Artificial Intelligence Review
https://doi.org/10.1007/s10462-021-10056-6

Measuring associational thinking through word embeddings

Carlos Periñán-Pascual[1]

Abstract
The development of a model to quantify semantic similarity and relatedness between words has been the major focus of many studies in various fields, e.g. psychology, linguistics, and natural language processing. Unlike the measures proposed by most previous research, this article is aimed at estimating automatically the strength of associative words that can be semantically related or not. We demonstrate that the performance of the model depends not only on the combination of independently constructed word embeddings (namely, corpus- and network-based embeddings) but also on the way these word vectors interact. The research concludes that the weighted average of the cosine-similarity coefficients derived from independent word embeddings in a double vector space tends to yield high correlations with human judgements. Moreover, we demonstrate that evaluating word associations through a measure that relies on not only the rank ordering of word pairs but also the strength of associations can reveal some findings that go unnoticed by traditional measures such as Spearman's and Pearson's correlation coefficients.

Keywords Association measure · Neural network · Word embedding · Word2Vec · GloVe · FastText

1 Introduction

Word associations have been a topic of intensive study in a variety of research fields, such as psychology, linguistics, and natural language processing (NLP). In psychology, word associations are closely related to free-association tasks (Van Rensbergen et al. 2015; Günther et al. 2016; Bhatia 2017; Rieth and Huber 2017; Dacey 2019; Gilligan and Rafal 2019), where word priming reflects a clear distinction between two types of information inherent in word relationships: associative vs. non-associative, and semantic vs. non-semantic (Harley 2014). Most studies of word priming have looked at pairs of words that are both associatively and semantically related. However, participants can produce words as associates of other words that are not related in meaning; for example, *waiting* can be generated in response to *hospital*. Moreover, there can be semantically related words that are not produced as associates; for example, *dance* and *skate* are related in meaning, but

✉ Carlos Periñán-Pascual
jopepas3@upv.es

[1] Universitat Politècnica de València, Paranimf 1, 46730 Gandia, Valencia, Spain

Published online: 14 August 2021

skate is rarely produced as an associate of *dance*. Therefore, words can be associatively related, semantically related, or both of them.

In linguistics, it is widely agreed that two essential types of lexical relations (i.e. syntagmatic and paradigmatic) are reflected in basic operations in the human brain (Higginbotham et al. 2015; Xiaosa and Wenyu 2016; Kang 2018; Playfoot et al. 2018; Ma and Lee 2019; Reyes-Magaña et al. 2019). On the one hand, syntagmatic relations take place between words with a different part of speech (POS) that frequently co-occur in natural language utterances. In this horizontal axis, we find the phenomena of collocations (e.g. *fine weather*, *torrential rain*, or *light drizzle*) and idioms (e.g. *bite the bullet*, *kick the bucket*, or *pull someone's leg*). On the other hand, paradigmatic relations hold between words that can replace each other in a given sentence without affecting its grammaticality or acceptability. In this vertical axis, we find semantic relations such as synonymy (e.g. *die–perish*, *handsome–pretty*, or *truthful–honest*), antonymy (e.g. *buy–sell*, *dead–alive*, or *hot–cold*), hypernymy (e.g. *adult–woman*, *mammal–horse*, or *vehicle–car*), co-hyponymy (e.g. *woman–man*, *horse–dog*, or *car–truck*) and meronymy (e.g. *bird–wing*, *finger–hand*, or *minute–hour*). Therefore, both types of lexical relations can be considered to be word associations.

Finally, NLP researchers prefer terms such as "semantic similarity" and "semantic relatedness" to refer to word associations (Banjade et al. 2015; Gross et al. 2016; Cattle and Ma 2017; Garimella et al. 2017; El Mahdaouy et al. 2018; Du et al. 2019; Grujić and Milovanović 2019). As stated by Budanitsky and Hirst (2001, p. 13 Budanitsky and Hirst (2001)), "computational applications typically require relatedness rather than just similarity". Whereas semantic similarity is a lexical relation of meaning resemblance (e.g. *bank–trust company*), semantic relatedness is a more general concept, which includes not only similarity but also other lexical-semantic relations (e.g. antonymy, hypernymy, and meronymy) and any kind of functional relationship or frequent association (e.g. *pencil–paper* or *penguin–Antarctica*). In this context, a variety of semantic similarity and relatedness measures have been developed in NLP over the past three decades. Broadly speaking, these measures have been traditionally devised from two different approaches. On the one hand, the weak-knowledge approach is based on the co-occurrence information of words in a corpus. For example, this approach is illustrated by the geometric model, where words are represented as points within a multi-dimensional vector space and semantic similarity is quantified as the spatial distance between two points (e.g. through the cosine coefficient). On the other hand, the strong-knowledge approach is based on the network model, which uses a semantic network–e.g. WordNet (Fellbaum 1998), to define the concept of a given word in relation to other concepts in the network. Figure 1 serves to summarize the terminology used in these research fields, where we employed "word association" as an umbrella term in this study.

The primary goal of this article is not to introduce a new measure of word association but to devise a model (WALE) to measure the associative strength between words by exploring different ways to integrate existing deep neural embeddings. The working hypothesis is that the performance of the model depends not only on the combination of multiple information sources but also on the way these sources are interlaced. In particular, we focus on Word2Vec (Mikolov et al. 2013a) GloVe (Pennington et al. 2014), and FastText (Bojanowski et al. 2017), as they are the most adopted neural language models in distributional semantics. Therefore, we are not concerned with looking into how the hyperparameters of the neural network need to be efficiently tuned or with proposing a new type of neural network to improve the accuracy of the model. This strategy could have led us to conduct this research in an ad-hoc manner. Instead, our work is motivated by the

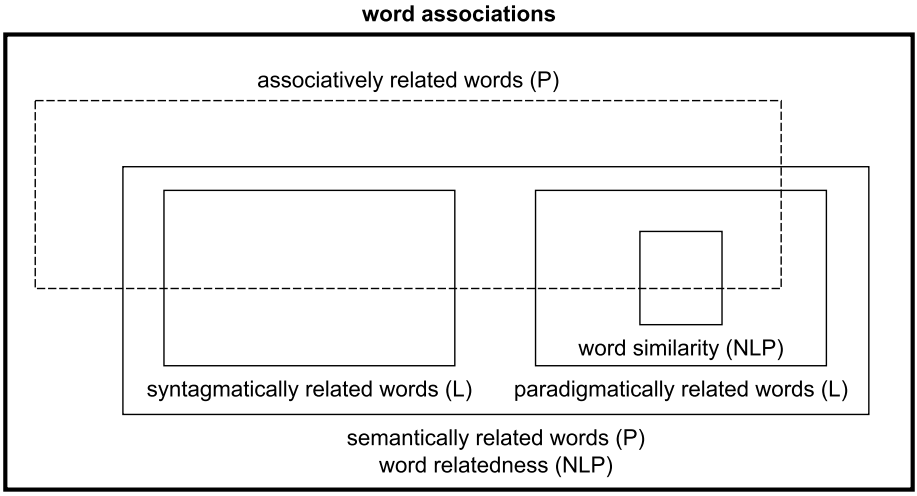

Fig. 1 Terminology on word associations in psychology (P), linguistics (L), and natural language processing (NLP)

assumption that the reuse of general-purpose resources such as pre-trained word embeddings is a critical issue in language engineering, where the development of new components requires considerable time and effort.

