*Natural Language Processing* (2025), **31**, pp. 1423–1449
doi:10.1017/nlp.2025.10005

ARTICLE

# Semantic enrichment of neural word embeddings: Leveraging taxonomic similarity for enhanced distributional semantics

Dongqiang Yang[1], Xinru Zhang[1], Tonghui Han[1,2] and Yi Liu[1]

[1]School of Computer Science and Technology, Shandong Jianzhu University, Jinan, China and [2]School of Computing, Engineering and the Built Environment, Ulster University, Belfast, UK
**Corresponding author:** Dongqiang Yang; Email: dongqiang.yang@gmail.com

(Received 5 June 2024; revised 30 March 2025; accepted 10 June 2025; first published online 30 July 2025)

**Abstract**
Data-driven neural word embeddings (NWEs), grounded in distributional semantics, can capture various ranges of linguistic regularities, which can be further enriched by incorporating structured knowledge resources. This work proposes a novel post-processing approach for injecting semantic relationships into the vector space of both static and contextualized NWEs. Current solutions to retrofitting (RF) word embeddings often oversimplify the integration of semantic knowledge, neglecting the nuanced differences between relationships, which may result in suboptimal performance. Instead of applying multi-thresholding to distance boundaries in metric learning, we compute taxonomic similarity to dynamically adjust these boundaries during the semantic specialization of word embeddings. Benchmark evaluations on both static and contextualized word embeddings demonstrate that our dynamic-fitting (DF) approach produces SOTA correlation results of 0.78 and 0.76 on SimLex-999 and SimVerb-3500, respectively, highlighting the effectiveness of incorporating multiple semantic relationships in refining vector semantics. Our approach also outperforms existing RF methods in both supervised and unsupervised semantic relationships recognition tasks. It achieves top accuracy scores for hypernymy detection on the BLESS, WBLESS, and BIBLESS datasets (0.97, 0.89, and 0.83, respectively) and an F1 score of over 0.60 on four types of semantic relationship classification in the shared Subtask-2 of CogALex-V, surpassing all participant systems. In the analogy reasoning task of the Bigger Analogy Test Set, our approach outperforms existing RF methods on inferring relational similarity. These consistent improvements across various lexical semantics tasks suggest that our DF approach can effectively integrate distributional semantics with symbolic knowledge resources, thereby enhancing the representation capacity of word embeddings in downstream applications.

**Keywords:** distributional semantics; semantic similarity; neural word embeddings; metric learning

## 1. Introduction

Distributional semantic models (DSMs) vectorize lexical terms mainly through counting or predicting co-occurrence patterns in context, among which neural word embeddings (NWEs), generated by self-supervised-training neural language models (NLMs) or large-scale language models (LLMs), are capable of capturing various levels of linguistic regularities in their geometry space (Mikolov, Yih, and Zweig, 2013c; Tenney, Das, and Pavlick, 2019). NWEs can be static or dynamic in terms of regulating word meanings in context. Static NWEs such as GloVe (Pennington, Socher, and Manning, 2014) and Skip-gram (Mikolov *et al.* 2013b) generate unified and context-independent word representations, effectively avoiding the issues of high-dimensional sparse representations and poor scalability associated with counting co-occurrences in bag-of-words models. They are more suitable for calculating semantic relatedness (Hill, Reichart, and Korhonen, 2015) and tend to construct similar representations for words with similar contextual distributions, which may dampen their semantic expressiveness (Peters *et al*., 2018) in semantic relation discrimination and inference. Dynamic NWEs, on the other hand, utilize large-scale NLMs like BERT (Devlin *et al.* 2018) and GPT-2 (Radford *et al.* 2018) to produce contextualized word representations. Their multilayered transformer framework can develop distinctive representations respectively suited for morphological, syntactic, semantic, and application-specific purposes, forming the foundation of current breakthroughs in natural language processing (NLP). They can also be converted into static NWEs (Ethayarajh, 2019, Bommasani, Davis, and Cardie, Bommasani *et al.* Bommasani *et al.* 2020, Gupta and Jaggi, 2021). In contrast with dynamic NWEs, static NWEs have demonstrated advantages in generalization, computing efficiency, and strong interpretability (Gupta and Jaggi, 2021). Apart from scaling up the parameters of NLMs or improving their neural architectures to learn more implicit knowledge from unstructured corpora, NWEs can be further optimized by integrating explicit knowledge resources, either directly regulating the ad hoc training loss functions of NLMs (Yu and Dredze, 2014; Xu *et al.* 2014; Liu *et al.* 2015) or semantically specializing NWEs in a post hoc manner (Faruqui *et al.* 2015; Mrkšić *et al.* 2016; Vulić and Mrkšić 2018; Arora, Chakraborty, and Cheung, 2020). In contrast to utilizing cross-entropy loss to maximize the conditional probability of token prediction within a common self-supervised pretraining paradigm, post-processing methods typically employ ranking loss such as contrastive loss (Chopra, Hadsell, and Lecun, 2005) and triplet loss (Schroff, Kalenichenko, and Philbin, 2015) to optimize the distributional distance between a token and its relata, a process also known as distance metric learning (Bellet, Habrard, and Sebban, 2015). The objective of metric learning is to pull semantically similar or related tokens closer in a vector space and push dissimilar or unrelated tokens farther apart. Since joint-training NWEs may consume substantial computational resources to simultaneously learn distributional features and encode relational knowledge, the post-processing methods can significantly reduce such computational demands while demonstrating versatility and flexibility in incorporating multiple categories of semantic relationships (Faruqui *et al.* 2015).

### 1.1. Metric learning

Metric learning, a form of contrastive representation learning, is also applicable for generating NWEs, where positive and negative samples for a target are processed contrastively during self-supervised training. For example, to improve training efficiency and avoid updating all parameters of Skip-gram, Mikolov *et al.* (2013b) introduced negative sampling, replacing the softmax function with a contrastive loss that maximizes the probability of a target and its positive context words, while minimizing the probability between the target and its randomly selected negative samples. Additionally, Nguyen *et al.* (2017) employed triplet loss for pretraining hierarchical word embeddings. Throughout this study, we use the terms metric learning and contrastive learning interchangeably, as both share the core objective of attracting related terms and repelling unrelated ones in a geometric space.

However, typical solutions to retrofitting (RF) NWEs are often ineffective at integrating relational knowledge resources. They either indiscriminately inject different types of semantic relationships (Faruqui *et al.* 2015), establish unified distance boundaries for hierarchical relationships (Vulić and Mrkšić 2018), or manually allocate distance boundaries for distinctive relationships (Vulić and Mrkšić 2018; Arora *et al.* 2020; Yang *et al.* 2022; Pan *et al.* 2024), which may result in interference of different semantic specializations on vector space models, thereby limiting their effectiveness on downstream applications (Arora *et al.* 2020). Distance boundaries in metric learning should be adjustable to accommodate various knowledge injections, inherently capturing association intensity within semantic relationships. To enforce pulling and

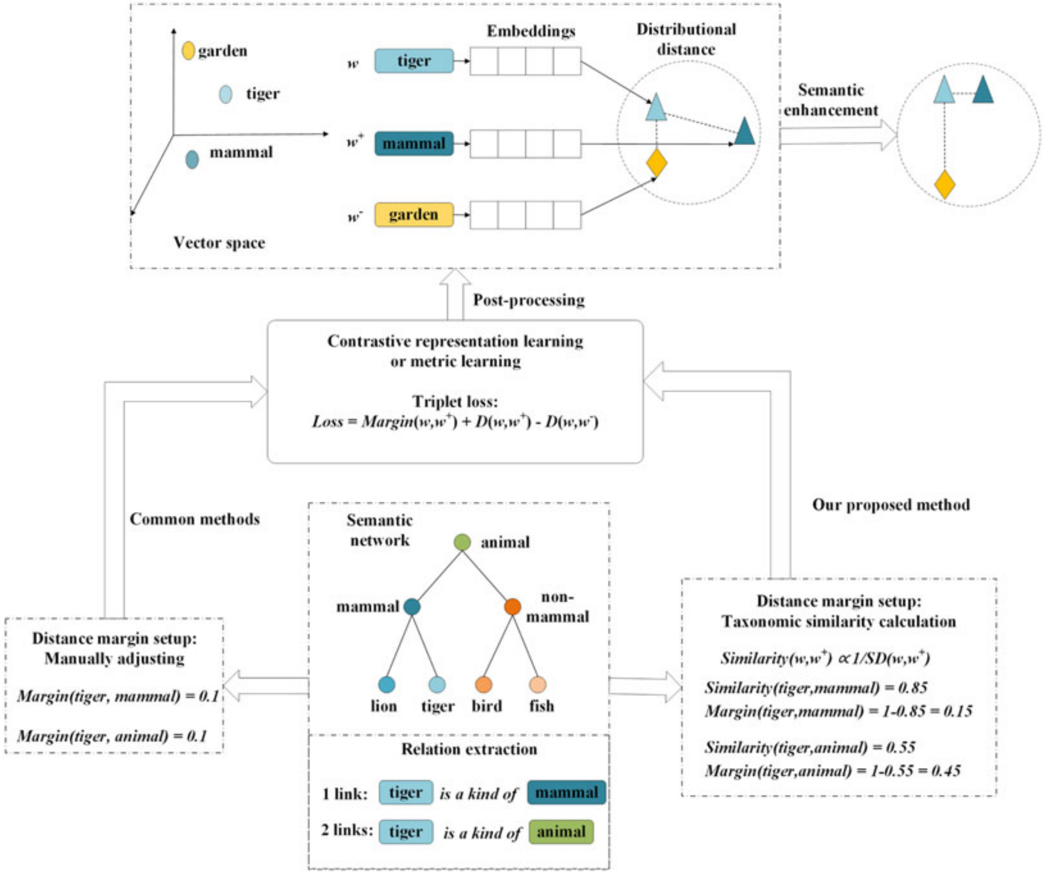
garden
tiger
mammal
Vector space
Embeddings
Distributional distance
w
w+
w-
tiger
mammal
garden
Semantic enhancement
Post-processing
Contrastive representation learning or metric learning
Triplet loss:
Loss = Margin(w,w+) + D(w,w+) - D(w,w-)
Common methods
Our proposed method
Semantic network
animal
mammal
non-mammal
lion
tiger
bird
fish
Distance margin setup: Manually adjusting
Margin(tiger, mammal) = 0.1
Margin(tiger, animal) = 0.1
Distance margin setup: Taxonomic similarity calculation
Similarity(w,w+) ∝ 1/SD(w,w+)
Similarity(tiger,mammal) = 0.85
Margin(tiger,mammal) = 1-0.85 = 0.15
Similarity(tiger,animal) = 0.55
Margin(tiger,animal) = 1-0.55 = 0.45
Relation extraction
1 link: tiger is a kind of mammal
2 links: tiger is a kind of animal

**Figure 1.** Illustration of our proposed method on semantically retrofitting word embeddings. In a vector space model, $w$, $w^+$, and $w^-$ stand for a target word, its positive or related sample, and its negative or unrelated sample, respectively. In distance metric learning, *Margin*$(w, w^+)$ refers to the distributional distance ($D$) boundary or margin between $D(w, w^+)$ and $D(w, w^-)$ in the triplet loss. We propose adjusting this margin adaptively using taxonomic similarity calculation, where *SD* indicates the shortest distance between $w$ and $w^+$ within a semantic network.

pushing operations in metric learning, RF NWEs necessitates setting multiple distance boundaries or margins to account for various semantic constraints. These constraints help guide the positioning of words in vector space, refining word embeddings based on their semantic similarity or relationships, such as synonymy, antonymy, and hypernymy.

While incorporating relational knowledge into word embeddings, instead of relying on fixed distance margins to impose corresponding semantic constraints in metric learning, we propose fully exploring the semantic nuances of the constraints to dynamically adjust their distance margins. As illustrated in Figure 1, metric learning first retrieves both positive and negative pairs to construct the triplet loss. Positive pairs represent semantically related tokens in knowledge bases, while negative ones lack significant connections. To impose semantic constraints on the vector space of NWEs, the triplet loss function seeks to minimize the distance of a positive pair while maximizing the distance of its corresponding negative pair. During the optimization of NWEs, distance margin often serves as a minimum threshold by which the negative pair must be separated from the positive one.

For example, with a target token like *tiger*, we employ semantic networks in WordNet (Miller *et al.* 1990; Fellbaum, 1998) to retrofit NWEs. We first extract its direct hypernym *"mammal"* as a positive sample and calculate their semantic similarity score as 0.85, based on counting their shortest link distance (one link) in the hierarchy. The indirect hypernym *"animal"* can also serve as a positive sample, but with a lower similarity score of 0.55, as they are two links apart in the hierarchy. In the mini-batch updating process of NWEs, we typically select a token at random or a token with the minimum distributional similarity to *tiger*, say, *garden*, as its negative sample. Assuming that such negative pairs have little semantic association, the similarity scores on these positive pairs are subsequently converted into distance margins of 0.15 and 0.45, respectively, matching their semantic nuances for metric learning. The training objective is to attract the positive pair, *tiger* and *mammal*, closer while repelling the negative pair, *tiger* and *garden*, apart, ensuring that their relative distance falls within the specified margin of 0.15, thus facilitating proper semantic specialization on word embeddings.

In contrast, current RF approaches address different semantic constraints in metric learning either by individually adjusting distance margins or by simply assigning them a uniform value. For example, while the positive pair, *tiger* and *mammal*, is semantically closer than *tiger* and *animal* in the IS-A hierarchy, existing research often treats them equivalently with the identical distance margin (0.1), as shown in Figure 1. The fixed margin inevitably restricts both direct and indirect hypernymy constraints within the same distance range in a vector space, misinterpreting their semantic disparity through inappropriate actions of attracting and repelling in metric learning. We first calculate taxonomic similarity in WordNet, which is then adapted as a proxy for distance margins or boundaries in metric learning. Geometric relationships of words in a vector space model can then be optimized to mirror their semantic association from knowledge bases.

