Memory & Cognition (2019) 47:1415–1430
https://doi.org/10.3758/s13421-019-00942-4

Learning about things that never happened: A critique and refinement of the Rescorla-Wagner update rule when many outcomes are possible

Geoff Hollis[1]

Published online: 31 May 2019
© The Psychonomic Society, Inc. 2019

Abstract
A vector-based model of discriminative learning is presented. It is demonstrated to learn association strengths identical to the Rescorla–Wagner model under certain parameter settings (Rescorla & Wagner, 1972, *Classical Conditioning II: Current Research and Theory*, *2*, 64–99). For other parameter settings, it approximates the association strengths learned by the Rescorla–Wagner model. I argue that the Rescorla–Wagner model has conceptual details that exclude it as an algorithmically plausible model of learning. The vector learning model, however, does not suffer from the same conceptual issues. Finally, we demonstrate that the vector learning model provides insight into how animals might learn the semantics of stimuli rather than just their associations. Results for simulations of language processing experiments are reported.

Keywords Rescorla–Wagner model · Discriminative learning · Associative learning · Language acquisition · Lexical processing

Discriminative learning accounts for a broad range of phenomena in language learning and processing. This includes changes in cognitive performance due to aging (Ramscar, Hendrix, Shaoul, Milin, Baayen, 2014; Ramscar, Sun, Hendrix, & Baayen, 2017), attention times in statistical learning paradigms (Baayen, Shaoul, Willits, & Ramscar, 2016), and morphological, lexical, and *n*-gram processing effects (Baayen, 2010; Baayen, Hendrix, & Ramscar, 2013; Baayen, Milin, Đurđević, Hendrix, & Marelli, 2011; Shaoul, Baayen, & Westbury, 2014). These findings are surprising considering that the aforementioned work uses models from a domain that has had little influence on modern language research: animal learning.

Numerous animal models of discriminative learning have been proposed (e.g., Courville, 2006; Gallistel & Gibbon, 2000; Pearce, 1994; Sutton & Barto, 1981; Wagner, 1981). Within psychology, the most well-known of these is the Rescorla–Wagner (R–W) model (Rescorla & Wagner, 1972). The R–W model learns to predict outcomes from available cues based on error correction. Formally, the R–W update rule is as follows:

$$\begin{aligned} &\text{If outcome i present}: \Delta V_{x,i} = \alpha_x \beta_i \left(\lambda_i - V_{tot,i}\right); \\ &\text{otherwise}: \quad \Delta V_{x,i} = \alpha_x \beta_i \left(0 - V_{tot,i}\right), \end{aligned} \tag{1}$$

where λ_i denotes the maximum associability to Outcome i (by convention, 1), $V_{tot,i}$ denotes the summed association strength to i of all cues present, $\Delta V_{x,i}$ is the change in association strength between Cue x and Outcome i, α_x is a learning rate parameter in range [0, 1] tied to Cue x (conceptually, its salience), and β_i is a learning rate parameter in range [0, 1] tied to Outcome i (conceptually, its salience).

The R–W model updates association strengths between cues and outcomes both in the case that a particular outcome is present and in the case that it is absent. When observing the temporal contingency between standing at a bus stop and a bus arriving, the available cues—the bus terminal, other people standing about, a particular time of day—become informative of the outcome of a bus arriving. However, since you did not burn your hand, someone did not serve you lunch, and a dog did not walk past, association strengths between the available cues and those nonoutcomes also need to be updated (towards $\lambda = 0$). Consider that there are always far more possible outcomes that did not occur, but which you have knowledge of, than outcomes that did occur. Because of this, the R–W model implies that the bulk of computational effort involved in learning is directed toward learning about nonoutcomes. It seems implausible that, effectively, all of the computational work in learning is directed toward explicitly updating knowledge about things that never happened.

✉ Geoff Hollis
hollis@ualberta.ca

[1] Department of Computing Science, University of Alberta, 3-39 Athabasca Hall, Edmonton, AB T6G 2E9, Canada

The conceptual issue of the R–W model—that far more learning effort is directed toward outcomes that never happened than toward the outcome that did—brings with it a computational burden. Researchers can reduce this computational burden by delimiting between outcomes that are relevant and outcomes that are irrelevant. If a researcher is interested in studying learned associations between tones and food under various patterns of association, the researcher need not consider how the animal's knowledge about the relationship between tones and predators has been affected by the nonoccurrence of those outcomes. If researchers are interested in studying language learning, they can ignore all but linguistic stimuli and linguistic outcomes. Doing so substantially delimits the space of possible outcomes, reducing the computational burden of the R–W model.

The problem is not so simple to address from an animal's perspective. Researchers have the luxury of knowing ahead of time what they want to study and can thus delimit between what is relevant and what is irrelevant, but the animal must learn the boundary between relevant and irrelevant on its own. It is unclear how that boundary could be learned if the animal does not first start by considering all possible outcomes. Learning to delimit between what is relevant and what is irrelevant is a nontrivial problem and bears resemblance to the philosophical frame problem[1]: In a sufficiently rich environment, there is no tractably identifiable boundary between (1) knowledge that is relevant to a particular context, and thus needs to be updated through learning, and (2) knowledge that is irrelevant to a particular context, and thus can be left alone (Dennett, 2006; Moore, 1981; Pylyshyn, 1987; Wheeler, 2008).

To make the problem concrete, consider the following solution to the limitations of the R–W update rule: When one or more cues occur, only update their relationship to outcomes with nonzero association strength to at least one of the cues. Our world is highly structured, meaning that most cue → outcome associations will be zero. If an animal only had to update nonzero associations, this would greatly reduce the burden of updating knowledge about nonoccurring outcomes. But, how does an animal know an association is nonzero unless that fact is verified? This proposal does not save any computational effort unless it is accompanied by a mechanism for distinguishing between zero and nonzero associations that does not require explicit verification of those facts. Possible mechanisms (e.g., tracking nonzero associations as lists tied to various cues) invoke their own problems, either in terms of burdens to memory size or burdens to memory search (see Pylyshyn, 1987).

In what follows, I present the vector learning model (VLM). Under specific parameter settings, the VLM learns association strengths identical to the R–W model. However, it only needs to update association strengths for observed outcomes. I argue, because of this, the VLM is algorithmically more plausible than the R–W model. I also demonstrate that the VLM is informative of how animals could come to learn the semantics of stimuli rather than just associations between stimuli, a topic on which other models of animal learning models have provided little insight.

[1] Not the AI frame problem, although the two problems are related.

The vector learning model

Consider cues and outcomes as occupying points in an *n*-dimensional Euclidean space. If association strength is a function of proximity between a cue and an outcome, and if vectors for different outcomes are orthogonal to each other, then a cue gaining association strength with one particular outcome (i.e., its point moving toward that of the outcome) implies a loss of association strength with other outcomes (i.e., moving away from the point occupied by them).[2] Learning about nonoccurring outcomes happens implicitly while learning about observed outcomes. Based on this observation, we propose a learning rule to approximate the R–W update rule:

$$\Delta \vec{V}_x = \alpha_x \beta_i \left(\vec{\lambda} - \vec{V}_{tot} \right) \tag{2}$$

Cues and outcomes are represented as vectors in an *n*-dimensional space. Upon conditioning, a set of cues on an outcome, i, the vector for an individual cue that is present, x, (V_x) is updated based on the difference between the outcome vector (λ_i) and the addition of vectors for all cues present (V_{tot}). Updating is scaled by parameters representing salience of the cue and outcome, α_x and β_i, respectively, just like with the R–W model. A low-dimensional visual example of how this learning rule works, using a single cue, is depicted in Fig. 1.

In an *n*-dimensional space, the maximum set of orthogonal vectors is *n*. Thus, to represent two outcomes as orthogonal vectors, at minimum a two-dimensional space would need to be used. To represent 45,000 outcomes as orthogonal vectors, outcomes would need to occupy points in a 45,000-dimensional space. This does not, in fact, address the R–W model's frame problem. Instead of having to update association strengths with 45,000 outcomes (44,999 of which did not occur), the VLM is instead updating one vector of dimensionality 45,000, which is the same number of calculations; as the environment becomes richer, longer vectors are required to represent outcomes orthogonally. The VLM does not have issues distinguishing between relevant and irrelevant stimuli. Instead, it updates a single multi-dimensional vector after each learning episode.

[2] Extinction of a cue → outcome association is then a necessary and implicit consequence of a cue gaining association strength with another outcome that occupies a different region of space. This has similarities to Matzel, Held, and Miller's (1998) explanation of extinction.

