COGNITIVE SCIENCE
A Multidisciplinary Journal

Cognitive Science 48 (2024) e13494

ISSN: 1551-6709 online
DOI: 10.1111/cogs.13494

Latent Relations at Steady-state with Associative Nets

Kevin D. Shabahang,[a] Hyungwook Yim,[b] Simon J. Dennis[a]

[a]*School of Psychological Sciences, The University of Melbourne*
[b]*Department of Cognitive Sciences, Hanyang University*

Received 28 September 2023; received in revised form 24 June 2024; accepted 6 August 2024

Abstract

Models of word meaning that exploit patterns of word usage across large text corpora to capture semantic relations, like the topic model and word2vec, condense word-by-context co-occurrence statistics to induce representations that organize words along semantically relevant dimensions (e.g., synonymy, antonymy, hyponymy, etc.). However, their reliance on latent representations leaves them vulnerable to interference, makes them slow learners, and commits to a dual-systems account of episodic and semantic memory. We show how it is possible to construct the meaning of words online during retrieval to avoid these limitations. We implement a spreading activation account of word meaning in an associative net, a one-layer highly recurrent network of associations, called a Dynamic-Eigen-Net, that we developed to address the limitations of earlier variants of associative nets when scaling up to deal with unstructured input domains like natural language text. We show that spreading activation using a one-hot coded Dynamic-Eigen-Net outperforms the topic model and reaches similar levels of performance as word2vec when predicting human free associations and word similarity ratings. Latent Semantic Analysis vectors reached similar levels of performance when constructed by applying dimensionality reduction to the Shifted Positive Pointwise Mutual Information but showed poorer predictability for free associations when using an entropy-based normalization. An analysis of the rate at which the Dynamic-Eigen-Net reaches asymptotic performance shows that it learns faster than word2vec. We argue in favor of the Dynamic-Eigen-Net as a fast learner, with a single-store, that is not subject to catastrophic interference. We present it as an alternative to instance models when delegating the induction of latent relationships to process assumptions instead of assumptions about representation.

Keywords: Retrieval dynamics; Associative net; Semantic memory; Process model; Words

Correspondence should be sent to Kevin D. Shabahang, School of Psychological Sciences, Melbourne Connect, The University of Melbourne, 700 Swanston Street, Carlton, VIC 3053, Australia. E-mail: k.shabahang@gmail.com

1. Introduction

Landauer and Dumais (1997) echoed Plato's observation that most of our knowledge about the meaning of words depends on the induction of latent relationships between words that never directly co-occur. A statistical learning account of the formation of the meaning of words from linguistic experience must specify how to exploit surface-level co-occurrence patterns to infer relationships between words that have not appeared together. How can the system infer a relationship between two words like EAGLE and HAWK if they never co-occur, when both occur in the context of other words like FEATHER and FLY?

In the absence of direct co-occurrence, the relationship between words like EAGLE and HAWK is latent. Here, we limit our scope to relations that are defined based on the patterns of word usage in natural language text, but emphasize that a complete specification of the relationships between words and the determinants of their meaning also depends on extralinguistic and denotative structures that may not be captured through intralinguistic co-occurrence patterns. One such relation links words that are similar based on the extent to which they are interchangeable in their usage, such as words linked through hyponymy (e.g., animal-feline-cat). Another kind of relation holds when two words are linked based on the probability that they will be used in the same context (e.g., cat-fur). de Saussure (1916) highlighted the distinction and proposed that the meaning of a word in an utterance is a function of its relations with other words along a syntagmatic dimension (relatedness) that constraint its associations to words that previously occurred in the same context and a paradigmatic dimension (similarity), which links words that need not have occurred in the same context, but simply require that the words can be substituted across contexts.

Syntagmatic relations encode surface-level information that can be derived between word pairs in proportion to the number of times they co-occur in the same contexts. They encode the contextual bindings that capture episodic details and can be encoded through direct associations between word-pairs. For instance, given the sentence "the eagle had white feathers" in one context, and "the hawk had black feathers" in another context, encoding the episodic details corresponds to strengthening syntagmatic connections between EAGLE, WHITE, and FEATHERS in one associative bundle and HAWK, BLACK, and FEATHERS in a different bundle.

Unlike syntagmatic relations, the co-occurrence patterns characteristic of paradigmatic relations depend on structural information that is not directly available in the surface-level co-occurrence. Whereas syntagmatic relations link word pairs in proportion to the number of prior learning episodes (contexts) where they both occurred, paradigmatic relations link word pairs in proportion to the overlap between all prior contexts of one word and all prior contexts of the other. Evaluating a word's syntagmatic affinity for another word can be evaluated directly by reading the associative strengths between the two, but evaluating a word's paradigmatic affinity with another word expands the scope of overlap beyond their intersecting contexts and depends on the estimation of overlap between the entire set of contexts in characterizing one word's history of usage and the entire set of contexts characterizing the other word's contextual representation.

Dennis (2005) adapted de Saussure's (1916) ideas to an instance-based memory model called the Syntagmatic-Paradigmatic (SP) model and showed how multiple sets of benchmarks in language comprehension, reasoning, and classic memory paradigms can be modeled without the need to reconfigure the control flow, the architecture of the model, or the parameters to switch between tasks. The SP model was directly trained on text corpora and encoded syntagmatic and paradigmatic associations from observable co-occurrence statistics. Different control flows that distinguish one task from another emerged through the retrieval of information from previous contexts that aligned with the retrieval context. The present work is one of many offshoots from the explorations spearheaded in Dennis (2005).

We begin with a review of several computational frameworks for exploiting direct structure from co-occurrence patterns and higher-order structure that is latent, and argue that most models of lexical semantics explicitly encode latent representations into memory during learning. After describing some limitations of latent representation accounts, we turn to retrieval accounts that defer the exploitation of latent structure until retrieval instead of relying on an abstractive encoding mechanism. Finally, we present a linear associative net with dynamic weights called the Dynamic-Eigen-Net (DEN) as an example of a generalization-at-retrieval account that overcomes the limitless memory problem in alternative instance-based accounts. After a detailed overview of the DEN and some toy simulations that demonstrate its key properties, we compare its match to human free association and word similarity judgment norms and show that it performs on par with latent representation models like word2vec, Topics, and Latent Semantic Analysis (LSA). Next, we use a set of word relation norms, previously collected by having participants classify a large set of cue-response pairs into a small set of exhaustive relation types, to explore whether models differ in the types of relations they capture between word pairs. Finally, we end with a set of simulations that explore the performance of the models over different hyperparameter configurations.

1.1. Overview of latent representation accounts of semantic processing

Latent representation models assume a set of transformations that map the surface-level co-occurrence patterns of words into a latent representation through an encoding process to compress each word's co-occurrence history into a lower-dimensional gist-level representation, abstracted from the episodic details. Latent representation accounts vary in how they designate contextual boundaries and the nature of the abstraction operation. Here, we use context to refer to the linguistic context of a word or the collection of other words that surround it during processing (cf., Lohnas et al., 2015). Some accounts treat discrete documents (e.g., a collection of news articles) as separate contexts, and assume that the surface-level information is a set of word vectors, where each element of each word's vector is proportional to the frequency of the word in a corresponding document. A vector is an ordered list of numbers. Geometrically, it is a point in space and the value of each element determines its offset relative to the origin along the corresponding dimension.

In LSA (Landauer & Dumais, 1997), the vectors are arranged into a word-by-document co-occurrence matrix, derived from a large collection of documents, each roughly matching a single news article in the number of tokens. Latent relationships between words are

emphasized by projecting the normalized word-by-document matrix into a lower-dimensional space. The shared variance with which different words occur in different documents is factorized into a set of orthogonal vectors, where each vector accounts for a unique amount of the variance in co-occurrence. The combination of the factorized vectors reconstructs the original word-by-document matrix, but Landauer and Dumais showed how only using a subset of the factorized vectors to reconstruct a low-dimensional projection of the origins matrix improves semantic representations.

In the topic model (Griffiths, Steyvers, & Tenenbaum, 2007), the latent representation corresponds to a set of discrete topics, whose combination is assumed to generate each of the observed word-by-document counts. The latent relationship between two words is mediated through unobserved topics, to which both words correspond. More specifically, the topic model is characterized by two sets of conditional probability distributions. One set of distributions estimates the conditional probability of topics, given each document. Another set of distributions estimates the conditional probability of each word, given each topic. A document is assumed to be generated by iteratively sampling a topic from a topic-given-document distribution, followed by sampling a word from the word-given-topic distribution. The two distributions can be estimated using Gibbs sampling.

Word2vec (Mikolov, Chen, Corrado, & Dean, 2013) defines context based on a fixed-size window of text from a corpus. At each slice of text, the word in the middle of the window is designated as the target and the surrounding words are used as its context. Therefore, each word in the corpus can take the role of target or context, depending on whether it is at the center of the window or the periphery. In the Continuous Bag of Words (CBoW) variant, latent representations are formed through gradual changes to the connectivity of a multi-layer neural net that is trained to learn the conditional probability of each target word given its context words.[1] The input and output layers are one-hot coded: words are represented with a vector that is the same size as the number of unique words in vocabulary and designates a separate element for each unique token. Cueing the system with a set of context words is accomplished through setting each word's corresponding one-hot element to one. The one-hot coded vector activates the input layer and is projected through a lower dimensional hidden layer and out through the output layer. Compressing the input through the lower-dimensional hidden layer increases the proximity of words based on the overlap between their contexts as in LSA. Whereas the compression operation is linear in LSA, word2vec's use of a sigmoidal activation function in the hidden layer introduces a nonlinearity in the compression operation (see Levy, Goldberg, & Dagan, 2015, for a more detailed discussion of the differences between dimensionality reduction between the two approaches).

The projection of a set of word indices to a lower-dimensional embedding through a feedforward network forms one of the fundamental building blocks of the Transformer architecture (Vaswani et al., 2017), a modular framework for constructing multi-layered feedforward neural networks that equip each layer with an alignment mechanism (Bahdanau et al., 2014) that enables it to condition the latent representation of each word token in an input sequence of word tokens on the latent representation of each other token. Variants of Transformers—most famously OpenAI's ChatGPT (e.g., Achiam et al., 2023)—have recently achieved state-of-the-art performance on various language comprehension and reasoning

benchmarks, particularly when massively scaled (e.g., in the number of layers) and trained on gigantic corpora containing samples of language material, both natural and formal (e.g., computer program source code, tabular data, and unfiltered markup).

Although Transformers are not cognitive models, the success of Large Language Models demonstrates how the interaction of a set of latent embeddings through a series of feedforward projections provides a solution to the language modeling problem: given a sequence of sampled word tokens, estimate a probability for each word token in vocabulary to infer its likelihood to be sampled as the next word token in the sequence. The simulations presented in the current work fall in the domain of lexical semantics, whereas the language modeling problem builds on top of lexical representations to expand into the domain of semantic composition—the construction of the meaning of a whole (e.g., a sentence) from interactions between the meaning of its parts (e.g., the words in the sentence). Since the Transformer architecture directly inherits fundamental assumptions from earlier work in lexical semantics, we will briefly describe some of its essential mechanisms.

Transformers are trained using backpropagation to iterate through a corpus with a sliding context window that rolls over slices of tokens. Each learning iteration consists of the estimation of probabilities over each token in vocabulary to form an expectation of the next likely token, followed by a readjustment of expectations through the backpropagating a signal based on the difference between expectation and outcome (i.e., the next word token in the sequence).[2] Over a large number of training iterations, the weights in the network gradually assimilate patterns of word co-occurrence that facilitate greater predictability. In a second phase of training, the network is further trained on a collection of tasks such as text summarization or question answering that are cast as a sequence-to-sequence mapping problems. A large collection of input-output sequence pairs are used to maximize the likelihood that the model will generate a particular output sequence when prompted with a previously paired input sequence. On the first iteration, only the words in the input sequence are used to probe the network. After backpropagation, the first word in the output sequence is added to the sequence used to prompt the network for the next word, and so forth. One main difference between pretraining and finetuning is the use of a smaller learning rate for the latter as a way to prevent catastrophic forgetting of the pretraining information.[3]

In a Transformer, inference begins through the projection of each token in the sampled sequence to a lower-dimensional embedding vector as in word2vec, but is further augmented with positional information by superimposing a vector of the same dimensionality whose elements are a function of a position-varying oscillator (cf., OSCAR; Brown, Preece, & Hulme, 2000). A placeholder vector for the next expected token ("padding") is also added at input using a position vector.[4] The resulting vectors—one for each item in the input and the placeholder—are arranged into a matrix as row vectors. The input matrix is then used as input that projects through a series of intermediate "attention" layers before the final latent output layer is mapped to the original vocabulary space. The probability distribution estimating each vocabulary token's likelihood of being sampled next is obtained based on the placeholder vector, which absorbs information from the other input vectors through each attention layer projection.

Each attention layer consists of a multi-head attention sublayer, followed by a feedforward sublayer. A multi-head attention sublayer contains around a dozen attention heads. An attention head solves the table-lookup problem, which requires matching a query (e.g., name of a person) to a key (an entry with the person's name) in a large list of keys (phonebook) to retrieve some corresponding information (phone number). Instead of assuming an exact match, the queries, keys, and values in an attention head are all specified as low-dimensional vectors and a fuzzy similarity-based match process is used to implement a continuous approximation to the discrete table-lookup scenario.

As in the case of word2vec, a word in a sequence can either be the to-be-predicted target or part of the context used to predict another word. In an attention head, the two modes are distinguished by two different linear projections. The query-projection vector of a target or to-be-predicted word is matched against the key-projection vectors of all preceding tokens in the sequence and the level of match (e.g., dot-product) between the query and each key is used to combine a weighted sum of the context words' value-projection vectors. Each head has a separate set of query, key, and value matrices and projects the embeddings to a dimensionality generally set lower by a factor of the number of attention heads (e.g., 64 dimensions for a head with 12 heads and an embedding size of 768). Given the additional compression, each attention-head projection will be receptive to a subset of the total variance in the original embedding. Information across the attention heads is then pooled through additional feedforward layers to yield the output of a single attention layer. The output is also a matrix with the same dimensionality as the input. The network is also equipped with direct connections between the input and output of each multi-head attention module to prevent the latent representations of tokens from unnecessary transformations as activation is fed upstream through a stack of attention layers. The input to the attention layer is added to the output after each transformation, followed by a normalization of the activations. The latent embeddings of the input converge to stable patterns as activations traverse up each attention layer.

1.2. Limitations of latent representation models of semantic processing

Neural embedding models that are trained with backpropagation like word2vec depend on a slow learning mechanism that requires several passes through the same input to form stable semantic representations. People are generally able to approximate the meaning of a new word, like WUG, based on exposure to just a few sentences like "the wug had gray feathers" (Borovsky, Elman, & Kutas, 2012; Lazaridou et al., 2017). Lazaridou et al. show that based on just two contexts, participants could infer that a nonword like WUG is more similar to EAGLE and AIRPLANE relative to CRANE and CUCUMBER. Extending neural embeddings with larger modular architectures like the Transformer is one possible resolution to the one-shot learning problem. The formation of the latent representation corresponding to an unfamiliar word like WUG would be conditioned on the latent representation of the other familiar tokens through each level of the network and constrain its meaning.