The main contributions of this article are as follows:

1. We devised a parametric model that can compute the association strength of two words from the combination of word-embedding matrices, leading to the creation of a single or double vector-space model. Indeed, after extensively experimenting with the integration of embeddings constructed from text corpora (i.e. external language model) with those constructed from a semantic network (i.e. internal language model), we demonstrate that the weighted average of the cosine-similarity coefficients derived from independent corpus- and network-based embeddings in a double vector-space model outperforms not only off-the-shelf embeddings but also other ways of integrating these embeddings. This is the first work that employs this approach to combine word embeddings.
2. We demonstrate that an evaluation measure derived from information-retrieval research can take advantage of not only the rank ordering of word pairs but also the strength of associations, as with the degrees of relevance represented by human annotators in test datasets. Therefore, a measure such as RankDCG can be viewed as more psychologically plausible than measures traditionally used to compute the correlation with human judgements, e.g. Spearman's rank or Pearson's product-moment correlation coefficients. Indeed, as we introduced the possibility to tune RankDCG to assess word associations on rank ordering only or taking into consideration also the associative strength, we managed to analyse the vector-space models generated by several word-embedding techniques through a different exploratory lens, going beyond the results provided by traditional measures. This is the first work that employs RankDCG to evaluate word embeddings.

The remainder of this article is organised as follows. Section 2 describes the most relevant works for this study. Section 3 provides an accurate account of the proposed research

method. Section 4 describes a variety of experiments, whereas Sect. 5 evaluates WALE and Section 6 interprets the results. Finally, Sect. 7 presents some conclusions.

2 Related work

2.1 Distributional semantics

2.1.1 Constructing word-vector models

Distributional semantics, or vector-space semantics, is a usage-based model to represent meaning since it "builds semantic representations from co-occurrence statistics extracted from corpora as samples of language usage" (Lenci 2018, p. 165). Distributional semantics is based on Harris' (Harris 1954) distributional hypothesis, which was famously summarized in Firth's (1957 , p. 11) statement "You shall know a word by the company it keeps". In this context, words are represented as real-valued numbers in vectors, where each number captures a dimension of the meaning of each word so that semantically similar words are mapped to proximate points in the vector-space model. More specifically, the weights that comprise a word vector are learned by making predictions on the probability that other words are contextually close to a given word. Therefore, semantic relatedness is determined by looking at word co-occurrence patterns in corpora so that "contextual similarity then becomes proximity in space" (Erk 2012, p. 635).

Distributional semantics can leverage computational methods to learn meaning representations from language data. There are two primary approaches to train word-vector models: count models and predict(ive) models (Baroni et al. 2014). On the one hand, distributed semantic models can use simple linear algebra on word-to-word co-occurrence counts to reflect the importance of contexts. Some classical weighting functions of count models are raw frequency, tf-idf, pointwise mutual information, or log-entropy. Moreover, as co-occurrence matrices are highly dimensional because the dimensions correspond to the hundreds of thousands of words in a given corpus, these matrices can be factorized to reduce dimensionality, e.g. by using Singular Value Decomposition (SVD) or Principal Component Analysis (PCA), among other techniques. In this way, word vectors are not only more compact but also contain more discriminative dimensions, which makes these representations more effective for semantic-relatedness detection. Concerning the psychological plausibility of this approach, Mandera et al. (Mandera et al. 2017, p. 58) explained that:

> the counting step and its associated weighting scheme could be seen as a rough approximation of conditioning or associative processes and that the dimensionality reduction step could be considered an approximation of a data reduction process performed by the brain

although "it cannot be assumed that the brain stores a perfect representation of word-context pairs or runs complex matrix decomposition algorithms in the same way as digital computers do" (ibid. Mandera et al. 2017). Some examples of count models are Latent Semantic Analysis (LSA) (Deerwester et al. 1990), Hyperspace Analogue to Language (HAL) (Lund and Burgess 1996), Latent Dirichlet Allocation (LDA) (Blei et al. 2003), and Hellinger PCA (Lebret and Collobert 2014).

On the other hand, predictive models, or neural-network models (Bengio and Senécal 2003; Bengio et al. 2003; Morin and Bengio 2005; Collobert and Weston 2008; Mnih and Hinton 2008; Mikolov et al. 2013c), use a non-linear function of word co-occurrences, where word embeddings capture more complex information than just co-occurrence counts.[1] Indeed, (Mandera et al. (2017)) recognized that predictive models are much better psychologically grounded than count models since the underlying principle of implicitly learning how to predict a word from other words is congruent with biologically inspired models of associative learning. One of the most popular neural-network models is Word2Vec, supported by Google (Mikolov et al. 2013a, b, c). Word2Vec is a neural network with a single hidden layer that takes a single word as input and returns the probability that the other words in the corpus belong to the context of the input word. The output of this process is a matrix of n words by k dimensions, or neurons of the hidden layer of the model. Therefore, the hidden layer is introduced to reduce dimensionality, where a non-linear activation function transforms the activations of outcomes to probabilities. Word2Vec can be implemented in two different architectures, i.e. CBOW, where the model attempts to predict the target word from a set of context words, and Skip-gram, where the model predicts the context words from a target word.

Since Word2Vec first came on the scene, other popular word-embedding training techniques have emerged, such as GloVe (Pennington et al. 2014), supported by the NLP research group at Stanford University, and FastText (Bojanowski et al. 2017), developed by Facebook. On the one hand, GloVe builds word embeddings by taking into consideration the frequency of co-occurrences over the whole corpus. It should be recalled that Word2Vec learns embeddings by relating target words to their context, but it ignores whether some context words appear more often than others. Therefore, instead of the log-linear model representations that use local information only in Word2Vec, GloVe exploits global statistical information by using a weighted least-squares model that trains on global word-word co-occurrence counts. It should be noted that GloVe can be considered as a dense count-based method (Riedl and Biemann 2017) since it is based on co-occurrence statistics and does not predict contexts from words directly, as performed in Word2Vec. Indeed, GloVe learns by constructing a co-occurrence matrix, which is factorized to achieve a lower-dimension representation, which brings it close to LDA. However, GloVe uses neural methods to decompose the co-occurrence matrix into more expressive and dense word vectors. As concluded by (Pennington et al. (2014)), GloVe is a model that employs the benefit of count-based methods to capture global statistics while simultaneously capturing the meaningful linear substructures prevalent in prediction-based methods.

On the other hand, FastText is an extension of the Skip-gram architecture implemented by Word2Vec that enriches embeddings with sub-word information using bags of character n-grams. In Word2Vec and GloVe, embeddings are constructed directly from words, which are the smallest units in the training. In contrast, FastText represents each word as a bag of character n-grams (i.e. sub-word units). A vector representation is associated with each character n-gram, and the average of these vectors provides the final representation of the word, from which a Skip-gram model is trained to learn the embeddings. One of the benefits of FastText is that it works well with rare words, or even with words that were not seen during training, since such words can be broken down into n-grams to get their embeddings.