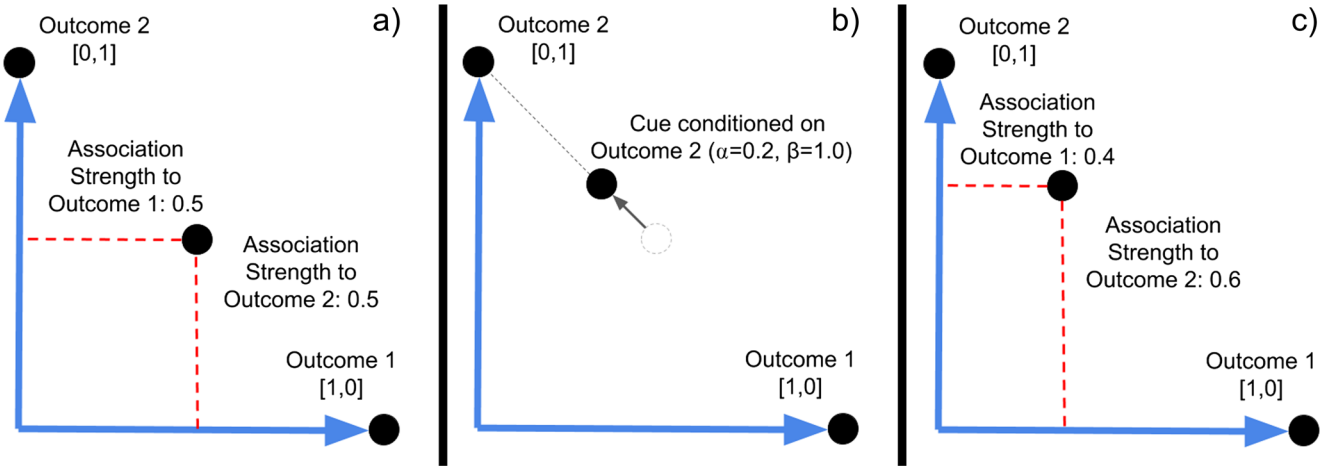

Fig. 1 The learning problem, as conceptualized by the vector learning model. Outcomes are points whose vectors are orthogonal to each other. Conditioning a cue on an outcome involves moving its point toward that of the outcome a proportion of the distance determined by α and β. Association strengths are determined by the proximity of a cue to an outcome. Because outcomes are orthogonal, when a cue moves toward an outcome and gains association strength with it, a loss of association with other outcomes also necessarily occurs. **a** Cue initially has association strength of 0.5 with both outcomes. **b** Cue is conditioned on Outcome 1. **c** Cue now has association strength of 0.6 with Outcome 1 and 0.4 with Outcome 2

Although vectors of dimensionality n are required to represent n outcomes orthogonally, it is possible to generate much more than n near-orthogonal vectors through random sampling (e.g., Jones & Mewhort, 2007). Consider this in terms of correlation: Two random sequences of numbers are, on average, uncorrelated with each other. If those sequences are considered as vectors, then the direction of those vectors will, on average, be orthogonal to each other.

Instead of requiring outcome vectors to be orthogonal, the same amount of near-orthogonal vectors can be generated through random sampling in a lower dimensional space. After proving the equivalence of association strengths learned by the two models when orthogonal vectors are used, we will demonstrate that using near-orthogonal vectors allows for a computationally efficient approximation of the Rescorla–Wagner update rule when many outcomes are possible.

Formal equivalence of learned association strengths

We define association strength between a cue and an outcome as the sum of the cue vector elements multiplied by the sign of the outcome vector elements, divided by the sum of the absolute values of the outcome vector elements (see Equation 3). Association strength is a function of whether the signs of cue elements match the signs of outcome elements, weighted by the magnitudes of those elements.

$$V_{cue,outcome} = \sum cue_i * sign\,(outcome_i) / \sum abs\,(outcome_i). \quad (3)$$

If a cue vector had values [0, 0.3, 0.2] and an outcome vector had values [0.5, 0.5, 0.5], the association strength between cue and vector would thus be (0 * 1 + 0.3 * 1 + 0.2 * 1) / (0.5 + 0.5 + 0.5) = 0.33. If the sign of the last element of the outcome vector were flipped, such that the vector was now [0.5, 0.5, −0.5], association strength would instead be (0 * 1 + 0.3 * 1 + 0.2 * −1) / (0.5 + 0.5 + 0.5) = 0.066. Using this second outcome vector and a cue vector where the sign of the third element is also flipped (i.e., [0, 0.3, −0.2]), the association strength would again be (0 * 1 + 0.3 * 1 + −0.2 * −1) / (0.5 + 0.5 + 0.5) = 0.33.

This way of representing cues, vectors, and their associations has the property that associations $V_{cue1,outcome} + V_{cue2,outcome} = V_{cue1\ +\ cue2,\ outcome}$. For purposes of calculating the total association between all available cues and an outcome, the vector for each cue can be added, and association to an outcome can be calculated over the composite cue vector. We have observed empirically, but not yet proven formally, that the proposed measure of association strength is equal to the ratio of magnitudes of the composite cue vector to the composite outcome vector, multiplied by the cosine angle between the composite cue vector and the composite outcome vector. This may be of note to the fields of information retrieval and computational semantics where it has been observed that cosine angle provides a useful measure of association strength between entities represented as points in geometric space.

Consider the case of an animal learning to expect the presence or absence of a single outcome based on available cues. The Rescorla–Wagner model treats the presence of the outcome as a single value, $\lambda = 1$. Its absence is also treated as a single value, $\lambda = 0$. Prior to being conditioned on an outcome, the association strengths of all cues are assumed to be $V = 0$.

Single values are one-dimensional vectors. We can restate this example by saying that the outcome is represented as the vector, [1], the absence of the outcome is represented as the

vector, [0], and the initial value of a cue is [0]. We can then use the update rule presented in Equation 2 to update the association strength between cues and outcome, conditioning cues on [1] when the outcome is present and on [0] when the outcome is absent. Formally, this is identical to applying the R–W update rule from Equation 1.

Next, consider the case where a cue might be conditioned on one of *n* outcomes. Let us represent each outcome as a vector of dimensionality *n*, such that element i of outcome vector i is equal to 1, and all other elements are equal to 0. With this representation, all outcome vectors are orthogonal to each other. Let us initialize cues as vectors of dimensionality *n* where all elements are initially 0. We will apply Equation 2 to update a cue vector when it is associated with an outcome. We will now demonstrate that association strengths learned by such a model are identical to those that would be learned by the R–W model.

The association strength between a cue and outcome i is simply the cue's value for element i. This is because element i in the outcome vector is the only nonzero value, and that value has a positive sign. Since all of the other elements are zero, any nonzero value in the cue vector at an index other than i will be multiplied by zero prior to summation along all dimensions (see Equation 3).

Now, note that outcome i is also the only outcome with a nonzero value for element i. Any time a cue is conditioned on outcome i, its value for element i will be updated toward 1. This is because 1 is the value of outcome i's element i, and any time a cue is conditioned on an outcome other than I, its value for element i will instead be updated toward 0, because all other outcomes have a value of 0 at element i. When applying the R–W update rule with $\lambda = 1$, this is exactly how the association strength between a cue and an outcome changes depending on the presence or absence of that outcome. Since all cues start with initial values of 0 for all elements, the strength of association between a cue and outcome i learned by the R–W model after any arbitrary pattern of conditioning will be identical to element i of a cue's vector after the same pattern of conditioning. Since we have already established that the association strength between a cue and outcome i is simply the cue's value for element i, we have now proven that there exists a version of the VLM that learns association strengths identical to that of the R–W model when $\lambda = 1$. This holds for any arbitrary number of outcomes and any arbitrary pattern of conditioning.

Reducing vector dimensionality

The above equivalence requires that outcome vectors are orthogonal and, hence, of dimensionality equal to the number of possible outcomes. We now demonstrate that the VLM also approximates the association strengths learned by the R–W model when using near-orthogonal vectors in a reduced-dimension space. Use of a reduced-dimension space is relevant to discussing how the frame problem applies to the VLM.

We adopt the convention of initializing outcome vectors such that their elements are sampled uniformly from [−1, 1] and then unit normalized. Cue vectors are initialized such that all of their elements are zero.

Baayen (2010) has demonstrated that the R–W model can be used to simulate lexical processing. Language learning was simulated by conditioning letters and bigrams on the words they appeared in over a corpus of written text. For the word *fair*, the cues *f, a, i, r, fa, ai,* and *ir* would be conditioned on the outcome, *fair*. Lexical processing times were then simulated as a function of word activation, given word cues, on the assumption that lexical processing is being driven by the strength of bottom-up support for a specific meaning that is available in the stimulus environment. We replicate this simulation using the R–W model and the VLM and then compare the association strengths learned by both models.

The R–W model and the VLM were both trained on the TASA corpus (Landauer, Foltz, & Laham, 1998). It contains introductory paragraphs from a broad sample of K–12 textbooks. The TASA corpus was first preprocessed by converting all words to lowercase and stripping all punctuation except for apostrophes and dashes placed between two letters. Models were trained on all words that occurred at least 10 times in the corpus. This included 10,764,128 tokens and 29,056 unique types. For each token that appeared in the corpus, letters and bigrams within the token were conditioned on its type. Training occurred in order of appearance of each token. Versions of the VLM were trained with vector dimensionalities of 100; 200; 400; 800; 1,600; 3,200; 6,400; 12,800; 25,600; and 29,056. Outcome vectors were created through random generation following the method described above. One additional case was tested using orthogonal outcome vectors (dimensionality = 29,056). Models were trained with $\alpha = 0.02$ and $\beta = 1.0$. Word activations, given cues present in the word, were compared between the R–W model and the VLM. Results are presented in Fig. 2.

Three things should be noted from these results. First, when orthogonal outcome vectors are used, predictions made by the VLM are identical to those of the R–W model (see Fig. 2d; *MSE* = 0.0). Second, lowering vector dimensionality reduces the computational cost of the model, but also reduces fidelity at replicating the R–W model's predictions (see Fig. 2a). Third, even if vector dimensionality is aggressively reduced, fidelity can still be quite high. At vector dimensionality 1,600 (5.51% of maximum dimensionality), word activations from the VLM correlate with those of the R–W model at $r[29{,}054] = 0.97$ ($p < 2.2e\text{-}16$), accounting for 94.1% of the variance in R–W activations.