Another problem with neural embedding approaches like word2vec, is their vulnerability to interference. McCloskey and Cohen (1989) showed how learning one set of input-output mappings in a three-layer neural network, trained with backpropagation, completely wipes

out information encoded from previously learned input-output mappings. More recently, Mannering and Jones (2020) used homonyms to show that word2vec suffers from similar problems due to interference. Homonyms have the same phonology and orthography, but signify different meanings depending on the context of their usage. Mannering and Jones found that word2vec favored one sense of homonymous words over another sense, depending on whether contexts portraying the one sense were trained before or after the contexts portraying the other sense. That is, if the network was trained with all the contexts using BANK to refer to a financial institution, followed by contexts that use BANK to refer to land surrounding a body of water, then the resulting word vector for BANK would be most similar to other words like RIVER or WATER instead of MONEY or ACCOUNTING. The word-sense is just one-way interference between new learning trials and stored information from previous trials impacts performance. The substantially greater number of parameters in a Transformer raises its tolerance to interference but catastrophic forgetting remains an issue, particularly during fine-tuning (Kotha et al., 2023).

The requirement for context-sensitivity is evident in the case of homonyms. A word like BANK has its dominant meaning as a financial institution and its secondary meaning as the land surrounding a body of water. The dependence of meaning on context presents itself more broadly when instantiating the sense of polysemous words. The word RECORD, for example, has a very different sense depending on whether it occurs in the context of a "world record" or "a record of an event." The instantiation of a word is not just defined by the contexts making up its past history of usage, but rather the subset of its history of usage that contextually overlaps with the current context of its instantiation. It makes sense for the system to evaluate contextual overlap during retrieval rather than encoding, since the current context can filter out previous usages that may not be relevant to the current usage. Transformer architectures resolve the problem by using the attention layers to contextualize a word's representation in an input sequence based on the other words in the sequence.

A complete model of human memory must not only abstract regularities from learning trials, but also match human performance in recollecting episodic details. All three latent representation models require a complementary representation to capture episodic details, because the contextual boundaries that distinguish one episode from another dissolve through the assumed generalization mechanisms. Therefore, using a latent representation model subscribes to a dual-systems theory of episodic and semantic memory. Alternatively, a system may only store direct co-occurrence information as an aggregate of associative bundles. At retrieval, a cue is integrated with all stored associative bundles to generate the superposition of both gist-level and episodic details. We consider retrieval-accounts of generalization more parsimonious with respect to a general theory of human memory compared to latent-representation accounts, because the former assumes a single memory system capable of processing both episodic and semantic information.

1.3. Generalization at retrieval with instance-based memory

Instance theories, like MINERVA2 (Hintzman, 1986), assume that each associative bundle is stored as a separate memory trace. Each bundle is aggregated using concatenation to a

limitless store. More specifically, the bundle is stored by appending a noisy copy of its corresponding feature vector, $\mathbf{x}$, to a *V*-by-*t* dimensional matrix, $\mathbf{E}$, of fixed dimensionality along one axis (*V*) and growing dimensionality along the other axis (*t*). During retrieval, the cue, $\mathbf{x}_0$ (a *V*-dimensional vector), is assumed to match against each of the *t* stored traces, in parallel. All the traces are summed component-wise, each weighted by a nonlinear function of its similarity with the cue vector, to yield the response pattern. The retrieved pattern takes the form, $\mathbf{x}_1 = [(1/\mathrm{N_{nnz}})\mathbf{x}_0\mathbf{E}]^{2m+1}\mathbf{E}^{\mathrm{T}}$, where *m* is any positive integer (usually between 1 and 10) and $\mathrm{N_{nnz}}$ is the number of nonzero elements in the cue. The scaled dot-product between the cue and each trace is raised to an odd power to maintain the original sign.

The construction of a generic pattern as a weighted sum of all stored associative bundles—during retrieval—affords MINERVA2 context-sensitivity; however, the assumption that memory is an explicit timeline of the system's past implies a limitless episodic store. In the present work, we relax the multi-trace assumption by collapsing across the explicit timeline during encoding (but see Kwantes, 2005, for similar work using MINERVA2). That is, instead of isolating the associative bundles, they are superimposed into a network. Formally, if the input to the system at each time point was a vector of all zeros, except for elements corresponding to words experienced at that particular instance (i.e., local-codes), then one way to collapse the timeline is to condense it into a word-by-word co-occurrence matrix, $\mathbf{C} = \mathbf{E}\mathbf{E}^{\mathrm{T}}$.

1.4. Generalization at retrieval with associative memory

The natural choice for a process model with a memory representation derived from word-by-word co-occurrence is an associative net. A normalization transformation is directly applied to the co-occurrence counts to approximate the cumulative result of thousands of learning trials. Our use of the co-occurrence counts is only a means to estimate the final weights derived through normalization. The assumption is that a learning rule exists, such that its repeated application through a large number of learning trials yields the normalized co-occurrence statistics. Reliance on Hebbian learning, instead of backpropagation, affords associative nets with one-shot learning. Using local codes also insulates the system from interference. An associative net is also context-sensitive because its mode of generalization varies based on the patterns in the cue like MINERVA2, but an associative net has the advantage of assuming a bounded memory.

One way to derive the associative strengths by normalizing the co-occurrence counts is to use the Shifted Positive Pointwise Mutual Information (SPPMI; Bullinaria & Levy, 2007; Levy & Goldberg, 2014). The Pointwise Mutual Information (PMI) obtains associative strength by down-weighting associations between pairs of words based on each word's base-rate probability. If two words co-occurred some number of times, their associative strength would be higher if they were low-frequency words than if both or one item had a high frequency. The direct PMI matrix contains negative associative strengths when the rate of co-occurrence between word pairs falls below the product of their base-rates. The negative associations can be problematic because the lack of co-occurrence between word-pairs is highly dependent on the idiosyncrasies of the corpus. Evidence for the absence of an association is much less informed by the absence of co-occurrence than evidence for its presence

based on nonzero co-occurrence. Setting all negative elements to zero to obtain a positive PMI matrix (PPMI) has been shown to better match human semantic similarity judgments compared to the original PMI (Bullinaria & Levy, 2007). Levy and Goldberg (2014) "shifted" the PMI matrix by subtracting a constant from every cell prior to zeroing out all the resulting negative weights (SPPMI) and showed that transforming a word-by-context[5] co-occurrence matrix using SPPMI approximates the same objective function used to train word2vec. As we later show, it is possible to obtain a similar level of performance as word2vec by using spreading activation through the SPPMI matrix instead of dimensionality reduction when matching human word similarity judgments and patterns of free association.

An associative net equipped with a spreading activation function provides a solution to the fast-learning problem. Assuming that associations strengthen between word pairs each time they are experienced together (e.g., WUG-FEATHERS), then it is possible to infer a novel word's meaning by evaluating its configuration within the higher-order structure in an associative network. Spreading activation provides a natural mechanism for evaluating the higher-order structure of the novel word in an associative network. The novel word first activates its neighbors with which it just became associated, but then activation would spread to its second-order associates—its neighbors' neighbors—to allow the novel word to inherit a similar history of usage (e.g., WUG activates FEATHERS, and FEATHERS activates EAGLE).

Spreading activation through an associative net provides an alternative to dimensionality reduction for exploiting latent structure, which should reduce cross-talk between encoded information (French, 1991). Cross-talk is minimized when the representations are mutually orthogonal, but at the expense of generalizability. For example, each hidden unit in word2vec has a distinct profile of excitation, given the activation of each of the input units. Some hidden units may be completely indifferent to the activation of one subset of input units, but highly sensitive to the activation of other input units. Each hidden unit has a different receptive field. The span of the receptive field of each hidden unit depends on the ratio of the number of hidden units to the number of input/output units. The receptive fields broaden as the number of hidden units drops relative to the number of input/output units, resulting in greater overlap in representations. On the other hand, as the number of hidden units approaches the number of input/output units, the receptive fields of each hidden unit tighten around the mode, resulting in less overlap in representations. The situation is analogous with LSA and the topic model, but with the ratio of the number of dimensions in the compressed space relative to the dimensionality of the original word-by-document space in LSA and the ratio of the number of topics relative to the number of words in the topic model.

The system must balance a tradeoff between generalizability and orthogonality. We present one solution to the generalizability-orthogonality dilemma by deferring the generalization process to retrieval. We define generalization as the process through which a higher-order structure that is implicit in patterns of surface-form co-occurrence is made explicit in the derived activations. A generic pattern is constructed during retrieval by averaging over relevant episodic details, given a cue. The dependence of the generic pattern on the cue's match to the stored episodic details balances the tradeoff and also equips the system with context-sensitivity.

In this paper, we show how latent relations can form in a one-hot coded associative net, using a spreading activation algorithm we developed previously (Shabahang, Yim, & Dennis, 2022). An associative net is a network of direct associations, stored in a weight matrix, coupled with a recurrence relation that specifies how activation spreads from an initially active set of nodes to the rest of the network. Spreading activation forces the structure in the network to interact with the initial activations to yield a representation that integrates those initial activations with the global associative structure. Geometrically, the state at each snapshot corresponds to a point in semantic space. Cueing is analogous to placing an initial point into the state-space and letting it stabilize to the nearest equilibrium point. In prior work, we explored generalization over serial-order associations by population coding words in a sequence through the concatenation of one-hot vectors that function as separate slots. The weights were learned through similar normalizations presented here, but for a larger matrix composed of submatrices reflecting different relative serial positions. After learning, we tested the model's generalizability by forming syntactically congruent (e.g., "the dog") and incongruent ("dog the") bigram pairs and using familiarity values derived with the DEN to discriminate the two classes of bigrams. The DEN attributed greater familiarity to the congruent bigrams over incongruent bigrams, even when the weight corresponding to each of the congruent bigrams was zeroed out prior to retrieval. The network exploited the global associative structure to generalize. Serially ordered associations span a combinatorial domain, and are especially relevant to syntactic processing. Here, we extend the model to order-independent associations, which have traditionally been considered as the basis for semantic processing.

Fig. 1 illustrates one-way associative bundles can be represented within a network. The top panel shows the input to the system during encoding across three different time points, with the three sentences assumed to have occurred in different contexts (color coded). We have excluded high-frequency words. The middle panel shows how the associative bundles corresponding to the sentences may form in the network, and the bottom panel shows how such a network would be implemented in a word-by-word weight matrix. The weights between words are color coded to emphasize the three different contexts. Collapsing the timeline into an associative net maintains the integrity of the episodic details since cueing the system with a symbol (e.g., HAWK) activates symbols that are in associative bundles with the symbol (e.g., FEATHERS and BLACK). Fig. 1 also illustrates how a single occurrence of a novel word like WUG with FEATHERS and a spreading activation mechanism enables rapid inference of its meaning through the evaluation of its configuration with respect to the global structure in the network.

2. Dynamic-Eigen Net

In Hebbian associative nets, synapses between pairs of "neurons" strengthen when the neurons activate within a short time interval (Hebb, 1949). If a single neuron encodes a single word in a sentence, then the strengthened synapses encode co-occurrence rates between words that keep the same company. When one of those words becomes activated at a later time, then it also activates other words it co-occurred with in the past. More broadly, Hebbian encoding

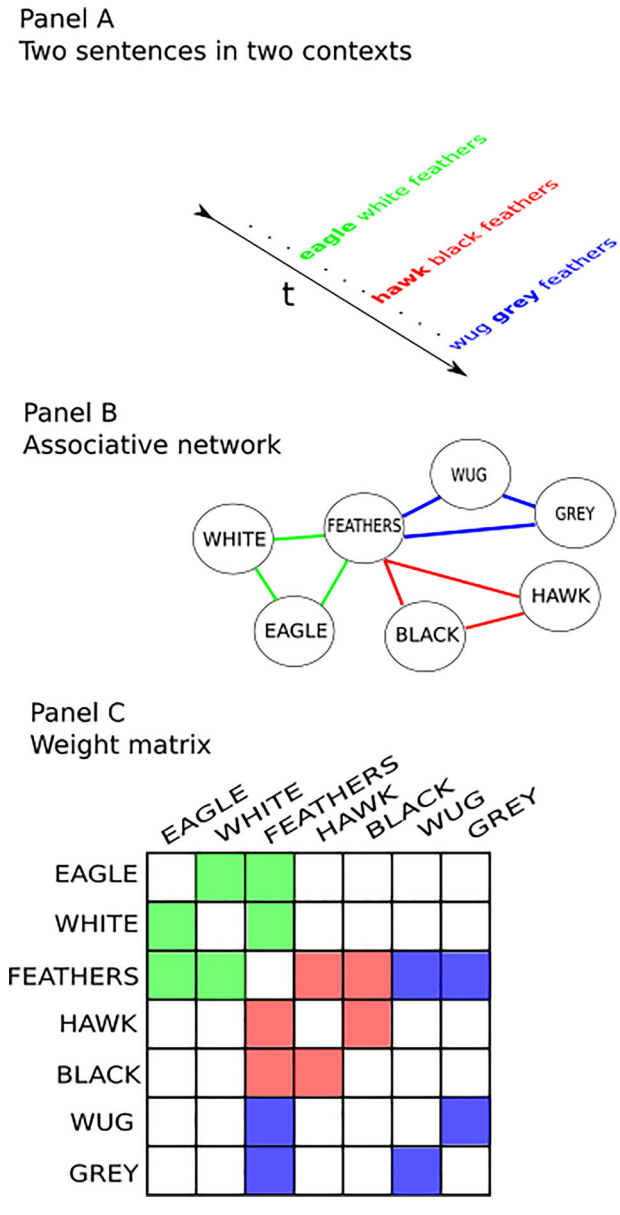

Fig. 1. Storing associative bundles into an associative network.

Note. Panel A shows a hypothetical timeline with three different snapshots. Panel B shows a network that could be constructed to encode the associations based on the three different snapshots. Panel C shows the weight matrix corresponding to the network in Panel B. The colors emphasize different contexts for illustrative purposes only.

of a pattern of activation over a collection of neurons at a point in the past increases the likelihood that the entire pattern is reactivated when the system is cued with a partially overlapping pattern at a point in the future. Some words will occur much more frequently than other words, and, therefore, end up acquiring a large number of associates. A general problem is to reduce the impact of noise resulting from the activations of high-frequency words during retrieval. Here, we prevent the force of the high-frequency patterns by temporarily conditioning the system on the cue, using a DEN. A DEN is a linear associative net with transient cue-driven weight changes. The transient weight changes occur each time a new cue is presented and last until the spread of activation through the network convergences. After convergence, the weights are reset. The transient weights temporarily bias the network's settling point toward the cue and prevent runaway toward the dominant steady-state of the static weight matrix.

Processing in an associative net is characterized by an initial state and an update function that propels the system forward in time. The state-transition law specifies how memory weights, $\mathbf{W}$, interact with the momentary state, $\mathbf{x}_t{}^{\mathrm{T}}$, to drive the system into the future, $\mathbf{x}_{t+1}{}^{\mathrm{T}}$. For retrieval, an input cue, $\mathbf{x}_0{}^{\mathrm{T}}$, is used to initialize the state for the first time point, and the state at the next time point is obtained as a function of the vector-matrix multiplication of the current state and the weight matrix, $\mathbf{x}_{t+1}{}^{\mathrm{T}} = f(\mathbf{x}_t{}^{\mathrm{T}}\mathbf{W})$. The letter f is a function that bounds the activations and can be simple vector normalization or a nonlinear saturation function like the sigmoid. The process is carried out iteratively until further iterations have no additional effect on the state vector (i.e., when $\mathbf{x}_{t+1}{}^{\mathrm{T}} \approx \mathbf{x}_t{}^{\mathrm{T}}$). Retrieval forces the interconnections between words encoded from previously learned patterns to interact with the cue until the system reaches an equilibrium state. The equilibrium state, $\mathbf{x}_\infty{}^{\mathrm{T}}$, is treated as the retrieved pattern, and distributes activations over all input tokens despite the original cue only having one or a handful of active nodes. We use the subscript, ∞, to represent the converged pattern because it approximates taking the limit as the number of iterations, t, approaches infinite, that is, $\mathbf{x}_\infty{}^{\mathrm{T}} = \mathbf{x}_{t+1}{}^{\mathrm{T}} \approx \mathbf{x}_t{}^{\mathrm{T}}$.