[1] In this article, we employ the term "word embedding" in a narrow sense, that is, to refer to distributional vectors built with neural networks.

It is worthwhile to mention that a new generation of algorithms based on neural language models is now able to construct contextualized word embeddings (Liu et al. 2020b; Pilehvar and Camacho-Collados 2020). These dynamic context-dependent representations are better suited to capture sentence-level semantics than static context-independent word embeddings (i.e. Word2Vec, GloVe, and FastText). In this regard, one of the most popular architectures is BERT (Devlin et al. 2019). In traditional neural embeddings, each word has a fixed real-valued vector representation regardless of the context within which the word appears or the different meanings it can have. In contrast, BERT produces word representations that are dynamically modelled by surrounding words, so it generates different embeddings for each occurrence of a given word in the corpus. As a result, contextualized word embeddings cannot be used directly for word-association tasks due to the lack of sentential contextualization. As explained by (Wang et al. (2020) , p. 1), there are several methods to obtain static embeddings from dynamic embeddings:

> For example, the contextualized vectors of a word can be averaged over a large corpus. Alternatively, the word vector parameters from the token embedding layer in a contextualized model can be used as static embeddings.

However, their experiments showed that these methods do not necessarily outperform traditional static embedding models, which is why our research only focused on the latter.

2.1.2 Combining word vectors

Over the last decade, some studies described semantic models developed from the integration of independent word vectors, motivated by the belief that:

> The plethora of measures available in the literature suggests that no single method is capable of adequately quantifying the similarity/relatedness between words. Therefore, combining different approaches may provide a better result. (Niraula et al. (2015) , p. 200)

(Agirre et al. (2009)) employed a hybrid model. On the one hand, they computed a personalized PageRank vector of probability distributions over the WordNet graph for each word. On the other hand, they constructed a corpus-based vector-space model from different approaches, i.e. bag of words, context window and syntactic dependency, where the method based on context windows provided the best results for similarity and the bag-of-words representation outperformed for relatedness. Finally, they demonstrated that distributional similarities can perform as well as the knowledge-based approach, and the combination of both models using a supervised learner can exceed the performance of results.

(Tsuboi (2014)) showed that the combination of Word2Vec and GloVe embeddings improves accuracy in POS tagging, outperforming the separate use of those embeddings.

(Faruqui and Dyer (2014)) proposed a technique based on Canonical Correlation Analysis (CCA) that first constructs independent vector-space models in two languages and then projects them onto a common vector space, where translation pairs can be maximally correlated. In particular, they constructed LSA word vectors for English, German, French, and Spanish, and then projected the English word vectors using CCA by pairing them with the vectors in the other languages. The experiment was also performed with Skip-gram vectors from the neural-network approach.

(Niraula et al. (2015)) explored how to combine heterogeneous semantic models of word representations. In particular, they experimented with count models such as LSA and

LDA and predictive models such as Word2Vec and GloVe, evaluating all the combinations of these models. They showed that measures of word relatedness and similarity can be improved by combining diverse representations in two different ways: (a) extend, where individual vectors are added to create a new vector, and (b) average, where semantic-similarity scores are computed and then the mean score is taken. In this regard, the average method yielded better results. For example, the average combination of LDA, Word2Vec and Glove outperformed individual vectors. The rationale behind this approach of combining individual word representations is the assumption that different models represent different aspects of the meaning of words. Their experiments also demonstrated that a given combination of models does not perform equally well in word similarity and word relatedness. The distributional hypothesis leads us to expect that it is more likely to give higher scores for *chicken–egg* than *chicken-hen* because the former has a higher number of co-occurrences in a text corpus compared to the latter. Consequently, they suggested that a knowledge-based approach is a must to improve similarity measures.

(Goikoetxea et al. (2016)) showed that the concatenation of word embeddings learned independently from different sources, e.g. a text corpus and WordNet, produces better performance than learning a representation space from one single source. On the one hand, corpus-based representations were derived from Word2Vec. On the other hand, the structure of WordNet was encoded by combining a random walk algorithm and dimensionality reduction to create compact contexts in the form of a pseudo-corpus, from which distributed representations were produced using Word2Vec. Moreover, they tried simple combination methods, e.g. averaging similarity results or concatenating vectors, and more complex methods, e.g. CCA (Faruqui and Faruqui and Dyer (2014)) and retrofitting (Faruqui et al. 2015), demonstrating that simple techniques outperform the more complex techniques in similarity and relatedness tasks.

(Lee et al. (2016)) proposed a novel approach for measuring semantic relatedness by combining the Word2Vec and GloVe word-embedding models, which were trained on Common Crawl and Google News respectively, with WordNet through a weighted composition function. The semantic-relatedness score was computed with Equation 1, where $cos(v_{w_i}, v_{w_j})$ is the cosine similarity between the vector representations of word w_i and w_j, $dist(S_{i,m}, S_{j,n})$ is the path distance between the sense m of w_i and the sense n of w_j in WordNet, and λ is a weighting factor between 0 and 1.

$$rel(w_i, w_j) = \max_{m,n} \lambda * cos(v_{w_i}, v_{w_j}) + (1 - \lambda)\frac{1}{dist(S_{i,m}, S_{j,n})} \tag{1}$$

Their experiments demonstrated that performance increased with the linear combination of word embeddings and WordNet. In particular, according to Equation 1, the best results were obtained with GloVe, rather than with Word2Vec, where $\lambda = 0.75$.

(Yin and Schütze (2016)) proposed methods for the generation of a "meta-embedding", i.e. ensembling distinct word embeddings to create a new embedding. The rationale for this approach is that there is a variety of methods for the production of word embeddings where the overall quality significantly depends on the neural-network model and the language resource. Therefore, meta-embeddings have two key benefits: enhancement and coverage. In other words, a meta-embedding is expected to contain more information and cover more words than the individual embeddings from which the meta-embedding was derived. The alternative is to directly improve the learning algorithm to produce better embeddings, but this strategy substantially increases the training time of embedding learning. These researchers introduced different ensemble approaches, from the simplicity of

word-embedding concatenation to the complexity of meta-embedding learning methods such as 1TON and 1TON+. In this context, (Coates and Bollegala (2018)) showed empirical evidence that averaging across distinct embeddings results in performance comparable to, and in some cases better than, concatenating embedding vectors.

Cross-lingual embedding models at the word level have also influenced our idea to combine word embeddings. On the one hand, bilingual vectors can be trained online (Chandar et al. 2014; Hermann and Blunsom 2013), where the source and target languages are learned together in a shared vector-space model. Typically, this approach makes use of two monolingual text corpora together with a smaller bilingual corpus of aligned sentences. On the other hand, bilingual vectors can be obtained offline (Mikolov et al. 2013b; Faruqui and Dyer 2014; Artetxe et al. 2016; Smith et al. 2017), after which a mapping-based approach is required:

> Mapping-based approaches [...] first train monolingual word representations independently on large monolingual corpora and then seek to learn a transformation matrix that maps representations in one language to the representations of the other language. They learn this transformation from word alignments or bilingual dictionaries. (Ruder et al. 2019, p. 581)

As the geometric constellation that holds between words is similar across languages, it is possible to transform the vector space of the source language to the vector space of the target language by employing a technique such as SVD or CCA to learn a linear projection between the languages.