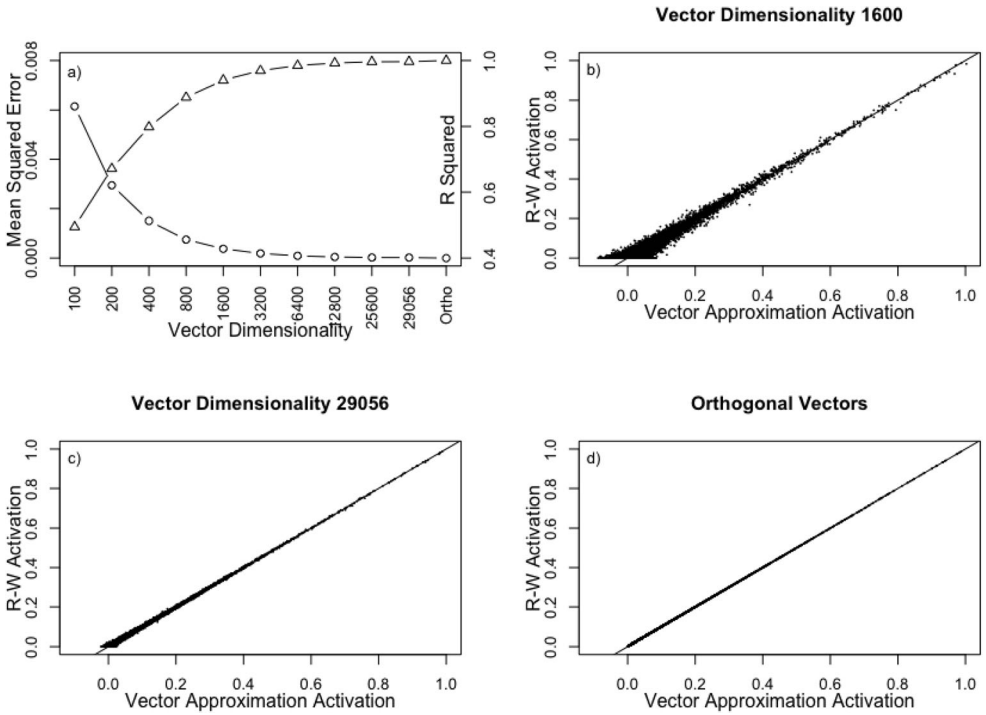

Fig. 2 Comparison of predictions made by the R–W model and the VLM on a simulated lexical decision task. Models were trained by conditioning letters and bigrams on words in the TASA corpus. **a** Similarity in word activations given their letter and bigram cues are plotted for the R–W model and the VLM. Similarity was measured by mean square error (dots) and *r* squared (triangles). Data are plotted for vector dimensionalities ranging from 100 to 29,056. For each unique word type that occurred in the training data, VLM activation for all words is plotted against R–W activation (vector dimensionality 1,600) (**b**), with nonorthogonal vectors of dimensionality 29,056 (**c**), and with orthogonal vectors of dimensionality 29,056 (**d**)

Discussion

We have presented a model of discriminative learning that approximates association strengths learned by the R–W model. (Python code that implements this new model can be found at https://github.com/hollisgf/reswag) .

In an environment that approaches the diversity of everyday experience, there will always be far more outcomes that did not occur than outcomes that did during any given learning episode. In sufficiently rich environments, the way the R–W model frames learning implies that effectively all updates to knowledge are for cue-outcome pairs that were unobserved. This seems like a dubious property for a model of learning.

The R–W model has plausibility as a *computational* model of learning (in the sense of Marr & Poggio's, 1976, levels of analysis), evidenced by the wide range of behavioral phenomena it captures, but it is lacking in *algorithmic* plausibility. Within the R–W model, there is no computationally tractable way for a learning agent to distinguish between knowledge that is and is not relevant to a particular learning context and, thus, does or does not need to be updated based on a particular learning episode. Knowledge about all outcomes must be updated after any outcome occurs, but the space of all outcomes is intractably large for an animal embedded within a diverse environment.

The VLM updates knowledge only using information about observed outcomes. The VLM provides a way of conceptualizing how a discriminative learning model could work in a more algorithmically plausible way, eschewing negative evidence by reframing learning as a geometric problem. The VLM also provides a means of trading off between learning fidelity (with reference to the R–W model) and learning efficiency. The VLM does this by using stimulus representations that can be made more or less informationally rich by lengthening or shortening them. The VLM can save substantially on computational effort by shortening its vector representations; when low-dimensional vectors are used, fewer prediction errors need to be calculated during each learning episode. Using vectors with as low as ~5% of the maximum possible dimensionality incurs surprisingly little loss in fidelity in the above example. These findings are interesting in terms of providing a reformulation of how to compute association strengths between stimuli. However, they do not directly address the frame problem.

To address the frame problem, we need to study how vector dimensionality affects predictive validity in complex learning environments, not how well vector dimensionality allows for reduplication of association strengths learned by the R–W model. If higher dimensional vectors always provide more precise predictions of learning data, then the vector model

does not address the frame problem; such a scenario would imply that the model parameters that best capture learning are the parameters that make for an intractable update rule. If, however, there is a vector dimensionality below the maximum that provides best fits to learning data, and that dimensionality is relatively short, then that would be evidence that the VLM does address the frame problem.

Experiment 1

This experiment has two main purposes. First, to provide evidence that increases to vector dimensionality do not necessarily increase the precision of modeling human learning in diverse environments. The second purpose is to test predictions about learning generated by the VLM. We start by providing an interpretation of what the vectors of the model correspond to in the world. We then use this interpretation to generate predictions of the model that are not made by the R–W model.

Properties of the vector learning model

Within the VLM, a single stimulus might have two vectors referring to it—one for when the stimulus acts as a cue and one for when it acts as an outcome. This distinction between an outcome vector and a cue vector is similar to a distinction made by the BEAGLE model of semantic memory (Jones & Mewhort, 2007). BEAGLE is a model that learns semantic representations for words. Each word has an environment vector, a context vector, and an order vector (the latter will be ignored in this discussion). A word's environment vector represents the physical properties of that word. A word's context vector is the accumulation of environment vectors of other words it occurs with. After many exposures to samples of language, context vectors become weighted sums of the contexts in which the represented word is expected to occur when it does occur. Contexts in this case are other words. Critically, the cosine similarity between context vectors provides information about the degree to which two words share similar contexts (and thus give information about similarity in meaning; Firth, 1957). Measures of similarity between context vectors and environment vectors provide information about the degree to which one word (context vector) predicts the occurrence of another (environment vector).

An interpretation of the VLM's functioning is available when considered in comparison to how BEAGLE frames learning: Outcome vectors (environment vectors in BEAGLE) are encodings of perceptual state when the outcome is observed, whereas cue vectors (context vectors in BEAGLE) are predictions about an animal's future perceptual state. The VLM's update rule aligns predictions of the world with observations of the world by minimizing prediction error encountered during learning episodes.

In the rest of this article, the terms *cue* and *outcome* are used to refer to stimuli that an animal would experience. The terms *expectation vector (EV)* and *state vector (SV)* refer to the vectors representing an animal's expectations and experience, respectively. To reiterate, a single stimulus might have two vectors referring to it—one encoding the perceptual details of that stimulus and the other encoding expectations that the stimulus generates. The cue-outcome/EV-SV distinction is relevant; I demonstrate that it is beneficial and theoretically motivated to update a cue's EV based on a combination of an outcome's SV *and* EV.

Previous sections focused on how the VLM can reproduce the R–W model's predictions about association strengths. However, the above interpretation of the VLM leads to predictions about learning that go beyond the functionality of the R–W model. Some of those predictions follow.

The VLM learns relations between cues even if the cues themselves never occur within the same learning episode. This is a consequence of cue EVs existing in the same geometric space. As one cue is learned about, the relationship of its EV necessarily changes to all other cue EVs. By learning that a cue produces certain expectations, it is also learned that those expectations are similar to or different from the expectations generated by other cues.

EVs are encoding predictions about future state, so the degree to which two EVs share association strength will be informative of the overlap in their distribution of outcomes. The distributional hypothesis, which is one of the main foundations of computational semantics, states that the degree to which two symbols' contexts of occurrence are similar is an indicator of the degree to which those symbols have similar meanings (Firth, 1957). We thus predict that two stimuli will have similar meanings to the extent that their EVs share association strength. This is a prediction that is beyond the scope of the R–W model, since the R–W model provides no means of assessing relationships between cues.

Second, so far we have only considered conditioning an EV on a, SV. When a stimuli's, s_1, EV is conditioned on s_2's SV, effectively the model is learning to "look ahead one step into the future." However, a further glimpse into the future is also available to s_1 via s_2's EV since that EV is an expectation about future state after s_2 occurs. The foresight of the VLM can be changed by conditioning a cue's EV on a weighted sum of the outcome's SV *and* EV. The more weight given to the outcome's EV, the more the model will come to value the distant future when learning to make predictions about future state. The R–W model does not allow for a way to learn from expectations rather than observations. We anticipate that this property of the VLM will aid learning in environments where there are complex temporal contingencies between stimuli (e.g., in language).

