties only if this is not sufficient to craft a fully distinguishable description. The set of properties selected by SD-PIA are returned to the Task Component, which generates and displays a text realization of that description (Figure 2, step 4).

Validation Methodology

Experimental Design

To validate our approach, we conducted a human-subjects study in which participants interacted with our architecture in the context of a "Guess Who" game. Each participant played two (order counterbalanced) games with the architecture: one in which the architecture used a decay model, and one in which the architecture used an interference model. For each game, this model was randomly parameterized in one of three ways: for the decay model, δ was set to either 10, 15, or 20 seconds; For the interference model, α was set to either 2, 3, or 4 (as under an *Entity-Based* WM model, even a small number of properties maintained per entity yield a large number of total remembered properties, depending on the total amount of entities in memory).

Experimental Context

In the Guess-Who game, the user and the robot architecture were each presented with a set of human faces. They took turns describing designated faces and then trying to guess the face the other player had been referring to. The game interface shown to human participants consisted of a grid of sixteen faces positioned above an interactive panel where the player would either (1) select properties to describe a face, if they were describing; or (2) select a face from a drop-down menu, if they were guessing.

On player turns, the interactive panel displayed 23 buttons with properties such as *long hair*, *no glasses*, *lab coat* in a randomized order. As new properties were clicked, a text box above the buttons was updated to reflect the changes in the sentence that was going to be sent to the robot. The program accounted for the selection of properties that contradicted each other, using newly selected properties to replace any previously selected properties they would contradict. For example, if the player had clicked the properties *sad*, *female*, *square glasses*, and *dark hair*, the text box would display "The sad woman with square glasses and dark hair." If, subsequently, the player selected the property *happy*, this would be updated to "The happy woman with square glasses and dark hair." The robot architecture would then guess which face was being described, and this guess was conveyed to the participant.

On robot turns, players were shown the sentence generated by the robot architecture to describe its designated face, and then selected the face they believed the robot was referring to. To generate these sentences, DIARC's WM-enabled REG algorithm was used to select a set of properties to describe the designated face, using the forgetting policy assigned to their condition. These properties were then fed into a template-based sentence realization system to create a standardized American English sentence string.

Measures

To validate our approach, we collected four key measures: **Human accuracy** was measured as the percentage of correct player guesses. **Response time** was used as a proxy for difficulty of cognitive processing, and was measured as the average time the player took to guess faces across rounds. **Naturalness** was measured through surveys administered every five rounds, which asked how natural the robot sen-

Table 1: Results

Model Parameter	Decay						Interference					
	10 seconds		15 seconds		20 seconds		2 items		3 items		4 items	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Accuracy	0.955	0.060	0.975	0.056	0.961	0.045	0.968	0.039	**0.979**	0.026	0.967	0.029
Naturalness	3.951	1.052	3.666	0.984	**4.124**	0.594	4.049	0.790	3.774	0.839	3.700	0.750
Response Time [s]	15.115	4.197	16.506	5.498	15.934	3.955	**14.941**	2.974	15.286	5.146	17.186	2.928
Human-likeness	0.638	0.236	0.645	0.198	0.640	0.226	0.629	0.222	**0.605**	0.232	0.632	0.216

tences had been in those rounds (highly unnatural, slightly unnatural, neutral, slightly natural, highly natural). Values for each game were translated to a 1-5 scale and averaged. **Human-likeness** was measured through Jensen-Shannon divergence (Lin, 1991) between the properties selected by the architecture and those selected by humans, across all games.

Procedure

After providing informed consent, participants were taken to an experiment room where a computer had the game interface open. The experimenter then explained that the game was going to take 15 minutes and that in some rounds, the participant would have to describe a face to the robot while in other rounds they would have to guess which face the robot was referring to. After answering participants' questions (if any), the experimenter started the game timer and headed to a control room. After the timer reached 15 minutes, the experimenter ushered the participant to a waiting room and returned to the control room to set up the second game, which used a different model of forgetting from the first game. The participant was then ushered back to the experiment room and played the second game for 15 minutes. After the second game was over, participants were debriefed and paid.

Participants

Students, faculty, and staff (39 female, 51 male) were recruited from the **anonymous** academic body, and reached a total of 90 participants. Each participant played two fifteen-minute games with the robot. Participants received $10 for participation at the end of their experiment.

Results

While the purpose of this paper is not to prove that certain parameterizations for forgetting models are better than others, for the reader's sake, we will briefly touch on how the different parameterizations used throughout our experiments compare to each other. Because the first game served as an opportunity for participants to get used to the interface and the game mechanics, only the second game's data were used for our final analysis. Our descriptive results are summarized in Table 1. Data files have been uploaded to OSF (https://bit.ly/cogsci2023-1785).