### 1.2. Distributional semantics in LLMs

We aim to leverage handcrafted relational knowledge to enhance distributional semantics distilled from prediction-based static embeddings, along with two contextualized embeddings: autoencoding BERT and autoregressive GPT. Different from counting-based DSMs, neural embeddings as the intermediate states of NLMs are essentially sub-symbolic, storing linguistic patterns implicitly in a continuous semantic space, but they still fundamentally ground on the distributional hypothesis (Harris, 1954; Firth, 1957) to capture and encode relationships between words, phrases, and contexts. Thanks to the significant scaling up on pretraining data size and transformer architecture, NLMs equipped with terabytes of tokens and billions of parameters have evolved into foundation models (Bommasani *et al.* 2021) for downstream applications across diverse domains, demonstrating remarkable zero-shot or few-shot performances as meta-learners. Nonetheless, data-driven and parameter-packed LLMs also face extraordinary challenges such as biases, limited knowledge, and hallucinations, while balancing memorization and generalization to overcome overfitting (Naveed *et al.* 2023). Incorporating external factual or world knowledge resources, for example, retrieval-augmented generation (Lewis *et al.* 2020), can strengthen the consistency and accuracy of distributional semantics in LLMs, rendering their "black-box" generation process more trustworthy and explainable. Further research into distributional semantics, for example, probing their internal representations or token embeddings (Kadavath *et al.* 2022; Li *et al.* 2024; Chen *et al.* 2024a), can refine how word meanings and contextual nuances are structured in vector space, which can deepen insights into the inner workings of LLMs, enhancing interpretability and control over these models.

Off-the-shelf LLMs have emerged as potential proxies for knowledge bases (Davison, Feldman, and Rush, 2019; Petroni *et al.* 2019; Roberts, Raffel, and Shazeer, 2020; Talmor *et al.* 2020), exhibiting several advantages over traditional symbolic systems, particularly in scalability on data consumption, unsupervised training, and minimal reliance on scheme engineering. The intermediate states or latent space of token embeddings within LLMs can facilitate the retrieval of relational knowledge, for example, learning hyperspherical relational embeddings (Wang, He, and

Zhou, 2019), masked token prompting (Petroni *et al.* 2019), and fine-tuning to optimize contextualized embeddings (Ushio, Camacho-Collados, and Schockaert, 2021). Additionally, LLMs can directly generate world knowledge, thereby enhancing their utility in knowledge-intensive applications (Chen *et al.* 2023). However, the effectiveness of these knowledge mining strategies is contingent upon the accuracy of factual and commonsense relationships extracted from LLMs. Current studies suggest that the internal embeddings may contain truthfulness information (Kadavath *et al.* 2022; Li *et al.* 2024; Chen *et al.* 2024a), which may be harnessed to predict error types associated with hallucinations or improve the overall truthfulness of LLMs' outputs (Chuang *et al.* 2024). Refining these internal embeddings may enhance LLMs' comprehension on nuanced semantic relationships.

LLMs, as foundational models, have extrapolated their near-human-level performance on certain artificial general intelligence tasks (Bubeck *et al.* 2023) to serve as AI evaluators. For example, LLMs can consistently align with human preferences when scoring natural language generation tasks (Wang *et al.* 2023; Fu *et al.* 2024), validating chatbots (Zheng *et al.* 2024), and evaluating information retravel (Thomas *et al.* 2024), leveraging strengths in scalability and explainability. However, data-driven LLMs inevitably introduce new biases in the outputs generated from evaluation prompts (Wu and Aji, 2023; Zheng *et al.* 2024; Koo *et al.* 2024; Chen *et al.* 2024b; Stureborg, Alikaniotis, and Suhara, 2024, Wang *et al.* 2024), with remedies such as chain-of-thought and fine-tuning yielding only limited improvements.

### *1.3. Our contributions*

With the aid of computing taxonomic similarity in WordNet, we describe a novel solution to injecting semantic knowledge into vector semantics, which is a clear improvement on the current RF strategies on semantically specializing NWEs (Mrkšić *et al.* 2016; Vulić and Mrkšić 2018; Faruqui *et al.* 2015; Arora *et al.* 2020). Our method can dynamically adjust the scope of margins while imposing corresponding semantic specialization on NWEs. It can facilitate amalgamating various semantic relationships into distributional semantics, with the effects of mitigating the interference of inconsistently enforcing different semantic specializations on NWEs. Arora *et al.* (2020) have highlighted such interference in LexSub, for which they proposed to learn a separate subspace for each type of semantic relationship. Note that they still followed the traditional RF methods (Mrkšić *et al.* 2016; Vulić and Mrkšić 2018; Faruqui *et al.* 2015) in manually tuning distance boundaries to learn those subspaces, with an additional cost of training a dedicated projection matrix for each semantic constraint. The benchmark tests show that our hybrid method of combining knowledge resources and distributional semantics can significantly improve lexical semantics applications, including semantic similarity calculation, lexical entailment (LE) detection, and word analogy reasoning.

Given the objective of probing the internal states of LLMs for relational knowledge, we address evaluation mainly through intrinsic and extrinsic tasks rather than relying on LLM-based judges. Lenci *et al.* (2022) observed that intrinsic evaluations correlate well with the performance of DSMs on extrinsic tasks. Although intrinsic evaluations may exhibit bias or subjectivity (Bakarov, 2018; Naveed *et al.* 2023), they often work as reliable metrics for assessing the internal representations of LLMs, providing valuable insights into the core properties of embeddings beyond task-specific outcomes. To thoroughly evaluate the distributional quality of NWEs, we employ three intrinsic benchmarks (Baroni, Dinu, and Kruszewski, 2014) on lexical and relational similarity computation, as well as a supervised extrinsic task (Santus *et al.* 2016) on semantic relationship classification.

## 2. Related work

Prediction-based neural embeddings are more capable of deriving semantic relatedness rather than semantic similarity (Hill *et al.* 2015; Lê and Fokkens 2015), in which antonyms are prone to learning proximate vector representation. Hill *et al.* (2015) also demonstrate that modeling semantic similarity from NWEs is inherently more challenging than modeling semantic relatedness or association under the assumption of similar words sharing similar contexts.

Given that lexical knowledge bases (LKBs) such as WordNet are superior in yielding semantic similarity rather than semantic relatedness (Lê and Fokkens 2015; Yang and Yin, 2022), various studies have attempted to leverage NWEs with handcrafted semantic relationships, aiming to enrich the distributional representation of word meanings with both first-order co-occurrences of syntagmatic association acquired in context and second-order co-occurrences of paradigmatic parallelism extracted from LKBs. In the following, we briefly introduce three popular LKBs in computing semantic similarity and then present major methodologies for improving vector semantics in similarity judgment.

### 2.1. Deriving semantic similarity from LKBs

To enrich distributional semantics derived from sub-symbolic word embeddings with human-curated symbolic knowledge, we first outline how to calculate taxonomic similarity in the semantic networks of WordNet. We then review current literature on knowledge-enhanced NWEs, highlighting advancements in integrating symbolic knowledge with word embeddings.

#### 2.1.1 WordNet

WordNet is an online lexical database for the English language, whose organization of concept relationships has a fundamental impact on multilingual lexical bases such as BabelNet (Navigli and Ponzetto, 2012) and the Open Multilingual WordNet (Bond and Foster, 2013). The typical properties of wordnets are the concept networks of the synsets that stand for the unique concepts that consist of a group of synonyms with the same parts of speech (PoS) tags. On assuming the interchangeability of the lexical units in a specific context, WordNet consists of some popular paradigmatic relations, including syn/antonym, hyper/hyponym (IS-A relations), and holo/meronym (PART-OF relations). Note that WordNet only contains paradigmatic relationships. The lack of connections between topically related words is also known as the tennis problem in WordNet, which means the tightly related concepts in tennis, racquet, ball, and net are located in different hierarchies.

#### 2.1.2 Taxonomic similarity

LKBs play a crucial role in soliciting word similarity from the provision of concept definitions and relationships. Aside from the shortest path length in semantic networks, the key difference among taxonomic similarity methods is how to estimate concept specificity, either using network structures in edge-counting or using concept frequencies in information content (Yang and Yin, 2022).

**Edge-counting** exclusively relies on semantic networks as the resource of retrieving conceptual relationships so that every possible concept contrast is exposed before the computation of word similarity. Edge-counting can calculate word similarity in different meaning combinations (Yang and Yin, 2022). Given that each link holds the same weight in searching paths in LKBs, edge-counting tends to traverse all possible paths between concepts without any consideration of feasibility costs. Although some concepts are unpopular or obsolete in literal and metaphorical usage, edge-counting can compute word similarity from every aspect of word senses.

**Information content (IC)** (Resnik, 1999) relies on concept frequencies in use to predict word similarity. It is inevitably biased toward using the predominant sense of a word because the unbalanced word-sense distribution fits well with Zipf's law (Zipf, 1965), which says that the most typical sense of a word prevails in reality. As IC can capture the predominant meanings of words in contexts, its result reflects the biases in word usage within specific domains. Assigning different weights for each link in IC results in a heuristically informed search that only predicts the economic path between concepts. Therefore, IC probably emphasizes only the literal connectedness between words in language usage. Without such additional restrictions on sense distribution, edge-counting tends to reveal both the literal and metaphorical proximity between words, and word similarity derived from edge-counting in WordNet can automatically determine the corresponding senses of words.

### 2.2. Synergy of distributional similarity and knowledge-based similarity

Concept representations in semantic memory hypothesize semantic modeling within association networks, distributed binary features, and statistically distributional semantics (Kumar, 2021). This framework highlights the complexity of computing similarity judgments, emphasizing its multifaceted nature in terms of knowledge-based or taxonomic similarity (Collins and Quillian, 1969; Collins and Loftus, 1975), contrast model on feature overlap (Tversky, 1977), and distributional similarity (Harris, 1985). Integrating these approaches within a unified model appears to be an optimal solution for capitalizing on their respective merits in similarity calculations. For example, Hassan and Mihalcea (2011) achieved significant improvement in measuring semantic relatedness through combining both Wikipedia-based distributional semantics, that is, explicit semantic analysis (Gabrilovich and Markovitch, 2007), and Wikipedia's hyperlink hierarchy. Using BabelNet, a multilingual comprehensive LKB that amalgamates various lexical resources including WordNet and Wikipedia, Camacho-Collados *et al.* (2016) constructed a joint vector semantic space that encompasses both semantic relationships and statistical co-occurrences. Instead of yielding a unified semantic representation, Banjade *et al.* (2015) trained a support vector regressor (SVR) to integrate similarity judgments from multiple resources, including taxonomic similarity on WordNet, explicit semantic analysis on Wikipedia, and distributional similarity on NWEs. Lee *et al.* (2020) also employed SVR to combine distributional features from word embeddings and multiple linguistic features extracted from WordNet to predict semantic relatedness.

### 2.3. Semantically specializing NWEs

According to the distributional hypothesis, geometric relations derived through vector distance and direction manipulation can effectively capture underlying semantic relationships between words. By leveraging semantic composition in word embeddings, we can enhance language understanding tasks such as similarity calculation, analogy reasoning, and LE recognition. The primary objective of contrastive learning for NWEs is to position words in similar contexts closer in geometric space, utilizing distance metrics in their loss functions. Euclidean metrics like dot product are well-suited for learning symmetric relationships such as synonymy and antonymy, while non-Euclidean metrics like hyperbolic distance can encode the hierarchy of asymmetric relationships such as hypernymy and meronymy (Nickel and Kiela, 2017; Nickel and Kiela, 2018). However, NWEs learned through non-Euclidean metrics demonstrate little superiority over Euclidean ones in semantic computing tasks (Ganea, Bécigneul, and Hofmann, 2018; Leimeister and Wilson, 2018). Additionally, Euclidean embeddings can also represent complex linguistic patterns, aided by high dimensionalities (Torregrossa *et al.* 2021). This study focuses on Euclidean NWEs, for example, SGNS (Mikolov *et al.* 2013b) and BERT (Devlin *et al.* 2018), generated from shallow to deep neural networks. They can be semantically enhanced through the incorporation of semantic relations from LKBs, potentially improving their representation capacities. Note that we seek to distill semantic relationships from word embeddings rather than from hyperspherical relation embeddings like SphereRE (Wang *et al.* 2019) or from relation embeddings extract through prompting pretrained LLMs (Petroni *et al.* 2019; Ushio *et al.* 2021).

### *2.3.1 Joint-training NWEs*

Apart from predicting co-occurrent words in context, most joint-training methods directly factor in semantic constraints to optimize the training objective of NLMs (Yu and Dredze, 2014; Nguyen *et al.* 2017; Alsuhaibani *et al.* 2018). For example, Yu and Dredze (2014) regularized the training objective of continuous bag-of-words (CBOW) (Mikolov *et al.* 2013a) through maximizing $\sum_{w \in R_w} \log p(w_i|w_j)$, in which $R_w$ is the set of semantic constraints, consisting of the target word $w_i$ and its synonym $w_j$ extracted from the paraphrase database (PPDB) (Ganitkevitch, Van Durme, and Callison-Burch, 2013) and WordNet. In a similar vein, Xu *et al.* (2014) enhanced the training objective of Skip-gram (Mikolov *et al.* 2013b) by incorporating both relational and categorical knowledge sourced from knowledge graphs and Freebase (Bollacker *et al.* 2008); Liu *et al.* (2015) augmented the pretraining process of NWEs by integrating both synonymy and LE from WordNet with an improvement in semantic similarity calculation and named entity recognition. Apart from incorporating semantic associations into the training objectives of NLMs, another approach to enhancing NWEs is through the utilization of more intricate neural architectures such as training graph convolutional networks with syntactic dependencies and semantic relationships (Vashishth *et al.* 2019) and employing attention mechanisms (Yang and Mitchell, 2017; Peters *et al.* 2019). Joint-training methods can customize NWEs for downstream applications, albeit at the expense of excessive computing demands.