Third, we predict that relaxing the near-orthogonality constraint for SVs can be beneficial for learning. Johns and Jones (2011) have demonstrated that learning can be facilitated by allowing BEAGLE's environment vectors to share similarity to the extent that the physical properties of the encoded words (i.e., phonology, spelling) are similar. The VLM can add similar functionality by representing SVs as the summation of vectors that reference subcomponents of the full stimulus. For example, the SV for *the* could be the summation of three vectors, one for each of *t, h,* and *e*. This should allow the VLM to generalize associations between a cue and a set of outcomes that share features. Allowing SVs to be nonorthogonal does not provide the VLM with new functionality over the R–W model (see compound stimuli in, e.g., Rescorla & Wagner, 1972). Rather, it allows for generalizations to be learned in a way that is internally consistent with the interpretation of SVs as encoding information about a real-valued perceptual state.

To demonstrate the functionality of the VLM, we report its performance on various measures of language-based knowledge.

Method

We take the stance that language learning is an associative process over the stimuli of language, and cues and outcomes are determined by temporal order. To simulate this process, the VLM was exposed to contiguous sequences of real-world language. Each successive pair of words in a sequence were treated as a cue–outcome pair, conditioning the preceding word on the following word.

Corpus

We use the TASA corpus (Landauer, Foltz, & Laham, 1998) as model input. The TASA corpus was preprocessed by converting all words to lowercase and stripping all punctuation with the exception of apostrophes and dashes placed between two letters. Words that occurred fewer than 10 times were removed from the corpus. This produced a corpus containing temporally ordered streams of written words, blocked into documents (each document coming from a different textbook, or textbook chapter). The corpus contained 10,764,128 tokens and 29,056 unique types.

Model parameters

Parameters were systematically manipulated to test model predictions. Vector dimensionalities of 100; 200; 300; 400; 800; 1,600; 3,200; 6,400; and 12,800 were used.

SVs were constructed using two methods: either they were randomly generated (see Reducing Vector Dimensionality section) or they were constructed so that words with similar spellings had similar SVs. For this second method, a random vector was generated for every possible three-letter sequence. A word's SV was the addition of vectors for all three-letter sequences contained within that word. Spaces at the beginning and ending of a word were also treated as a letter in that word. We will denote spaces as #. For example, *#dog#*'s SV would be the addition of vectors for *#do, dog,* and *og#*. This vector would have some correlation with the vector for *#frog#* due to the shared trigram, *og#*. We call this method of vector construction the *orthographically correlated* method because it creates a dependency between a word's form and its state vector.

Finally, a cue's EV was conditioned on a weighted addition of the following word's SV and EV. The outcome's SV was always given a weight of 1.0. The weight of the outcome's EV was manipulated within the range of 0.0 to 1.0 (inclusive), in steps of 0.2. For example, if the sequence *the boy* were encountered and the outcome EV weight was set to 0.4, the EV for *the* would be conditioned on 1.0 times the SV for *boy*, plus 0.4 times the EV for *boy*.

Since the construction of SVs is stochastic, each parameter set was run independently 20 times. All models were trained on the exact same corpus, experienced in the exact same order.

Model evaluation

Models were evaluated on three tests of language-based knowledge. These tests are meant to evaluate the extent to which a particular model has acquired structure in its knowledge (i.e., EVs) that approximates the organization of human knowledge.

TOEFL The test of English as a foreign language (TOEFL) was used by Landauer and Dumais (1997) to evaluate the quality of knowledge acquired by their model, latent semantic analysis (LSA). The TOEFL consists of 80 questions. Each question has a reference word (e.g., *enormously*) and four options (e.g., *appropriately; uniquely; tremendously; decidedly*). Test-takers are instructed to choose the option that is most similar in meaning to the reference. LSA also represents stimuli as vectors. It answered questions by choosing the option whose vector had the highest cosine similarity to the reference's vector. The VLM was tested on the TOEFL, using association strength to select among options; the option whose EV had the highest association strength to the reference's EV was chosen.

Word similarity A common method for evaluating models that learn word meanings as vectors is to measure how closely cosine similarities between vectors capture variation in human judgments of word similarity. One of the most widely used similarity norms is Wordsim353 (Finkelstein et al., 2002), which contains 353 words pairs and human judgments of similarity for those pairs. The VLM was tested on its ability to

capture human judgments of word similarity by measuring the cosine similarity between EVs of the word pairs present in Wordsim353.

Priming in lexical processing Priming effects are pervasive in psychology. If we assume that priming effects are due to physical or semantic similarity (e.g., spreading activation; Collins & Loftus, 1975), or instead assume that they reflect enhanced memory availability due to expectations of future state generated by available cues (e.g., needs probability; Anderson & Milson, 1989), then we should hope that models of lexical semantics will capture priming effects in lexical processing. It is thus vexing that vector-based semantic models offer little to no predictive validity of priming effects in lexical processing tasks (e.g., Hutchison, Balota, Cortese, & Watson, 2008; but cf. Johns, Jones, & Mewhort, 2016).

The data set of 300 prime–target facilitation effects for word naming and lexical decision times released by Hutchison et al. (2008) were modeled. Association strength between EV–SV (prime–target) pairs were used as the measure of facilitation. EV–SV associations were used for this task rather than EV–EV associations, under the assumption that priming effects reflect animals generating expectations about future environmental state and preparing to act effectively in that future environmental state (Anderson & Milson, 1989).

Alternative models The VLM is not intended to be a model of *just* language learning. Its relationship to the R–W model suggests it should be applicable to modeling a broader set of learning problems. However, we subject the model to tests of lexical processing because that is a domain where many learning outcomes are present and data for modeling are readily available.

That said, we were interested in comparing the VLM's performance to that of popular lexical-semantic models in the literature: LSA (Landauer & Dumais, 1997), BEAGLE (Jones & Mewhort, 2007), skip-gram, and CBOW (Mikolov, Chen, Corrado, & Dean, 2013). Like the VLM, each of these models represent stimuli as vectors of real-valued numbers.

Each of these models was also trained on the TASA corpus and subjected to the above three tests. Their parameters were not optimized. These models each have some reasonable ranges of parameter settings. In each case, parameter values were chosen to be within these ranges, as judged by the first author.

LSA parameter set Vectors of 300 dimensionality were used. The association window spanned an entire document.

BEAGLE parameter set Vectors of 1,024 dimensionality were used. Context and order vectors were used to construct memory vectors over which associations were performed. The association window was a full sentence. Two-gram, three-gram, and four-gram sequences were used to construct order vectors.

Skip-gram parameter set Vectors of 300 dimensionality were used. The association window was four words to either side of the target. The association window did not span beyond the beginning or end of sentences. The down-sampling parameter for high-frequency words was set to 1e-5. Ten negative samples were used in each learning episode.

CBOW parameter set The same parameters were used for constructing the CBOW model as were used for constructing the skip-gram model. The details of each of the model's parameters goes beyond the scope of this work. If readers would like to understand what these parameters do, they are referred to articles describing these models: Landauer and Dumais (1997), Jones and Mewhort (2007), and Mikolov et al. (2013). Parameter values are presented here only to aid replication.

Results

TOEFL

Results for the TOEFL test are displayed in Fig. 3. There are four results of note. First, the VLM's performance plateaus at a vector dimensionality of approximately 300 (1.03% of the maximum dimensionality of 29,056; Fig. 3a). We have previously demonstrated that increases to vector dimensionality increase the fidelity of the vector model insofar as it correctly reproduces association strengths learned by the R–W model. However, these results provide evidence that increased fidelity of R–W association strengths does not translate to better prediction of behavioral data. A 2 × 9 × 6 (Vector Type × Vector Dimensionality × Memory Weight) ANOVA was conducted on model performance. A reliable effect for vector dimensionality was found, $F(8, 2095) = 21.05$, $p < 2e\text{-}16$. However, when models with vector dimensionality less than 300 were omitted from the analysis, this effect was no longer observed, $F(6, 1633) = 1.13$, $p = .34$. Increases in vector dimensionality beyond 300 do not appear to influence model performance on the TOEFL test.

Second, the VLM performed better when SVs were constructed such that words with similar spellings had similar SVs (see Fig. 3a), $F(1, 2095) = 214.53$, $p < 2e\text{-}16$. This result was expected and is interpreted as a benefit of generalizing a learned cue–outcome association to outcomes with similar spellings.

Third, increasing the weight of the outcome's EV produces a complex pattern of results. It was anticipated that increasing the outcome's EV weight would improve model performance. This prediction was made on the basis that increasing the EV weighting allows the model to learn temporally distant associations (by allowing both observations *and* expectations to drive learning), and that temporally distant associations are characteristic of the structure of language. We expected that