2.1.3 Word embeddings in text classification

With the exponential increase in text content on the Web (e.g. news articles, customer reviews, tweets, etc.), automatic text classification plays a critical role. To this end, many studies have chosen to use static word embeddings in a wide variety of NLP tasks, e.g. topic categorization (Zhang et al. 2020), sentiment analysis (Smetanin and Komarov 2019; Demotte et al. 2020), fake-news detection (Goldani et al. 2021), and natural language understanding (Pylieva et al. 2019), among others. In this context, our research, which is aimed at generating high-quality word embeddings, can contribute to significantly improving the underlying model of such text-classification systems. In particular, pre-trained word embeddings have been primarily employed as part of topic models and deep neural network-based methods in the last few years.

On the one hand, LDA is by far the most popular topic model in current use, which can infer the probability distribution of hidden topics in a given document and that of words in a given topic. Some of the latest research efforts in topic modelling have been aimed at improving LDA with semantic similarity. Bhutada et al. (2016) proposed Semantic LDA, where they computed topic membership by including in the LDA process two new matrices constructed from the attribute values derived from word- and synonym-frequency information, from which a new measure was used to find the similarity between documents. Poria et al. (2016) presented Sentic LDA, which integrates word distributions with word similarities through the common-sense knowledge in SenticNet (Cambria et al. 2014). Jingrui et al. (2017) proposed a method of optimizing the purity of the topics discovered by LDA based on the semantic similarity between the topics and the categories of news. Moreover, several proposals have been recently presented to integrate LDA with word embeddings. Yu et al. (2017) proposed the Multilayered Semantic LDA, which relies on Word2Vec embeddings

to obtain the semantic similarity of words and thus extract the dimension hierarchies of tweeters' interests. Budhkar and Rudzicz (2019) combined LDA probabilities with Word2Vec representations to increase the accuracy of clinical-text classification. Akhtar et al. (2019) proposed fuzzy document representations generated by LDA, where each document is represented as a fuzzy bag of words using Word2Vec to calculate word-level semantic similarity. Zhang et al. (2020) described the FastText-based Sentence-LDA model. Specifically, cosine-based similar words from FastText are integrated into Sentence-LDA (Jo and Alice 2011), which relies on the idea that all words in a single sentence are generated from one topic, thus producing significant improvements in topic modelling over short texts.

On the other hand, according to the most commonly used architectures of deep-learning models for text classification (Minaee et al. 2021), pre-trained word embeddings tend to be explored by the following categories of neural networks: recurrent neural networks (RNNs), convolutional neural networks (CNNs), siamese neural networks (SNNs), and capsule networks. First, one of the most popular RNN-based models, which regard the text as a sequence of lexical structures, is long short-term memory (LSTM), which was designed to better capture long-term word dependencies. Indeed, Pylieva et al. (2019) tested several RNN architectures to identify French medical words that are difficult to be understood by non-expert users. They found that adding FastText embeddings to the set of features substantially improves the performance of LSTM. Demotte et al. (2020) demonstrated that the sentiment analysis of Sinhala news comments performs better when sentence-state LSTM (Zhang et al. 2018) is trained with FastText embeddings. Second, many studies have also focused on CNN-based models, which are trained to recognize patterns in text. Smetanin and Komarov (2019) employed Word2Vec embeddings as the input of a CNN architecture for the sentiment analysis of product reviews in Russian. Kulkarni et al. (2021) performed several experiments to evaluate the classification of Marathi texts using FastText embeddings in conjunction with deep-learning models such as CNN, LSTM, and BERT. They found that CNN and LSTM coupled with FastText embeddings perform on par with BERT, which is computationally more complex. Third, SNNs are usually exploited to compute semantic textual similarity in NLP. For example, De Souza et al. (2019) trained an SNN architecture with Word2Vec embeddings and a set of lexical, semantic, and distributional features to perform semantic textual similarity in Portuguese texts. Finally, capsule networks, which have shown great performance in image recognition, deal with the information-loss problem suffered by the pooling operations of CNNs. Goldani et al. (2021) employed Word2Vec embeddings as the input to capsule networks to detect fake news in short news items.

2.2 Word associations

2.2.1 Measuring word associations

The measures of semantic similarity and relatedness in NLP have been devised from a knowledge- and/or corpus-based model. In this section, we focus on the variety of methods that leverage knowledge bases, word embeddings, or both of them to measure the semantic association between words.

First, the knowledge-based model is aimed at computing semantic associations from the information stored in lexical knowledge bases, where WordNet (Fellbaum 1998) has become the most commonly used resource. In particular, this model primarily relies on the structure of ontologies or semantic networks (i.e. topology-based methods), the definitions

of words (i.e. gloss-based methods), or the vectors that encode lexical meanings. On the one hand, topology-based methods deal with the path distance between words (Rada et al. 1989; Wu and Palmer 1994; Leacock and Chodorow 1998; Li et al. 2003; Pedersen et al. 2007) and/or the information content (IC) of words (Resnik 1995; Lin 1998; Jiang and Conrath 1997; Seco et al. 2004; Zhou et al. 2008; Jiang et al. 2017). In topology-based methods, the knowledge base is considered as a graph, where word senses are nodes and semantic relations are edges. According to Rada et al. (1989), if *A* and *B* are two concepts represented by the nodes *a* and *b*, respectively, then *distance(A, B)* returns the minimum number of edges that separate *a* and *b*. In this context, Wu and Palmer (1994) introduced the notion of the Least Common Subsumer (LCS), which is the lowest concept shared by two given concepts in an ontology. In IC-based methods, the association between two words is determined by the IC that both words have in common. Most of these methods are grounded on Resnik's (1995) notion of IC, which is based on the number of occurrences of words in a corpus and the number of senses of words in the ontology. Moreover, IC takes into consideration the IS-A hierarchy; in particular, two words are semantically associated in proportion to the amount of information that is shared, which is determined by the IC of the LCS. Therefore, the standard method to measure the IC of words consists in combining the knowledge of the hierarchical structure of an ontology with the statistics about the real use of words in a corpus. It should be noted, however, that some researchers, e.g. Seco et al. (2004) and Zhou et al. (2008), managed to compute the IC without recourse to corpora. On the other hand, gloss-based methods (Lesk 1986; Banerjee and Pedersen 2003) primarily rely on the definitions of words. Lesk (1986) proposed computing word associations through the overlap between the definitions or glosses of words, on the assumption that the words that frequently co-occur in linguistic realizations are semantically related because they are used together to convey a particular idea. Banerjee and Pedersen (2003) extended Lesk's algorithm by including neighbouring words found in the glosses of related meanings. Finally, vector-based methods are aimed at representing the meaning of words as vectors derived from the relational information in the graph-based representation of the knowledge base. Patwardhan (2003) presented a measure of semantic relatedness based on gloss vectors, i.e. context vectors constructed from WordNet glosses and augmented using WordNet relations. Therefore, the semantic relatedness of two words is simply the cosine similarity between their normalized gloss vectors. Agirre and Agirre and Soroa (2009) applied a random-walk algorithm based on Personalized PageRank to WordNet, where each word was finally represented as a vector in a multi-dimensional conceptual space, with one dimension for each concept in WordNet. Goikoetxea et al. (2015) also employed random walks based on PageRank over WordNet, thus creating synthetic contexts for words. The corpus of such pseudo-sentences was then fed into Word2Vec to create word embeddings. In this context, researchers such as Tang et al. (2015) and Grover and Leskovec (2016) also explored how to compress the structural information of large semantic networks into a few hundred dimensions representing latent semantic features.