One problem with the linear associative net is that the state vector will always converge to the dominant eigenvector (Anderson, 1995). Dropping normalization, the state at the second iteration is given by,

$$\mathbf{x}_2^T = \sum_{i=1}^{V} \lambda_i \mathbf{x}_0^T \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T \mathbf{W} = \left(\sum_{i=1}^{V} \lambda_i \mathbf{x}_0^T \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T \right) \left(\sum_{i=1}^{V} \lambda_i \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T \right).$$

Since $\dot{\mathbf{e}}_\mathrm{i}$ are mutually orthogonal, $\dot{\mathbf{e}}_\mathrm{i} \cdot \dot{\mathbf{e}}_\mathrm{j} = 0$, for all $i \neq j$, and of unit-length, $\dot{\mathbf{e}}_\mathrm{i} \cdot \dot{\mathbf{e}}_\mathrm{i} = 1$, distributing the summands through gives,

$$\mathbf{x}_2^T = \sum_{i=1}^{V} \lambda_i^2 \mathbf{x}_0^T \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T.$$

In the general case, the state after t iterations, $\mathbf{x}_t{}^{\mathrm{T}}$, is given by,

$$\mathbf{x}_t^T = \sum_{i=1}^{V} \lambda_i^t \mathbf{x}_0^T \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T.$$

After t iterations, the state-vector is the sum of each of the eigenvectors, weighted by the dot-product of the initial cue and that eigenvector, scaled by the corresponding eigenvalue, raised to the t'th power. Let $\mathbf{x}_\infty{}^{\mathrm{T}}$ be the equilibrium point and of unit-length. Because the eigenvalues monotonically decrease from the dominant eigenvalue, $\lambda_{\max}$, to the least dominant eigenvalue, the equilibrium in the linear associative net is identical to the dominant eigenvector,

$$\mathbf{x}_\infty^T = \lim_{t\to\infty} \sum_{i=1}^{V} \lambda_i^t \mathbf{x}_0^T \dot{\mathbf{e}}_t \dot{\mathbf{e}}_i^T = \lambda_{max} \dot{\mathbf{e}}_{max}^T.$$

Based on the eigenvalue equation, $\mathbf{u}^{\mathrm{T}}\mathbf{A} = \lambda\mathbf{u}^{\mathbf{T}}$,

$$\mathbf{x}_\infty^T \mathbf{W} = \sum_{i=1}^{V} \lambda_i \mathbf{x}_\infty^T \dot{\mathbf{e}}_i \dot{\mathbf{e}}_i^T = \lambda_{max} \dot{\mathbf{e}}_{max}^T.$$

Earlier generations of associative nets (Anderson, Silverstein, Ritz, & Jones, 1977; Hopfield, 1982) proposed nonlinear alternatives to normalization to prevent runaway toward the dominant eigenvector; however, the resulting systems were limited in their ability to generalize. For example, in the Brain-State-in-a-Box (Anderson et al., 1977), the encoded patterns were mutually orthogonal and the main characteristic of the associative net was to complete partially overlapping input patterns to a previously encoded pattern. The set of valid[6] steady-states in the system were restricted to previously encoded patterns; however, generalization requires interpolating based on multiple correlated patterns. Amari (1977) explored threshold activations in associative nets to facilitate encoding correlated patterns, but found that cross-talk became catastrophic when noise was introduced during encoding.

In a DEN, the equilibrium is shifted toward the cue by adding the cue vector's outer-product, with itself, to the weight matrix before recurrence takes place. Each cue shifts the equilibrium toward itself during recurrence. Temporarily adding the outer-product of a cue adds another term, which persistently pulls the state toward the initial representation to prevent flight toward the dominant eigenvector. The update function for a DEN is given by,

$$\mathbf{x}_{t+1}^T = \frac{\mathbf{x}_t^T \left(\mathbf{W} + \mathbf{x}_0 \mathbf{x}_0^T\right)}{\left\| \mathbf{x}_t^T \left(\mathbf{W} + \mathbf{x}_0 \mathbf{x}_0^T\right) \right\|}.$$

If we let λ_∞ be the primary eigenvalue of the modified weight matrix, $\mathbf{W} + \mathbf{x}_0\mathbf{x}_0{}^{\mathrm{T}}$, the following equation holds:

$$\mathbf{x}_\infty^T \left(\mathbf{W} + \mathbf{x}_0 \mathbf{x}_0^T\right) = \sum_{i}^{V} \lambda_i \left(\mathbf{x}_\infty^T \dot{\mathbf{e}}_i\right) \dot{\mathbf{e}}_i^T + \left(\mathbf{x}_\infty^T \mathbf{x}_0\right) \mathbf{x}_0^T = \lambda_\infty \mathbf{x}_\infty^T.$$

The right-hand side, $\lambda_\infty\mathbf{x}_\infty{}^{\mathrm{T}}$, follows from the eigenvalue equation and because $\mathbf{x}_\infty{}^{\mathrm{T}}$ is the primary eigenvector of $\mathbf{W} + \mathbf{x}_0\mathbf{x}_0{}^{\mathrm{T}}$. The equation shows how transient weights shift the system's equilibrium toward the initial cue. In general, the activation pattern converges toward the direction of each of the eigenvectors, weighted by its dot-product with the current state, plus the initial pattern, weighted by its dot-product to the current state.

Table 1
Toy corpus

dog played bone
cat played leaf
truck drove factory
car drove garage
flower grew field
tree grew hill
liberal increased taxes
conservative decreased taxes
eagle white feathers
hawk black feathers
bird grassy river bank
investors stood bank
traders deposited money bank
salesman withdrew money bank
bank raised interest rate
wug grey feathers

Given two sentences such as "the dog played with the bone" and "the cat played with the leaf," cueing the system with CAT would activate PLAYED and LEAF in the first iteration. In the next iteration, the word PLAYED would activate DOG and BONE, and so forth. A simple three-dimensional system is explored analytically in the Supplementary Appendix.

Table 1 shows a small corpus of 15 different sentences that will be used to provide a toy demonstration of some of the DEN's properties. First, each unique word in the corpus was assigned a unique index. In a Hebbian system, the cumulative result of a set of learned associations can be captured in a word-by-word co-occurrence matrix. Therefore, the number of times each word co-occurred with each other word within the same context was collected into a word-by-word co-occurrence matrix, $\mathbf{C}$, where each cell, $\mathbf{C}_{ij}$, corresponded to the number of times the i'th word co-occurred with the j'th word in the same context. For simplicity, we define context as the sentence for the toy demonstration. In the next section, context is defined as a fixed-sized sliding window.

The co-occurrence matrix is used to estimate the joint-probability of pairs of words occurring in the same context. We used a smoothing parameter, α, to estimate the joint and base-rate probabilities. The base-rate probability of the i'th word was estimated with α as an additive smoothing parameter using,

$$p_i = \frac{\mathbf{C}_j + \alpha V}{\sum_{k=1}^{V} [\mathbf{C}_k + \alpha V]}.$$

The base-rate probability of the j'th word was estimated by using α as both an additive and multiplicative smoothing parameter using,

$$p_i = \frac{\left(\mathbf{C}_j + \alpha V\right)^{\alpha}}{\sum_{l=1}^{V} \left[(\mathbf{C}_l + \alpha V)^{\alpha}\right]}.$$

Incorporating a multiplicative smoothing parameter on the base-rate probabilities over the columns has been referred to as *context distribution smoothing* in the broader distributional semantics literature (Levy et al., 2015). Context distribution smoothing has an analogous effect as smoothing the distribution of negative samples in word2vec (Mikolov et al., 2013) and reduces bias toward rare words.

The joint-probability of word pairs is estimated with α as an additive smoothing parameter using,

$$p_{ij} = \frac{\mathbf{C}_{ij} + \alpha}{T + \alpha V^2},$$

where T is the total count, $T = \sum_{i=1}^{V} \sum_{j=1}^{V} \mathbf{C}_{ij}$.

We used a smoothing parameter $\alpha = 1$ for the toy demonstration.

Words like THE and ON occur much more often than words like DOG and CAT. The associative strength between two words should not only take into account their joint-probability, but also each of the words' base-rates. The SPMI was applied to the probabilities to normalize for the different base-rate probabilities of different words. For the simulations in the next section, we also remove all negative values to increase sparsity to reduce the memory load for computational reasons.

The weight connecting the *i*'th word to the *j*'th word is given by,

$$\mathbf{W}_{ij} = \log_2\left(\frac{p_{ij}}{p_i p_j}\right) - \log_2(k),$$

where p_{ij} is the probability that the *i*'th word occurs with the *j*'th word and p_{i} and p_{j} are the base-rate probabilities of the *i*'th word and the *j*'th word, respectively. Taking the ratio of the joint probability of a pair of words and the product of their base-rates ensures that the associative strength between each pair of words is proportional to the magnitude with which their joint probability exceeds their expected probability, under the assumption that they occur independently.

The parameter, *k*, corresponds to the amount of "negative evidence" used to discount each association by a constant to reduce spurious associative strengths (see Levy & Goldberg, 2014). The shift parameter, *k*, was set to 1 (i.e., no negative evidence since $log(1) = 0$) for the toy example.

The contribution of each eigenvector (i.e., corresponding eigenvalues) of the weight matrix rapidly drops with its rank. During retrieval, each recurrent iteration that follows the first retrieval cycle introduces a higher-order polynomial term due to the nested structure of recurrence, leading to growth in the relative contribution of each eigenvector with every recurrence iteration. The DEN applies partial inhibition of the dominant eigenvector (cf., Mu et al., 2017), by subtracting some proportion,[7] η, of the dominant eigenvector's outer-product, with itself, $\dot{\mathbf{e}}_{\max}\dot{\mathbf{e}}_{\max}{}^{\mathrm{T}}$, weighted by its corresponding eigenvalue, $\lambda_{\max}$, from the weight matrix,

$$\widehat{\mathbf{W}} = \mathbf{W} - \eta\lambda_{max}\dot{\mathbf{e}}_{max}\dot{\mathbf{e}}_{max}^{T}.$$

The parameter, η, was set to 0 for the toy demonstration, but a nonzero value was used in later simulations. Before cueing the system, the transient weights, $\mathbf{x}_0\mathbf{x}_0{}^{\mathrm{T}}$, are added to the weight matrix with a weight set to, $\lambda_{\max} + \beta\lambda_{\max}$ to ensure that the attraction of the initial cue dominates during recurrence. The parameter, β, was set to 0.001. The complete update function is given by,

$$\mathbf{x}_{t+1}^{T} = \frac{\mathbf{x}_t^T \left(\widehat{\mathbf{W}} + (\lambda_{max} + \beta\lambda_{max})\, \mathbf{x}_0\mathbf{x}_0^T\right)}{\left\|\mathbf{x}_t^T \left(\widehat{\mathbf{W}} + (\lambda_{max} + \beta\lambda_{max})\, \mathbf{x}_0\mathbf{x}_0^T\right)\right\|}.$$

Overall, the model has four tunable parameters, α (smoothing), k (negative evidence), η (dominant eigenvector inhibition), and β (excess force of the transient weights over the dominant eigenvector).

The six panels in Fig. 2 show the trajectory of activations (vertical axis) over recurrence iterations (horizontal axis) in the DEN, trained on the toy corpus, given various cues (shown in bold-face near the top left corner of each panel). The top-left panel shows the activations when cued with the word CAT. After a single iteration, the words that co-occurred with CAT in the same context—the episodic details—are activated (PLAYED and LEAF). The activation then spreads from those first-order associates to the second-order associates (BONE and DOG) over the next set of iterations, until the system reaches equilibrium at a state that has both the first- and second-order associates active. A similar pattern is observed when the system is cued with the word EAGLE, top-middle panel, where FEATHERS and WHITE are first activated, followed by BLACK and HAWK. Likewise, the top-right panel shows how the first-order associates (INCREASED and TAXES) are activated when the system is cued with LIBERAL, followed by the second-order associates like CONSERVATIVE and DECREASED. Spreading activation through the network is sufficient to activate latent relations in all three cases.

The bottom-left panel of Fig. 2 shows the trajectory of activations when the system is cued with the homonymous word, BANK, which was used in the context of the financial sense of the word with greater frequency relative to the other sense of the word: land surrounding a body of water. In the steady-state, the words with the dominant sense of the word are most active (e.g., MONEY, DEPOSITED, TRADERS, etc.). Words related to the secondary sense of the word are also active (i.e., GRASSY, RIVER, and BIRD), but rank lower in activations. The bottom-middle panel shows the activations when given the compound cue, BANK and MONEY. A subset of words including WITHDREW, SALESMAN, TRADERS, and DEPOSITED form a high activation cluster and another subset of words form a lower activation cluster (i.e., RAISED, INTEREST, BIRD, etc.). Cueing with the compound leads to the suppression of the secondary sense of the word relative to simply cueing with BANK. Finally, the bottom-right panel shows the activations when given the compound cue, BANK and BIRD, which result in RIVER and GRASSY being the most active, followed by words corresponding to the primary sense of the word (i.e., MONEY, SALESMEN, DEPOSITED, etc.). The patterns of activation illustrate DEN's context-sensitivity. The DEN does not treat meaning as a static representation, but as a dynamically constructed set of activations that are dependent on the context.[8]

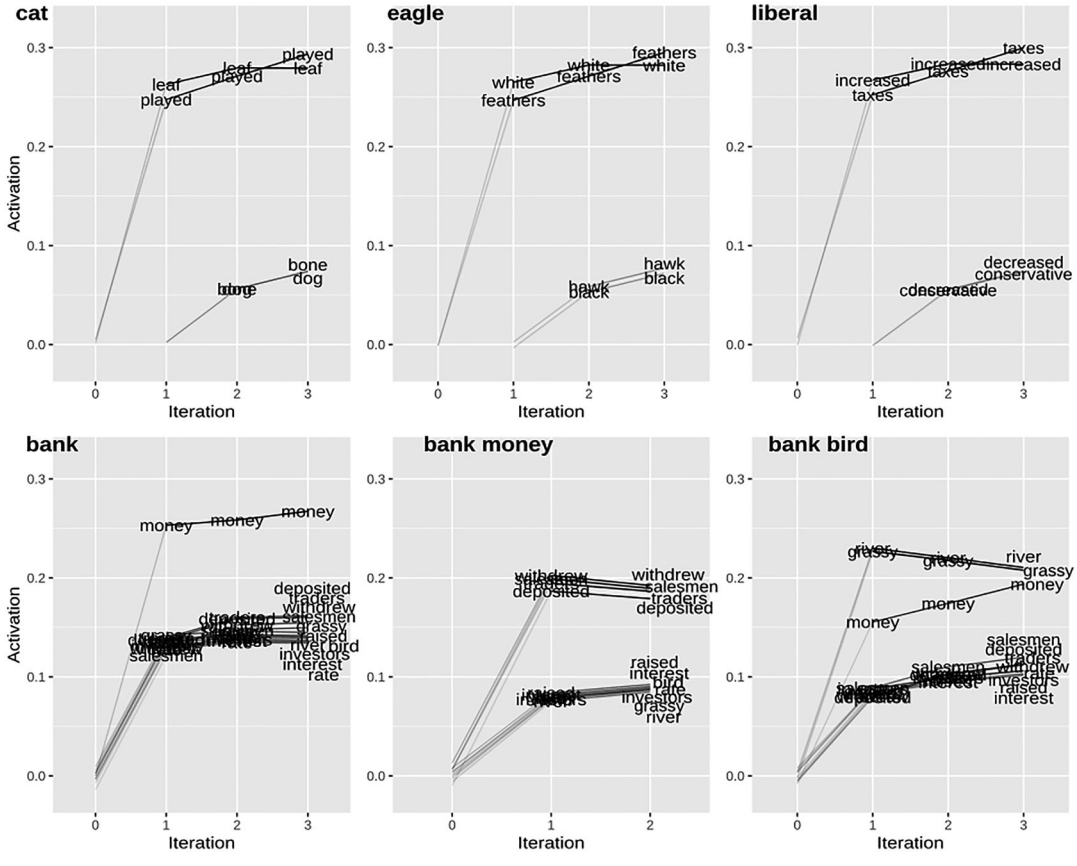

Fig. 2. Trajectory of activations during recurrence in the Dynamic-Eigen-Net.
Note. The six panels show the evolution of activations over recurrence iterations in the Dynamic-Eigen-Net, trained on a toy corpus, under various cues. The activation spreads from first- to second-order associates, reaching equilibrium with both active. The bottom panels demonstrate the context-sensitivity of the Dynamic-Eigen-Net, with different cues resulting in activations related to different senses of the word "BANK." The activations illustrate that meaning is a dynamically constructed set of activations dependent on context.