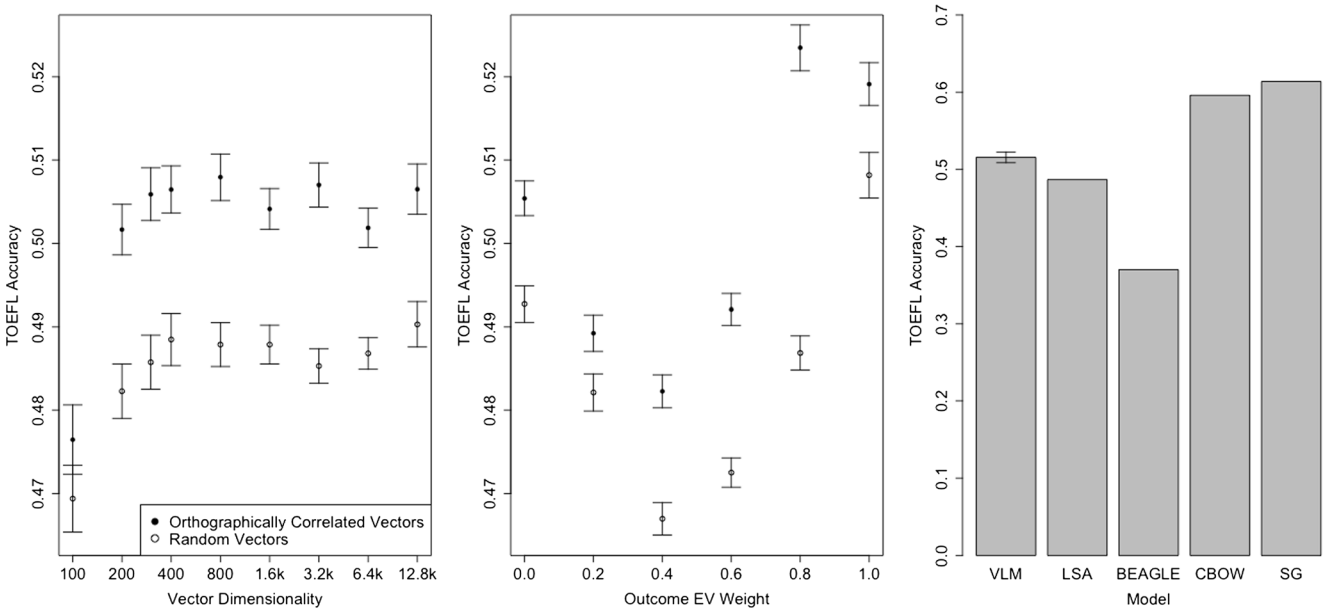

Fig. 3 Changes in model performance on the TOEFL test when (**a**) vector dimensionality is manipulated and (**b**) outcome EV weight is manipulated. Results are presented for two different methods of SV construction. **c** Comparison with four distributional semantic models. Results are reported for the 20 runs where VLM parameters were set to vector dimensionality 800, outcome EV weight 0.8, and orthographically correlated SVs. Error bars represent ±2 standard errors. *Note.* Landauer and Dumais (1997) report accuracies of 54% on this test when trained on the same corpus. This discrepancy has to do with how models were tested. There were some questions for which models had no entries for the relevant words, due to them not appearing in the TASA corpus. Landauer and Dumais assigned their model a correct response for 25% of these questions (i.e., assumed the model guessed). These questions were instead omitted from the analysis

increasing outcome EV weight would improve model performance (i.e., giving more weight to future expectations rather than just observed outcomes) up to a threshold, and then begin to decrease model performance (too much weight given to distant expectations). Instead, we observe a clear U-shape pattern. The model performs best with a large weight being given to the outcome's EV (0.8–1.0), suggesting that weighting the outcome's EV is in fact beneficial for some weights. However, diminished performance was observed for middling weights compared with no weighting at all. The effect for outcome EV weighting was reliable, $F(5, 2095) = 109.37$, $p < 2e\text{-}16$.

Fourth, the model's overall performance on the TOEFL is relatively good compared with the other models tested. It does not achieve the same accuracies as CBOW and skip-gram, which are the best performing semantic models across a range of tasks in natural language processing (e.g., Baroni, Dinu, & Kruszewski, 2014), but it does have better performance than LSA and BEAGLE, which are more commonly seen in the psychological literature. We stress that model parameters were only optimized for the VLM; it could be that better performance would have been seen for other models with parameter optimization.

Wordsim

There were some notable similarities and differences between the TOEFL results and the results for predicting Wordsim353 similarity measures (see Fig. 6 for Wordsim353 results). First, a plateau in the predictive validity gained from increased vector dimensionality is again seen. There is a clear effect of vector dimensionality when all dimensionalities are considered, $F(8, 2095) = 10.18$, $p = 5e\text{-}14$. However, this effect disappears when considering just vector dimensionalities of 800 and above, $F(4, 1159) = 1.33$, $p = .26$. This is still a small fraction of the maximum vector dimensionality of 29,056 (2.75%).

Second, there is again a clear effect for the way SVs were constructed. The model has better performance at predicting similarity judgments when SVs were constructed such that words with similar spellings had similar vectors, $F(1, 2095) = 2770.01$, $p < 2e\text{-}16$.

There is also a clear effect for the weighting given the outcome's EV when conditioning a cue on that outcome, $F(5, 2095) = 547.51$, $p < 2e\text{-}16$. Unlike with the TOEFL data, the anticipated pattern was seen with the Wordsim353 pattern: EV weighting appears to have a monotonic effect on predictive validity, possibly up to a threshold that exceeds the maximum EV weight tested in this experiment (Fig. 4).

Finally, the poor performance of the VLM at predicting word similarities should be noted. LSA, CBOW, and skip-gram models all perform reasonably well given the size of the corpus on which they were trained (rs range between .37 and .48). The VLM ($r = .12$) and BEAGLE ($r = .082$) perform much poorer by this test. All models were trained on the same input; variation in predicting word similarity must then stem from variations in the learning algorithms and their parameter settings.

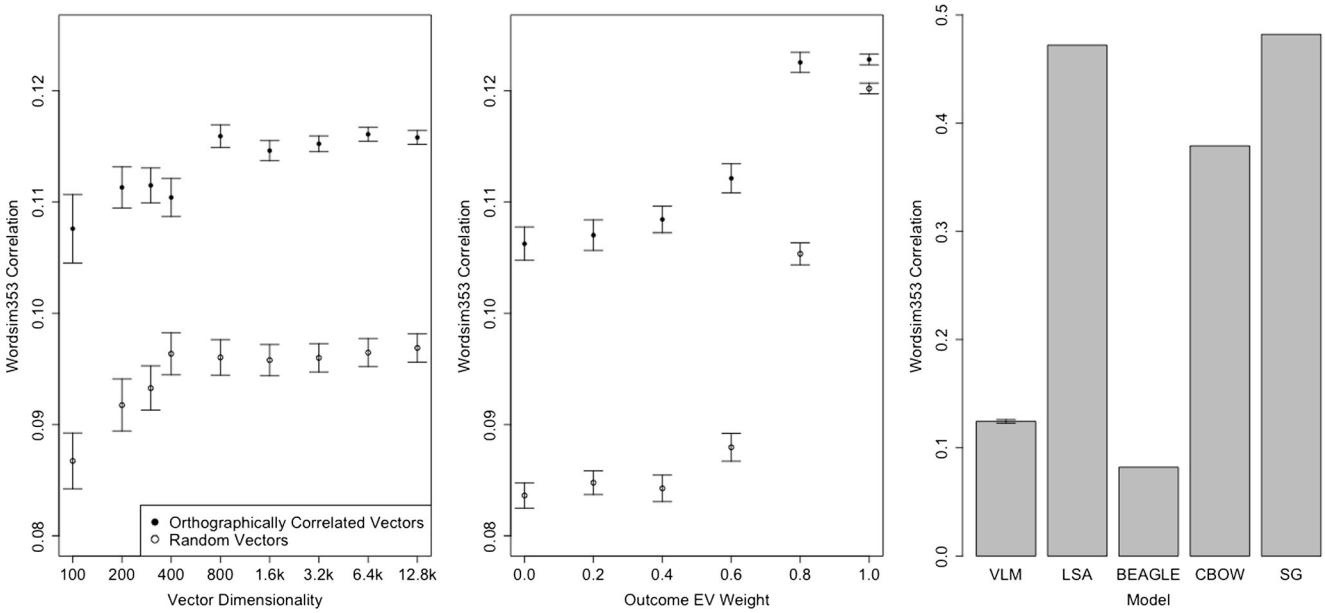

Fig. 4 Changes in model performance on predicting word similarity judgments when (**a**) vector dimensionality is manipulated, and (**b**) outcome EV weight is manipulated. Results are presented for two different methods of SV construction. **c** Comparison with four distributional semantic models. Results are reported for the 20 runs where VLM parameters were set to vector dimensionality 800, outcome EV weight 0.8, and orthographically correlated SVs. Error bars represent ± 2standard errors

Predicting priming effects

The results of the priming analysis were substantially different from either of the other two tests. Results for word-naming data are displayed in Fig. 5. Lexical decision results are displayed in Fig. 6. Only results for word naming are discussed; the pattern of results are similar for both measures.

The first difference is that the model performs better at predicting priming effects if SVs are *not* correlated with word orthography, $F(1, 2095) = 631.86$, $p < 2e\text{-}16$. Upon visual