Second, the corpus-based model of semantic similarity and relatedness is inspired by distributional semantics, where one of the latest approaches is based on neural networks (Sect. 2.1.1). In this case, semantic associations are quantified as the spatial distance between the embeddings of two words through the cosine coefficient. It should be noted that the vector-space model is not able to discriminate among different meanings of a word, what Camacho–Collados and Pilehvar (2018 Camacho-Collados and Pilehvar (2018)) called "meaning conflation deficiency". In other words, each word type has a single word vector, so polysemy and homonymy are ignored. A solution to deal with the meaning conflation deficiency of word embeddings is to construct an independent representation for

each meaning of a given word. Such multi-sense embedding models can be generated from annotated corpora, but producing sense-annotated data on a large scale is a labour-intensive and time-consuming task. For this reason, some researchers deconflated words into specific word-sense vectors from non-annotated text documents. For example, Iacobacci et al. (2015) applied word-sense disambiguation to Wikipedia texts with BabelNet (Navigli and Ponzetto 2012) to create an annotated corpus, which was then processed with Word2Vec. Ruas et al. (2019) devised Most Suitable Sense Annotation (MSSA), an unsupervised algorithm based on WordNet that can process a collection of articles from Wikipedia to identify the synset for each word in the corpus; in the training step, they employed Word2Vec to obtain multi-sense embeddings. However, there have also been other studies where single-vector representations of word meaning have exhibited strong performance on NLP tasks (Salehi et al. 2015; Iacobacci et al. 2016; Kober et al. 2017). For example, Kober et al. (2017) demonstrated that a single vector that conflates the different senses of a polysemous word is sufficient for recovering sense-specific information and thus discriminating the meaning of a word in context in tasks such as phrase similarity and word-sense disambiguation. They concluded that additive composition helps to perform local disambiguation for any lexeme in a phrase, and thus "the act of composition contextualises or disambiguates each of the lexemes thereby making the representations of individual senses redundant" (Kober et al. (2017), p. 80).

Third, word-embedding models that complement distributional information from corpora with relational information from knowledge bases have received much attention in the last decade. Such hybrid models can be categorized into three groups. On the one hand, information fusion can take place during the construction of word embeddings, so the method jointly learns from both the corpus and the knowledge base. For example, Xu et al. (2014) introduced a method called RC-NET, which models relational and categorical knowledge from Freebase (Bollacker et al. 2008) as regularization functions, combining both types of knowledge with the original objective function in the Skip-gram architecture of Word2Vec in the training of a Wikipedia corpus. Yu and Dredze (2014) presented the Relation Constrained Model, which incorporates prior knowledge contained in WordNet and the Paraphrase Database (Ganitkevitch et al. 2013) to extend the objective function in the CBOW architecture of Word2Vec. Bollegala et al. (2016) proposed a method that uses the relational constraints provided by WordNet to regularize corpus-derived word embeddings learned by GloVe. Nguyen et al. (2016) integrated lexical contrast information (i.e. antonym-synonym distinction) into the objective function of the Skip-gram architecture of Word2Vec. On the other hand, pre-trained word embeddings can be enriched with relational information from knowledge bases in a post-processing stage. For example, Faruqui et al. (2015) applied a technique called retrofitting to fine-tune word embeddings through the structure of a knowledge graph, so that words that are connected in the semantic network become closer in the vector space. It is noteworthy to mention that several researchers experimented with different variants of retrofitting, e.g. graph-based retrofitting and skip-gram retrofitting (Kiela et al. 2015), expanded retrofitting (Speer and Lowry-Duda 2017), and functional retrofitting (Lengerich et al. 2017), among others. Rothe and Schutze (2015) created AutoExtend, a system that extends standard word embeddings to embeddings of WordNet synsets in the same space. Although the system originally focused on WordNet, it can also be used with other knowledge bases. Johansson and Pina (2015) constructed sense vectors by embedding the graph structure of a semantic network into the corpus word space based on the assumption that (a) the embeddings of polysemous words can be decomposed into a convex combination of sense embeddings, and (b) these sense embeddings should preserve the structure of the semantic network; indeed, these two assumptions constitute an

optimization problem, where the first is a constraint and the second is the objective. Mrkšić et al. (2017) presented the Attract-Repel algorithm, which injects synonymy and antonymy constraints from mono- and cross-lingual resources to yield specialized vector spaces, thus improving their ability to capture semantic similarity. Pilehvar and Collier (2017) proposed a technique that exploits lexical resources to expand the vocabulary of pre-trained word embeddings, which is very useful to infer the meaning of infrequent domain-specific terms. In particular, Personalized PageRank (Haveliwala 2002) can process lexical resources to extract a set of semantic landmarks, which are employed to place rare words in the most significant region of the semantic space. Finally, there are some models (e.g. Goikoetxea et al. 2016) that combine word embeddings learned independently from different types of sources, i.e. corpus and knowledge base.

2.2.2 Evaluating word associations

In recent years, there has been a revival of interest in the research of word-vector models together with word associations in fields such as NLP and psycholinguistics, which view the issue from different but complementary perspectives. On the one hand, the high-quality vector representation of words is extremely important for many NLP tasks that can be improved by using word-embedding similarities, e.g. in text summarization (Gross et al. 2016) or information retrieval (El Mahdaouy et al. 2018), among others. Moreover, various evaluation methods have been proposed to test the quality and coherence of a given vector-space model, where word similarity and relatedness tests are currently the most popular (and computationally inexpensive) methods (Pilehvar and Camacho-Collados 2020). In this regard, the semantic proximity of two words in a vector-space model is evaluated against the actual distance derived from human judgements. Typically, a set of word pairs is ranked according to the cosine-similarity scores computed through word vectors, and then the correlation with the ratings of human annotators is measured (e.g. Spearman's and/or Pearson's correlation coefficients). The best model is the one that comes closest to human ratings. In this context, a large number of studies on testing word associations through embeddings have been conducted. For example, Cattle and Ma (2017) undertook some incipient research into cosine similarities derived from Word2Vec and GloVe to predict associative strengths in word-association norms. However, in all of these studies, research results are reported using evaluation measures that do not focus on the strengths.