The meaning of a word is determined by its differences and similarities with other words along various axes corresponding to the eigenvectors of the weight matrix. Similarity forms between words most when they co-occur, but also to a lesser extent when they co-occur with the same neighbors, through alignment on nearby points along the axes defined by the eigenvectors. Differences form between words most when they do not co-occur and do not co-occur with the same neighbors, and to a lesser extent when they simply do not co-occur. The granularity with which each axis differentiates words is most general for the top eigenvectors and becomes finer for each additional eigenvector. For instance, EAGLE and HAWK never co-occur in the toy corpus, but they both co-occur with FEATHERS, whereas MONEY and FEATHERS neither co-occur nor share any neighbors.

Table 2
Four of the most active and four of the most suppressed words based on loadings on eigenvectors corresponding to the top six largest eigenvalues after encoding the toy corpus

λ_i	1	2	3	4	−4	−3	−2	−1
3.38	bank	money	withdrew	traders	factory	drove	car	garage
	0.54	0.4	0.25	0.25	0	0	0	0
2.34	played	leaf	cat	dog	factory	garage	truck	drove
	0.52	0.34	0.34	0.34	−0.01	−0.01	−0.01	−0.02
2.34	grew	tree	hill	flower	eagle	hawk	black	feathers
	0.55	0.36	0.36	0.36	0	0	0	0
2.34	taxes	**increased**	**decreased**	conservative	black	**eagle**	**hawk**	feathers
	0.48	**0.31**	**0.31**	0.31	−0.21	**−0.21**	**−0.21**	−0.32
2.34	drove	garage	car	factory	tree	field	flower	grew
	0.53	0.34	0.34	0.34	−0.19	−0.19	−0.19	−0.3
2.34	taxes	feathers	**conservative**	**liberal**	bone	leaf	cat	played
	0.42	0.39	**0.27**	**0.27**	−0.14	−0.14	−0.14	−0.21
1.95	river	grassy	bird	rate	deposited	withdrew	traders	money
	0.24	0.24	0.24	0.24	−0.28	−0.28	−0.28	−0.42
1.79	grassy	bird	river	truck	front	rate	raised	interest
	0.42	0.42	0.42	0	−0.02	−0.4	−0.4	−0.4
1.79	stood	investors	front	bank	interest	river	grassy	bird
	0.47	0.47	0.47	0	−0.24	−0.24	−0.24	−0.24
0.96	**eagle**	white	**decreased**	**conservative**	**liberal**	**increased**	black	**hawk**
	0.35	0.35	**0.25**	**0.25**	**−0.25**	**−0.25**	−0.35	**−0.35**

Table 2 shows the differentiation of words along each of the top 10 eigenvectors, in addition to the corresponding eigenvalue. The four highest and lowest valued words provide a general glimpse into the encoded structure. The left-most column shows the eigenvalues, from the largest (dominant) to the 10th largest. The next four columns show the most highly loaded words (the positive pole), whereas the last four columns show the most negatively loaded (the negative pole) or the null-loadings if the eigenvector is constrained to the positive quadrant. The dominant eigenvector is pointed toward the words BANK, MONEY, WITHDREW, and TRADERS and lies in the positive quadrant. BANK, the most frequent word in the toy corpus, has a strong influence on the first dominant eigenvector in the weight matrix due to its high frequency. The next eigenvector is pointed toward the word PLAYED and away from DROVE, and so forth. Each additional eigenvector adds an additional level of granularity in carving up semantic space. Tracking a collection of reference words (shown in bold font in Table 2) along the eigenvectors, the fourth eigenvector segments the space into a region around TAXES and another about FEATHERS. Despite not having occurred together, the words INCREASED and DECREASED are aligned on the positive pole and the words EAGLE and HAWK are aligned on the negative pole. Likewise, the words CONSERVATIVE and LIBERAL are aligned on the positive pole of the sixth eigenvector. Whereas the second-order associates like EAGLE and HAWK, DECREASED and INCREASED, and CONSERVATIVE and LIBERAL were aligned in the fourth and sixth eigenvectors, they are differentiated along the 10th eigenvector because of their lack of a first-order association.

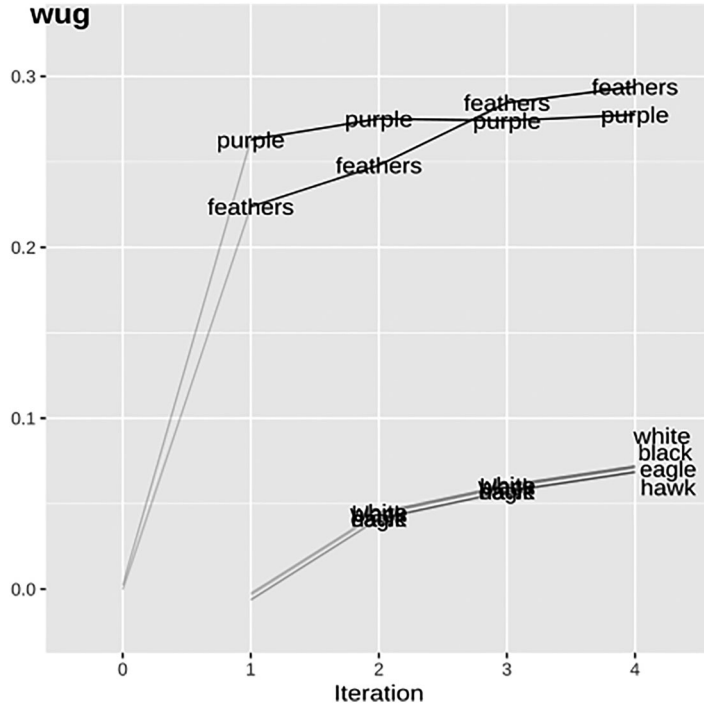

Fig. 3. Trajectory of activations when cued with a novel word in the Dynamic-Eigen-Net.
Note. Activation of second-order associates, including WHITE, BLACK, EAGLE, and HAWK, following a probe of the novel word WUG, encoded with associations to PURPLE and FEATHERS. A single presentation aligns WUG with relevant words like HAWK and EAGLE.

In LSA, second-order associations are made explicit by disregarding the lower-ranking eigenvectors, which eliminates highly specific differentiation in a manner similar to ignoring the lower-ranking eigenvectors (e.g., the 10th eigenvector) shown in Table 2. In the DEN, the second-order associations become explicit in activation instead of representation, and reflect the gravitation of the state vector toward the higher-ranking eigenvectors of the system. Since higher-order associations result from processing during retrieval, the DEN can easily assimilate novel words through their alignment in the higher-ranking eigenvectors. Fig. 3 shows activations after the system has been probed with the novel word WUG, upon encoding associations between the words WUG, PURPLE, and FEATHERS. In the steady-state, the second-order associates like WHITE, BLACK, EAGLE, and HAWK are activated. With just a single presentation, the system aligns WUG with relevant words like HAWK and EAGLE.

In the following sections, we make contact between models and human performance by exploring each model's ability to match patterns of human free word associations and word-pair evaluation judgments. In a free word association task, participants are asked to respond with the first word that comes to mind in response to a given cue word, without any restrictions on the type of response. In the word-pair evaluation task, participants are presented with

pairs of words and are asked to rate the similarity between the two words on a scale, such as from 1 to 7. The task measures the similarity of word pairs based on the participants' subjective judgment of their semantic relatedness. The patterns of free association and word-pair evaluation judgments across participants provide windows into the mental representation of language and the way words are related to one another.

3. Simulations

To compare our account with other commonly employed models, we used the same corpus to train LSA, the topic model, word2vec (CBOW), and the DEN, and examined each model's performance on the free association task, using the University of South Florida norms (USF; Nelson, McEvoy, & Schreiber, 2004), and several word-pair evaluation datasets where participants are asked to rate pairs of words on their similarity. Our results demonstrate how spreading activation using the DEN outperforms the other models in several cases and present it as a more parsimonious account. Finally, we used a set of norms, where raters categorized the relation between pairs of words into six different classes, to qualitatively profile each model in terms of their differential proclivity toward particular types of relations. Our main contribution is the demonstration that spreading activation using the DEN is capable of capturing the meaning of words as well as commonly used alternatives without relying on latent representations.

To construct LSA vectors, we first normalized a raw word-by-document co-occurrence matrix, $\mathbf{C}$, into the transformed matrix, $\mathbf{G}$, using,

$$\mathbf{G}_{ij} = \log_2\left(\mathbf{C}_{ij} + 1\right)\left(1 - \mathbf{H}_i\right),$$

where,

$$\mathbf{H}_i = -\frac{1}{\log_2(D)} \sum_{j=1}^{D} \left[\frac{\mathbf{C}_{ij}}{\sum_{k=1}^{D} \mathbf{C}_{ik}} \log_2 \left(\frac{\mathbf{C}_{ij}}{\sum_{k=1}^{D} \mathbf{C}_{ik}} \right) \right].$$

The variable D is the number of documents. We then applied Singular Value Decomposition (SVD) to reduce the dimensionality of each $\mathbf{G}_i$ from the original 37,650 to 700.

To train the topic model, we used the same procedure as Griffiths et al. (2007). We fixed the number of topics at $K = 1700$ and set the smoothing parameters over documents and words to $50/K$ and $200/V$, respectively. For the Gibbs sampling, we ran three separate chains, each with 800 burn-in trials. After the burn-ins, we used eight samples, each separated by 100 thinning samples, from each chain to estimate the posterior.

For training word2vec, we set the embedding dimensionality to 200, and used a sliding window—same as the DEN—over the corpus. We used the negative sampling optimization algorithm, and trained the network over 40 epochs. The parameters for the DEN mirror those used with word2vec. The parameters used in the following simulations are shown together in Table 2. The resulting SPPMI weight matrix was very sparse, and contained about 0.4% nonzero entries.

We used the USF norms for the free association task in order to match results from Griffiths et al. (2007). We used the same procedure that they describe to preprocess the training corpus and kept it fixed across models. That is, we used the TASA corpus and filtered out any word that either occurred less than 10 times or was in a stop-list to follow Griffiths et al. (2007). Inhibiting the first eigenvector in the DEN eliminates the need to use a stop-list; however, we train all models on the same corpus for the sake of comparison. This left us with a corpus of $V = 52{,}046$ word types over $D = 37{,}650$ documents; the total number of word tokens was 11,518,807 before preprocessing and 4,402,747 afterward.

With LSA and word2vec, to predict the free associates of a given cue, the cue's vector cosine with every other word was obtained and the word with the largest cosine was treated as the response. We rank-ordered the words in decreasing order, and checked the most probable human free associate's rank. A rank of 1 corresponds to a perfect match, whereas a rank of 2 means that the most common associate is below another word, and so forth. We also calculated the percentage of times the most probable free associate, based on the USF norms, was also the top most active word. We followed a similar procedure for the second-most, third-most, n'th most probable free associates according to the USF norms. Overall, lower values for ranks and higher values for the percentage of n'th associates imply a better match to the human free association norms.

The same procedure was applied to the topic model by swapping cosines for probabilities. The probability of seeing a response, given a cue, was obtained by marginalizing over the topics,

$$p(response|cue) = \sum_{topic} p(response|topic)p(topic|cue).$$

In addition to the four models, we also provide results from the direct associative strengths corresponding to the weight matrix (referred to as SPPMI). That is, for a given cue, the associative strengths in the corresponding row in the weight matrix were treated as activation strengths. Better performance in the DEN relative to SPPMI demonstrates a performance advantage gained by spreading activation. For simplicity, we will refer to the associative strengths obtained from all models as strengths.

Following Griffiths et al. (2007), the ranks in activation were restricted to the intersection of the words that were used as either cues or responses in the free association norms, with the words in the corpus. The intersection of the word types in the USF norms and TASA was 4566.

The conditional probability of each of the most probable free associates from the USF norms, given the corresponding cue, ranges greatly between highly probable cue-response pairs like SHOVE-PUSH, $p(\text{PUSH} \mid \text{SHOVE}) \approx .936$, to very improbable cue-response pairs like BENEATH-OCEAN, $p(\text{OCEAN} \mid \text{BENEATH}) \approx .013$. We used a separate free association dataset to estimate the expected probability that a random participant would generate the n'th most probable associate in the USF norms, given each cue, to obtain an estimate based on stable linguistic regularities that are not dependent on the spatiotemporal context characterizing the collection of data in the USF norms. We used the Small-World-of-Word's (SWOW;

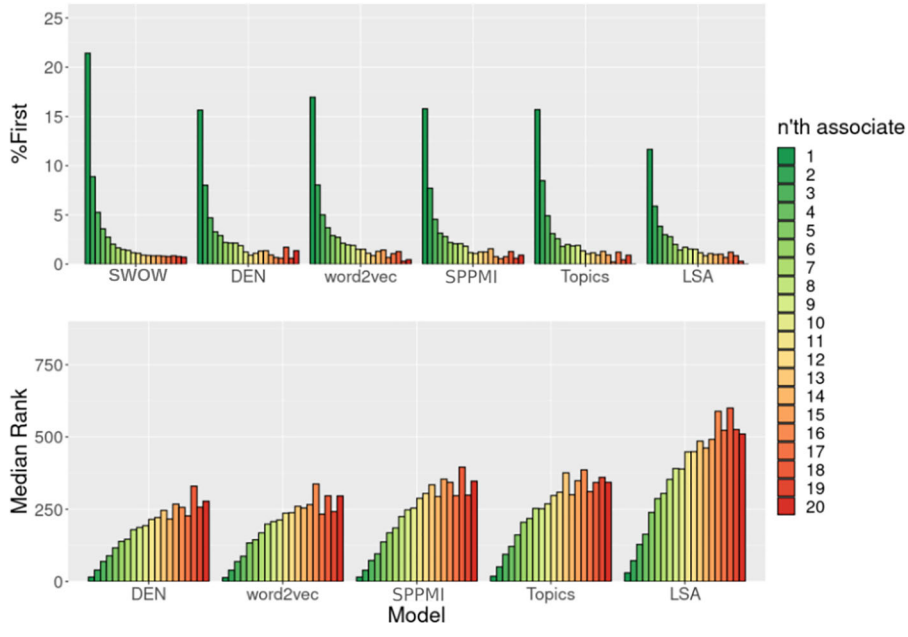

Fig. 4. Predicting free associates.