Human Accuracy

Mean human accuracy was above 95% across all six configurations, demonstrating that regardless of model our approach enabled high accuracy. Among the parameterizations considered, the interference parameterization using an α value of 3 enabled the highest accuracy (M=0.979, SD=0.029), while the decay parameterization using a δ value of 10 enabled the lowest accuracy (M=0.955, SD=0.06).

Perceived Naturalness

Mean perceived naturalness was above 3.5 in all conditions, demonstrating that regardless of model our approach was perceived as fairly natural. Among the parameterizations considered, the decay parameterization using a δ value of 20 enabled the highest perceived naturalness (M=4.124, SD=0.594), and the decay parameterization using a δ value of 15 enabled the lowest perceived naturalness (M=3.666, SD=0.984).

Response Time

Participants were slow to respond regardless of model, with a mean listener response time of over 14 seconds in all conditions. Among the parameterizations considered, the interference parameterization using a α value of 2 enabled the shortest response times (M=14.941, SD=2.974), and the interference parameterization using a α value of 4 enabled the longest response times (M=17.186, SD=2.928).

Perceived Human-likeness of Referring Expression Generation

Mean Jensen-Shannon Divergence was above 0.500 in all conditions, suggesting significant deviation from human response patterns regardless of model. Among the parameterizations considered, the interference parameterization using a α value of 3 enabled the most human-like property distributions (M=0.605, SD=0.232), and the decay parameterization using a δ value of 15 enabled the least human-like property distributions (M=0.645, SD=0.198).

Discussion

Assessment of Results

Human Accuracy Our human accuracy results suggest that our approach allowed users to successfully identify target referents, as all configurations enabled mean accuracy above 95%. However, our accuracy results also suggest a clear ceiling effect. These results could have been influenced by the lack of time constraints. That is, when guessing which face the robot was referring to, players had unlimited time to search for the correct referent before making their final

guess. In contexts where time is limited, differences in accuracy across models might be more pronounced.

Perceived Naturalness Perceived naturalness results manifested differently for our two model categories. This difference can be further investigated in future work. Models of decay with a δ value of 20, which stored items in WM for longer, were perceived to be slightly more natural than the other decay parameterizations, although the differences between these parameterizations was not pronounced. In contrast, when the interference model of forgetting was used, perceived naturalness results dropped as α values increased, suggesting that lower resource caps may have enabled more natural speech. We note that despite the low values of α considered, even the lowest value of α used in our study puts the total amount of items in the robot's WM above human storage capacity (i.e. with $\alpha = 2$ and 16 entities within our domain, there can be a maximum of 32 properties stored in working memory at any given point). These limitations are inherent to our Entity-Based resource management framework. To explore whether naturalness might further increase by imposing more significant resource caps, researchers would need to consider alternate WM architectures where each entity representation is not guaranteed space in WM.

Listener's Response Time For interference models, lower α values enabled faster average response times, suggesting that smaller WM storage capacities might lead to generation of more easily comprehensible utterances due to inability to rely on "stale" properties. Similarly, decay models achieved faster average response times when features decayed out of memory more quickly. That being said, all models yielded slow mean response times, close to 15 seconds. We believe this outcome is due to the lack of time pressure; imposing time constraints may have led to more immediately visible differences between parameterizations.

Perceived Human-likeness of Referring Expression Generation All model parameterizations had mean Jensen-Shannon divergence values suggesting a moderate difference between the set of properties used by the robot architecture and by human players. However, it was not clear whether this difference was inherent to our approach, or due to individual differences between the descriptions provided by each participant. We thus conducted a supplemental analysis using the same methodology, and compared the set of properties used by each participant to the set of properties used by all other participants. Interference models with $\alpha = 4$ produced the best Jensen-Shannon results (M=0.585, SD=0.241) and decay models with $\delta = 20$ produced the worst results (M=0.660, SD=0.226). These results suggest similar results for both the robot architecture and the human players, and suggests that the low values obtained from the Jensen-Shannon divergence might be a product of individual differences between player descriptions rather than being endemic to our approach.

Overall, our results provide promising support for our architectural approach, and motivate further research to more formally interrogate differences in parameterizations and, most importantly, compare the proposed architecture to a baseline model that does not implement WM forgetting.