On the other hand, the relevance of word embeddings in psycholinguistics is recently reflected in works such as Günther et al. (2016), who concluded that lexical priming effects can be predicted from distributional semantics models (e.g. LSA and HAL), or Bhatia (2017), who demonstrated that pre-trained vector representations based on techniques such as Word2Vec and GloVe can predict the associations involved in a large range of judgement problems. After conducting several experiments with word similarity and relatedness tests, (Gladkova and Drozd 2016, 2016: p. 38) stated that they did not know "to what extent word embeddings are cognitively plausible, but they do offer a new way to represent meaning that goes beyond symbolic approaches". In this regard, (Mandera et al. 2017, 2017: p. 57) suggested that the learning mechanisms of neural-network models might resemble how humans learn the meaning of words, so "these models bridge the gap between traditional approaches to distributional semantics and psychologically plausible learning principles". To this end, they compared the performance of predictive models with that of the methods currently used in psycholinguistics, performing a variety of experiments involving not only word association norms but also semantic similarity and relatedness ratings. In line with

previous findings (Baroni et al. 2014; Levy and Goldberg 2014), they demonstrated that predictive models were generally superior to count models.

Finally, another psycholinguistic study that influenced our research was De Deyne et al. (2016), who suggested that, when people judge word similarity, they may be relying more on networks of semantic associations than on statistics calculated from the distributional patterns of words, thus drawing on Taylor's (2012) distinction between external and internal language models. On the one hand, an external language model (e.g. word embeddings generated from text corpora) treats language as an "external" object consisting of all the utterances made in a given speech community. On the other hand, an internal language model (e.g. a network of semantic associations) sees language as the body of knowledge residing in the brains of its speakers. De Deyne et al. (2016) relied on the idea that word associations capture representations that cannot be reflected in the distributional properties of an external language model, which is shaped by pragmatic and communicative considerations. In other words:

> word associations are not merely propositional but tap directly into the semantic information of the mental lexicon [...]. They are considered to be free from pragmatics or the intent to communicate some organized discourse, and thought to be simply the expression of thought. (De Deyne et al. (2015), p. 1646)

For example, *yellow* is strongly associated with *banana*, but the two words rarely co-occur in discourse because most bananas are yellow, so mentioning *yellow* together with *banana* is uninformative. In their experiments, they used several standard datasets of word similarity and relatedness to evaluate external language models constructed from text corpora and internal language models constructed from a semantic graph derived from the English Small World of Words (SWOW-EN) De Deyne et al. (2019), consisting of over 12,000 cue words and 300 associations for each cue resulting from judgements from over 90,000 participants. They showed, for example, that an internal language model grounded on Word2Vec embeddings substantially outperformed an external language model grounded on a random-walk semantic graph. However, the superior performance of this internal language model is unsurprising: the model was constructed from data derived from free-association tasks and then compared with human judgements on word associations, inevitably resulting in a biased evaluation.

2.3 Ensemble application of symbolic and sub-symbolic approaches to natural language processing

For several decades, semantic systems have been predominantly developed around knowledge graphs (e.g. semantic networks and ontologies), which usually store logically sound structured representations of manually encoded knowledge. In the last decade, sub-symbolic artificial intelligence, which typically relies on some form of automatic learning from numerical, statistical or distributed data by machine-learning or neural-network models, has also become a mainstream area of research. Indeed, most of the current research in artificial intelligence is sub-symbolic, where neural language models aimed at exploring large amounts of data to make categorizations and predictions, e.g. ELMo (Peters et al. 2018), BERT (Devlin et al. 2019) and GPT-2 (Radford et al. 2019), among others, have revolutionized the field of NLP. It should be noted, however, that transforming lexical items into numbers enables us to discover hidden patterns in data but does not provide much information about the items themselves. Advances in real-world natural language understanding

applications should be grounded on hybrid systems that combine large-scale symbolic representations of knowledge with sub-symbolic methods. As explained by Gomez-Perez et al. (2020), the combination of symbolic and sub-symbolic approaches will be critical for the next leap forward in NLP, where language models capture how sentences are constructed and knowledge graphs contain a conceptualization of the entities and relations in a given domain. In this context, our research focuses on the word-embedding enrichment resulting from the combination of distributional information from corpora and relational information from knowledge bases. As word embeddings have been lately explored by deep-learning language models (Sect. 2.1.3), the remainder of this section presents the most recent efforts in enhancing language models with external knowledge for a variety of NLP tasks.

In text classification, Zhang et al. (2019) and Ostendorff et al. (2019) enhanced BERT with Wikidata embeddings (Vrandecic and Krotzsch 2014), and Meng et al. (2019) improved classification accuracy when semantic information from DBpedia (Bizer et al. 2009) was used with a multi-level CNN. In zero-shot text classification, where the model can detect classes that are not included in the training dataset, Liu et al. (2020a) employed the category knowledge from ConceptNet (Speer and Lowry-Duda 2017) to construct semantic connections between the seen and unseen classes, so that a CNN could classify the unseen classes by information propagation over the connections.

In story generation, some researchers demonstrated that common-sense knowledge can contribute to generating more coherent texts. Yang et al. (2019a) devised a memory-augmented neural model with adversarial training to incorporate knowledge from ConceptNet into an automatic topic-to-essay generation system. Guan et al. (2020) proposed a knowledge-enhanced pre-training model for story generation by extending GPT-2 with knowledge from ConceptNet and ATOMIC (Sap et al. 2019). Yang and Tiddi (2020) developed a story-generation system named DICE, which injects knowledge from ConceptNet, WordNet, and DBpedia into a GPT-2 model.

In machine reading comprehension, Mihaylov and Frank (2018) employed WordNet and ConceptNet to enrich text representations, which were learned by a Bi-directional Gated Recurrent Unit to infer the answer of common-noun and named-entity questions. Wang and Jiang (2018) proposed Knowledge Aided Reader, which relies on the general knowledge extracted from passage-question pairs with the aid of WordNet to assist the attention mechanisms of a bidirectional LSTM model. Yang et al. (2019b) introduced KT-NET, which employs an attention mechanism to select knowledge from WordNet and NELL (Carlson et al. 2010) and then injects the selected knowledge into BERT to enable context- and knowledge-aware predictions. Gong et al. (2020) proposed KCF-NET, a system that employs a BERT embedding layer containing two encoding methods that compute the context-aware representation and the knowledge-graph representation of the input text, respectively, and then a fusion layer that integrates context information with external knowledge.

In question answering, Goodwin and Demner-Fushman (2020) presented OSCR (Ontology-based Semantic Composition Regularization), which can inject world knowledge from Wikipedia into BERT during pre-training to improve the performance of the system. Similarly, Phan and Do (2020) combined BERT with a knowledge graph to enhance a Vietnamese question-answering system about tourism.

In text summarization, Gunel et al. (2020) injected entity-level knowledge from Wikidata into a Transformer-XL encoder-decoder Dai et al. (2019) to enhance abstractive summaries.