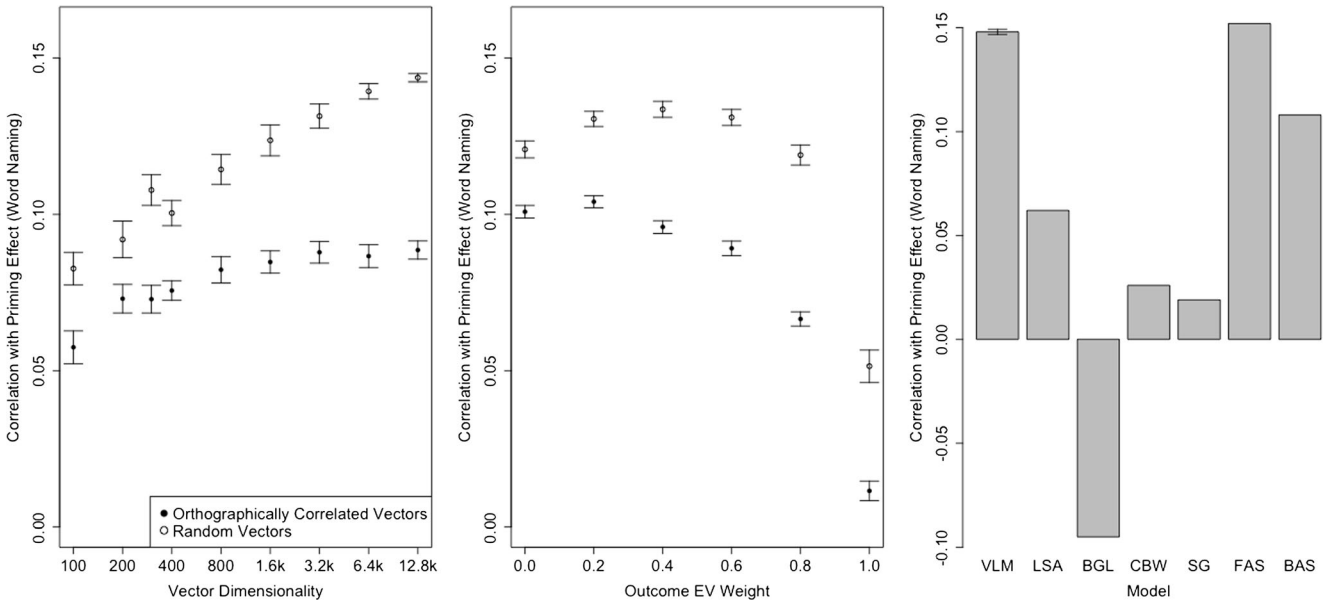

Fig. 5 Changes in model performance on predicting the magnitude of priming effects in word naming when (**a**) vector dimensionality is manipulated, and (**b**) outcome EV weight is manipulated. Results are presented for two different methods of SV construction. **c** Comparison with four distributional semantic models, as well as measures of forward and backward association strengths. Results are reported for the 20 runs where VLM parameters were set to vector dimensionality 12,800; outcome EV weight 0.4; and random SVs. Error bars represent ±2 standard errors

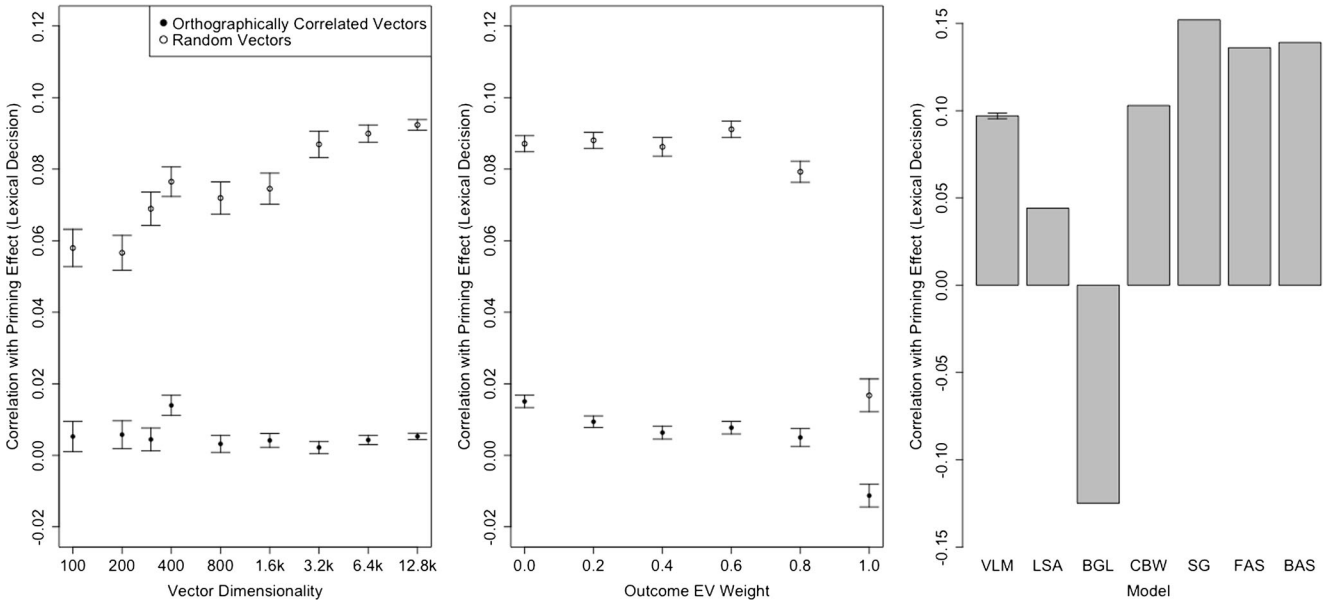

Fig. 6 Changes in model performance on predicting the magnitude of priming effects in lexical decision when (**a**) vector dimensionality is manipulated, and (**b**) outcome EV weight is manipulated. Results are presented for two different methods of SV construction. **c** Comparison with four distributional semantic models, as well as measures of forward and backward association strengths. Results are reported for the 20 runs where VLM parameters were set to vector dimensionality 12,800; outcome EV weight 0.4; and random SVs. Error bars represent ±2 standard errors

inspection, the effect of vector dimensionality does appear to plateau when orthographically correlated SVs are used, but not when random vectors are used. When considering only vectors of dimensionality 800 or more, there is an effect for dimensionality, $F(4, 1145) = 21.61$, $p < 2e\text{-}16$. There is also an interaction with SV construction method, $F(4, 1145) = 9.64$, $p = 7e\text{-}08$. Figure 5 suggests the effect may exist for random vectors, but not for orthographically correlated vectors. However, when only orthographically correlated vectors are considered, the effect remains reliable, $F(4, 573) = 2.72$, $p = .029$.

The effect for outcome EV weight also appears to be qualitatively different compared with previous tests. The effect is reliable, $F(5, 2095) = 350.14$, $p < 2e\text{-}16$, and suggests a reduction in performance as outcome EV weight is increased. The function of this parameter is to give the model "look-ahead" into the future by conditioning it on its own expectations. The decrement in performance may indicate that priming effects reflect immediate-future expectations, not distant-future expectations.

Finally, although the model has poor *absolute* performance at predicting priming effect magnitudes, it does have good *relative* performance compared with other models (see Figs. 7c and 8c). These figures include correlation strengths for forward association strength (FAS) and backward association strength (BAS) taken from Nelson, McEvoy, and Schreiber (2004), which Hutchison et al. (2008) report as their best predictors of item-level priming effects. Our results suggest that the VLM predicts priming effects in word naming about as well as forward association strength. The model's relative performance is not as high for lexical decision priming effects, but still substantially better than the two of four models that are more common in the psychological literature (LSA and BEAGLE). In fact, BEAGLE predicts effects to be in the opposite direction than they are empirically observed to be.

Discussion

The VLM was motivated by the observation that the R–W model suffers from the frame problem. The R–W model provides no delimitation between knowledge that needs to be updated after a learning episode and knowledge that can be let alone. The VLM was presented as a possible solution to this problem. It addresses the problem by eliminating the need to update association strengths to nonoccurring outcomes when a cue → outcome contingency is observed. The VLM's learning is driven entirely by observed outcomes.

It is possible that the VLM may simply be trading off one computational problem for another. The VLM has complex stimulus representations (i.e., high-dimensional vectors). Using maximum dimensionality, the vector model requires just as much computational effort to update as the R–W model does; rather than updating n association strengths, the model instead updates a vector of dimensionality n. The current experiment provides evidence that in some cases there exists a threshold that, above which, further increases to dimensionality have no effect on the predictive validity of the model. When this threshold exists, it is very low compared with the

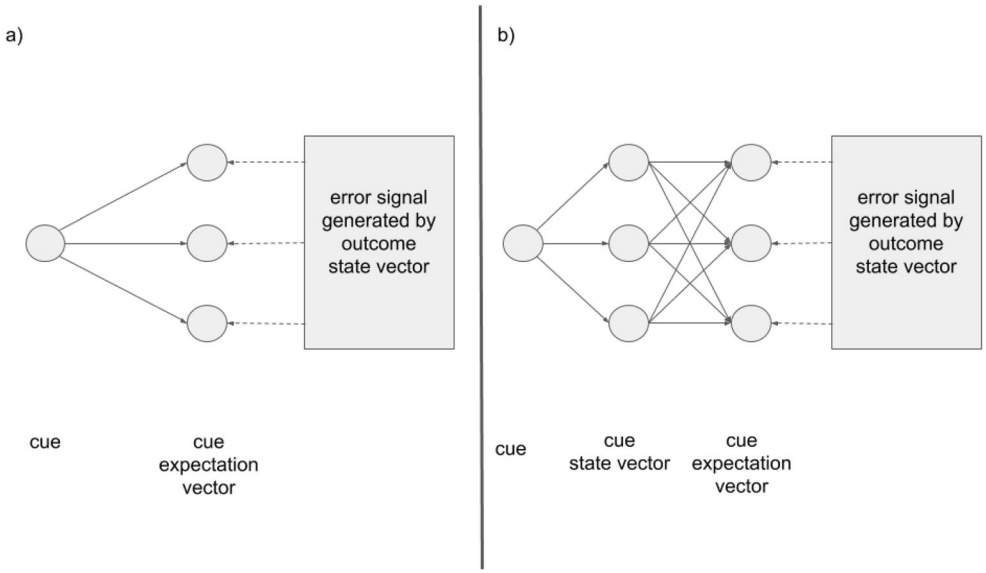

Fig. 7 Neural network representations of (**a**) the vector learning model and (**b**) a proposed refinement to the model that represents cues as real-valued, multidimensional entities rather than as symbolic abstractions

maximum possible vector dimensionality (1.03%–2.75% of maximum). Increasing vector dimensionality increases the degree to which the vector model accurately reproduces the association strengths learned by the R–W model, but this does not improve the model's ability to predict behavioral data in learning contexts where many outcomes are possible.