Note. The top panel shows how the expected percent of times that the *n*'th most probable response from the USF free association norms is reported. The estimates using the Small-World-of-Words (Human) norms drop when shifting from the most probable response to the second-most probable response and so-forth up to the 20'th most probable response. The percent of times the *n*'th most probable response is the most active across the Dynamic-Eigen-Net (DEN), word2vec, direct associations (SPPMI), the topic model, and LSA show the same pattern. The lower panel shows the median rank of activation for the *n*'th most probable response across cues, the pattern rises when shifting from the most probable response to the less probable responses. We do not show the median ranks for SWOW because their estimation is less direct than the estimation of response probabilities.

De Deyne et al., 2019) free word-association dataset to estimate the expected rate using,

$$\mathbb{E}(r^n|c) = \frac{1}{N}\sum_{i=1}^{N} p\left(r_i^n|c_i\right),$$

where $r_i{}^n$ denotes the *n*'th most probable response to the *i*'th cue, c_{i}, based on the USF norms and the conditional probabilities, $p(r_i{}^n|\ c_{\mathrm{i}})$, correspond to estimates from the SWOW free association norms. According to our estimate, a random participant in the SWOW dataset only had about a 21.5% chance of generating the most probable first associate in the USF norms. Our choice of USF for this purpose aligns the current work with previous modeling work by Griffiths et al. (2007).

Fig. 4 summarizes the overall match to the USF norms across models. The top panel shows the percent of the time the *n*'th most probable free associate had the highest strength across the models, in addition to the expected percentage of times that a random participant would

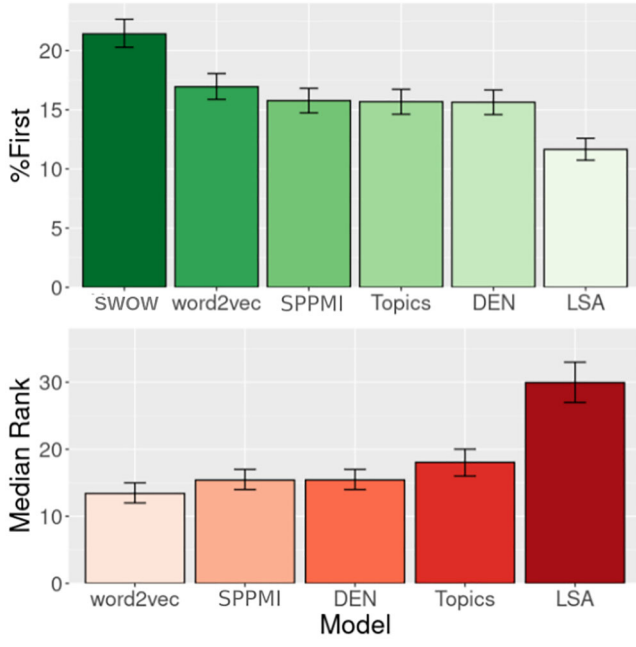

Fig. 5. Predicting the most probable free associate.
Note. The top panel shows the expected percent of times the most probable response in the USF free association norms is reported using the Small-World-of-Words (Human) norms only somewhat exceeds the percent of times the most probable response is activated across word2vec, direct associations (SPPMI), the topic model, Dynamic-Eigen-Net, and LSA. The error-bars are derived from a proportions test. The lower panel shows the median rank of the most probable response in the USF free association norms across the models. The error-bars are based on 1000 bootstrap samples. We do not show the median ranks for SWOW because their estimation is less direct than the estimation of response probabilities.

generate the response when given the corresponding cue. Overall, all models lag slightly behind the human estimates (SWOW) when predicting the most probable free associate; however, both models and the human estimates show a similar pattern of reducing rate of generating the *n*'th associate as the probability of generating the associate in the USF norms falls. The bottom panel shows the median rank of the *n*'th associates in their strength across the models, where lower values indicate higher strengths. Taken together, Fig. 4 shows how co-occurrence-based statistics provide a good basis for predicting human free associates, regardless of the specific choice of representational and processing assumptions. Out of the models, LSA yields the lowest percentage of the free associates and the largest median ranks. SPPMI shows a similar profile as DEN for the percentage of the free associates, but the median ranks show how DEN slightly pushes the free-associates closer to the top in strength.

Fig. 5 shows the percentage of the first associates that are most activated in the top panel (in addition to the expected percentage of times a random participant should report the first associate) and the median ranks of the first associate, in the bottom panel, across the models. We

conducted a Bayesian proportions test (generalization of the binomial test to more than two groups) to obtain 95% credible intervals of the differences between model and human predictions for the percentage of first associates; the error-bars show the 95% credible intervals. On the bottom panel, 1000 bootstrap samples were drawn to obtain the 95% confidence intervals for the median ranks, indicated by the error-bars. Overall, the percentage of first associates is lower for the models relative to estimates of human associates. With the exception of LSA, the difference in the percentage of first associates that have the highest strength is not very different from one model to another. LSA is worse at predicting free associations. The pattern is mirrored in the median ranks, with all models yielding comparable median ranks, except for LSA, where the median rank of the first associate is higher. First associate strengths derived from the topic model also appear to rank higher relative to word2vec. Strengths derived from word2vec have a slight tendency to be more likely to favor the first associates than the DEN and SPPMI.

3.1. Matching human word-pair evaluation judgments

Providing another perspective for comparing human semantic processing to models, several word-pair evaluation judgment norms have accumulated, each compiled by asking human raters to judge the *relatedness* or *similarity* of word pairs. We followed a similar procedure for applying the models to the word-pair evaluation datasets from Miller and Charles (1991; MC, $n = 30$), Bruni, Tran, and Baroni (2013; MEN, $n = 3000$), Radinsky, Agichtein, Gabrilovich, and Markovitch (2011; MKTurk1, $n = 771$), Halawi, Dror, Gabrilovich, and Koren (2012; MKTurk2, $n = 287$), Luong, Socher, and Manning (2013; RareWord, $n = 2034$), Rubenstein and Goodenough (1965; RG, $n = 65$), Hill et al. (2015; SimLex, $n = 999$), Gerz et al. (2016; SimVerb, $n = 3500$), Yang and Powers (2006; YP, $n = 130$), Finkelstein et al. (2001; WS1, $n = 353$), and Agirre et al. (2009; WS2, $n = 353$). For each word pair, we used both words as cues and averaged the corresponding strength of the other word. The match to each dataset was quantified as the Spearman's correlation between the model-derived strengths and corresponding human word-pair evaluation ratings.

Instructions provided to raters vary greatly across the word evaluation datasets. The MC, MKTurk1, RareWord, WS1, and YP datasets emphasize similarity, whereas WS2 emphasizes relatedness. The MEN and RG datasets do not differentiate between similarity and relatedness. Two of the datasets, SimLex and SimVerb, not only emphasize similarity, but also explicitly direct participants to give low ratings antonyms (e.g., HOT and COLD).

Fig. 6 shows the Spearman's correlation between strengths derived from each of the models and the ratings obtained from the 11 word evaluation datasets. The top-left panel shows correlations with the two datasets that emphasize similarity and stipulate low ratings to antonyms and the top-right panel shows correlations with the four datasets that only emphasize similarity. The bottom-left panel shows correlations with the dataset that emphasizes relatedness, and the bottom-right panel shows correlations with the datasets without explicit emphasis on the type of evaluation. The error-bars show the 95% confidence intervals obtained from drawing 1000 bootstrap samples.

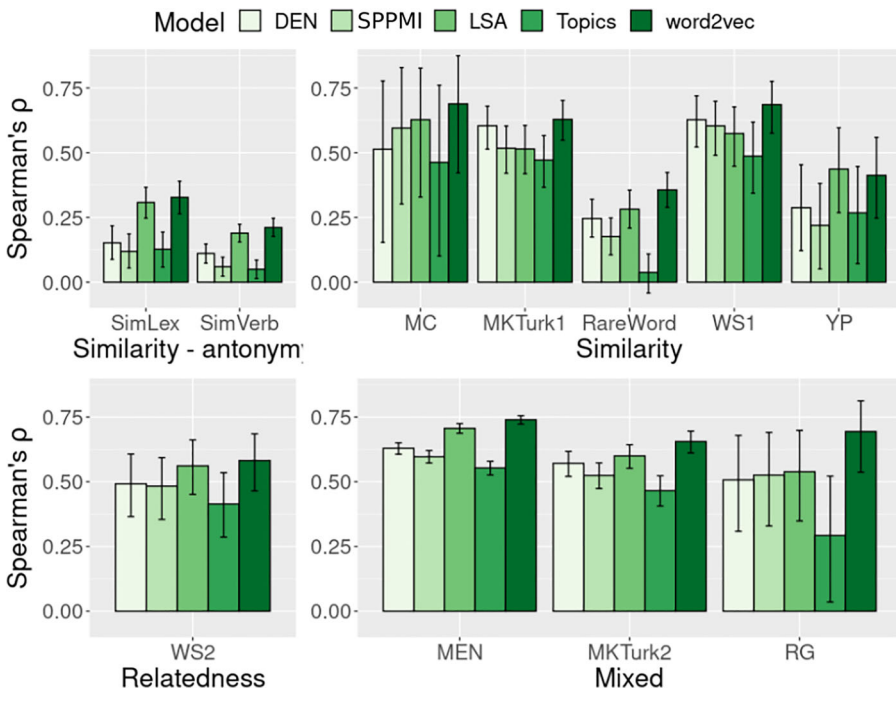

Fig. 6. Predicting human word similarity norms across models.

Note. Spearman's correlation between model-derived activations and nine commonly used word similarity datasets is shown across models. The correlations are separated into four groups based on the instructions given to human judges. The top-left panel shows correlations for the datasets that emphasize similarity (e.g., cup and mug) over relatedness and direct raters to treat antonyms as dissimilar. The top-right panel shows correlations on datasets that emphasize similarity. The bottom-left panel shows correlations with the dataset that emphasizes relatedness (e.g., cup-coffee). The bottom-right panel shows correlations with datasets that do not instruct participants to differentiate between similarity and relatedness. The error-bars are derived from 1000 bootstrap samples.

Overall, the correlations are lowest for the SimLex and SimVerb datasets, likely due to the lack of differentiation between antonyms in co-occurrence-based models. Interestingly, strengths derived from LSA and word2vec show higher correlations with the SimLex ratings compared to strengths derived from the other models. A similar pattern is present for the SimVerb dataset, with word2vec and LSA leading, but the magnitude of correlations is lower across models. When similarity is emphasized without the explicit suppression of antonyms, strengths from all models yield approximately similar correlations with human ratings, although word2vec trends toward slightly higher correlations. Correlations are lowest for the RareWord dataset, which uses low-frequency words, followed by the YP dataset, which uses verbs. Correlations with model-derived strengths and word evaluation datasets that emphasize relatedness and those that contain a mixed number of correlations are moderate-to-high. Overall, strengths derived from the topic model yield the lowest correlations, strengths

from the DEN yield larger correlations than SPPMI, and strengths from word2vec and LSA show the highest correlations.

To explore the kinds of relations the different models capture, we obtained a set of norms[9] where five raters categorized a set of cue-response pairs into one of six categories, including "syntagmatic," "paradigmatic," "forward," "backward," "form," and "other." The raters were asked to categorize pairs that tend to occur in the same context (e.g., WEB and SPIDER) as syntagmatic (cf., relatedness), and pairs that can occur in place of one another (TROUSERS and PANTS or EAST and WEST) as paradigmatic (cf., similarity). When a response (PICKLE) tends to succeed the cue in serial order (DILL), the raters were asked to label the pair as having a forward association and when a response (DILL) tends to precede the cue (PICKLE), it was to be labeled as a backward association. If the two words had phonetic (EYE and I) or orthographic (CHOIR and CHORE) overlap, their relation was labeled as form-based. A final category, "other," was included to make the classification exhaustive. For any given pair, each rater was asked to choose one relation. Affinity toward form-based and the "other" category are not expected by any models. We designated the relation with the most votes across the five raters to be the dominant relation for each of the word-pairs, and probed the models with one member of the pair and tallied the median rank of the other word into bins corresponding to the dominant relation. Overall, there were 151 cue words and 1794 response words, totaling 4619 cue-response pairs. Of those pairs, 654 were binned as paradigmatic, 412 were binned as forward, 2345 were binned as syntagmatic, 615 were binned as backward, 201 were binned into the "other" category, and 392 were binned as form-based. We base the ranks on the intersection of the words in the norm-set and those in the TASA corpus, totaling 1371 words.

Fig. 7 shows the median rank of the responses, given the cues, across word relations and models. As with the free-association ranks, lower valued ranks indicate higher strengths for the responses relative to other words, when each model was probed by the corresponding cues. The error-bars show the 95% confidence intervals derived by drawing 1000 bootstrap samples. Responses that are syntagmatically linked to the cues have equivalent ranks for all models, but LSA, which yields poorer ranks. Responses that are paradigmatically linked to the cues show the highest ranks for word2vec relative to the other models. Responses that follow or precede cues yield similar ranks across models, except for LSA, which yields poorer ranks. The forward associations yield stronger ranks relative to backward associations across all models. Response ranks that overlap with the cues in form have no identifiable overlap since the model lacked sublexical representations.

3.2. Robustness of DEN, word2vec, and SPPMI

Since the word2vec, SPPMI, and DEN had analogous hyperparameters, we performed a grid search over a set of hyperparameters for all three models and evaluated their ability to match the USF norms with different configurations. Specifically, we varied the size of the context window, weight of the context smoothing, and the number of negative samples and tallied the median rank of first associates in their derived response strengths, when probed with the

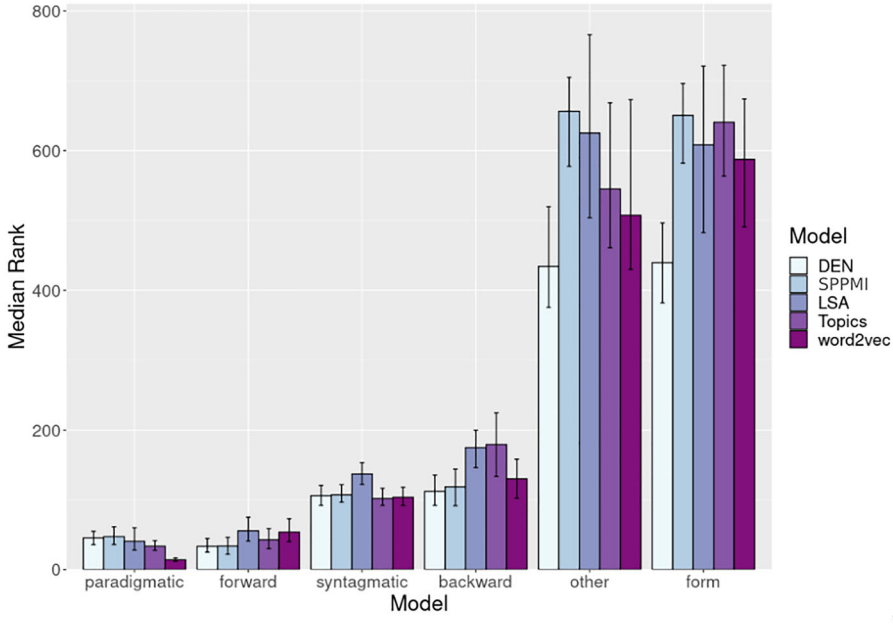

Fig. 7. Affinity for different relations across models.
Note. Median ranks in activation for different types of relations across models. Error-bars are derived from 1000 bootstrap samples.

corresponding cues, in addition to the percent of times the most probable free associate was ranked highest in strength.