The above examples serve to illustrate that top-down knowledge derived from semantic networks and ontologies can effectively be combined or integrated with bottom-up knowledge learned from text documents through neural networks, leading to a breakthrough in

natural language understanding. Finally, a different case of the synergy of symbolic and sub-symbolic approaches can be found in Cambria et al. (2020), who integrated logical reasoning within deep learning architectures (i.e. bidirectional LSTM and BERT) to build SenticNet.

3 Proposed method

3.1 Combining word embeddings

In line with Taylor's (2012) distinction between external and internal language models, there are two approaches to represent lexical semantics that have been instrumental for major advances in language technology, even though they were primarily motivated by psycholinguistic research. On the one hand, the semantic-space approach represents the meaning of a lexical unit through a vector in a high-dimensional space, where each component is generated on the co-occurrence with the other units in contexts of language usage. On the other hand, the semantic-network approach represents the meaning of a lexical unit within a graph, whose nodes represent words and edges between nodes encode different types of semantic relations holding among lexical units (e.g. synonym, hyponym, meronym, etc.). In this context, one of the goals of this research is to combine both approaches by integrating embeddings derived from text corpora with embeddings derived from a semantic network. Corpus-based embeddings represent a semantic space based on an external language model, namely a collection of texts that were produced by English-language speakers. In turn, network-based embeddings represent a semantic space based on an internal language model, thus being closely aligned with the lexical knowledge in the minds of speakers. The rationale behind this decision is that the complementarity of both approaches can help us determine word associations that, for example, are rarely or never evidenced in relevant context windows in the text collection but are likely to be encoded in a semantic network. It should be noted that addressing a semantic network as a vector-space model is just a notational issue. Indeed, as we managed to put both language models on equal grounds, we facilitated the integration with corpus-based embeddings.

To implement both approaches computationally, we chose to reuse existing language resources in the form of readily available pre-trained word vectors generated by different techniques. In this case, let $X \in \mathbb{R}^{|V|*D}$ be an embedding matrix, where V is the set of words and D is the dimensionality of the embeddings, so X_i^W is the embedding of the i-th word in the given matrix. On the one hand, we leveraged off-the-shelf deep neural embeddings to develop our corpus-based model. Indeed, we employed three types of corpus-based embeddings:

(a) X^{WV}, which contains vectors trained on part of Google News dataset (about 100 billion words) using Word2Vec,[2] where $|V^{WV}|$ is 3 million lexical units and D is 300,
(b) X^{GV}, which contains vectors trained on English Common Crawl Corpus using GloVe,[3] where $|V^{GV}|$ is 2 million words and D is 300, and

[2] The word embeddings were downloaded from https://code.google.com/archive/p/word2vec/.
[3] The word embeddings were downloaded from http://vectors.nlpl.eu/repository.

(c) X^{FT}, which contains vectors trained on English Common Crawl Corpus and Wikipedia using FastText,[4] where $|V^{FT}|$ is 2 million words and D is 300. This model was trained using CBOW with character n-grams of length 5, a window of size 5 and 10 negatives (Grave et al. 2018).

On the other hand, we also used X^{WN}, containing word embeddings trained on the WordNet semantic graph, where the strength of the semantic association between words was determined based on the following premise: the larger the number of paths and the shorter the paths connecting any two nodes, the stronger their association (Saedi et al. 2018).[5] The original WordNet-based embedding matrix (WNet2Vec) was finally obtained by extracting a subgraph containing 60,000 words that supported all parts of speech and all types of semantic relations, where each relation was assigned the same weight.[6] As a result, the lexical knowledge encoded in the semantic graph was re-encoded as a word-embedding matrix. We reduced the 850 dimensions of WNet2Vec to 300 through PCA so that network-based embeddings could be easily integrated with the above corpus-based embeddings. After dimensionality reduction, word embeddings in WNet2Vec were unit-length normalized.

Finally, together with these resources, we devised WALE (**W**ord **A**ssociation through mu**L**tiple **E**mbeddings), a parametric model that allows two views (i.e. WALE-1 and WALE-2) to calculate the association strength of two words (i.e. cue and target) based on the combination of two word-embedding matrices: the corpus-based matrix (X^C, which can take the form of X^{WV}, X^{GV}, or X^{FT}) and the network-based matrix (X^{WN}). Equation 2 and Equation 3 are used to calculate WALE-1 and WALE-2, respectively, where α and β are parameters, being $\alpha + \beta = 1$, and $distance[X](cue, target)$ calculates the cosine distance between the embeddings corresponding to the cue and target words in the matrix X.

$$\begin{aligned} &WALE\text{-}1(cue, target) = 1 - distance[X^{C',WN'}](cue, target), \\ &\text{where } X^{C',WN'} \text{ results from } \sum_{k}^{|V^{C'}|} (\alpha * X_k^{C'}) + (\beta * X_k^{WN'}) \end{aligned} \tag{2}$$

$$\begin{aligned} WALE\text{-}2(cue, target) = &(\alpha * (1 - distance[X^{C'}](cue, target))) \\ &+ (\beta * (1 - distance[X^{WN'}](cue, target))) \end{aligned} \tag{3}$$

To facilitate the combination between X^C and X^{WN}, we only took into consideration the unigrams that were found in $V^{WV} \cap V^{GV} \cap V^{FT} \cap V^{WN}$ and that fell into the POS categories of noun, verb, or adjective, where named entities were discarded. As a result, both X^C and X^{WN} were reduced to $X^{C'}$ and $X^{WN'}$, respectively, each one consisting of 18,475 lemmas with their corresponding embeddings.

WALE-1 and WALE-2 mainly result from the convergence of two factors: (a) how to integrate the semantic-space approach (i.e. external language model) with the

[4] The word embeddings were downloaded from https://fasttext.cc/docs/en/crawl-vectors.html.

[5] The word embeddings were downloaded from https://github.com/nlx-group/WordNetEmbeddings.

[6] Saedi et al. (2018) also ran an experiment where different weights were assigned to different relations: hypernymy, hyponymy, antonymy and synonymy got 1, meronymy and holonymy 0.8, and other relations 0.5. However, better results were obtained when the same weight was assigned to all types of semantic relation.

semantic-network approach (i.e. internal language model), and (b) how to combine the word-embedding matrices (i.e. single or double vector-space model). Suppose that we want to determine the association strength between *car* and *vehicle* as cue and target words, respectively, and that, for the sake of simplicity, we assume that the corpus- and network-based vectors corresponding to these words are as follows:

$$X_{car}^{C'} = \left[0.63\ \ 0.32\ \ 0.56\ \ 0.48\right] \tag{4}$$

$$X_{vehicle}^{C'} = \left[0.87\ \ 0.65\ \ 0.24\ \ 0.31\right] \tag{5}$$

$$X_{car}^{WN'} = \left[0.75\ \ 0.22\ \ 0.45\ \ 0.51\right] \tag{6}$$

$$X_{vehicle}^{WN'} = \left[0.71\ \ 0.57\ \ 0.31\ \ 0.43\right] \tag{7}$$

On the one hand, with regard to (a), we can assign relative weights to $X^{C'}$ and $X^{WN'}$ to explore the impact of each type of approach on the performance of the system. In this regard, we use the parameters α and β in conjunction with $X^{C'}$ and $X^{WN'}$, respectively. For example, suppose that we intend to give more weight to the semantic representations constructed from the corpus rather than to those derived from the semantic network. In this case, we could choose 0.7 and 0.3 for α and β, respectively. On the other hand, with regard to (b), we can consider integrating $X^{C'}$ and $X^{WN'}$ into a single or double vector-space model. The single vector-space model consists in ensembling the word embeddings in $X^{C'}$ with those in $X^{WN'}$ to create a new $X^{C',WN'}$ so that we can compute a single similarity coefficient between the meta-embedding representing the cue and that of the target in $X^{C',WN'}$. Following the previous example, the meta-embeddings corresponding to *car* and *vehicle* are computed in Equation 8 and Equation 9, respectively, assuming that we set α to 0.7 and β to 0.3.