It may be that, in practice, behavior in complex learning environments is best modeled by (relatively) low-dimensional representations. Landauer and Dumais (1997) find that high-dimensional representations in their model of semantics actually hinders its performance. They suggest this is because making too many distinctions prevents a learner from finding useful generalizations. The adequacy of low-dimensional vectors in the VLM is supported by the results of the TOEFL analysis and the word similarity analysis. However, it is not supported by analysis of word priming effects. On the whole, we take this as weak evidence that the VLM avoids the frame problem.

Poor predictive validity of word similarities

The VLM was subject to a word similarity task. Across all parameter settings, the model performed notably poorly compared with computational semantic models. This is likely due to a difference between how computational semantic models determine context and how the VLM determines which observed contingencies to learn from.

Sequential word pairs acted as cue → outcome contingencies in learning. In the sequence *the boy ran*, *the* would be conditioned on *boy*, and *boy* would then be conditioned on *ran*. The model is learning to make predictions about future state. In contrast, computational semantic models derive their semantic representations from trying to predict future *and* prior state. In the example just given, *boy* would be used to predict both the prior word, *the*, and the following word, *ran*. Such models can be trained using only forward prediction of state, but forward and back prediction improves their predictive validity in most cases.

Reading moves in one direction along a sequence of words (more than less). However, as people are reading, they might also be imagining the scenes, objects, and actions referred to by the text. This allows entities that were referenced earlier in a document to still be present in the attentional awareness of the reader. In such a case, the referents of antecedent words would still be available to act as outcomes for learning. Backward conditioning is a known phenomenon (i.e., an unconditioned stimulus precedes a conditioned stimulus, but the conditioned stimulus' association strength to the unconditioned stimulus still changes; e.g., Tait & Saladin, 1986). It would be reasonable to consider the model's performance if prior and posterior words acted as outcomes. Although these findings were not presented in the results of Experiment 1, a version of the VLM was trained using words on either side of the cue as outcomes. This improved the model's performance from a mean r of 0.11[3] to an *r* of 0.26 (by Fisher *r*-to-*z* transformation, $z = -1.89$, $p = .03$).

Expectation vector weight

When an EV is conditioned on an SV, the model is learning to make an expectation about immediate future state. But that observed state itself generates expectations, and those

[3] The following model parameters were used: vector dimensionality 800, outcome EV weight 0.0, orthographically correlated SVs. Better model performance was observed in Experiment 1 with an outcome EV weight of 0.8, but having this parameter be above zero makes little conceptual sense when predicting prior state.

expectations generate other expectations, and so forth. Chaining expectations allows for the prediction of distant future state. Adding some proportion of an outcome's EV to its SV during conditioning allows us to model the process of using expectations to generate other expectations. It was anticipated that nonzero values of this parameter would facilitate performance in tests that the model was subjected to. This hypothesis was made on the basis that language contains temporal contingencies that span further than immediate word–word relationships; words have consequences that span sentences, paragraphs, and further.

In two of the three tests (TOEFL, word similarity), clear evidence was found that model performance was improved by allowing a portion of the outcome's EV to contribute to learning. In the third test (word priming), the parameter appeared to decrease model performance.

It could be that word similarity judgments and performance on the TOEFL reflects the type of distant-future predictions that were anticipated to be characteristic of language learning and that word priming effects instead reflect local predictions. Different tasks are best modeled by different parameter settings. This interpretation would suggest that animals can dynamically adjust how far into the future they wish to try to predict, based on task constraints. We do not think this to be an unreasonable speculation. However, it is inconsistent with how the VLM uses the outcome EV weight parameter. This parameter is fixed at the beginning of learning and affects the contingencies that the model learns, not how it recalls relationships. Likely, this reflects a limitation of the model.

We should expect that too much emphasis on prediction about distant-future state should be a hindrance to an animal. Planning for the future has utility, but only to the extent that it does not impede fulfillment of immediate needs. Thus, we expected to see an inverted U-shaped relationship between the parameter that controls the model's depth of future prediction and its performance on modeling behavioral data. Instead, the pattern of effects for this parameter varied substantially between tests. This might reflect a failing of the model, an incorrectness in the presented hypothesis, or a failure to test a large enough range for this parameter setting. More work is required to fully test how this parameter controls model behavior.

Word priming data

Word priming simulations produced results that were inconsistent with the other two tests. The benefit of increased vector dimensionality did not plateau. Using orthographically correlated SVs decreased model performance. Increasing the weighting given to the outcome's EV vector during conditioning hindered model performance. It is also notable that predictions made by the BEAGLE model were in the opposite direction of empirical data. This latter finding has no bearing on an evaluation of the VLM, but it is another point to the fact that the results observed for this test were unexpected.

The semantic models tested in this work are anticipated to perform well at modeling priming effects. Their most common use is to provide a measure of word similarity, and in the case of one model, CBOW, its measure of word similarity can be directly mapped onto Anderson and Milson's (1989) concept of needs probability (Hollis, 2017) Yet the performance of all these models on the Hutchison et al. (2008) data is quite poor. This is not by necessity; Johns et al. (2016) have demonstrated that manipulations to a model's training data can substantially improve its performance on this test. Ultimately, we have no good explanation to offer as to why this data set is such an historic challenge for distributional semantic models, or why the results we observed were so wildly inconsistent with hypotheses. A better understanding of the relationship between item-level priming effects and models of learning and semantics is clearly required.

Limits of the R–W update rule

There are good reasons to think that the R–W model is an incomplete model of learning (e.g., Miller, Barnet, & Grahame, 1995), and that there might even be fundamentally unaddressable problems with elemental models as a group (e.g., Pearce, 1994). Within the classical conditioning literature, numerous people have moved away from the R–W model and have begun pursuing other models (Courville, 2006; Pearce, 1994; Wagner, 1981), some of which focus more on temporal elements to conditioning rather than on discrete events (e.g., Gallistel & Gibbon, 2000; Sutton & Barto, 1981), which is an important aspect of learning that the R–W model neglects. As a reviewer put it, "the present use of the R–W model is like using a 50-year old locomotive to pull a modern train." The R–W model is most certainly flawed, limited, and outdated. That is acknowledged. However, its antiquity is overstated by the above quote.

A large portion of this work has demonstrated how animal learning models can be applied to model language processing and comprehension. It is only recently that language researchers have started explicitly considering the applications of animal learning models to simulate language learning. For good or ill, most of the early work was done using the R–W model (e.g., Baayen, 2010; Baayen et al., 2011). The fact that the R–W model *is* successful at capturing such a wide range of linguistic phenomena should be a point of intense interest to both learning theorists and linguists. It suggests the possibility that language learning is well-described by simple animal learning models. Because of its recent adoption in the language learning literature, the R–W model is an appropriate model for the presented work. Using the R–W model does not preclude future application of other models to the problem

of language learning. There is clearly much more work to be done.

It is worth pointing out the possibility that some of the apparent limits of models like the R–W model (i.e., elemental representations, using error correction) may be a failure of rigorous study rather than a necessary flaw. An historic failure of the R–W model is its inability to account for retrospective revaluation effects. Retrospective revaluation occurs when the pattern of responding to a stimulus is modified by later experience with a different stimulus (e.g., Wasserman & Berglan, 1998). Ghirlanda (2005) has demonstrated that the R–W update rule, with a simple and principled change to the way stimuli are represented, is capable of accounting for retrospective revaluation effects.

It has also been thought that the R–W model must make use of configural cues to solve linearly nonseparable learning problems. A configural cue is said to exist when the simultaneous presence of two cues, in a particular configuration, acts as a third cue for an animal. However, the R–W model presents no basis for deciding when configural cues are and are not involved in learning (Rescorla & Wagner, 1972). Without a basis for use or exclusion, the only reasonable option for the R–W model is to assume that configural cues *always* enter into learning. This path quickly leads to absurdity; there are combinatorially more *configurations* of observed elements than elements themselves, and combinatorial problems quickly become computationally intractable (Baayen et al., 2013). However, Baayen and Hendrix (2017) have demonstrated that dependence on configural cues is (again) largely a matter of stimulus representation, and not a fundamental feature of the R–W update rule. We echo Ghirlanda's (2005) conclusion that "even simple elemental models are not yet fully understood" (p. 110). Further exploration of the R–W model, and related models, may reveal other surprises about what such models can and cannot do.

Finally, the R–W model is not the only animal learning model that suffers from the frame problem. This issue is also present in the more recent configural model of Pearce (1994) and for the exact same reasons that it is present in the R–W model. Demonstrating how a problem can be addressed in one model, regardless of how antiquated it is, is useful knowledge for advancing other models that suffer from the same problem for the same reasons.

The outcomes of learning episodes

The current model has similarities to naïve discriminative learning (NDL; e.g., Baayen et al., 2011). Broadly, NDL is the R–W update rule applied to various aspects of language learning. A key insight of NDL is that the R–W update rule is very good at modeling human behavior in many-outcome learning problems like language. The fact that a model of learning designed to capture patterns of variation in nonlinguistic animal behavior can, with zero modification and often zero free parameters, be scaled up to human language and do an excellent job at capturing a variety of phenomena within that domain is of intense interest, at least to this researcher.