When the associations between each target word and its contexts are computed using SPPMI with context distribution smoothing, setting the context smoothing parameter to 0 treats the contexts (columns in a word-by-word matrix) of each target (rows) as equally probable:

$$p_j = \frac{\left(\mathbf{C}_j + \alpha V\right)^{\alpha}}{\sum_{l=1}^{V}\left[\left(\mathbf{C}_l + \alpha V\right)^{\alpha}\right]} = \frac{\left(\mathbf{C}_j\right)^{0}}{\sum_{l=1}^{V}\left[\left(\mathbf{C}_l\right)^{0}\right]} = 1/V,$$

$$\mathbf{W}_{ij} = \log_2\left(\frac{p_{ij}}{p_i p_j}\right) - \log_2(k) = \log_2\left(\frac{p_{ij}}{p_i \frac{1}{V}}\right) - \log_2(k).$$

The contribution of the contextual base-rates based on the corpus derived co-occurrence counts increases as the smoothing parameter approaches 1, at which point it reduces to add-one smoothing. We varied the context smoothing parameter from 0.1 to 0.9 to cover the range in between the two extremes. For the range of parameters of the context window size and negative evidence parameters, we incrementally increased each parameter's magnitude

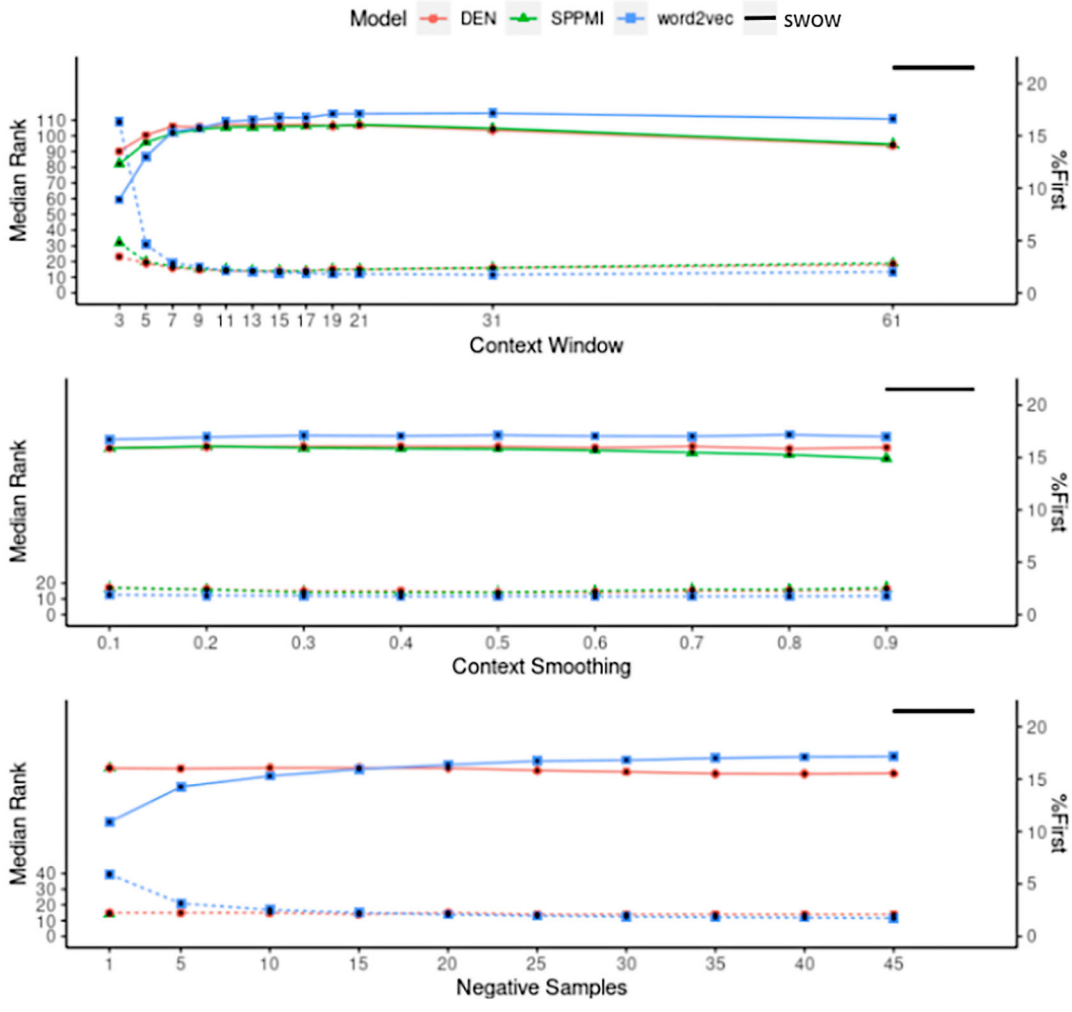

Fig. 8. Parameter sweep across three key parameters.

Note. Median rank (dashed lines; left vertical axis) and the percent of first associates that are most active (solid lines; right vertical axis) across the size of the context window (top panel), context smoothing (middle panel), and the number of negative samples (lower panel) for word2vec, Dynamic-Eigen-Net (DEN), and the direct associations (SPPMI). The parameters include the size of the context window, the magnitude of the context distribution smoothing, and the number of negative samples. The horizontal lines near the right y-axes correspond to the expected percent of first associates from human participants.

until asymptote. Every level of each parameter was combined with each level of every other parameter in a grid.

Fig. 8 shows the results of the grid search over parameters that were used for word2vec, SPPMI, and the DEN. Each of the three panels shows the median rank of first associates

(i.e., responses with the highest probability, conditioned on the corresponding cues) on the left vertical axis (dashed lines) and the percentage of times the first associates ranked highest in activation on the right vertical axis (solid lines). The top panel shows how the DEN (red square) reaches asymptotic performance with a much smaller context window relative to SPPMI (green triangle) and word2vec (blue square) which start reaching asymptotic levels with a context window of size 7. Word2vec surpasses the DEN in performance with context window sizes greater than 9 and peaks with a context window of size 31. SPPMI is always below the DEN in performance; however, both reach very similar levels as the context window is increased. One reason the DEN can reach asymptotic levels with a smaller context window is that the spread of activation implicitly expands the scope of associations to reach beyond the explicit window size. That is, as activation is spread, neighbors of neighbors are also activated, leading to the satisfaction of constraints that reach beyond SPPMI's scope of associations. The middle panel shows a relatively flat level of performance across various values of context smoothing; however, small gains are achieved across all models when the smoothing parameter is near 0.5. Finally, the bottom panel shows how performance is relatively stable across different numbers of negative samples for the DEN and SPPMI, but varies greatly for word2vec. Word2vec surpasses the DEN and SPPMI in its match to the USF norms with 20, or greater, negative samples. Taken together, the parameter sweep shows that the DEN is more robust than word2vec and SPPMI with respect to the size of the context window, and neither the DEN nor SPPMI are impacted by the number of negative samples, whereas word2vec depends on negative sampling to match human free associations.

The DEN's ability to reach asymptotic performance more rapidly than word2vec and SPPMI, as the size of the context window increased from three to larger values, motivated us to explore how quickly the match to the free association norms reached its asymptote as a function of the proportion of the corpus encoded. Fig. 9 shows the median rank of the first associates and the percent of times the first associates were the most active across different snapshots during training, for the DEN, SPPMI, and word2vec. The figure illustrates the fast learning capacities of the DEN relative to the two other models.

The DEN approaches asymptotic performance much more rapidly than both SPPMI and word2vec, and the pattern is most stark when examining the median ranks. The three models converge after about half of the corpus has been encoded, and word2vec moderately exceeds the DEN in its match to the free association data when the entire corpus has been encoded. Out of the three models, the DEN requires the least amount of training input to capture the relevant statistical regularities in the data.

Levy and Goldberg (2014; also see Levy, Goldberg & Dagan, 2015) provide derivations that show how word2vec's objective function enforces dimensionality reduction through the implicit factorization of a word-by-word matrix, when treating the *i*'th cell as the target and all corresponding *j* columns as its context. They showed how using SVD to apply linear dimensionality reduction on an SPPMI matrix improves correlations between vector similarities and human word similarity ratings. In the earlier simulations, we used a commonly used entropy-based normalization prior to applying SVD to form LSA vectors in order to match prior work, particularly by Griffiths et al. (2007). In a final set of simulations, we further explore the impact of dimensionality reduction to the SPPMI matrix.

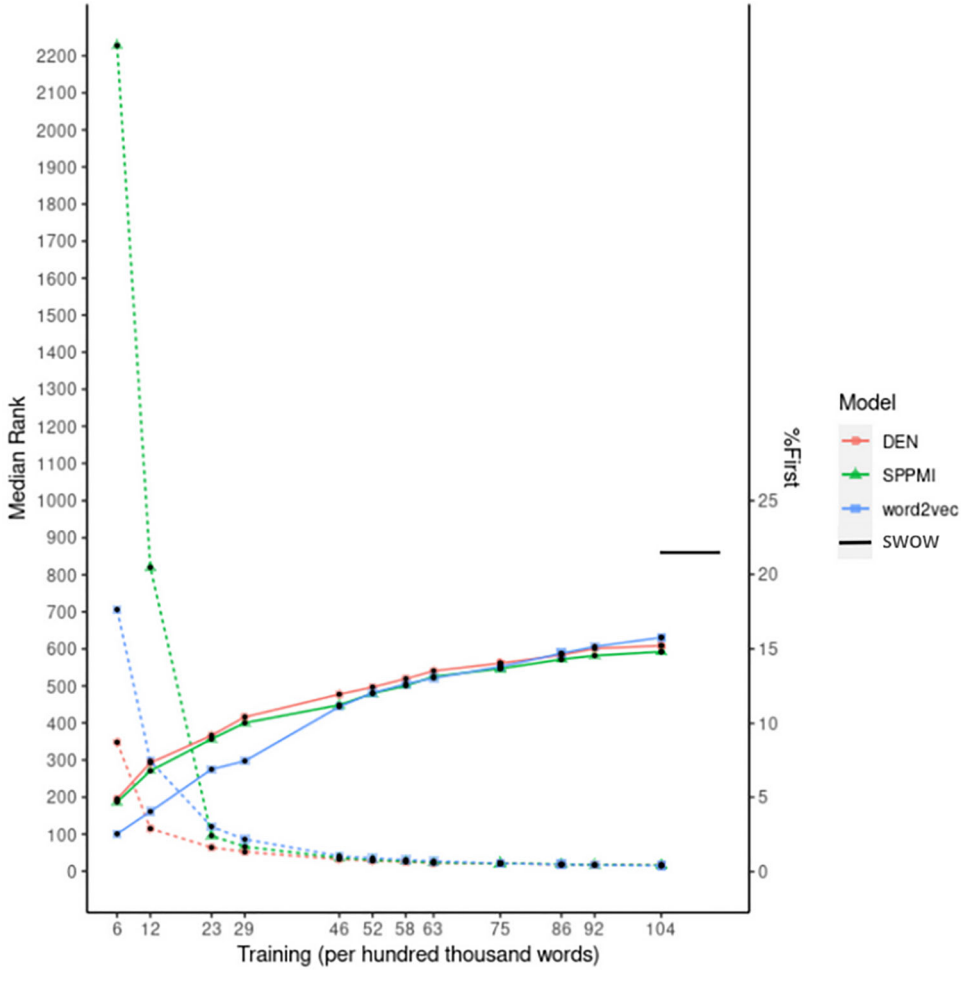

Fig. 9. Match to USF free association norms and number of training tokens.
Note. The left vertical axis corresponds to the median ranks (dashed lines) and the right vertical axis corresponds to the percent of times the most probable free associate ranked highest in activation (solid lines) for Dynamic-Eigen-Net, direct associations (SPPMI), and word2vec. The horizontal line near the right y-axis corresponds to the expected percent of first associates from human participants.

We constructed an SPPMI matrix using the same smoothing, context window size, and "negative samples" as we used in the DEN in the earlier simulations (presented in Table 3).[10] Spearman correlation coefficients were then computed to capture the degree of match between model-derived strengths (vector cosines) and human judgments from the set of similarity judgment norms presented in Fig. 6. We calculated the correlations for each norm set, over a range of dimensionalities from 2 to 16,384. Fig. 10 shows the correlation (vertical axis)

Table 3
Model hyperparameters used in simulations

Model	Context	α	k	η	β	Median rank	%First
DEN	12	0.50	15	0.90	0.001	15.00	15.64
word2vec	30	0.40	45	–	–	12.55	17.06

Note. The parameter α is the smoothing parameter, k corresponds to the number of negative samples, η corresponds to the weight of the dominant eigenvector inhibition, and β is the bias toward the initial cue during recurrence.

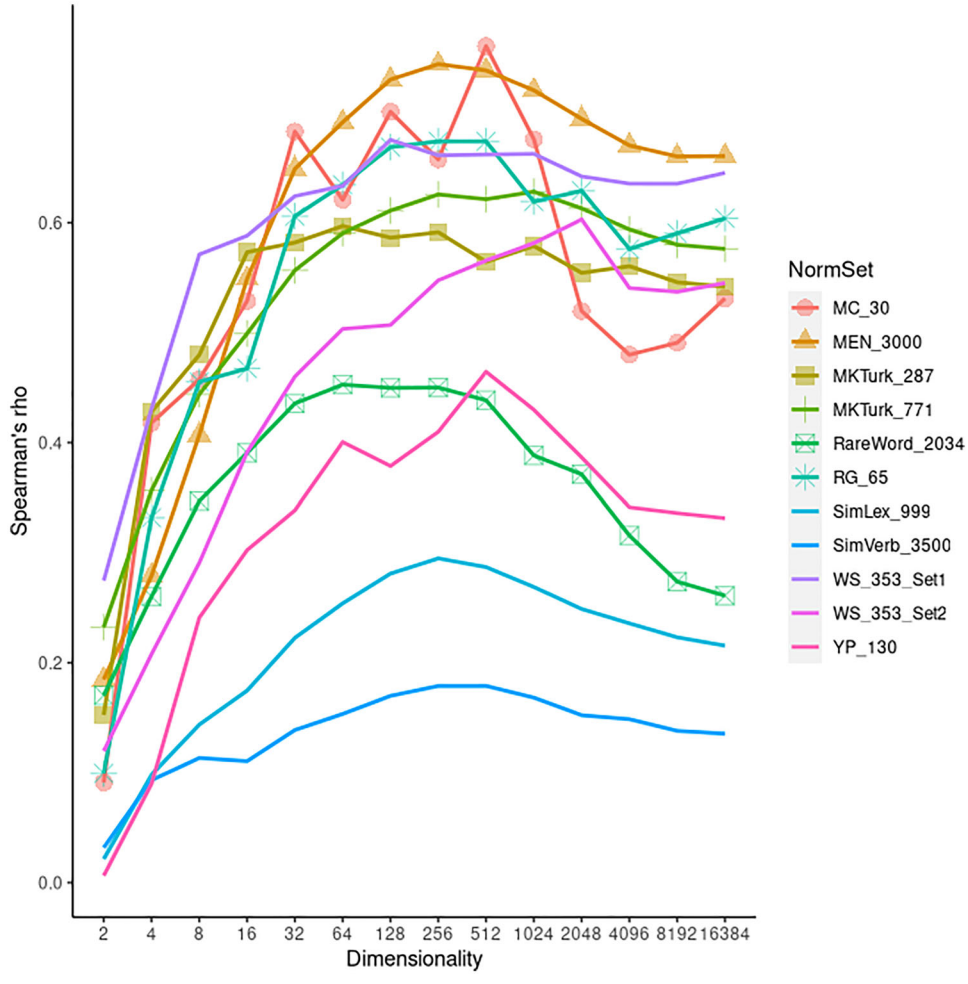

Fig. 10. Correlation between the compressed SPPMI matrix and human word similarity norms at varying dimensionalities.

between each norm set (different symbols and lines) and model-derived similarities, over the range of dimensionalities (horizontal axis on a log scale).