### *2.3.2 Post-processing NWEs*

Post-processing methods frequently employ ranking loss in metric learning (Kaya and Bilge, 2019) to update NWEs, hypothesizing that associated terms in LKBs should remain closer in a vector space. Given a target $w_i$ and its semantically linked counterpart $w_j$, along with the unlinked one $w_j^{'}$ in the n-dimensional space of NWEs: $f_\theta\ (x) : x \in R^n$, metric learning aims to learn an updated $f_\theta^{'}$ with ranking loss such as contrastive loss (Chopra *et al.* 2005) and triplet loss (Schroff *et al.* 2015). The goal is to minimize the distance of $w_i$ and its positive sample $w_j : D\left(f_\theta\ (w_i), f_\theta\ (w_j)\right)$ to pull them closer, while simultaneously maximizing $D\left(f_\theta\ (w_i), f_\theta\ \left(w_j^{'}\right)\right)$ to push $w_i$ and its negative sample $w_j^{'}$ farther apart.

**Retrofitting (RF).** To refine NWEs in RF, Faruqui *et al.* (2015) collectively integrated various semantic relations including synonymy and LE from WordNet, word association from FrameNet (Baker, Fillmore, and Lowe, 1998), and lexical paraphrasing from PPDB (Ganitkevitch *et al.* 2013).

**Counter-fitting (CF).** Mrkšić *et al.* (2016) established respective distance boundaries for synonymy and antonymy in contrastive loss to specialize the geometry space of NWEs. Subsequently, Mrkšić *et al.* (2017) improved CF with triplet loss in ATTRACT-REPEL, which employed semantic constraints extracted from both mono- and cross-lingual resources, including PPDB and BabelNet.

**Lexical entailment attract-repel (LEAR).** Apart from synonymy and antonymy used in CF and ATTRACT-REPEL, Vulić and Mrkšić (2018) introduced LE in WordNet in RF NWEs. LEAR does not differentiate between direct and indirect hypernymy, establishing the same distance margin in the triplet loss function for both hypernymy and synonymy.

**Hierarchy-fitting (HF).** Inspired by LEAR, Yang *et al.* (2022) proposed HF to distinguish semantic nuances between synonymy and direct hypernymy in the quadruplet loss function.

**LexSub.** Different from RF, CF, LEAR, and HF in integrating various semantic relations to establish a unified vector space, Arora *et al.* (2020) trained a projection matrix to create a separate subspace for each semantic constraint. LexSub aims to reduce the interference of different semantic constraints in RF NWEs, in contrast with other post-processing methods such as ATTRACT-REPEL and LEAR. Note that apart from syn/antonymy and IS-A relations commonly used in existing methods, LexSub incorporates PART-OF relations from WordNet as additional semantic constraints.

In comparison with the joint-training paradigm, RF NWEs effectively integrates various types of semantic resources into vector space models. Additionally, it enables the construction of a unified word embedding model using multilingual lexicons (Mrkšić *et al.* 2017).

Regardless of its benefits, the indiscriminate application of multiple semantic constraints in post-processing may underestimate their semantic distinctions and ultimately undermine RF effects on NWEs. While some methods attempted to align semantic constraints with corresponding distance boundaries, the intricate nature of diverse concept relationships poses challenges in adjusting margins within a single vector space, potentially exacerbating interference with semantic specialization. Our contribution lies in introducing taxonomic similarity computation to dynamically adjust margins or distance boundaries in metric learning, through which we can combine both lexical similarity and distributional similarity in RF NWEs.

## 3. Dynamic fitting

The dynamic-fitting (DF) method we propose consists of two main parts: (1) calculating semantic similarity or distance using multiple relationships in semantic networks and (2) integrating semantic similarity and distributional similarity within metric learning to retrofit NWEs, as depicted in Figure 2.

### 3.1. Taxonomic similarity

We adopt WUP (Wu and Palmer, 1994), one of the edge-counting methods, to compute semantic similarity in WordNet, which has demonstrated effectiveness in diverse application tasks. WUP estimates similarity by measuring the depths of two conceptual nodes, $w_i$ and $w_j$, in the semantic network, as well as the depth of their least common ancestor node, $LCS\left(w_i, w_j\right)$, which is as follows:

$$sim_{wup}\left(w_i, w_j\right) = -\log \frac{depth\left(LCS\left(w_i, w_j\right)\right)}{depth\left(w_i\right) + depth\left(w_j\right)} \tag{1}$$

WUP is part of the similarity calculation toolkit (Pedersen, Patwardhan, and Michelizzi, 2004), which is subsequently used to define distance boundaries for metric learning. Note that the methods in this package are also applicable to other lexical resources like Roget's thesaurus (Jarmasz and Szpakowicz, 2003), which encompasses both syntagmatic and paradigmatic relations within a relatively shallow taxonomy and can be treated as a hybrid network of semantic similarity and relatedness. Additionally, these methods can be extended to the Gene Ontology (Pedersen *et al.* 2007; Guzzi *et al.* 2011) in the bioinformatic domain, as well as for computing verb similarity (Yang and Powers, 2006)

### 3.2. Semantic specialization function

DF for semantically RF NWEs mainly employs a triplet loss function:

$$L_s = \tau\left(m\left(w, w^+\right) + D\left(w, w^+\right) - D\left(w, w^-\right)\right) \tag{2}$$

Here, for a target word $w$, its positive sample $w^+$, and negative sample $w^-$, $D$ denotes the distributional distance in the vector space model of NWEs, $\tau(x) = max(0, x)$, and $m$ is the dynamic distance boundary or margin between the positive pair $(w, w^+)$ and the negative pair of $(w, w^-)$. If $w$ and $w^+$ are measured as semantically similar by WUP, they form a positive pair, and vice versa for a negative pair, where $w$ and $w^-$ are dissimilar. To convert $L_s$ from the learning distance metric into a similarity metric, we use:

$$L_s = \tau\left(sim_{wup}\left(w, w^+\right) + S\left(w, w^-\right) - S\left(w, w^+\right)\right) \tag{3}$$

where $m\left(w, w^+\right) = 1 - sim_{wup}(w, w^+)$ and $S$ stands for the cosine similarity $(1 - D)$ in a distributional space.

In addition to synonymy and antonymy, the semantic constraints for DF also consist of LE, covering both direct and indirect hypernymy, organized as the main taxonomy in WordNet. These constraints are derived from LEAR (Vulić and Mrkšić 2018). When injecting synonymy or LE into the vector space of NWEs, we utilize $sim_{wup}$ to score semantic similarity between $w$ and its semantic associate $w^+$, serving as the similarity boundary in the loss function.

During the mini-batch updating process of NWEs, DF first computes cosine similarity across each batch to retrieve the hard negative samples closest to $w$ or $w^+$. In our experiment, DF fetches only one negative sample for $w$ or $w^+$.

To incorporate antonymy into the loss function and push $w$ or its antonym $w^-$ apart, the corresponding loss function becomes:

$$L_s = \tau\left(sim_{wup}\left(w, w^-\right) + S\left(w, w^-\right) - S\left(w, w^+\right)\right) \tag{4}$$

Here, the positive sample $w^+$ in antonymy specialization should be farthest apart from $w$ or $w^-$ or have the minimum distributional similarity with $w$ or $w^-$ in a batch.

After incorporating taxonomic similarity into the loss functions, DF aims to optimize distributional semantics in the vector space of NWEs, so that words semantically similar to $w$ should be clustered together based on their proximity in semantic networks, while words dissimilar to $w$ should remain separable. DF dynamically adjusts distance boundaries between $w$ and its positive samples according to their semantic similarity, aiming to simulate semantic hierarchies in WordNet. By eliminating the need for manually selected margins in metric learning, DF enriches the range of semantic specializations for post-processing NWEs, effectively fusing distributional semantics with lexical semantic knowledge.

Unlike previous joint-training or RF methods to enhance embeddings, DF not only employs semantic constraints to fine-tune a vector space model but also weighs semantic similarity in quantifying distance margins to differentiate semantic nuances among word usage patterns.

### 3.3. Hypernymy directionality

Unlike the symmetric relations of syn/antonymy, the IS-A link between a subordinate and its superordinate or hypernym holds directionality or is asymmetric. This implies that we should encode the order of LE into an Euclidean vector space model during RF NWEs. Apart from computing semantic boundaries in RF NWEs, we follow LEAR to embed the directionality of LE into NWEs, which is as follows:

$$L_{LE} = \frac{\left\|w_{hypo}\right\| - \left\|w_{hyper}\right\|}{\left\|w_{hypo}\right\| + \left\|w_{hyper}\right\|} \tag{5}$$

Here, under the assumption that the $L2$ vector norm of a subordinate $w_{hypo}$ in a Euclidean space should be less than the norm of its superordinate $w_{hyper}$, the training objective is to adjust the magnitude of $\|w\|$ to learn hypernymy's directionality from the subordinate $w_{hypo}$ to $w_{hyper}$, and vice versa for hyponymy.

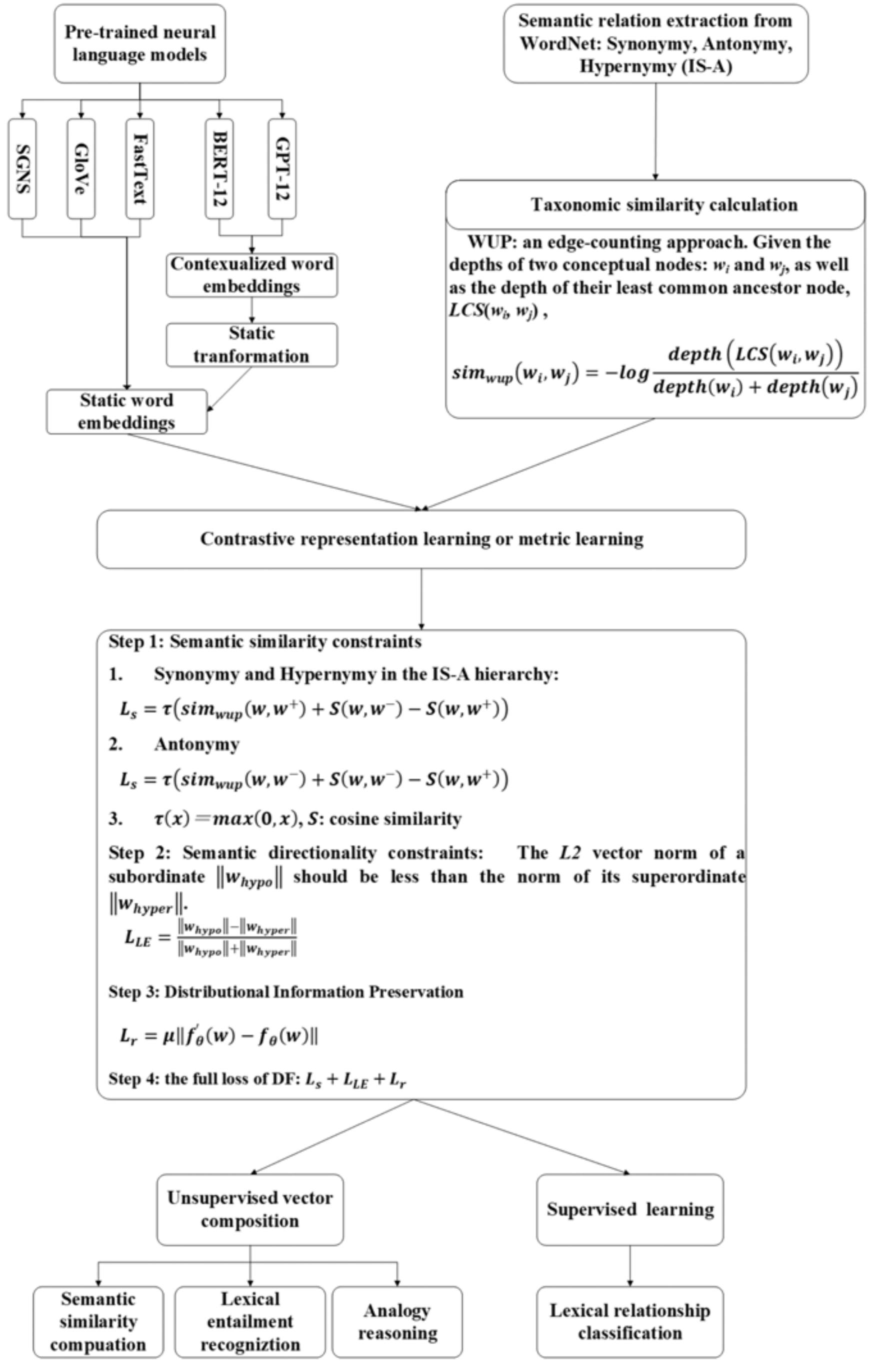
Pre-trained neural language models
SGNS
GloVe
FastText
BERT-12
GPT-12
Contexualized word embeddings
Static tranformation
Static word embeddings
Semantic relation extraction from WordNet: Synonymy, Antonymy, Hypernymy (IS-A)
Taxonomic similarity calculation
WUP: an edge-counting approach. Given the depths of two conceptual nodes: $w_i$ and $w_j$, as well as the depth of their least common ancestor node, $LCS(w_i, w_j)$ ,
$sim_{wup}(w_i,w_j) = -log\frac{depth(LCS(w_i,w_j))}{depth(w_i)+depth(w_j)}$
Contrastive representation learning or metric learning
Step 1: Semantic similarity constraints
1. Synonymy and Hypernymy in the IS-A hierarchy:
$L_s = \tau(sim_{wup}(w,w^+) + S(w,w^-) - S(w,w^+))$
2. Antonymy
$L_s = \tau(sim_{wup}(w,w^-) + S(w,w^-) - S(w,w^+))$
3. $\tau(x) = max(0,x)$, $S$: cosine similarity
Step 2: Semantic directionality constraints: The $L2$ vector norm of a subordinate $\|w_{hypo}\|$ should be less than the norm of its superordinate $\|w_{hyper}\|$.
$L_{LE} = \frac{\|w_{hypo}\|-\|w_{hyper}\|}{\|w_{hypo}\|+\|w_{hyper}\|}$
Step 3: Distributional Information Preservation
$L_r = \mu\|f'_\theta(w) - f_\theta(w)\|$
Step 4: the full loss of DF: $L_s + L_{LE} + L_r$
Unsupervised vector composition
Supervised learning
Semantic similarity compuation
Lexical entailment recogniztion
Analogy reasoning
Lexical relationship classification

**Figure 2.** A flowchart of dynamic fitting on neural word embeddings.

### 3.4. Distributional information preservation

Following other post-processing methods such as LEAR and LexSub, DF also incorporates a regularization function in RF:

$$L_r = \mu \left\| f_\theta^{'}(w) - f_\theta(w) \right\| \tag{6}$$

Here, $\mu$ is the adjustable coefficient for $L2$ regularization. $L_r$ aims to partially preserve the distributional semantics of $w$ learned in its original vector space $f_\theta(w)$, preventing over-correction in the specialized space $f_\theta^{'}(w)$.