It is worth discussing the notion of a learning outcome in NDL, and how it relates to the VLM. Cues within NDL are sublexical units. Outcomes are a concept that has been termed *lexome* (Baayen et al., 2011). A lexome is an underlying semantic entity that the word evokes. We can discuss the word *ran* as evoking the lexomes "run" and "past tense." Lexomes are thought of as dynamic concepts that are themselves constantly being recalibrated due to learning; you probably learned about "fairness" as a child and also about "fairness" as an adult, but your concept of fairness has likely changed from then until now.

NDL conceptualizes cues as physical stimuli that generate expectations. The VLM is consistent on this point. However, NDL conceptualizes outcomes as semantic entities of which those expectations are about. This is different from the VLM, where outcomes are perceptual states. We think that both stances are reasonable within limits.

Experiment 1 demonstrated that it can be beneficial to condition a cue's EV on a combination of the outcome's SV and EV. We have discussed EVs as encoding expectations of future state, and that semantics can be operationalized as an expectation about context. Future state is one aspect of context. As the weight given to the outcome's EV is increased, the VLM becomes like NDL in its assertions about what outcomes are.

The two cases where outcomes are *only* perceptual states or *only* semantics are both unlikely; self-evidently, cues are followed by perceptual entities. Empirically, giving weight to expectations generated by those perceptual entities can improve learning about the cues that precede them (Experiment 1). The VLM provides a way to systematically explore how perceptual and semantic outcomes differentially affect learning. The VLM also provides an algorithmic means for implementing the core feature of a lexome—that outcomes are themselves constantly being recalibrated as a result of learning. This is a consequence of outcome EVs contributing to learning, and the fact that EVs are themselves changing as a consequence of learning about the referenced stimulus. This is a new contribution; NDL currently treats lexomes as black boxes.

Representing cues

Both the R–W model and the VLM treat cues as if they are symbolic. An "F" and an "R" are unrelated cues, despite the fact that they share feature overlap. Both models can force a feature-based representation by treating "F" as the presence of (e.g., a long vertical line, a short horizontal line, and a medium

horizontal line). All cues would contribute to model predictions and have their association strengths updated when "F" is presented as a cue. The value of such a representation is that it allows for generalization of learning to other cues that share similar features like, for example, the long vertical line shared by "R." However, this approach also leads to incorrect predictions about rates of discrimination between cues with high versus low feature overlap (Pearce, 1994).

Use of symbolic and featural representations of cues in the VLM both lead to a conceptual puzzle: Why are cues treated as symbolic when state vectors represent the perceptual properties of outcomes as real valued and high dimensional? The expectations generated by cues are subsymbolic, as are the outcomes that follow those expectations, but cues themselves are symbolic.

This issue can be addressed with further elaboration to the model. Rather than generating expectations directly from a symbolic representation of a stimulus (see Fig. 7a), that stimulus could be first transduced into a perceptual encoding (i.e., a state vector) and then that encoding could be used to generate an expectation about future perceptual state (see Fig. 7b). The proposed model could be implemented as a three-layer, fully connected neural network that uses the R–W update rule to back propagate error between the third layer and the second. Connections between the first layer and the second would represent communication channels between the external environment and an animal's perceptual system. The resulting model would use current perceptual state to predict future perceptual state, and future perceptual state to error correct predictions. The proposed model bears similarity to Clark's (2013) argument that the brain is a perceptual prediction machine in which learning is driven by error correction.

Open practices statement All software used in this research is available at http://www.github.com/hollisgf/reswag. All data used for simulations reported is publicly available.

References

Anderson, J. R., & Milson, R. (1989). Human memory: An adaptive perspective. *Psychological Review*, *96*(4), 703.

Baayen, R. H. (2010). Demythologizing the word frequency effect: A discriminative learning perspective. *The Mental Lexicon*, *5*(3), 436–461.

Baayen, R. H., & Hendrix, P. (2017). Two-layer networks, non-linear separation, and human learning. In M. Wieling, M. Kroon, G. van Noord, & G. Bouma (Eds.), From semantics to dialectometry: Festschrift in honor of John Nerbonne (pp. 13–22). London, UK: College Publications.

Baayen, R. H., Hendrix, P., & Ramscar, M. (2013). Sidestepping the combinatorial explosion: An explanation of n-gram frequency effects based on naive discriminative learning. *Language and Speech*, *56*(3), 329–347.

Baayen, R. H., Milin, P., Đurđević, D. F., Hendrix, P., & Marelli, M. (2011). An amorphous model for morphological processing in visual comprehension based on naive discriminative learning. *Psychological review*, *118*(3), 438.

Baayen, R. H., Shaoul, C., Willits, J., & Ramscar, M. (2016). Comprehension without segmentation: A proof of concept with naive discriminative learning. *Language, Cognition and Neuroscience*, *31*(1), 106–128.

Baroni, M., Dinu, G., & Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 238–247). doi:https://doi.org/10.3115/v1/P14-1023

Clark, A. (2013). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences*, *36*(3), 181–204.

Courville, A. C. (2006). A latent cause theory of classical conditioning (Unpublished doctoral dissertation). Carnegie Mellon University, Pittsburgh, PA.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological review*, *82*(6), 407.

Dennett, D. C. (2006). The frame problem of AI. *Philosophy of Psychology: Contemporary Readings*, *433*, 67–83.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2002). Placing search in context: The concept revisited. *Proceedings of the 10th international conference on World Wide Web* (pp. 406–414). New York, NY: ACM.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930–1955. In *Studies in linguistic analysis*. Oxford, UK: Basil Blackwell.

Gallistel, C. R., & Gibbon, J. (2000). Time, rate, and conditioning. *Psychological Review*, *107*(2), 289.

Ghirlanda, S. (2005). Retrospective revaluation as simple associative learning. Journal of Experimental Psychology: Animal Behavior Processes, 31(1), 107.

Hollis, G. (2017). Estimating the average need of semantic knowledge from distributional semantic models. *Memory & Cognition*, *45*(8), 1350–1370.

Hutchison, K. A., Balota, D. A., Cortese, M. J., & Watson, J. M. (2008). Predicting semantic priming at the item level. *The Quarterly Journal of Experimental Psychology*, *61*(7), 1036–1066.

Johns, B., & Jones, M. (2011). Construction in semantic memory: Generating perceptual representations with global lexical similarity. *Proceedings of the Annual Meeting of the Cognitive Science Society, 33*(33). Retrieved from https://escholarship.org/uc/item/3jk6d4pk

Johns, B. T., Jones, M. N., & Mewhort, D. J. K. (2016). Experience as a free parameter in the cognitive modeling of language. *Proceedings of the 37th Annual Conference of the Cognitive Science Society.* Retrieved from https://mindmodeling.org/cogsci2016/papers/0397/paper0397.pdf

Jones, M. N., & Mewhort, D. J. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological review*, *114*(1), 1.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, *104*(2), 211.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to latent semantic analysis. *Discourse Processes, 25,* 259–284.

Marr, D., & Poggio, T. (1976). From understanding computation to understanding neural circuitry (A. I. memo). Cambridge, MA: Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. Retrieved from https://arxiv.org/abs/1301.3781

Miller, R. R., Barnet, R. C., & Grahame, N. J. (1995). Assessment of the Rescorla–Wagner model. *Psychological Bulletin*, *117*(3), 363.

Moore, R. C. (1981). Reasoning about knowledge and action. In B. L. Webber & N. J. Nilsson (Eds.), *Readings in artificial intelligence* (pp. 473–477). https://doi.org/10.1016/B978-0-934613-03-3.50037-4

Nelson, D. L., McEvoy, C. L., & Schreiber, T. A. (2004). The University of South Florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, *36*(3), 402–407.

Pearce, J. M. (1994). Similarity and discrimination: a selective review and a connectionist model. *Psychological review*, *101*(4), 587.

Pylyshyn, Z. W. (1987). The robot's dilemma. Norwood, NJ: Ablex Publishing Corporation.

Ramscar, M., Hendrix, P., Shaoul, C., Milin, P., & Baayen, H. (2014). The myth of cognitive decline: Non-linear dynamics of lifelong learning. *Topics in Cognitive Science*, *6*(1), 5–42.

Ramscar, M., Sun, C. C., Hendrix, P., & Baayen, H. (2017). The mismeasurement of mind: Life-span changes in paired-associate-learning scores reflect the "cost" of learning, not cognitive decline. *Psychological Science*, *28*(8), 1171–1179.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical Conditioning II: Current Research and Theory*, *2*, 64–99.

Shaoul, C., Baayen, R. H., & Westbury, C. F. (2014). N-gram probability effects in a cloze task. *The Mental Lexicon*, *9*(3), 437–472.

Sutton, R. S., & Barto, A. G. (1981). Toward a modern theory of adaptive networks: Expectation and prediction. *Psychological Review*, *88*(2), 135.

Tait, R. W., & Saladin, M. E. (1986). Concurrent development of excitatory and inhibitory associations during backward conditioning. *Learning & Behavior*, *14*(2), 133–137.

Wagner, A. R. (1981). SOP: A model of automatic memory processing in animal behavior. *Information Processing in Animals: Memory Mechanisms*, *85*, 5–47.

Wasserman, E. A., & Berglan, L. R. (1998). Backward blocking and recovery from overshadowing in human causal judgment: The role of within-compound associations. *Quarterly Journal of Experimental Psychology, 51B*(2), 121–138.

Wheeler, M. (2008). Cognition in context: phenomenology, situated robotics and the frame problem. *International Journal of Philosophical Studies*, *16*(3), 323–349.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.