Future Work Guidelines

The purpose of this paper was not to prove whether specific forgetting models are better or worse than others, but rather to show how they can be effectively implemented within a robotic cognitive architecture and explore the effects of different parameterizations. We see five key directions for future work. First, we are interested in comparing these results to a model that does not use WM at all. Second, we will address our possible ceiling effects by improving task complexity and investigate why decay and interference presented opposite trends for perceived naturalness. Third, future work should consider the way that different decay strategies may or may not unintentionally align with the boundaries between interaction turns. Fourth, a wider variety of parameterizations should be considered. Finally, in this work, we only use one consultant because we are measuring performance at an *entity-level*. However, multi-modal contexts in which robots interact with people, locations, and objects present an opportunity to explore a *global* resource management strategy for situations in which multiple consultants are needed.

Conclusion

In this paper, we investigated (1) how working memory modules based on the decay and interference models of forgetting could be integrated into a robot cognitive architecture, and (2) whether a robot cognitive architecture using these models would produce effective natural language in terms of human accuracy, ease of the listener's cognitive processing, perceived naturalness, and perceived human-likeness.

To do so, we use a cognitive architectural approach in which we (1) demonstrated how an *Entity-Based* model of WM can be integrated into a robot cognitive architecture, (2) demonstrated how different WM forgetting models can be implemented within that approach, and (3) experimentally validated this approach through a user study with an autonomous robotic cognitive architecture.

Through this work we were able to provide two key classes of insights for both technical and experimental cognitive HRI research: our architectural approach provides new insights into how these models can be integrated into robot cognitive architectures, and our experimental validation provides assurance that even with forgetting dynamics employed, robots can produce natural, accurate, and human-like behavior and language, regardless of which forgetting model is used or the way that model is parameterized.

Overall, this work provides the foundation for future technical work to explore a wider array of model parameterizations and resource management strategies, and for future experimental work to more acutely investigate the performance differences between these different possible approaches.

Acknowledgments

This work was funded in part by NSF CAREER grant IIS-2044865.

References

Attamimi, M., Ando, Y., Nakamura, T., Nagai, T., Mochihashi, D., Kobayashi, I., & Asoh, H. (2016). Learning word meanings and grammar for verbalization of daily life activities using multilayered multimodal latent dirichlet allocation and bayesian hidden markov models. *Advanced Robotics*, *30*(11-12), 806–824.

Baddeley, A. D. (1983). Working memory. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, *302*(1110), 311–324.

Baddeley, A. D. (1992). Working memory. *Science*, *255*(5044), 556–559.

Baddeley, A. D. (2000). The episodic buffer: a new component of working memory? *Trends in cognitive sciences*, *4*(11), 417–423.

Baddeley, A. D. (2010). Working memory. *Current biology*, *20*(4), R136–R140.

Baddeley, A. D., Gathercole, S. E., & Papagno, C. (1998). The phonological loop as a language learning device. *Exploring Working Memory*, 164–198.

Bannon, J. (2019). *Working memory and referential communication: An investigation of the cognitive factors affecting the production of overspecified referring expressions* (Unpublished master's thesis). McMaster University.

Barrouillet, P., Bernardin, S., & Camos, V. (2004). Time constraints and resource sharing in adults' working memory spans. *Journal of Experimental Psychology: General*, *133*(1), 83.

Baxter, P., & Browne, W. N. (2010). Memory as the substrate of cognition: A developmental cognitive robotics perspective. In *Epirob.*

Brown, J. (1958). Some tests of the decay theory of immediate memory. *Quarterly journal of experimental psychology*, *10*(1), 12–21.

Chella, A., & Pipitone, A. (2020). A cognitive architecture for inner speech. *Cognitive Systems Research*, *59*, 287–292.

Cowan, N. (2017). The many faces of working memory and short-term storage. *Psychonomic bulletin & review*, *24*(4), 1158–1170.

Culpepper, W., Bennett, T. A., Zhu, L., Sousa Silva, R., Jackson, R. B., & Williams, T. (2022). Ipower: Incremental, probabilistic, open-world reference resolution. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 44).

Dale, R., & Reiter, E. (1995). Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive science*, *19*(2), 233–263.

Daneman, M., & Merikle, P. M. (1996). Working memory and language comprehension: A meta-analysis. *Psychonomic bulletin & review*, *3*(4), 422–433.

Denhovska, N., Serratrice, L., & Payne, J. (2016). Acquisition of second language grammar under incidental learning conditions: The role of frequency and working memory. *Language Learning*, *66*(1), 159–190.

EbbingHaus, H. (1885). 1885: Memory: a contribution to experimental psychology. *Über das Gedächtnis: Untersuchungen zur experimentellen Psychologie*.

Gatt, A., Goudbeek, M., & Krahmer, E. (2011). Attribute preference and priming in reference production: Experimental evidence and computational modeling. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 33).