The full loss function of DF comprises $L_s$, $L_{LE}$, and $L_r$. Different from the previous post-processing methods that establish constant distance margins or handpick them separately for each category of semantic constraints, we propose to replace them with taxonomic similarity computation, dynamically adjusting them in metric learning. Hence, DF not only can accommodate multiple semantic relationships into a vector space model but also can augment NWEs with semantic taxonomy through learning the IS-A hierarchy in WordNet.

## 4. Experiment setup

Validating the distributional hypothesis in LLMs presents a complex challenge (Bommasani *et al.* 2021). Little general agreement exists on how to validate the quality of NWEs (Bakarov, 2018). Intrinsic evaluations directly measure the correlation between distributional semantics and human judgments on lexical and relational semantics, whereas extrinsic evaluations typically need an extra layer of "supervised learner" for downstream benchmark tasks. Due to the self-supervised way of pretraining and data usage, extrinsic evaluation of NWEs may only be suitable for certain downstream tasks (Schnabel *et al.* 2015). Conversely, intrinsic evaluation is often considered a robust quality indicator for NWEs but may potentially contain subjectivity or bias (Bakarov, 2018).

Given that we combine taxonomic similarity with distributional similarity in semantically specializing NWEs, we prefer to utilize intrinsic tasks (Baroni *et al.* 2014) to evaluate DF, which include semantic similarity calculation, LE recognition, and lexical analogy reasoning. These benchmark tasks have been widely used in validating distributional semantics (Bakarov, 2018). We compared DF with other popular RF methods, including RF, CF, LEAR, HF, and LexSub.

### 4.1. Neural embeddings

Apart from three popular pretrained static embeddings: GloVe[a], fastText[b] (Bojanowski *et al.* 2017), and SGNS,[c] we also chose two typical contextualized ones: BERT and GPT-2 in evaluation.

Note that it is not feasible to generate static embeddings by simply feeding a single word into BERT or GPT-2 (Bommasani *et al.* 2020), as contextualized embeddings typically exhibit anisotropic characteristics or occupy a relatively small conical space across layered transformers (Ethayarajh, 2019). Gupta and Jaggi (2021) adapted the CBOW-like pretraining method to convert contextualized embeddings into their static representations, which notably outperforms two common distilling methods—calculating the first principal component (Ethayarajh, 2019) and average-pooling (Bommasani *et al.* 2020)—in semantic similarity calculation tasks. We thus opted for the two best static embeddings,[d] based on sentence contexts and the last layer of a 12-layer BERT and GPT-2 in their study.

[a] https://nlp.stanford.edu/projects/glove
[b] https://fasttext.cc/docs/en/english-vectors.html
[c] https://github.com/eyaler/word2vec-slim
[d] https://zenodo.org/record/5055755

**Table 1.** Evaluation tasks in the lexical entailment recognition and lexical relationship classification

| | Data size | Relationship types | Task | Method |
|---|---|---|---|---|
| BLESS | 1,337 | **hypernymy(*hyper*):** *fox vs carnivore: hyper.* | directionality detection | unsupervised |
| WBLESS | 1,168 | **hypernymy, other:** *stove vs artifact: hyper; stove vs migraine: other.* | bi-classification | unsupervised |
| BIBLESS | 834 | **hypernymy, hyponymy (*rhyper*), other:** *rifle vs gun: hyper; rifle vs revolver: other; device vs rifle: rhyper.* | 3-way classification | unsupervised |
| CogALex-V | 3,054/4,260 (training/testing) | | | |
| Subtask-1 | | **semantically related (*T*) and unrelated (*F*):** *milk vs drink: T; milk vs cloudy: F.* | bi-classification | supervised |
| Subtask-2 | | **hypernymy, synonymy (*syn*), antonymy (*ant*), meronomy (*part-of*), unrelated (*random*):** *nation vs country: syn; brain vs organ: hyper; brain vs head: part-of; bright vs dark: ant; brain vs island: random.* | 5-way classification | supervised |

## 4.2. *Evaluation task*

### *4.2.1 Semantic similarity calculation*

This task focuses on directly evaluating NWEs with the Spearman rank correlation coefficient ($\rho$) in the gold-standard test sets: SimLex-999 (Hill *et al.* 2015) and SimVerb-3500 (Gerz *et al.* 2016). Note that we only used the test set of SimVerb-3500, consisting of 3,000 verb pairs, denoted as SimVerb-3000.

### *4.2.2 Lexical entailment recognition*

This task encompasses three subtasks within the HyperVec toolkit (Nguyen *et al.* 2017), as outlined in Table 1. Specifically, BLESS focuses on detecting hypernymy directionality, WBLESS addresses binary classification to distinguish hypernymy from the other relationships such as hyponymy, meronymy, co-hyponym, and random, while BIBLESS tackles a challenging three-way classification task, identifying hypernymy, hyponymy, and the others. In addition, the toolkit utilizes HyperScore, an unsupervised entailment recognition metric, to assess the impact of the aforementioned post-processing methods on NWEs. HyperScore is defined as follows:

$$HyperScore(w_{hypo}, w_{hyper}) = cosine(w_{hypo}, w_{hyper}) * \frac{\left\|w_{hyper}\right\|}{\left\|w_{hypo}\right\|} \tag{7}$$

It factors in both distributional similarity and the ratio between the vector norms of a subordinate $w_{hypo}$ and its superordinate $w_{hype}$ to recognize LE. This approach provides more interpretability in comparison with supervised methods that mainly seek to "*memorize*" unique contextual feature patterns, as highlighted by Levy *et al.* (2015). Given that only hypernymy is present in BLESS, it is unnecessary to calculate distributional similarity, cosine $(w_{hypo}, w_{hyper})$, within HyperScore. Instead, the ratio $\left\|w_{hyper}\right\| / \left\|w_{hypo}\right\|$ can be directly applied, based on the assumption that $\left\|w_{hyper}\right\|$should be larger than $\left\|w_{hypo}\right\|$ to capture the inherent asymmetry in hypernymy. However, a full HyperScore calculation is required to differentiate hypernymy from other relationships in WBLESS and BIBLESS.

We employ the default settings in the toolkit to determine the respective HyperScore threshold for WBLESS and BIBLESS, where 2% of the data is randomly selected for learning the threshold, and the remainder is reserved for testing. This thresholding process is repeated 1,000 times, and the average precision (AP) is reported (Nguyen *et al.* 2017).

#### *4.2.3 Lexical relationship classification*

In addition to the three unsupervised learning tasks in HyperVec, we include CogALex-V (Santus *et al.* 2016), a supervised benchmark task for lexical relationship classification, for comparison. As summarized in Table 1, CogALex-V consists of two shared tasks: Subtask-1 aims to predict whether two terms are semantically related, while Subtask-2 further classifies their relationship into one of the following categories: synonymy, antonymy, hypernymy, part-whole meronymy, or unrelated (*random*). The dataset of CogALex-V is highly imbalanced, with approximately 73% of the training data and 71.8% of the testing data tagged as *random*. These random pairs are treated as noise and excluded during evaluation, where overall classification performance is reported as the weighted precision, recall, and F1 score. CogALex-V is intentionally designed to be challenging (Santus *et al.* 2016), due to its dataset's diverse resources, such as WordNet, ConceptNet, and crowdsourcing, and lack of morphological or PoS information, which complicates lexical relationship identification. Note that, apart from the monolingual CogALex-V (English), the CogALex-VI shared task (Xiang *et al.* 2020) focuses on multilingual identification of semantic relationships and may serve as a benchmark for evaluating multilingual NWEs in future research.

We train a multilayer perceptron (MLP) model with a single hidden layer containing 128 neurons for the two subtasks in CogAlex-V. The rectified linear unit (ReLU) activation is applied to both the input and hidden layers. For the final output layer, we employ the Sigmoid function for Subtask-1 and the Softmax function for Subtask-2. During training with cross-entropy loss, we set the learning rate to 0.001 and epoch and batch size to 50 and 200, respectively.

We feed both numerical and vectorial features into the MLP classifier. The numerical features consist of the Euclidean norms of the embeddings for the input pair, along with their cosine similarity score. The vectorial feature is derived through their vector subtraction. Each numerical feature is transmuted into its corresponding vector using the Gaussian-based feature vectorization (Maddela and Xu, 2018), with an optimal dimension size of 30.

#### *4.2.4 Analogy reasoning*

It primarily examines whether the inter-conceptual relationship between $a$ and $a^{'}$ also holds between $b$ and $b^{'}$, which is applicable to tasks such as word-sense disambiguation and semantic relation detection. Unsupervised methods for analogy reasoning often involve computing both relation and word similarity. For example, 3CosAdd (Mikolov *et al.* 2013b) computes $b^{'}$ as $argmax_{b' \in V}( \cos (b^{'}, b - a + a^{'}))$, while PairDistance (Levy and Goldberg, 2014) computes $b^{'}$ as $argmax_{b' \in V}( \cos (b^{'} - b, a^{'} - a))$. Bouraoui *et al.* (2020) proposed fine-tuning NLMs for analogy reasoning, achieving much higher accuracy than unsupervised methods. However, supervised methods may not generalize as well as unsupervised ones in dealing with new types of relationships. For this task, we use 3CosAdd, a simple vector arithmetic method.

Given that we semantically specialize NWEs with syn/antonymy and hypernymy, we used a subset of the bigger analogy test set (BATS) (Gladkova, Drozd, and Matsuoka, 2016) to validate DF on relation inference, which consists of L01, L02, and L03 for hypo/hypernymy and L07 and L08 for synonymy, along with L09 and L10 for antonymy.

### *4.3. Training parameter settings*

Similar to LEAR and HF, we chose a standard dataset to assess the training results of DF, consisting of 201 pairs of nouns from WordSim-353-similarity (Agirre *et al.* 2009) and 500 pairs of verbs from the training part of SimVerb-3500 (Gerz *et al.* 2016).

**Table 2.** Measuring semantic similarity using different retrofitting methods on NWEs

| | SimLex-999 / SimVerb-3000 | | | | |
|---|---|---|---|---|---|
| | SCNS | GloVe | fastText | BERT | GPT-2 |
| Vanilla | 0.13/0.05 | 0.37/0.22 | 0.18/0.14 | 0.55/0.44 | 0.54/0.47 |
| RF | 0.27/0.14 | 0.50/0.31 | 0.38/0.28 | 0.62/0.51 | 0.62/0.54 |
| CF | 0.29/0.19 | 0.55/0.41 | 0.38/0.31 | 0.64/0.57 | 0.65/0.60 |
| LEAR | 0.66/0.65 | 0.71/0.69 | 0.68/0.67 | 0.67/0.68 | 0.70/0.69 |
| HF | 0.77/0.75 | 0.79/0.75 | 0.78/0.75 | 0.79/0.77 | 0.80/0.77 |
| DF | 0.76/0.75 | 0.78/0.76 | 0.77/0.76 | 0.78/0.76 | 0.79/0.76 |

We optimized the cost function of DF with AdaGrad, along with hyperparameters such as learning rate and mini-batch size in a grid search. The initial learning rate was set to 0.002, the small batch dataset size to 32, and the L2 regularization constant to 0.001. We only retrieved one positive or negative sample in each mini-batch, depending on the type of semantic constraints in DF.

## 5. Results and analysis

### *5.1. Semantic similarity calculation*

As for the static NWEs: SGNS, GloVe, and FastText, the mean Spearman rank correlation coefficient ($\bar{\rho}$) of DF on SimLex-999 and SimVerb-3000 is 0.77 and 0.76, respectively, as shown in Table 2. They are notably higher than those of Vanilla, RF, and CF, which stand at 0.23 and 0.14, 0.39 and 0.24, 0.36 and 0.30, respectively. For contextualized embeddings: BERT and GPT-2, DF attains $\bar{\rho}$ of 0.79 and 0.76, respectively, outperforming Vanilla, RF, and CF by at least 0.14 and 0.18. In comparison with RF and CF, DF integrates both syn/antonymy and hypo/hypernymy in RF NWEs, indicating that injecting additional semantic relationships can enhance vector space models in computing semantic similarity.