Overall, the match between human similarity norms and vector similarities peaks when the compressed representations are projected to around 512 dimensions. For the two norm sets that emphasize similarity and prevent the attribution of high similarity judgments to antonyms through instructions to raters (SimLex and SimVerb), latent semantic vectors reach peak performance at lower dimensionalities (256). For the WS2 dataset, which emphasizes relatedness, peak performance is obtained at much higher dimensions (2048). Of the paradigmatic ("similarity")-based datasets, a dimensionality of 512 yields the best correlations for MC, WS1, and YP, but much lower dimensionalities are needed to reach peak correlation for the MKTurk1 (32) and RareWord (64) datasets.

Across the range of dimensionalities explored, peak correlations between human ratings and vector similarities based on compressing the SPPMI fall within the regions of error of correlations with vector similarities using an entropy-based normalization shown in Fig. 7. The one exception is for the RareWord dataset, which appears slightly higher when applying dimensionality reduction to SPPMI to project into a space of around 64 dimensions. LSA vectors highly correlated with most word similarity datasets relative to other models; however, they were clearly inferior to the other models when used to predict free associations using the USF norms. In the next simulation, we explored whether applying dimensionality reduction to the SPPMI matrix instead of an entropy-weighted word-by-document matrix facilitates a greater match between model-derived similarities and free association norms.

Fig. 11 shows both the median ranks and percentages of first associates based on predictions from vectors derived at different dimensionality reductions applied to the same SPPMI matrix used in the word similarity simulations (i.e., matching Table 3 parameters for DEN). As in Fig. 9, the median ranks of the first associates (dashed line) correspond to the left vertical axis and the percentages of first associates predicted by the model (solid line) correspond to the right vertical axis, but as a function of dimensionality instead of training size. The gray horizontal line marks the best median rank achieved using the DEN (15). The median rank of the first associates sinks and the percentage of first associates predicted rise as embedding dimensionality increases from 16 to around 512, at which point the match between the model and human free associations reaches asymptote, with marginal improvements dimensionality is increased to around 4096. The lowest median rank (14) is obtained with dimensionality set to 4096, a 1-point improvement over the DEN but a substantial improvement over the entropy-based LSA presented earlier which yielded a median rank around 30.

In general, reducing dimensionality to 512 achieves a good match to human norms in both word similarity and free association tasks, but peak performance in the free association tasks is reached with higher dimensions around 4096. Since both words are presented in each trial of a word similarity judgment tasks, the demands simply require a match operation, whereas in the free association task, only a single cue word is presented and the response depends on undirected retrieval. It is possible, that on average, a richer set of associative structures are evoked during a free association trial compared to a word similarity judgment trial. A higher embedding dimensionality being a precondition to the formation of those associative struc-

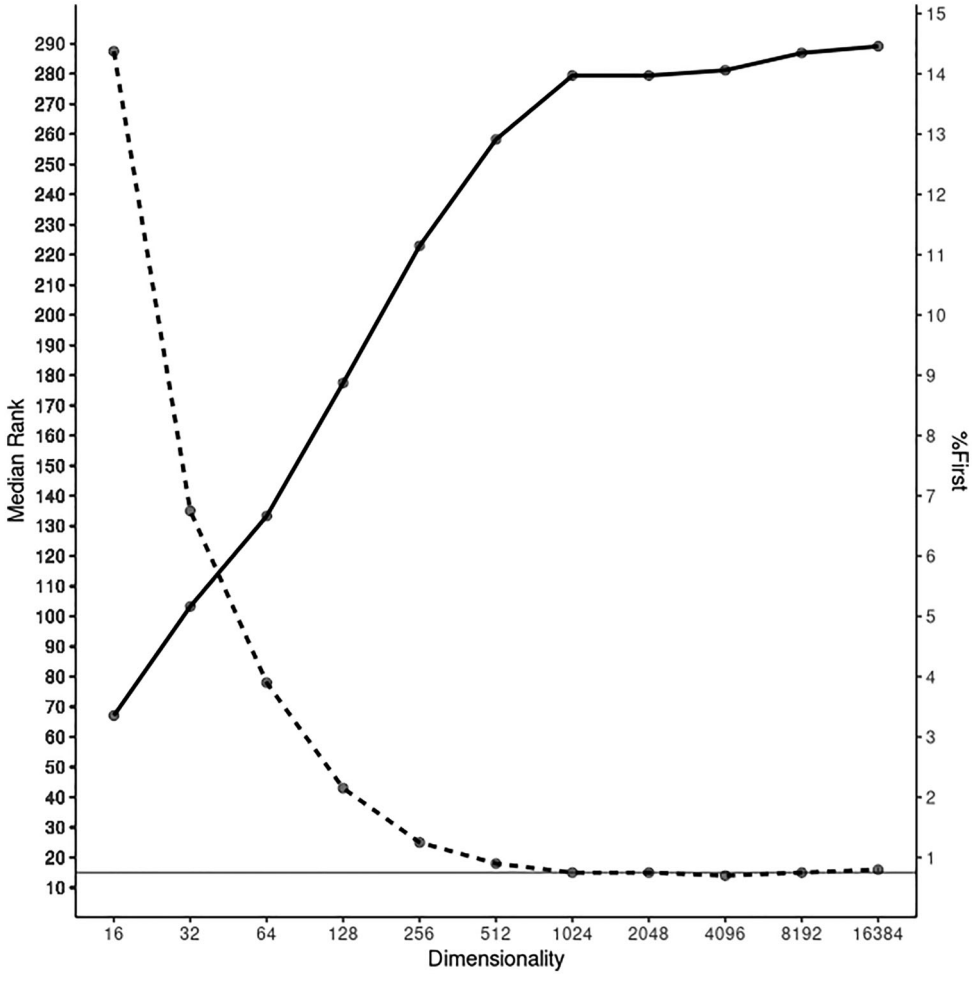

Fig. 11. Median rank and percent of first associate match from compressed SPPMI matrix at varying dimensionalities.

Note. The left vertical axis corresponds to the median ranks (dashed lines) and the right vertical axis corresponds to the percent of times the most probable free associate ranked highest in activation (solid lines). The horizontal line shows the median rank derived from the Dynamic-Eigen-Net.

tures can explain why a larger embedding size led to peak performance in the free association results presented in Fig. 11 compared to the peak word similarity results presented in Fig. 10.

In an SPPMI matrix, each word's history of usage—each row—is encoded as a vector of weights that estimate its conditional dependence on other words. Each element of a word's vector is nonzero, only if the word directly co-occurred with the context word corresponding to the element. With greater levels of compression, each word's representation of its contextual history is further abstracted away from a record of its direct associates. As a relatively

small number of dimensions must approximate a much larger number of dimensions, both surface-level (e.g., EAGLE → FEATHERS) and higher-order (e.g., EAGLE → HAWK) structure is blended to maximize sensitivity to the highest varying dimensions of variance.

In a final set of simulations, we replicated the word relation profiling first presented in Fig. 7 at multiple compression dimensionalities of the SPPMI matrix to determine if maximum affinity of the constructed semantic spaces to different kinds of relations between words (e.g., syntagmatic vs. paradigmatic) fall along different embedding sizes. Assuming that greater levels of dimensionality reduction lead to further abstraction from surface-level associative structure, we expect greater affinity for paradigmatic relations at lower embedding dimensionalities. Since two words' paradigmatic affinity for one another is proportional to the frequency with which one word is substituted for the other in similar contexts, optimality favors the reconfiguration of points in the original uncompressed semantic space to points in the compressed space that merge paradigmatically related words together. Therefore, we expect greater affinity for paradigmatic relations at lower embedding dimensionalities. In contrast, relations that are syntagmatic, forward-serial, or backward-serial are more likely to depend on surface-level information.

Fig. 12 shows the median rank (vertical axis) of responses to cues when using SPPMI vectors compressed to different embedding dimensions (horizontal axis). The median ranks are grouped by the type of relation (different lines and symbols) each response had to the cue based on word relation norms presented earlier in Fig. 7. Lower values of median ranks for each relation class indicate greater affinity for the relation. With the exception cue-response pairs classified to have a relation of type "form" or "other," affinity for "backward," "forward," "paradigmatic," and "syntagmatic" relations accelerates as the embedding dimension increases from 2 to 64, after which point affinity for different relations approaches asymptote. A dimensionality of 512 yields the maximum level of affinity for paradigmatic relations, with slight decrements in paradigmatic affinity with more dimensions. Affinity for syntagmatic, forward, and backward relations peaks at a dimensionality of 1024.

4. Discussion

After a brief review of commonly used models that learn the meaning of words using word-by-word co-occurrence statistics derived from large text corpora, we noted how all models assume that first-order associative structure is transformed into a latent representation in order to capture higher-order associative structure. We argued that the assumption of a latent representation has several problems. First, latent representation models are slower to train and require large training data to capture semantic relations between words. Second, the latent representations of words are formed without taking into account the context in which the words will be used in the future. Third, the transformation of surface-level associations into a latent structure compresses representations into a form that leaves them vulnerable to cross-talk. Fourth, latent representation models entail a dual-process account of human memory since the retrieval of episodic details requires a separate episodic store.

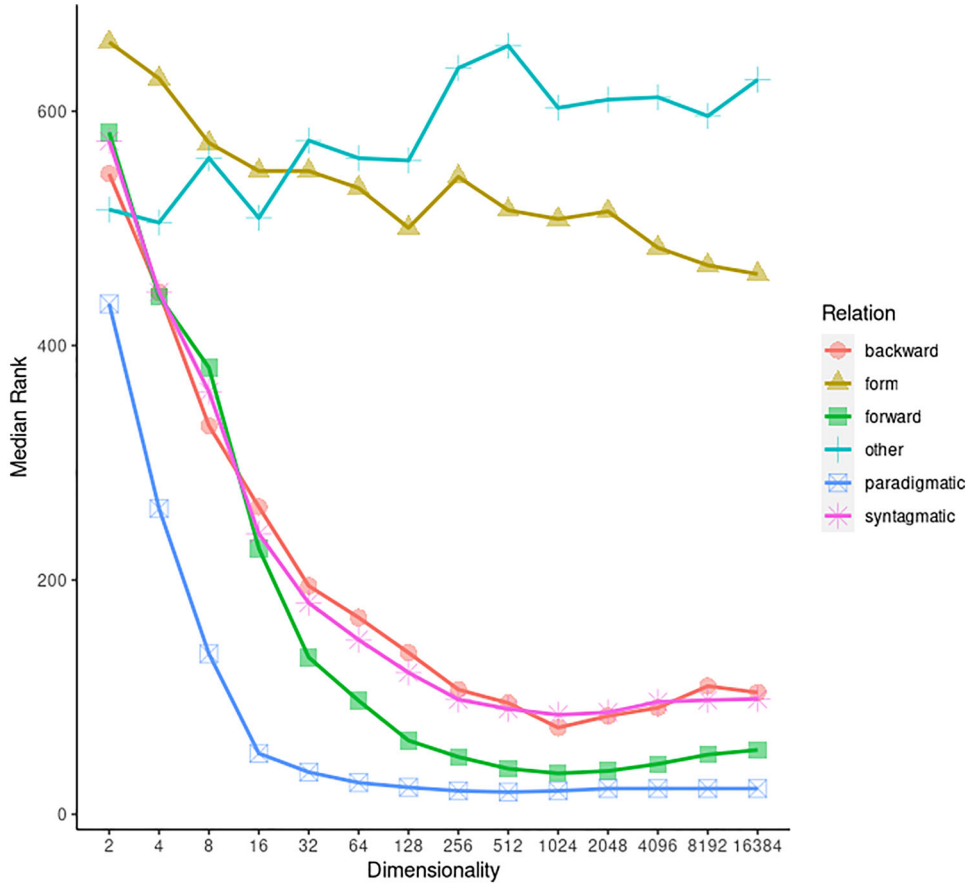

Fig. 12. Median rank of different relation types capturing different relational affinities of the compressed SPPMI matrix at varying dimensionalities.

We made a distinction between latent representations and latent relations, and suggested that the former is not necessary for the acquisition of the latter. Instead of relying on a latent representation to obtain latent relations, we provided a spreading activation account of how such relations can form through a retrieval process. We implemented a spreading activation account of generalization using a new variant of associative nets we call the DEN, which address generalization problems that plagued earlier variants of associative nets when dealing with correlated memory patterns. Whereas a linear associative net has a static equilibrium state, the equilibrium state in the DEN changes for each cue to facilitate the integration of co-occurrence statistics encoded into the weight matrix with each input pattern.

After demonstrating the DEN's ability to construct meaning representations online through a retrieval process, using a toy example, we scaled up the system by training it on the TASA corpus and compared it with other commonly used models of word meaning. In the first simulation, we matched model predictions against the USF free association norms and showed

that the DEN was as good at capturing the pattern of human free word association as other latent representation models. With the exception of LSA, which provided a poorer match than the other models, the topic model, word2vec, the DEN, and SPPMI performed comparably.

Next, we turned to a set of word evaluation norms where human raters have decided the similarity and relatedness of word pairs. We correlated model-derived strengths with human ratings across 11 different datasets and found that the DEN provided similar correlations with the ratings as other models. In some cases, performance was superior for the DEN relative to extant models like the topic model, whereas in other cases, it performed at a slightly lower level (i.e., relative to LSA and word2vec). Compared to SPPMI, the DEN yielded larger correlations overall.

We carried out a more direct comparison between DEN, SPPMI, and word2vec through a grid search over a set of hyperparameters. Three of the hyperparameters including the size of the context window, smoothing parameter, and the number of negative samples were interchangeable between the DEN and word2vec, and one of the hyperparameters, the weight of the first eigenvector inhibition, was specific to the DEN. Overall, the DEN's match to the free association norms was the most robust compared to word2vec and SPPMI. A defining characteristic of word2vec was its reliance on a large number of negative samples; however, neither the DEN nor SPPMI depended on the negative sample size. The smoothing parameter had a very minor effect on the models' match to the free association norms, but their tuning provided slight improvements. Finally, varying the size of the context window had the strongest impact on word2vec and SPPMI. Critically, performance was near asymptote for the DEN, even with the smallest context window—just three words. A final comparison was carried out between models by evaluating each model's performance at different stages during training. The results paralleled those obtained by changing the size of the context window, with the DEN requiring the least amount of training data before approaching asymptotic performance relative to word2vec and SPPMI.