Gilchrist, A. L., & Cowan, N. (2011). Can the focus of attention accommodate multiple, separate items? *Journal of experimental psychology: learning, memory, and cognition*, *37*(6), 1484.

Giorgi, I., Cangelosi, A., & Masala, G. L. (2021). Learning actions from natural language instructions using an on-world embodied cognitive architecture. *Frontiers in Neurorobotics*, *15*, 48.

Goudbeek, M., & Krahmer, E. (2011). Referring under load: Disentangling preference-based and alignment-based content selection processes in referring expression generation. *Proceedings of PRE-Cogsci: Bridging the gap between computational, empirical and theoretical approaches to reference*.

Gundel, J. K., Hedberg, N., & Zacharski, R. (1993). Cognitive status and the form of referring expressions in discourse. *Language*, 274–307.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and brain sciences*, *21*(6), 803–831.

Ivanova, I., & Ferreira, V. S. (2019). The role of working memory for syntactic formulation in language production. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *45*(10), 1791.

Jackson, R. B., & Williams, T. (2022). Enabling morally sensitive robotic clarification requests. *ACM Transactions on Human-Robot Interaction (THRI)*, *11*(2), 1–18.

Jonides, J., Lewis, R. L., Nee, D. E., Lustig, C. A., Berman, M. G., & Moore, K. S. (2008). The mind and brain of short-term memory. *Annual review of psychology*, *59*, 193.

Kane, M. J., & Engle, R. W. (2003). Working-memory capacity and the control of attention: the contributions of goal neglect, response competition, and task set to stroop interference. *Journal of experimental psychology: General*, *132*(1), 47.

Kiyonaga, A., & Egner, T. (2013). Working memory as internal attention: Toward an integrative account of internal

and external selection processes. *Psychonomic bulletin & review*, *20*(2), 228–242.

Kurup, U., & Lebiere, C. (2012). What can cognitive architectures do for robotics? *Biologically Inspired Cognitive Architectures*, *2*, 88–99.

Kyllonen, P. C., & Christal, R. E. (1990). Reasoning ability is (little more than) working-memory capacity?! *Intelligence*, *14*(4), 389–433.

Laird, J. E., Kinkade, K. R., Mohan, S., & Xu, J. Z. (2012). Cognitive robotics using the soar cognitive architecture. In *Workshops at the twenty-sixth aaai conference on artificial intelligence.*

Laird, J. E., Lebiere, C., & Rosenbloom, P. S. (2017). A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *Ai Magazine*, *38*(4), 13–26.

Lin, J. (1991). Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, *37*(1), 145–151.

Lindes, P., & Laird, J. E. (2017). Cognitive modeling approaches to language comprehension using construction grammar. In *2017 aaai spring symposium series.*

Liu, M., Xiao, C., & Chen, C. (2022). Perspective-corrected spatial referring expression generation for human-robot interaction. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

Ma, W. J., Husain, M., & Bays, P. M. (2014). Changing concepts of working memory. *Nature neuroscience*, *17*(3), 347–356.

Martín, F., Rodríguez Lera, F. J., Ginés, J., & Matellán, V. (2020). Evolution of a cognitive architecture for social robots: Integrating behaviors and symbolic knowledge. *Applied Sciences*, *10*(17), 6067.

Miyazawa, K., Horii, T., Aoki, T., & Nagai, T. (2019). Integrated cognitive architecture for robot learning of action and language. *Frontiers in Robotics and AI*, *6*, 131.

Moulin-Frier, C., Fischer, T., Petit, M., Pointeau, G., Puigbo, J.-Y., Pattacini, U., ... others (2017). Dac-h3: a proactive robot cognitive architecture to acquire and express knowledge about the world and the self. *IEEE Transactions on Cognitive and Developmental Systems*, *10*(4), 1005–1022.

Muter, P. (1980). Very rapid forgetting. *Memory & Cognition*, *8*(2), 174–179.

Myachykov, A., Scheepers, C., Garrod, S., Thompson, D., & Fedorova, O. (2013). Syntactic flexibility and competition in sentence production: The case of english and russian. *The quarterly journal of experimental psychology*, *66*(8), 1601–1619.

Newell, A., & Simon, H. (1956). The logic theory machine–a complex information processing system. *IRE Transactions on information theory*, *2*(3), 61–79.

Nozari, N., & Novick, J. (2017). Monitoring and control in language production. *Current Directions in Psychological Science*, *26*(5), 403–410.

Oberauer, K. (2013). The focus of attention in working memory—from metaphors to mechanisms. *Frontiers in human neuroscience*, *7*, 673.