Across all NWEs in Table 2, DF improves LEAR in $\bar{\rho}$ by 14.7% and 11.8% on the two datasets. Although both methods employ the same types of semantic relations and triplet loss to impose semantic constraints on NWEs, LEAR presets distance boundaries in metric learning, whereas DF dynamically adjusts distance margins using taxonomic similarity. The results suggest that DF can further improve vector semantics and more effectively enforce the specialization effect of different relationships on NWEs in comparison with LEAR.

The performance of DF slightly lags HF in SimLex-999 but is on par with HF in SimVerb-3000 ($\bar{\rho}$: 0.76). Wilcoxon rank-sum tests indicate no significant difference between them ($P > 0.05$). While both HF and DF employ triplet loss functions and the same semantic constraints as LEAR, HF also adds a complex quadruplet loss function to incorporate hierarchical relationships into NWEs. HF coordinates two preset margins to account for the semantic distance between different IS-A relations in RF NWEs. In contrast, DF tallies with HF by dynamically adjusting distance boundaries using taxonomic similarity calculation.

To further explore the RF effects of different semantic constraints employed by DF on NWES, we conducted an ablation study to assess how these constraints affect the derivation of semantic similarity from NWEs. We progressively eliminated the semantic constraints of hypernymy, antonymy, and synonymy in post-processing NWEs, with the results presented in Figure 3. As for the static embeddings: SGNS, GloVe, and fastText, the removal of the hypernymy constraint is detrimental to DF in deriving semantic similarity, with an average performance decline of ($\bar{\rho}$ = 0.06 across the two benchmark tasks. The subsequent removal of antonymy and synonymy further hampers DF's performance, resulting in additional reductions of 0.08 and 0.44, respectively. For contextualized embeddings: BERT and GPT-2, DF exhibits almost no deterioration on computing semantic similarity after the removal of IS-A relations. However, its performance

**Table 3.** Results of different retrofitting methods in lexical entailment recognition

| | | Vanilla | RF | CF | LEAR | HF | DF |
|---|---|---|---|---|---|---|---|
| BLESS | SGNS | 0.53 | 0.59 | 0.52 | 0.96 | 0.87 | **0.97** |
| | GloVe | 0.28 | 0.55 | 0.52 | 0.96 | 0.88 | **0.97** |
| | fastText | 0.34 | 0.56 | 0.43 | 0.95 | 0.89 | **0.97** |
| | BERT | 0.50 | 0.49 | 0.48 | 0.95 | 0.96 | **0.97** |
| | GPT | 0.41 | 0.52 | 0.36 | 0.96 | 0.96 | **0.97** |
| WBLESS | SGNS | 0.52 | 0.52 | 0.51 | 0.88 | 0.75 | **0.89** |
| | GloVe | 0.46 | 0.52 | 0.52 | **0.88** | 0.74 | **0.88** |
| | fastText | 0.48 | 0.52 | 0.50 | **0.89** | 0.75 | **0.89** |
| | BERT | 0.53 | 0.51 | 0.52 | 0.88 | 0.86 | **.89** |
| | GPT | 0.53 | 0.52 | 0.48 | **0.89** | 0.86 | 0.88 |
| BIBLESS | SGNS | 0.38 | 0.36 | 0.38 | **0.84** | 0.60 | **0.84** |
| | GloVe | 0.34 | 0.36 | 0.37 | **0.85** | 0.58 | 0.82 |
| | fastText | 0.33 | 0.38 | 0.37 | **0.85** | 0.59 | 0.84 |
| | BERT | 0.38 | 0.37 | 0.38 | **0.85** | 0.78 | 0.84 |
| | GPT | 0.37 | 0.37 | 0.35 | **0.85** | 0.78 | 0.83 |

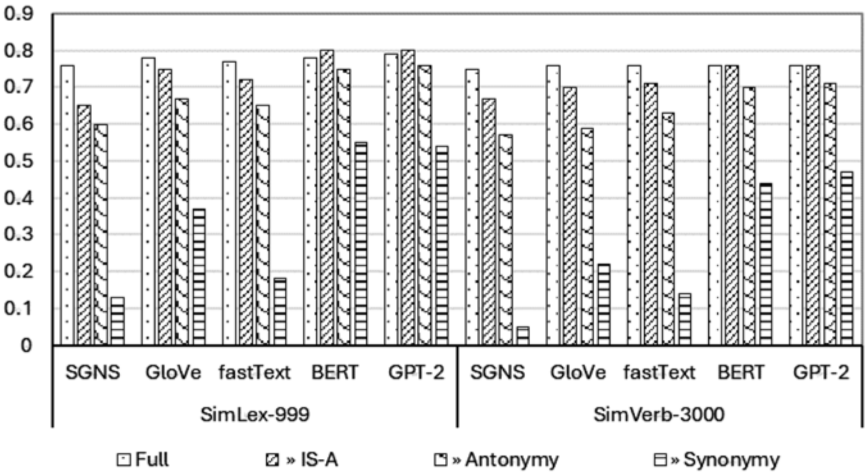

**Figure 3.** Ablation study on the different effects of semantic constraints on retrofitting vector semantics. Full stands for using all the semantic relationships: the IS-A link (hypernymy hierarchy) and syn/antonymy in DF.

gradually declines, with average reductions of 0.05 for antonymy and 0.23 for synonymy. Among the three semantic constraints, synonymy plays a more significant role than the others in semantically constraining both static and contextualized embeddings via DF. This underscores DF's effectiveness in incorporating diverse semantic knowledge into the geometric space of NWEs, enabling static embeddings to perform comparably to contextualized embeddings in deriving semantic similarity.

### 5.2. *Lexical entailment recognition*

As shown in Table 3, DF achieves an AP of 0.91 for SGNS, GloVe, and fastText and 0.88 for BERT and GPT-2 across the three benchmark tasks. DF outperforms RF by 85.7% and 91.3%,

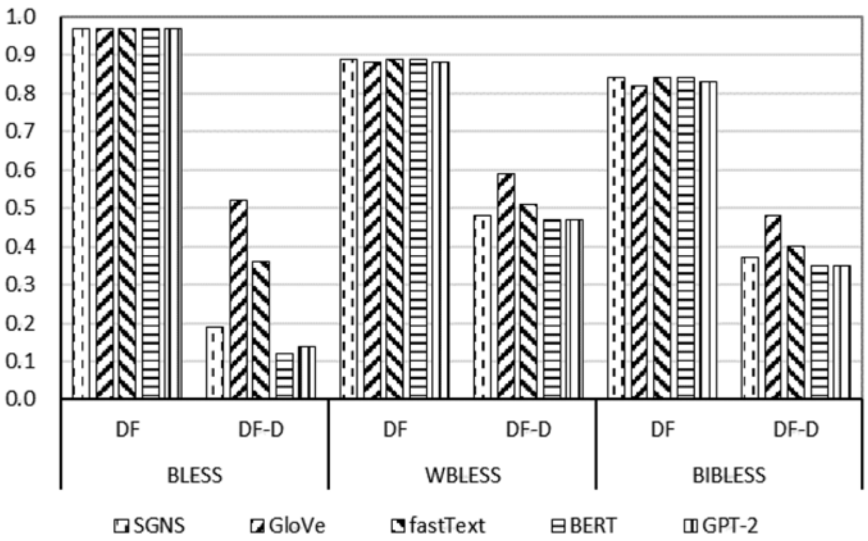

**Figure 4.** The effect of hypernymy directionality function in DF on identifying lexical entailment.

together with CF by 97.8% and 104.7%. There is no significant difference among DF-specified NWEs in recognizing LE (one-way ANOVA, $P > 0.05$). The absence of hypernymy constraints in RF and CF, along with the lack of directionality optimization for LE, underscores the advantage of DF, LEAR, and HF. DF achieves AP of 0.97, 0.89, and 0.83 for the three tasks in Table 3, only slightly inferior to LEAR on BIBLESS, but LEAR can surpass HF by a considerable margin. HF only uses direct hypernyms in RF, whereas DF and LEAR also cover indirect hypernyms, highlighting the utility of hierarchical IS-A relations in facilitating LE recognition. These results emphasize the validity of DF in dynamically adapting distance boundaries for the hypernymy taxonomy.

To further examine the significance of the hypernymy directionality function, we excluded it from the cost function of DF (denoted as DF-D) to investigate its impact on LE recognition. As shown in Figure 4, APs attained by DF-D on NWEs decrease by 72.6%, 43.1%, and 53.2% in BLESS, WBLESS, and BIBLESS, respectively, in comparison with DF. However, NWEs retrofitted by DF-D maintain $\bar{\rho}$ at 0.76 and 0.74 in SimLex-999 and SimVerb-3000, respectively, with negligible deviation from DF. This underscores the importance of simultaneously learning both directionality and similarity metrics when injecting asymmetric semantic constraints into a vector space model, which is also observed in HF (Yang *et al.* 2022).

### *5.3. Lexical relationship classification*

Additionally, to investigate the impact of semantic specialization on neural embeddings, we complement the unsupervised LE recognition in HyperVec with the supervised lexical relationship classification results on CogALex-V, as shown in Table 4. Across both static and contextualized embeddings, all the RF approaches, on average, consistently exceed Vanilla, demonstrating a positive impact on enhancing distributional semantics. For the binary classification of Subtask-1, DF performs comparably to HF, with both achieving an average F1 score above 0.85 across NWEs. RF, LEAR, HF, and DF all outperform GHHH (F1 = 0.79) (Attia *et al.* 2016), the top-performing participant system that employs cosine similarity feature and simple logistics. Subtask-2 poses a more challenging classification task than Subtask-1, with a notable drop in F1 scores in Table 4. Specifically, LEAR, HF, and DF all yield average F1 scores above 0.51, outperforming LexNet

**Table 4.** Results of different retrofitting methods on CogALex-V. Each cell indicates the weighted F1 scores for Subtask-1/Subtask-2. The best scores for NWEs are highlighted in bold, with SOTA results from respective papers included for comparison

| | SGNS | GloVe | FastText | BERT | GPT |
|---|---|---|---|---|---|
| Vanilla | **0.88**/0.43 | 0.81/0.43 | **0.90**/0.46 | 0.77/0.41 | 0.76/0.42 |
| RF | 0.73/0.30 | **0.89**/0.41 | 0.84/0.36 | **0.90**/0.45 | **0.92**/0.42 |
| CF | 0.72/0.32 | 0.79/0.38 | 0.63/0.34 | 0.84/0.44 | 0.83/0.41 |
| LEAR | 0.83/0.54 | 0.82/0.54 | 0.82/0.53 | 0.83/0.54 | 0.81/0.53 |
| HF | 0.86/**0.58** | 0.88/**0.56** | 0.87/**0.55** | 0.85/0.53 | 0.84/0.52 |
| DF | 0.84/0.55 | 0.81/0.40 | 0.82/0.42 | 0.88/**0.60** | 0.89/**0.61** |
| GHHH | 0.790/0.287 | | | | |
| LexNET | 0.765/0.445 | | | | |
| STM | N/0.453 | | | | |
| SphereRE | N/0.471 | | | | |

(Shwartz and Dagan, 2016), the leading participant system in Subtask-2, which uses an MLP classifier on GloVe. In comparison with the static embeddings: SGNS, GloVe, and FastText, DF attains the highest F1 scores on the contextualized embeddings of BERT (0.60) and GPT (0.61), well beyond other RF methods.

Note that we also include two additional state-of-the-art approaches in Table 4 for comparison on Subtask-2: STM (Glavaš and Vulić 2018) and SphereRE (Wang *et al.* 2019). STM utilizes multiple tensor functions to extract features from unspecialized NWEs, followed by a feedforward neural network to discriminate semantic relationships. Rather than using Euclidean embeddings of individual terms for lexical relation recognition, as seen in GHHH, LexNet, and STM, SphereRE constructs a hyperspherical vector space to embed relationships directly and then employs a feedforward neural network for classification. In a similar vein to STM and SphereRE, which employ straightforward neural networks for classification, DF-retrofitted embeddings such as BERT and GPT demonstrate clear benefits over these SOTA methods on this task.

As shown in Table 5, we delve deep into the prediction result of each type of relationship on DF-retrofitted GPT. The overall weighted recall remains modest at 0.55, highlighting substantial misclassification of semantically related pairs. DF-retrofitted GPT achieves a precision of 0.92 in identifying antonymy but only 0.53 for synonymy. Notably, 48.7% of antonymy predictions are labeled as *random*, whereas this error rate falls to 8.4% for synonymy. As indicated in the confusion matrix in Table 5, after the removal of *random* pairs, 22.6% of hypernymy pairs are misclassified as *syn*, while nearly 24.1% of synonymy pairs are misidentified as *hyper*, partly revealing the complexity in differentiating these relationships. Regarding meronymy predictions, since we incorporate no such relationship in RF GPT, nearly 45.4% of them are labeled as *hyper*, *ant*, and *syn*, and this error rate rises to 72.9% if including *random* pairs, with 50.4% classified as *random*.