In a final set of simulations, we constructed semantic spaces with embeddings of various dimensionality derived by applying SVD to an SPPMI matrix and found that compressing down to 4096 dimensions yields semantic vectors that predict free associations on comparable levels to word2vec and DEN. Overall, whereas match to word similarity judgment norms peaked with embeddings of size 512, match to the free association norms peaked with around 1024 dimensions. We further explored the extent to which different dimensionalities facilitate affinity for qualitatively distinct types of relation, and used the word relation norms first presented in Fig. 7 to profile affinity of the resulting semantic spaces for different relations. The general pattern seemed consistent with our assumption that contextual blending through dimensionality reduction should facilitate affinity for paradigmatic relations and reduce affinity for syntagmatic, forward-serial, and backward-serial relations, since affinity for the former peaked at a smaller dimensionality as shown in Fig. 12.

Our results show that assuming a latent representation is not necessary for capturing the meaning of words. The appropriate process assumption can suffice to reveal relations between words that are not explicitly observed through the surface-level regularities in the system's record of past experience. In the DEN, information stored in memory directly corresponds to the co-occurrence statistics of words across contexts, but the spreading of activation enables

the system to induce more generic relations between words through a relaxation process. We echo our claim about parsimony with respect to a solution to the long running question: how do general ideas come to form from experience? General ideas are abstractions over multiple experiences, or specific episodic details, and fall within the scope of semantic memory. Latent representation accounts assume that the stored information is itself generic, and, therefore, depend on the assumption of a complementary representation to explain how memories of individual experiences are stored and retrieved. Since associative nets assume the formation of general ideas as a retrieval process and contain direct episodic details in the stored weights, they make fewer assumptions regarding the types of memory systems necessary for capturing both episodic and semantic information.

The DEN, being linear, facilitates the characterization of the system's dynamics based on the eigenspectrum of the weight matrix. The activation of a particular word aligns the state vector with a subset of eigenvectors in the weight matrix and gradually shifts the state vector in their direction. Since the top eigenvectors of the weight matrix have the strongest attractive force, the state is generally pulled toward those dimensions of high variance. The high-level dynamics of spreading activation correspond to the gradual integration of the input state into the global associative structure of the entire memory system. In a similar way that LSA induces latent representations by approximating the original word-by-document matrix based on the singular vectors that capture the most variance, spreading activation drives the system's state toward the dominant eigenvectors of the weight matrix. The eigenspace characterization of spreading activation provides a principled link between local and global structure encoded into the weight matrix.

The DEN presented in the current work is not meant as a complete cognitive model, but provides one possible module in a broader system. For instance, the DEN algorithm may be used within the Construction-Integration (CI; Kintsch, 1988; Kintsch, 1974; Kintsch, 1998; Van Dijk & Kintsch, 1983) framework as a way to integrate input with prior knowledge. The CI framework provides a high-level blueprint for characterizing the representations and processes involved in language comprehension. The construction phase requires parsing an utterance into a semantic representation, and then using the semantic representation to cue other relevant knowledge in memory. The retrieved knowledge can take the form of both elaborative or inferential processes. A final integration phase is introduced for settling to a consistent set of active representations—the meaning. The CI framework rests upon a fundamental assumption about the dynamic nature of comprehension—meaning is not static but emerges from the momentary interplay of context, knowledge, and one's immediate object of awareness. Whereas latent representation models of meaning provide static vectors as the basis for representing meaning, the DEN provides a more consistent account of meaning in the context of the CI framework.

The simulations presented in the current manuscript adopt bag-of-word representations that discard order information, such that the two sentences "John loves Mary" and "Mary loves John" are not distinguished from one another. In contrast, sequence representation models like the Simple Recurrent Net (SRN; Elman, 1990) and Long-Short Term Memory (LSTM; Hochreiter & Schmidhuber, 1997) incorporate sequential information by accumulating the hidden representations of words earlier in the sequence as the source of information for

predicting each subsequent word in the sequence. Given the sentence "John loves Mary," a recurrent net will encode the first word "John" in a similar fashion as word2vec, by projecting the corresponding one-hot vector through a low-dimensional hidden layer and use the hidden representation to predict the next word "loves" at output. The hidden representations of both "John" and "loves" are then combined into a single hidden context vector to predict the next word, "Mary." Predicting each subsequent word is based on the accumulated hidden representations of each previous word into a single context representation. Information from earlier words is lost as subsequent words are accumulated into the context representation, motivating Bahdanau, Cho, and Bengio (2014) to preserve the hidden representation of each previous word, and separately constructing the accumulated context representation for each subsequent word's prediction. That is, the accumulated context was based on a different weighted combination of the hidden representations for each word, with the weighting function trained using backpropagation. Bahdanau et al. (2014) still maintained the overall recurrent architecture, but Vaswani et al. (2017) later showed that stacks of attention layers can fully replace recurrence.

The general logic of late accumulation of information over words at the time of each prediction, instead of early accumulation like in recurrent nets, is consistent with the generalization-at-retrieval property of the DEN presented in the current manuscript. In the same vein, work in neural machine retrieval has demonstrated superior relevance scores when retrieving from a large collection of documents, if the latent embeddings for the tokens in each document are pooled into a single embedding during retrieval, where the interaction between the token embeddings in the document is further conditioned on the latent representations corresponding to the search query relative to pooling the token embeddings prior to retrieval (Khattab et al., 2020).

Another similarity between the DEN and the Transformer lies in the nature of the residual connections as the activations are projected through each of the self-attention modules upstream. As the activations flow upstream, the projection through each subsequent self-attention module approaches a fixed-point in the same way the state-vector in the DEN reaches steady-state during recurrence (Bai, Kolter, & Koltun, 2019). Some work with earlier Transformer models has shown that fixing the weight matrices across all layers preserves the same level of performance as when each of the transformations is allowed to be distinct (e.g., Dehghani et al., 2018), leading to more recent developments that reduce the computational load of inference by incorporating early-exiting strategies (Schuster et al., 2021; also see Raposo et al., 2024).

Our generalization-at-retrieval account is similar to instance-based accounts like MINERVA2 (Hintzman, 1986, Kwantes, 2005); however, the models make different assumptions about the nature of encoding. That is, whereas MINERVA2 assumes that episodic memories are separated across time to approximate an explicit mnemonic timeline of the system, the DEN assumes that the encoding process collapses across the timeline by blending associative bundles together. Future work should systematically explore the consequences of timeline-collapse-at-encoding, and the extent to which the system's timeline is implicitly represented in the network's structure.

5. Conclusion

The DEN provides an alternative for capturing word meaning, relative to latent-representation accounts, and its context-sensitivity makes it promising for semantic composition. Our retrieval account shares the assumption of latent-representation accounts that exploiting latent structure in directly observable co-occurrence is critical to the kind of generalization necessary for learning word meaning, but provides an alternative to the assumption that the latent structures must crystallize into corresponding engrams as latent representations. Instead, we cast the locus of word meaning from representation to processing and address the paucity of explorations into process accounts of word meaning.

Author contributions

Kevin D. Shabahang carried out the modeling. Hyungwook Yim was the secondary supervisor who provided feedback on various drafts and helped guide some of the decisions for data collection and modeling. Simon J. Dennis was the primary investigator and provided theoretical guidance.

Acknowledgments

We would like to thank Vladimir Sloutsky at The Ohio State University, and Olivera Savic at Basque Center on Cognition, Brain & Language for collecting the semantic relations norms used to characterize patterns of model responses in the current work.

Open access publishing facilitated by The University of Melbourne, as part of the Wiley - The University of Melbourne agreement via the Council of Australian University Librarians.

Funding information

This research was funded by the Australian Research Council's Discovery Projects funding scheme to SJD (DP150100272), and the National Research Foundation of Korea funded by the Ministry of Science and ICT to HY (No. RS-2022-00166828).

Conflict of interest statement

The authors have no relevant financial or nonfinancial interests to disclose.

Data availability statement

The data collected for this manuscript will be made available soon.

Open Research Badges

This article has earned Open Materials badge. Materials is available at https://osf.io/pq2j7/

Code availability statement

The code for the model, analyses, and figures will be made available soon.

Notes

1 The situation is flipped in the Skip-gram variant, where context is predicted from items; however, both cases involve compression through a hidden layer and substitution of one variant for the other does not change the arguments made in the current paper. We will be referring to the CBoW variant whenever we refer to word2vec in the following simulations.
2 For simplicity, we are ignoring other regimes like predicting masked words in the center of a sequence.
3 A third step is often included using reinforcement learning techniques to directly use human feedback to adjust the weights to better match their preferences.
4 For simplicity, we are ignoring encoder-decoder architectures and describing decoder-only Transformers. In addition, more than a single padding is often used, but assuming a single placeholder suffices for describing the general mechanism.
5 Here, the term word-by-context is used as the superset of both word-by-document and word-by-word matrices, since in the latter case, a word's contextual history is encoded into its associative strengths with every other word.
6 Steady-states that did not correspond to previously encoded patterns were treated as anomalies.
7 In practice, we can orthogonally project the state vector, at each recurrence iteration, onto the dominant eigenvector to maintain the sparsity in the weight matrix.
8 *Note*. The toy corpus does not capture more complex network structures, such as variable neighborhood densities, that may be obtained from a realistic corpus.
9 The norms were collected by Vladimir Sloutsky and Olivera Ilic at Ohio State University.
10 We also explored performance across the range of dimensionalities using an SPPMI matrix corresponding to the best-performing word2vec model; however, the parameters from the Dynamic-Eigen-Net resulted in better overall performance.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum,

J., Bello, I., … Zoph, B. (2023). GPT-4 Technical Report (Version 6). arXiv. https://doi.org/10.48550/ARXIV.2303.08774

Amari, S. I. (1977). Neural theory of association and concept-formation. *Biological Cybernetics*, *26*(3), 175–185.

Anderson, J. A., Silverstein, J. W., Ritz, S. A., & Jones, R. S. (1977). Distinctive features, categorical perception, and probability learning: Some applications of a neural model. *Psychological Review*, *84*(5), 413–451.

Anderson, J. A. (1995). *An introduction to neural networks*. MIT Press.

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., & Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 19–27).

Bai, S., Kolter, J. Z., & Koltun, V. (2019). Deep equilibrium models. *Advances in Neural Information Processing Systems*, *32*.

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

Borovsky, A., Elman, J., & Kutas, M. (2012). Once is enough: N400 indexes semantic integration of novel word meanings from a single exposure in context. *Language Learning and Development*, *8*(3), 278–302.

Brown, G. D., Preece, T., & Hulme, C. (2000). Oscillator-based memory for serial order. *Psychological Review*, *107*(1), 127–181.

Bruni, E., Tran, N.-K., & Baroni, M. (2013). Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, *48*, 1–47.

Bullinaria, J. A., & Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, *39*, 510–526. https://doi.org/10.3758/BF03193020

De Deyne, S., Navarro, D., Perfors, A., Brysbaert, M., & Storms, G. (2019). The Small World of Words: English word association norms for over 12,000 cue words.

de Saussure, F. (1916). *Course in General Linguistics*. Duckworth.

Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., & Kaiser, Ł. (2018). Universal transformers. arXiv preprint arXiv:1807.03819.

Dennis, S. (2005). A Memory-Based Theory of Verbal Cognition. *Cognitive Science*, *29*(2), 145–193. Portico. https://doi.org/10.1207/s15516709cog0000_9

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th International Conference on World Wide Web* (pp. 406–414).

French, R. M. (1991). Using semi-distributed representations to overcome catastrophic forgetting in connectionist networks. In *Proceedings of the 13th Annual Cognitive Science Society Conference* (Vol. 1, pp. 173–178).

Gerz, D., Vulić, I., Hill, F., Reichart, R., & Korhonen, A. (2016). Simverb-3500: A large-scale evaluation set of verb similarity. ArXiv preprint.

Griffiths, T. L., Steyvers, M., & Tenenbaum, J. B. (2007). Topics in semantic representation. *Psychological Review*, *114*(2), 211.

Halawi, G., Dror, G., Gabrilovich, E., & Koren, Y. (2012). Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1406–1414).

Hebb, D. (1949). *The organization of behavior*. New York.

Hill, F., Reichart, R., & Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, *41*(4), 665–695.

Hintzman, D. L. (1986). "Schema abstraction" in a multiple-trace memory model. *Psychological Review*, *93*(4), 411.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, *79*, 2554–2558.

Khattab, O., & Zaharia, M. (2020). ColBERT. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 39–48. https://doi.org/10.1145/3397271.3401075

Kintsch, W. (1974). The representation of meaning in memory.

Kintsch, W. (1988). The role of knowledge in discourse comprehension: A construction-integration model. *Psychological Review*, *95*(2), 163.

Kintsch, W. (1998). *Comprehension: A paradigm for cognition*. Cambridge University Press.

Kotha, S., Springer, J. M., & Raghunathan, A. (2023). Understanding Catastrophic Forgetting in Language Models via Implicit Inference (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2309.10105

Kwantes, P. J. (2005). Using context to build semantics. *Psychonomic Bulletin & Review*, *12*(4), 703–710.

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.

Lazaridou, A., Marelli, M., & Baroni, M. (2017). Multimodal Word Meaning Induction From Minimal Exposure to Natural Text. *Cognitive Science*, *41*(S4), 677–705. Portico. https://doi.org/10.1111/cogs.12481

Levy, O., & Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (Volume 2: Short Papers) (pp. 302–308).

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, *3*, 211–225.

Lohnas, L. J., Polyn, S. M., & Kahana, M. J. (2015). Expanding the scope of memory search: Modeling intralist and interlist effects in free recall. *Psychological Review*, *122*(2), 337–363. https://doi.org/10.1037/a0039036

Luong, M. T., Socher, R., & Manning, C. D. (2013). Better word representations with recursive neural networks for morphology. In *Proceedings of the 17th Conference on Computational Natural Language Learning* (pp. 104–113).

Mannering, W., & Jones, M. N. (2020). Catastrophic Interference in Predictive Neural Network Models of Distributional Semantics.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In (Eds.), *Psychology of Learning and Motivation* (pp. 109–165). Psychology of Learning and Motivation

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Miller, G. A., & Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, *6*, 1–28.

Mu, J., Bhat, S., & Viswanath, P. (2017). All-but-the-top: Simple and effective postprocessing for word representations. arXiv preprint arXiv:1702.01417.

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.

Radinsky, K., Agichtein, E., Gabrilovich, E., & Markovitch, S. (2011). A word at a time: Computing word relatedness using temporal semantic analysis. In Proceedings of the 20th International Conference on World Wide Web (pp. 337–346).

Raposo, D., Ritter, S., Richards, B., Lillicrap, T., Humphreys, P. C., & Santoro, A. (2024). Mixture-of-Depths: Dynamically allocating compute in transformer-based language models. arXiv preprint arXiv:2404.02258.

Rubenstein, H., & Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, *8*, 627–633.

Schuster, T., Fisch, A., Jaakkola, T., & Barzilay, R. (2021). Consistent Accelerated Inference via Confident Adaptive Transformers. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.18653/v1/2021.emnlp-main.406

Shabahang, K. D., Yim, H., & Dennis, S. J. (2022). Generalization at retrieval using associative networks with transient weight changes. *Computational Brain & Behavior*, *5*(1), 124–155.

Van Dijk, T. A., & Kintsch, W. (1983). Strategies of discourse comprehension.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need (Version 7). arXiv. https://doi.org/10.48550/ARXIV.1706.03762

Yang, D., & Powers, D. M. (2006). Verb similarity on the taxonomy of WordNet. In *The Third International WordNet Conference: GWC 2006*. Masaryk University.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Appendix