Oberauer, K. (2019). Working memory and attention—a conceptual analysis and review. *Journal of cognition*.

Oberauer, K., & Lewandowsky, S. (2011). Modeling working memory: A computational implementation of the time-based resource-sharing theory. *Psychonomic bulletin & review*, *18*(1), 10–45.

Oberauer, K., & Lewandowsky, S. (2014). Further evidence against decay in working memory. *Journal of Memory and Language*, *73*, 15–30.

Oberauer, K., Lewandowsky, S., Farrell, S., Jarrold, C., & Greaves, M. (2012). Modeling working memory: An interference model of complex span. *Psychonomic bulletin & review*, *19*(5), 779–819.

Pickering, M. J., & Garrod, S. (2014). Self-, other-, and joint monitoring using forward models. *Frontiers in human neuroscience*, *8*, 132.

Reiter, E., & Dale, R. (1997). Building applied natural language generation systems. *Natural Language Engineering*, *3*(1), 57–87.

Rodgers, S. M., Myers, C. W., Ball, J., & Freiman, M. D. (2013). Toward a situation model in a cognitive architecture. *Computational and Mathematical Organization Theory*, *19*(3), 313–345.

Rönnberg, J., Rudner, M., Lunner, T., & Zekveld, A. A. (2010). When cognition kicks in: Working memory and speech understanding in noise. *Noise and Health*, *12*(49), 263.

Scheutz, M. (2006). Ade: Steps toward a distributed development and runtime environment for complex robotic agent architectures. *Applied Artificial Intelligence*, *20*(2-4), 275–304.

Scheutz, M., Briggs, G., Cantrell, R., Krause, E., Williams, T., & Veale, R. (2013). Novel mechanisms for natural human-robot interactions in the diarc architecture. In *Workshops at the twenty-seventh aaai conference on artificial intelligence.*

Scheutz, M., Williams, T., Krause, E., Oosterveld, B., Sarathy, V., & Frasca, T. (2019). An overview of the distributed integrated cognition affect and reflection diarc architecture. *Cognitive architectures*, 165–193.

Schwering, S. C., & MacDonald, M. C. (2020). Verbal working memory as emergent from language comprehension and production. *Frontiers in human neuroscience*, *14*, 68.

Süß, H.-M., Oberauer, K., Wittmann, W. W., Wilhelm, O., & Schulze, R. (2002). Working-memory capacity explains reasoning ability—and a little bit more. *Intelligence*, *30*(3), 261–288.

Van Deemter, K. (2016). *Computational models of referring: a study in cognitive science*. MIT Press.

Waugh, N. C., & Norman, D. A. (1965). Primary memory. *Psychological review*, *72*(2), 89.

Williams, T. (2017). A consultant framework for natural language processing in integrated robot architectures. *IEEE Intell. Informatics Bull.*, *18*(1), 10–14.

Williams, T. (2019). A givenness hierarchy theoretic approach. *The Oxford handbook of reference*, 457.

Williams, T., Acharya, S., Schreitter, S., & Scheutz, M. (2016). Situated open world reference resolution for human-robot dialogue. In *2016 11th acm/ieee international conference on human-robot interaction (hri)* (pp. 311–318).

Williams, T., Johnson, T., Culpepper, W., & Larson, K. (2020). Toward forgetting-sensitive referring expression generationfor integrated robot architectures. *arXiv preprint arXiv:2007.08672*.

Williams, T., Krause, E., Oosterveld, B., & Scheutz, M. (2018). Towards givenness and relevance-theoretic open world reference resolution. In *Rss workshop on models and representations for natural human-robot communication.*

Williams, T., & Scheutz, M. (2015). Power: A domain-independent algorithm for probabilistic, open-world entity resolution. In *2015 ieee/rsj international conference on intelligent robots and systems (iros)* (pp. 1230–1235).

Williams, T., & Scheutz, M. (2016). A framework for resolving open-world referential expressions in distributed heterogeneous knowledge bases. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 30).

Williams, T., & Scheutz, M. (2017). Referring expression generation under uncertainty: Algorithm and evaluation framework. In *Proceedings of the 10th international conference on natural language generation* (pp. 75–84).

Williams, T., Thielstrom, R., Krause, E., Oosterveld, B., & Scheutz, M. (2018). Augmenting robot knowledge consultants with distributed short term memory. In *International conference on social robotics* (pp. 170–180).

Williams, T., Yazdani, F., Suresh, P., Scheutz, M., & Beetz, M. (2019). Dempster-shafer theoretic resolution of referential ambiguity. *Autonomous Robots*, *43*(2), 389–414.