### 5.4. Analogy reasoning

Among the static NWEs used to derive semantic similarity in Table 2, GloVe consistently outperforms SGNS and FastText across all RF methods, with LEAR-, HF-, and DF-retrofitted GloVe performing on par with BERT and GPT, the contextualized NWEs. Furthermore, since 3CosAdd, employed in analogy reasoning, mainly measures distributional similarity in vector space, we chose GloVe for this task to assess the effectiveness of various RF methods. Top 1 precision (P@1)

**Table 5.** Performance of DF-retrofitted GPT on Subtask-2 of CogALex-V. The left shows precision (P), recall (R), and F1 scores for each relationship after removing random pairs as noise, while the right presents the confusion matrix of classification, where each row corresponds to the actual counts of each relationship in the dataset, and each column corresponds to the predicted counts

| | P | R | F1 |
|---|---|---|---|
| *hyper* | 0.69 | 0.56 | 0.62 |
| *ant* | 0.92 | 0.52 | 0.66 |
| *part-of* | 0.55 | 0.56 | 0.55 |
| *syn* | 0.53 | 0.58 | 0.55 |
| Weighted | 0.70 | 0.55 | 0.61 |

| | *random* | *hyper* | *ant* | *part-of* | *syn* |
|---|---|---|---|---|---|
| *random* | 2501 | 108 | 193 | 233 | 24 |
| *hyper* | 24 | 215 | 7 | 55 | 81 |
| *ant* | 124 | 12 | 186 | 20 | 18 |
| *part-of* | 42 | 29 | 3 | 125 | 25 |
| *syn* | 7 | 55 | 7 | 29 | 137 |

**Table 6.** Results of different retrofitting methods on lexical analogy reasoning conditioning on hypernymy. We only list the top 6 candidates in the varied distributional spaces of GloVe. We colored the same candidates

| Question | *What is to cat as canine is to dog?* | | | | | |
|---|---|---|---|---|---|---|
| Correct answer | ***feline*** | | | | | |
| Method | Prediction (similarity score) | | | | | |
| Vanilla | ***feline(0.82)*** | *canines (0.72)* | *felines (0.70)* | *fanciers (0.69)* | *cats (0.69)* | *distemper (0.69)* |
| RF | ***feline(0.82)*** | *cats (0.79)* | *felines (0.77)* | *canines (0.74)* | *kitten (0.74)* | textitfanciers (0.74) |
| CF | ***feline(0.83)*** | *canines (0.73)* | *felines (0.70)* | *cats (0.70)* | *distemper (0.69)* | *equinel (0.68)* |
| LEAR | *cheetah (0.89)* | ***feline (0.88)*** | *puma (0.88)* | *lioness (0.87)* | *felines (0.87)* | *bobcats (0.87)* |
| HF | ***feline(0.85)*** | *jaguar (0.83)* | *leopard (0.82)* | *cheetah (0.82)* | *lynx (0.82)* | *panther (0.80)* |
| DF | ***feline(0.86)*** | *panther (0.86)* | *leopard (0.85)* | *lynx (0.85)* | *jaguar (0.85)* | *cheetah (0.84)* |

**Table 7.** Results of different retrofitting methods on lexical analogy reasoning conditioning on synonymy. We only list the top 6 candidates in the varied distributional spaces of GloVe. We colored the same candidates

| Question | *What is to mother as kid is to child?* | | | | | |
|---|---|---|---|---|---|---|
| Correct answer | ***mom*** | | | | | |
| Method | Prediction (similarity score) | | | | | |
| Vanilla | *dad (0.80)* | ***mom (0.79)*** | *grandmother (0.78)* | *aunt (0.75)* | *remembers (0.75)* | *father (0.74)* |
| RF | ***mom (0.86)*** | *dad (0.84)* | *grandmother (0.82)* | *mama (0.8)* | *aunt (0.78)* | *grandma (0.78)* |
| CF | *grandmother (0.91)* | *daughter (0.90)* | *wife (0.90)* | ***mom (0.89)*** | *mama (0.89)* | *mamma (0.87)* |
| LEAR | *mothers (0.74)* | *sire (0.74)* | *mamas (0.73)* | *momma (0.73)* | *mama (0.73)* | *mummies (0.73)* |
| HF | ***mom (0.86)*** | *mommy (0.84)* | *momma (0.84)* | *mama (0.84)* | *ma (0.83)* | *mammy (0.82)* |
| DF | ***mom (0.86)*** | *mommy (0.84)* | *mama (0.83)* | *momma (0.83)* | *mammy (0.82)* | *ma (0.8)* |

serves as the evaluation metric, assessing whether the first candidate term after 3CosAdd reasoning matches the correct answer, as illustrated in Figure 5. We also randomly sampled an instance from each type of semantic relation in BATS, including IS-A and syn/antonymy, to examine the reasoning process of 3CosAdd, as demonstrated in Tables 6, 7, and 8.

### *5.4.1 Hypernymy analogy*

DF achieves an AP (P@1) of 26.6% for IS-A analogy reasoning in Figure 5, which is comparable to HF (25.7%) but significantly outperforms Vanilla (6.5%), RF (9.8%), CF (3.6%), and LEAR (5.1%). The absence of IS-A relationships in RF and CF results in less desirable results in both LE recognition and analogy reasoning. Although LEAR performs similarly to DF in LE recognition, it lags behind DF by 21.5% in this task. In contrast to DF, LEAR fails to differentiate between direct and indirect hypernymy in its training objective, opting instead for a uniform distance margin for IS-A links.

**Table 8.** Results of different retrofitting methods on lexical analogy reasoning conditioning on antonymy. We only list the top 6 candidates in the varied distributional spaces of GloVe. We colored the same candidates

| | | | | | | |
|---|---|---|---|---|---|---|
| Question | *What is to long as small is to large?* | | | | | |
| Correct answer | ***Short*** | | | | | |
| Method | Prediction (similarity score) | | | | | |
| Vanilla | ***short (0.81)*** | *few (0.78)* | *longer (0.77)* | *just (0.77)* | *little (0.76)* | *way (0.76)* |
| RF | ***short (0.82)*** | *longer (0.79)* | *few (0.78)* | *little (0.78)* | *just (0.76)* | *only (0.75)* |
| CF | ***little (0.87)*** | *tiny (0.76)* | *minor (0.76)* | *longer (0.75)* | *limited (0.74)* | *few (0.74)* |
| LEAR | *slim (0.76)* | *smaller (0.75)* | *inconsiderable (0.75)* | *slender (0.75)* | *half-size (0.74)* | *puny (0.74)* |
| HF | *slim (0.75)* | *slender (0.75)* | *tiny (0.75)* | *smaller (0.73)* | *puny (0.72)* | *little (0.72)* |
| DF | *smaller (0.78)* | *slim (0.77)* | *tiny (0.77)* | *little (0.77)* | *small-scale (0.76)* | *lesser (0.76)* |

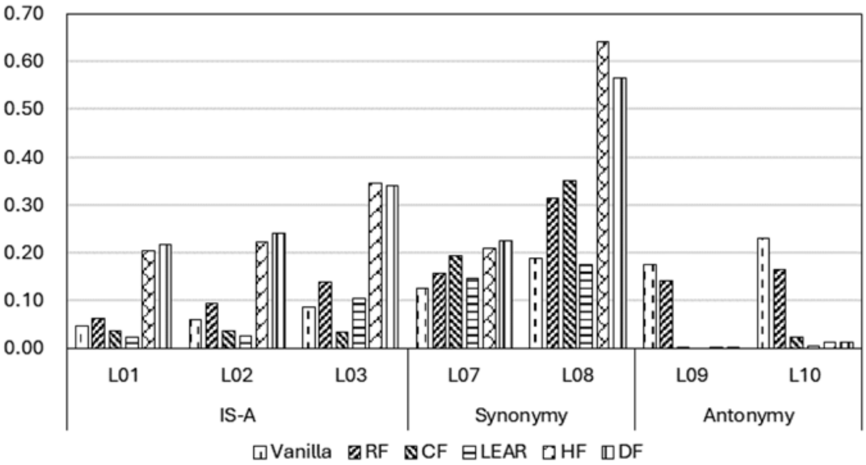

**Figure 5.** Results of retrofitted GloVe for three types of semantic relations in lexical analogy reasoning.

We presented in Table 6 an example of analogy reasoning of

$$b^{'} = argmax_{b^{'} \in V}( \cos (b^{'}, cat - dog + canine)) \tag{8}$$

in different vector spaces of GloVe. Only LEAR finds the correct answer "*feline*" at the second position; all other methods can correctly locate "*feline*" at the top 1. In the top 6 results of Vanilla, RF, and CF, the words "*canines*," "*fanciers*," and "distemper" appear at least twice or more, which are semantically similar to "*feline*" in WordNet. It indicates that GloVe retrofitted by RF and CF may still resemble Vanilla regarding vector semantics. DF and HF both identify the same candidates in the top 6 results but with different ranking orders. HF aims to learn the relational similarity between synonymy and hypernymy by establishing two margins in the quadruplet loss function, whereas DF dynamically adjusts distance margins in metric learning to integrate taxonomy similarity into specializing vector semantics, which is more adaptable for learning richer analogies.

### *5.4.2 Synonymy analogy*

DF reaches $\overline{\text{P@1}}$ of 39.5% in Figure 5, surpassing all other RF methods except for HF (42.6%). LEAR falls behind all other RF methods with $\overline{\text{P@1}}$ of 16.1%, comparable to Vanilla (15.6%).

As depicted in Table 7, the top 1 candidate predicted for synonymy analogy reasoning in Vanilla is "*dad*," which is an antonym of the correct answer "*mom*" "*mom*" only appears in the second position with a high distributional similarity of 0.79. For CF-retrofitted GloVe, the top 1 candidate "*grandmother*" is semantically close to "*mom*," while "*mom*" only appears in the 4th candidate position. "*sire*," the top 1 candidate of LEAR, is semantically contrary to "*mom*," and not correct answer can be found in its top 6 list. These results confirm the findings of Hill *et al.* (2015) on computing distributional similarity in NWEs, namely, semantically contradicted words often have similar vector representations and are distributionally related.

The top 6 candidates predicted in the DF- and HF-retrofitted spaces are fully overlapped, all of which are semantically similar to "*mom*." No correct answer turns up for LEAR. We found that the hypernymy constraint set in LEAR consists of "*mom*" vs "*mother*" and "*mom*" vs "*mothers*." Since LEAR employs unified distance margins for both hypernymy and synonymy in metric learning, not surprisingly, it might predict "*mothers*" instead of "*mom*" in analogy reasoning. In contrast, both DF and HF can distinguish the relational distance between hypernymy and synonymy, which indicates that their capacity to adapt semantic variation while injecting different semantic relations into NWEs may be the main factor affecting the performance of these RF methods in analogy reasoning.

### 5.4.3 Antonymy analogy

Vanilla scores $\overline{\text{P@1}}$ of 20.3% on antonymy analogy reasoning, whereas RF only gains 15.3%, and the other methods perform poorly with less than 1.0% precision. Although both Vanilla and RF predict "*short*" as the top 1 candidate in Table 8, their top 6 lists also include the antonym of the correct answer, "*longer*." Moreover, "*longer*" is present in the prediction list of CF but not in LEAR, HF, and DF, where these methods fail to identify the correct answer and instead produce a group of terms semantically similar to "*small*."

While using antonymy constraints, for example, "*long*" vs "*short*," to retrofit embeddings, if "*large*" is selected as a positive sample in a mini-batch, DF tries to decrease the similarity between "*long*" and "*short*" while increasing the similarity between "*long*" and "*large*." Consequently, "*long*" and "*large*" move closer in the retrofitted space, potentially resulting in $argmax_{b' \in V}(\cos(b', long - largr + small))$ to extract words that are semantically close to "*small*." This partially explains why antonymy analogy reasoning may be challenging for DF, together with LEAR and HF.

The quadruplet loss in HF or the triplet loss in LEAR and DF tend to prioritize synonymy or hypernymy constraints in refining a distributional space, suggesting their preference for synonymy or hypernymy analogy reasoning. We thus replaced the triplet loss of DF with the contrastive loss, which can exclusively reduce semantic similarity between antonyms. The updated results of the task show that DF achieves 36.6% and 42.7% on L09 and L10, respectively, outperforming Vanilla by 19.2% and 19.6%. As DF employs contrastive loss to deal with antonymy separately, it shows no interference with the synonymy and hypernymy analogy reasoning tasks.

### 5.4.4 Related work on word analogy reasoning

In addition to our method, denoted as 3CosAdd_DF + GloVe, we listed SOTA results on BATS for analysis, as shown in Table 9. As for the unsupervised vector arithmetic calculation for analogy reasoning, 3CosAdd_GloVe and 3CosAdd_3-gram stand for the best results reported by Gladkova *et al.* (2016). Note that our Vanilla result (13.0%) is close to 3CosAdd_GloVe.

Additionally, 3CosAvg_GloVe (Drozd, Gladkova, and Matsuoka, 2016) instead uses another vector arithmetic:

$$b' = argmax_{b' \in V}(\cos(b', b + avg_{offset})) \tag{9}$$

for analogy reasoning, where

$$avg_{offset} = \frac{\sum_{i=0}^{m} a_i}{m} - \frac{\sum_{j=0}^{n} a'_i}{n} \tag{10}$$

**Table 9.** Analogy reasoning methods on BATS. The best results are in black font. The results of $BERT_{100}^{max}$ are F1 values

| | L01 | L02 | L03 | L07 | L08 | L09 | L10 | Mean |
|---|---|---|---|---|---|---|---|---|
| 3CosAdd_GloVe | 6.0 | 4.0 | 5.0 | 11.0 | 13.5 | 21.0 | 37.0 | 13.9 |
| 3CosAdd_3-gram | 10.0 | 4.5 | 8.0 | 11.5 | 16.5 | 19.0 | 34.0 | 14.8 |
| 3CosAvg_GloVe | 2.5 | 2.5 | 7.0 | 10.0 | 9.0 | 18.0 | 34.0 | 11.9 |
| LRCos_GloVe | 35.0 | 23.0 | 12.0 | 18.0 | 7.0 | 22.0 | 28.0 | 20.7 |
| 3CosAdd DF + GloVe | 21.8 | 24.1 | 34.0 | 22.5 | **56.5** | 6.6 | 42.7 | 34.0 |
| $\text{BERT}_{100}^{\text{max}}$ | **71.8** | **78.8** | **61.3** | **50.8** | 48.7 | **48.5** | **54.5** | **59.2** |

$a_i$ and $a_i^{'}$ are similar words of $a$ and $a^{'}$, respectively. LRCos_GloVe (Drozd *et al.* 2016) infers with

$$b^{'} = argmax_{b^{'} \in V}(P_{b^{'} \in T} * \cos(b^{'}, b)) \tag{11}$$

where $P$ denotes the logistic regression probability of $b^{'}$ and $b$ in the same class.

Instead of relying on pairwise vector calculation like 3CosAdd, 3CosAvg and LRCos use a group of homogeneous vectors for analogy reasoning, avoiding dependence on individual words. Consequently, their results in Table 9 outperform 3CosAdd: GloVe and 3CosAdd: 3-gram. Our method attains the best results among the unsupervised vector composition methods, with an average accuracy of 34.0%.

Furthermore, we compared DF with a supervised learning method (Bouraoui *et al.* 2020), $\text{BERT}_{100}^{\text{max}}$, which is based on fine-tuning BERT with a binary analogy classifier. $\text{BERT}_{100}^{\text{max}}$ achieves the best results on nearly every sub-dataset except for L08 in Table 9. As suggested by Rogers *et al.* (2020), although such BERTology methods may excel in some downstream applications through fine-tuning NLMs, they may generalize less effectively in dealing with new types of analogy reasoning, in which unsupervised vector composition may show potential adaptability.

## 6. Conclusion

We have proposed leveraging both co-occurrence patterns in distributional semantics and hand-crafted relationships in lexical semantics to construct a unified semantic space. Unlike common practices of thresholding distance boundaries for different semantic constraints in metric learning, we simplify this process through automating taxonomy similarity computation to dynamically differentiate the impact of the semantic distinctiveness of each category of relationships on vector semantics. This effectively imposes semantic constraints on both static and contextualized NWEs and injects semantic hierarchy into a distributional semantic space. In comparison with the popular RF methods on semantically specializing NWEs, our DF model has shown no limitations on utilizing various types of semantic relationships and their association intensity in LKBs, advancing previous methods by avoiding incomplete or improper specialization on vector space models.

To further enhance distributional semantics in NWEs, future research can explore the fusion of multiple knowledge resources, such as Wiktionary and PPDB, to circumvent issues of conceptual obsolescence and lexical omission in WordNet. Additionally, integrating grounded semantics from multimodal knowledge bases such as BabelNet and Wikipedia into NWEs can be beneficial. Given that computing semantic relatedness rather than semantic similarity is more widely applicable in NLP, it is worthwhile to investigate augmenting NWEs with ConceptNet, which contains a rich array of semantic associations.

As NWEs heavily rely on self-supervised prediction on co-occurrences in context and a large volume of online linguistic corpora, they are susceptible to contamination by social biases, which can detrimentally affect the credibility of both static and contextualized embeddings in downstream applications. Post-processing NWEs according to the guidelines of semantic knowledge in LKBs may offer a potential solution for minimizing the occurrence of these biases.

## References

**Agirre E.**, **Alfonseca E.**, **Hall K.**, **Strakova J.**, **Pasca M.** and **Soroa A.** (2009). A study on similarity and relatedness using distributional and WordNet-based approaches. In The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 19–27.

**Alsuhaibani M.**, **Bollegala D.**, **Maehara T.**, **Kawarabayashi K.I.**, **Couto F. M.** (2018). Jointly learning word embeddings using a corpus and a knowledge base. *PLOS ONE* **13**, e0193094.

**Arora K.**, **Chakraborty A.** and **Cheung J. C. K.** (2020). Learning lexical subspaces in a distributional vector space. *Transactions of the Association for Computational Linguistics* **8**, 311–329.

**Attia M.**, **Maharjan S.**, **Samih Y.**, **Kallmeyer L.** and **Solorio T.** (2016). CogALex-V shared task: GHHH - detecting semantic relations via word embeddings. In Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V), The COLING 2016 Organizing Committee. pp. 86–91.

**Bakarov A.** (2018). A survey of word embeddings evaluation methods. arXiv preprint arXiv:1801.09536.

**Baker C. F.**, **Fillmore C. J.** and **Lowe J. B.** (1998). The berkeley framenet project. In The 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Association for Computational Linguistics, pp. 86–90. 980860.

**Banjade R.**, **Maharjan N.**, **Niraula N.**, **Rus V.** and **Gautam D.** (2015). Lemon and tea are not similar: measuring word-to-word similarity by combining different methods. In Computational Linguistics and Intelligent Text Processing (CICLing 2015), Springer International Publishing, pp. 335–346.

**Baroni M.**, **Dinu G.** and **Kruszewski G.** (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In The 52nd Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, vol **1**, pp. 238–247.

**Bellet A.**, **Habrard A.** and **Sebban M.** (2015). *Metric Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning, Springer Cham., vol 9, pp. 1–151.

**Bojanowski P.**, **Grave E.**, **Joulin A.** and **Mikolov T.** (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* **5**, 135–146.

**Bollacker K.**, **Evans C.**, **Paritosh P.**, **Sturge T.** and **Taylor J.** (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data, Association for Computing Machinery, pp. 1247–1250.

**Bommasani R.**, **Davis K.** and **Cardie C.** (2020). Interpreting pretrained contextualized representations via reductions to static embeddings. In The 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 4758–4781.

**Bommasani R., Hudson D. A., Adeli E., Altman R., Arora S., Arx S. v., Bernstein M. S., Bohg J., Bosselut A., Brunskill E., Brynjolfsson E., Buch S., Card D., Castellon R., Chatterji N. S., Chen A. S., Creel K. A., Davis J., Demszky D., Donahue C., Doumbouya M. K. B., Durmus E., Ermon S., Etchemendy J., Ethayarajh K., Fei-Fei L., Finn C., Gale T., Gillespie L., Goel K., Goodman N. D., Grossman S., Guha N., Hashimoto T., Henderson P., Hewitt J., Ho D. E., Hong J., Hsu K., Huang J., Icard T. F., Jain S., Jurafsky D., Kalluri P., Karamcheti S., Keeling G., Khani F., Khattab O., Koh P. W., Krass M. S., Krishna R., Kuditipudi R., Kumar A., Ladhak F., Lee M., Lee T., Leskovec J., Levent I., Li X. L., Li X., Ma T., Malik A., Manning C. D., Mirchandani S., Mitchell E., Munyikwa Z., Nair S., Narayan A., Narayanan D., Newman B., Nie A., Niebles J. C., Nilforoshan H., Nyarko J. F., Ogut G., Orr L. J., Papadimitriou I., Park J. S., Piech C., Portelance E., Potts C., Raghunathan A., Reich R., Ren H., Rong F., Roohani Y., Ruiz C., Ryan J., Ré C., Sadigh D., Sagawa S., Santhanam K., Shih A., Srinivasan K. P., Tamkin A., Taori R., Thomas A. W., Tramèr F., Wang R. E., Wang W., Wu B., Wu J., Wu Y., Xie S. M., Yasunaga M., You J., Zaharia M. A., Zhang M., Zhang T., Zhang X., Zhang Y., Zheng L., Zhou K.** and **Liang P.** (2021). On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258.

**Bond F.** and **Foster R.** (2013). Linking and Extending an Open Multilingual Wordnet. In The 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, pp. 1352–1362

**Bouraoui Z.**, **Camacho-Collados J.** and **Schockaert S.** (2020). Inducing relational knowledge from bert. In **Bouraoui, Z.**, **Camacho-Collados, J.** and **Schockaert, S.** (eds), Proceedings of the AAAI Conference on Artificial Intelligence, **34**, pp. 7456–7463.

**Bubeck S., Chandrasekaran V., Eldan R., Gehrke J. A., Horvitz E., Kamar E., Lee P., Lee Y. T., Li Y. F., Lundberg S. M., Nori H., Palangi H., Ribeiro M. T.** and **Zhang Y.** (2023). Sparks of artificial general intelligence: early experiments with GPT-4. arXiv preprint arXiv:2303.12712.

**Camacho-Collados J.**, **Pilehvar M. T.** and **Navigli R.** (2016). Nasari: integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence* **240**, 36–64.

**Chen C.**, **Liu K.**, **Chen Z.**, **Gu Y.**, **Wu Y.**, **Tao M.**, **Fu Z.** and **Ye J.** (2024a). Inside: Llms' internal states retain the power of hallucination detection. arXiv preprint arXiv:2402.03744.

**Chen G. H.**, **Chen S.**, **Liu Z.**, **Jiang F.** and **Wang B.** (2024b). Humans or llms as the judge? a study on judgement biases. arXiv preprint arXiv:2402.10669.

**Chen L.**, **Deng Y.**, **Bian Y.**, **Qin Z.**, **Wu B.**, **Chua T. S.** and **Wong K. F.** (2023). Beyond factuality: A comprehensive evaluation of large language models as knowledge generators. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 6325–6341.

**Chopra S.**, **Hadsell R.** and **Lecun Y.** (2005). Learning a similarity metric discriminatively, with application to face verification. In The 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), **1**, pp. 539–546.

**Chuang Y. S.**, **Xie Y.**, **Luo H.**, **Kim Y.**, **Glass J.** and **He P.** (2024). Dola: Decoding by contrasting layers improves factuality in large language models. In The Twelfth International Conference on Learning Representations. arXiv preprint arXiv:2309.03883.

**Collins A. M.** and **Loftus E. F.** (1975). A spreading activation theory of semantic priming. *Psychological Review* **82**, 407–428.

**Collins A. M.** and **Quillian M. R.** (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior* **8**, 240–247.

**Davison J.**, **Feldman J.** and **Rush A.** (2019). Commonsense knowledge mining from pretrained models. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, pp. 1173–1178.

**Devlin J.**, **Chang M. W.**, **Lee K.** and **Toutanova K.** (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. In The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, vol **1**, pp. 4171–4186.

**Drozd A.**, **Gladkova A.** and **Matsuoka S.** (2016). Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers, The COLING 2016 Organizing Committee. pp. 3519–3530.

**Ethayarajh K.** (2019). How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, pp. 55–65.

**Faruqui M.**, **Dodge J.**, **Kumar Jauhar S.**, **Dyer C.**, **Hovy E.** and **Smith A. N.** (2015). Retrofitting word vectors to semantic lexicons. In The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 1606–1615.

**Fellbaum C.** (1998). *WordNet: An Electronic Lexical Database.* Cambridge, MA: The MIT Press.

**Firth J. R.** (1957). *A Synopsis of Linguistic Theory 1930–1955.* London: Longman, pp. 1–32.

**Fu J.**, **Ng S. K.**, **Jiang Z.** and **Liu P.** (2024). Gptscore: Evaluate as you desire. In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, pp. 6556–6576.

**Gabrilovich E.** and **Markovitch S.** (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In The 20th International Joint Conference for Artificial Intelligence, pp. 1606–1611.

**Ganea O.**, **Bécigneul G.** and **Hofmann T.** (2018). Hyperbolic neural networks. In *Advances in Neural Information Processing Systems*, pp. 31.

**Ganitkevitch J.**, **Van Durme B.** and **Callison-Burch C.** (2013). Ppdb: The paraphrase database. In The 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 758–764.

**Gerz D.**, **Vuli'c I.**, **Hill F.**, **Reichart R.** and **Korhonen A.** (2016). Simverb-3500: A large-scale evaluation set of verb similarity. In The 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 2173–2182.

**Gladkova A.**, **Drozd A.** and **Matsuoka S.** (2016). Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In **Gladkova A.**, **Drozd A.** and **Matsuoka S.** (eds), *Naacl Student Research Workshop*, pp. 8–15.

**Glavaš G.** and **Vulić I.** (2018). Discriminating between lexico-semantic relations with the specialization tensor model. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. **2** (Short Papers). Association for Computational Linguistics, pp. 181–187.

**Gupta P.** and **Jaggi M.** (2021). Obtaining better static word embeddings using contextual embedding models. In The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, pp. 5241–5253.

**Guzzi P. H.**, **Mina M.**, **Guerra C.** and **Cannataro M.** (2011). Semantic similarity analysis of protein data: assessment with biological features and issues. *Briefings in Bioinformatics* **13**, 569–585.

**Harris Z.** (1985). *Distributional Structure.* New York: Oxford University Press, pp. 26–47.

**Harris Z. S.** (1954). Distributional structure. *Word-Journal of the International Linguistic Association* **10**, 146–162.

**Hassan S.** and **Mihalcea R.** (2011). Semantic relatedness using salient semantic analysis. *Proceedings of the AAAI Conference on Artificial Intelligence* **25**, 884–889.

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

**Jarmasz M.** and **Szpakowicz S.** (2003). Roget's thesaurus and semantic similarity. In *Recent Advances in Natural Language Processing (RANLP 2003)*. John Benjamins Publishing Company, pp. 212–219.

**Kadavath S.**, **Conerly T.**, **Askell A.**, **Henighan T.**, **Drain D.**, **Perez E.**, **Schiefer N.**, **Dodds Z.**, **Dassarma N.**, **Tran-Johnson E.**, **Johnston S.**, **El-Showk S.**, **Jones A.**, **Elhage N.**, **Hume T.**, **Chen A.**, **Bai Y.**, **Bowman S.**, **Fort S.**, **Ganguli D.**, **Hernandez D.**, **Jacobson J.**, **Kernion J.**, **Kravec S.**, **Lovitt L.**, **Ndousse K.**, **Olsson C.**, **Ringer S.**, **Amodei D.**, **Brown T. B.**, **Clark J.**, **Joseph N.**, **Mann B.**, **McCandlish S.**, **Olah C.** and **Kaplan J.** (2022). Language models (mostly) know what they know. arXiv preprint arXiv:2207.05221.

**Kaya M.** and **Bilge H. S.** (2019). Deep metric learning: a survey. *Symmetry* **11**, 1066.

**Koo R.**, **Lee M.**, **Raheja V.**, **Park J. I.**, **Kim Z. M.** and **Kang D.** (2024). Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics ACL 2024*. Association for Computational Linguistics, pp. 517–545.

**Kumar A. A.** (2021). Semantic memory: a review of methods, models, and current challenges. *Psychonomic Bulletin & Review* **28**, 40–80.

**Lê M.** and **Fokkens A.** (2015). Taxonomy beats corpus in similarity identification, but does it matter? In The International Conference Recent Advances in NLP 2015, Shoumen, BULGARIA: INCOMA Ltd, pp. 346–355.

**Lee Y. Y.**, **Ke H.**, **Yen T. Y.**, **Huang H. H.** and **Chen H. H.** (2020). Combining and learning word embedding with WordNet for semantic relatedness and similarity measurement. *Journal of the Association for Information Science and Technology* **71**, 657–670.

**Leimeister M.** and **Wilson B. J.** (2018). Skip-gram word embeddings in hyperbolic space. arXiv preprint arXiv:1809.01498.

**Lenci A.**, **Sahlgren M.**, **Jeuniaux P.**, **Cuba Gyllensten A.** and **Miliani M.** (2022). A comparative evaluation and analysis of three generations of distributional semantic models. *Language Resources and Evaluation* **56**, 1269–1313.

**Levy O.** and **Goldberg Y.** (2014). Linguistic regularities in sparse and explicit word representations. In **Levy**, **O.** and **Goldberg**, **Y.**, (eds), Proceedings of the Eighteenth Conference on Computational Natural Language Learning, Association for Computational Linguistics, pp. 171–180.

**Levy O.**, **Remus S.**, **Biemann C.** and **Dagan I.** (2015). Do supervised distributional methods really learn lexical inference relations? In The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 970–976.

**Lewis P.**, **Perez E.**, **Piktus A.**, **Petroni F.**, **Karpukhin V.**, **Goyal N.**, **Küttler H.**, **Lewis M.**, **Yih W. t.**, **Rocktäschel T.**, **Riedel S.** and **Kiela D.** (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In Proceedings of the 34th International Conference on Neural Information Processing Systems, Article 793, Curran Associates Inc.

**Li K.**, **Patel O.**, **Viégas F.**, **Pfister H.** and **Wattenberg M.** (2024). Inference-time intervention: eliciting truthful answers from a language model. In Proceedings of the 37th International Conference on Neural Information Processing Systems, Article 1797, Article 1797, Curran Associates Inc.

**Liu Q.**, **Jiang H.**, **Wei S.**, **Ling Z.-H.** and **Hu Y.** (2015). Learning semantic word embeddings based on ordinal knowledge constraints. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, pp. 1501–1511.

**Maddela M.** and **Xu W.** (2018). A word-complexity lexicon and a neural readability ranking model for lexical simplification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 3749–3760.

**Mikolov T.**, **Chen K.**, **Corrado G. S.** and **Dean J.** (2013a). Efficient estimation of word representations in vector space. In The 1st International Conference on Learning Representations (ICLR) Workshop Track, ICLR, pp. 1301–3781.

**Mikolov T.**, **Sutskever I.**, **Chen K.**, **Corrado G.** and **Dean J.** (2013b). Distributed representations of words and phrases and their compositionality. In The 26th International Conference on Neural Information Processing Systems (NIPS), vol. **2**, pp. 3111–3119. 2999959.

**Mikolov T.**, **Yih W. T.** and **Zweig G.** (2013c). Linguistic regularities in continuous space word representations. In **Mikolov**, **T.**, **Yih**, **W. T.** and **Zweig**, **G.** (eds). Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 746–751,

**Miller G. A.**, **Beckwith R.**, **Fellbaum C.**, **Gross D.** and **Miller K. J.** (1990). Introduction to wordnet: an online lexical database. *International Journal of Lexicography* **3**, 235–244.

**Mrkšić N.**, **Séaghdha Ó.**, **Thomson D.**, **Gašić B.**, **Rojas-Barahona M.**, **L. M.**, **Su P. H.**, **Vandyke D.**, **Wen T. H.** and **Young S.** (2016). Counter-fitting word vectors to linguistic constraints. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 142–148.

**Mrkšić N.**, **Vulić I.**, **Séaghdha D. Ó.**, **Leviant I.**, **Reichart R.**, **Gašić M.**, **Korhonen A.**, **Young S.** (2017). Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics* **5**, 309–324.

**Naveed H.**, **Khan A. U.**, **Qiu S.**, **Saqib M.**, **Anwar S.**, **Usman M.**, **Barnes N.** and **Mian A. S.** (2023). A comprehensive overview of large language models. arXiv preprint arXiv:2307.06435.

**Navigli R.** and **Ponzetto S. P.** (2012). Babelnet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* **193**(Supplement C), 217–250.

**Nguyen K. A.**, **Köper M.**, **Schulte im Walde S.** and **Vu N. T.** (2017). Hierarchical embeddings for hypernymy detection and directionality. In The 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 233–243.

**Nickel M.** and **Kiela D.** (2017). Poincaré embeddings for learning hierarchical representations. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc, pp. 6341–6350.

**Nickel M.** and **Kiela D.** (2018). Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In Proceedings of the International Conference on Machine Learning, pp. 3776–3785.

**Pan J. S.**, **Wang X.**, **Yang D.**, **Li N.**, **Huang K.** and **Chu S. C.** (2024). Flexible margins and multiple samples learning to enhance lexical semantic similarity. *Engineering Applications of Artificial Intelligence* **133**, 108275.

**Pedersen T.**, **Pakhomov S. V. S.**, **Patwardhan S.** and **Chute C. G.** (2007). Measures of semantic similarity and relatedness in the biomedical domain. *Journal of Biomedical Informatics* **40**, 288–299.

**Pedersen T.**, **Patwardhan S.** and **Michelizzi J.** (2004). Wordnet: Similarity - measuring the relatedness of concepts. In The Nineteenth National Conference on Artificial Intelligence (AAAI-04), AAAI Press, pp. 1024–1025.

**Pennington J.**, **Socher R.** and **Manning C.** (2014). Glove: Global vectors for word representation, In The 2014 Conference on Empirical Methods in Natural Language Processing, vol **14** of Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543.

**Peters M. E.**, **Neumann M.**, **Iyyer M.**, **Gardner M.**, **Clark C.**, **Lee K.** and **Zettlemoyer L.** (2018). Deep contextualized word representations. In The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 2227–2237.

**Peters M. E.**, **Neumann M.**, **Logan R.**, **Schwartz R.**, **Joshi V.**, **Singh S.** and **Smith N. A.** (2019). Knowledge enhanced contextual word representations. In The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, pp. 43–54.

**Petroni F.**, **Rocktäschel T.**, **Riedel S.**, **Lewis P.**, **Bakhtin A.**, **Wu Y.** and **Miller A.** (2019). Language models as knowledge bases? In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, pp. 2463–2473.

**Radford A.**, **Wu J.**, **Child R.**, **Luan D.**, **Amodei D.** and **Sutskever I.** (2018). Language models are unsupervised multitask learners. Technical report, OpenAi.

**Resnik P.** (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Artificial Intelligence Research* **11**, 95–130.

**Roberts A.**, **Raffel C.** and **Shazeer N.** (2020). How much knowledge can you pack into the parameters of a language model? In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, pp. 5418–5426.

**Rogers A.**, **Kovaleva O.** and **Rumshisky A.** (2020). A primer in bertology: what we know about how bert works. *Transactions of the Association for Computational Linguistics* **8**, 842–866.

**Santus E.**, **Gladkova A.**, **Evert S.** and **Lenci A.** (2016). The CogALex-V shared task on the corpus-based identification of semantic relations. In Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V), The COLING 2016 Organizing Committee. pp. 69–79.

**Schnabel T.**, **Labutov I.**, **Mimno D.** and **Joachims T.** (2015). Evaluation methods for unsupervised word embeddings. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 298–307.

**Schroff F.**, **Kalenichenko D.** and **Philbin J.** (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815–823.

**Shwartz V.** and **Dagan I.** (2016). Cogalex-v shared task: Lexnet - integrated path-based and distributional method for the identification of semantic relations. In Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V), The COLING 2016 Organizing Committee. pp. 80–85.

**Stureborg R.**, **Alikaniotis D.** and **Suhara Y.** (2024). Large language models are inconsistent and biased evaluators. arXiv preprint arXiv:2405.01724.

**Talmor A.**, **Elazar Y.**, **Goldberg Y.** and **Berant J.** (2020). oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics* **8**, 743–758.

**Tenney I.**, **Das D.** and **Pavlick E.** (2019). BERT rediscovers the classical NLP pipeline. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 4593–4601.

**Thomas P.**, **Spielman S.**, **Craswell N.** and **Mitra B.** (2024). Large language models can accurately predict searcher preferences. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval 1930–1940, Association for Computing Machinery.

**Torregrossa F.**, **Allesiardo R.**, **Claveau V.**, **Kooli N.** and **Gravier G.** (2021). A survey on training and evaluation of word embeddings. *International Journal of Data Science and Analytics* **11**, 85–103.

**Tversky A.** (1977). Features of similarity. *Psychological Review* **84**, 327–352.

**Ushio A.**, **Camacho-Collados J.** and **Schockaert S.** (2021). Distilling relation embeddings from pretrained language models. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 9044–9062.

**Vashishth S.**, **Bhandari M.**, **Yadav P.**, **Rai P.**, **Bhattacharyya C.** and **Talukdar P.** (2019). Incorporating syntactic and semantic information in word embeddings using graph convolutional networks. In The 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 3308–3318.

**Vulić I.** and **Mrkšić N.** (2018). Specialising word vectors for lexical entailment. In The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, vol **1**, pp. 1134–1145.

**Wang C.**, **He X.** and **Zhou A.** (2019). Spherere: Distinguishing lexical relations with hyperspherical relation embeddings. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 1727–1737.

**Wang J.**, **Liang Y.**, **Meng F.**, **Sun Z.**, **Shi H.**, **Li Z.**, **Xu J.**, **Qu J.** and **Zhou J.** (2023). Is ChatGPT a good NLG evaluator? a preliminary study. In Proceedings of the 4th New Frontiers in Summarization Workshop, Association for Computational Linguistics, pp. 1–11.

**Wang P.**, **Li L.**, **Chen L.**, **Cai Z.**, **Zhu D.**, **Lin B.**, **Cao Y.**, **Kong L.**, **Liu Q.**, **Liu T.** and **Sui Z.** (2024). Large language models are not fair evaluators. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, pp. 9440–9450.

**Wu M.** and **Aji A. F.** (2023). Style over substance: evaluation biases for large language models. In Proceedings of the 31st International Conference on Computational Linguistics, Association for Computational Linguistics, pp. 297–312.

**Wu Z.** and **Palmer M.** (1994). Verb semantics and lexical selection. In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 133–138.

**Xiang R.**, **Chersoni E.**, **Iacoponi L.** and **Santus E.** (2020). The CogALex shared task on monolingual and multilingual identification of semantic relations. In Proceedings of the Workshop on the Cognitive Aspects of the Lexicon, pp. 46–53.

**Xu C.**, **Bai Y.**, **Bian J.**, **Gao B.**, **Wang G.**, **Liu X.** and **Liu T. Y.** (2014). Rc-net: A general framework for incorporating knowledge into word representations. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, Association for Computing Machinery, pp. 1219–1228.

**Yang B.** and **Mitchell T.** (2017). Leveraging knowledge bases in LSTMs for improving machine reading. In The 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),, Association for Computational Linguistics, pp. 1436–1446.

**Yang D.**, **Li N.**, **Zou L.** and **Ma H.** (2022). Lexical semantics enhanced neural word embeddings. *Knowledge-Based Systems* **252**, 109298.

**Yang D.** and **Powers D. M.** (2006). Verb similarity on the taxonomy of WordNet. In The 3rd International WordNet Conference (GWC-06), pp. 121–128.

**Yang D.** and **Yin Y.** (2022). Evaluation of taxonomic and neural embedding methods for calculating semantic similarity. *Natural Language Engineering* **28**, 733–761.

**Yu M.** and **Dredze M.** (2014). Improving lexical embeddings with semantic knowledge. In The 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, pp. 545–550.

**Zheng L.**, **Chiang W. L.**, **Sheng Y.**, **Zhuang S.**, **Wu Z.**, **Zhuang Y.**, **Lin Z.**, **Li Z.**, **Li D.**, **Xing E. P.**, **Zhang H.**, **Gonzalez J. E.** and **Stoica I.** (2024). Judging llm-as-a-judge with mt-bench and chatbot arena. In Proceedings of the 37th International Conference on Neural Information Processing Systems, Curran Associates Inc, pp. 46595–46623.

**Zipf G. K.** (1965). *Human Behavior and the Principle of Least Effort: an Introduction to Human Ecology*. New York: Hafner Pub. Co, facsim. of 1949 edition.

**Cite this article:** Yang D, Zhang X, Han T and Liu Y (2025). Semantic enrichment of neural word embeddings: Leveraging taxonomic similarity for enhanced distributional semantics. *Natural Language Processing* **31**, 1423–1449. https://doi.org/10.1017/nlp.2025.10005