$$X_{car}^{C',WN'} = (0.7 * X_{car}^{C'}) + (0.3 * X_{car}^{WN'}) = \left[0.67\ \ 0.29\ \ 0.53\ \ 0.49\right] \tag{8}$$

$$X_{vehicle}^{C',WN'} = (0.7 * X_{vehicle}^{C'}) + (0.3 * X_{vehicle}^{WN'}) = \left[0.82\ \ 0.63\ \ 0.26\ \ 0.35\right] \tag{9}$$

In this case, the similarity between both meta-embeddings is 0.904. In contrast, the word-embeddings in $X^{C'}$ and $X^{WN'}$ are not ensembled in the double vector-space model, but we compute the weighted average of the cosine-similarity coefficients derived from the vectors corresponding to the cue and the target in each matrix. In this case, the similarity between $X_{car}^{C'}$ and $X_{vehicle}^{C'}$ is 0.88 and that between $X_{car}^{WN'}$ and $X_{vehicle}^{WN'}$ is 0.93. Therefore, the association strength between *car* and *vehicle* is calculated in this model as $(0.7 * 0.88) + (0.3 * 0.93) = 0.895$, using the same previous values for α and β.

3.2 Evaluating word associations

After more than four decades, agreement with the human ratings in a dataset of n pairs of words is usually measured using Pearson's product-moment correlation coefficient (Equation 10), and/or Spearman's rank correlation coefficient (Equation 11).

$$r = \frac{\sum_i^n (x_i - \bar{x})((y_i - \bar{y})}{\sqrt{\sum_i^n (x_i - \bar{x})^2}\sqrt{\sum_i^n (y_i - \bar{y})^2}} \tag{10}$$

$$\rho = 1 - \frac{6\sum_i^n (rank(x_i) - rank(y_i))^2}{n(n^2 - 1)} \tag{11}$$

In our case, x_i is the score computed by WALE for the word pair $< w_i, w_i' >$, y_i is the score provided by human annotators for the same pair of words, $\bar{x}$ is the mean of all values x_i, $\bar{y}$ is the mean of all values y_i, and $rank(x_i)$ and $rank(y_i)$ represent the rank value of the i-th pair of words according to the overall ranking of scores provided by WALE and human annotators, respectively. Zesch (2010) explained that Pearson's correlation suffers from some limitations: (a) it is sensitive to outliers, (b) it can only measure a linear relationship between the human-provided scores and those computed by the measure, and (c) the two variables need to be normally distributed. To overcome these limitations, he recommended using Spearman's rank correlation coefficient instead, which is the non-parametric version of Pearson's product-moment correlation coefficient. Indeed, Spearman's correlation does not use the actual values to compute a correlation but the ranking of the values. Therefore, it is not sensitive to outliers, non-linear relationships, or non-normally distributed data.

In contrast to all previous studies, we evaluated the effectiveness of a model for word associations through a measure that can take advantage of not only the rank ordering of word pairs, as in Spearman's correlation coefficient, but also the strength of associations, as with the degrees of relevance represented by human annotators in test datasets. To this end, we focused on a suite of measures that have gained much popularity in the field of information retrieval over the last decade, namely the cumulated gain-based techniques introduced by Järvelin and Kekäläinen (2000), Järvelin and Kekäläinen (2002), i.e. cumulative gain, discounted cumulative gain (DCG), and normalized discounted cumulative gain (NDCG).

In this type of techniques, a gain value must be assigned to each relevance level, where these gain values should be chosen to reflect the relative differences between the levels. Therefore, supposing that Q is a ranked list of pairs, the first step in the computation of NDCG is the construction of the gain vector G, i.e. $G_Q = \langle s_1, s_2, s_3, ..., s_k, ...s_q \rangle$, where $G[k]$ represents the score assigned to the cue-target pair at the k rank in Q, being q the total number of pairs in Q. The second step is the calculation of the cumulative-gain vector, where $CG[k]$, i.e. the value of the element k in CG, is the sum from 1 to k of the elements in G, as shown in Equation 12.

$$CG[k] = \sum_{i=1}^{k} G[i] \tag{12}$$

Before computing the cumulative-gain vector, a discount function can also be applied at each rank so that the relevance values are discounted progressively as one moves down the document ranking (i.e. the denominator in Equation 13).

$$DCG[k] = \sum_{i=1}^{k} \frac{G[i]}{log_2(1+i)} \tag{13}$$

As shown in Equation 14, the final step normalizes the DCG vector against the "ideal" DCG vector (DCG'), which is constructed from the ideal gain vector G', containing the scores from the ordering of the word pairs in a gold-standard benchmark.

Table 1 Sample of word pairs and their association scores

Cue	Target	Test	Reference
Jazz	Music	0.564	0.367
Champagne	Bubble	0.291	0.163
Adult	Responsible	0.086	0.041
Cancer	Kill	0.488	0.020
Athlete	Player	0.103	0.014

$$NDCG[k] = \frac{DCG[k]}{DCG'[k]} \tag{14}$$

As explained by Katerenchuk and Rosenberg (2016), NDCG has some drawbacks. Indeed, two issues could have a critical impact on the results of this research. On the one hand, NDCG was originally designed for the evaluation of information-retrieval systems rather than for rank-ordering evaluation. This means that NDCG takes into consideration the number of relevant and irrelevant elements. However, virtually all cue-target pairs involved in word-association tasks are relevant elements to a certain degree. As a result, the lower bound is rarely equal to 0, so this measure would return a value whose range is from 1 to some arbitrary number between 1 and 0. This could mean that a score such as 0.56 might be returned by the worst ordering, which can lead us to misinterpret the results. On the other hand, the discount function in DCG was originally designed to reward relevant search results when they appear close to the top. However, the rank-ordering problem needs a relative function with respect to the remaining elements. Otherwise, a strong bias towards top-ranked elements can be introduced. To address both issues, Katerenchuk and Rosenberg (2016) modified NDCG to design RankDCG, which not only outperforms conventional rank-ordering measures but also correctly handles multiple ties and produces a consistent and meaningful scoring range [0, 1], among many other advantages.[7]

To illustrate RankDCG, which can be used with any number of elements, we take the pairs of words in Table 1, which is supposed to contain the scores computed by our system and the reference scores in a gold standard.

Therefore, the ideal gain vector *G'* and the gain vector *G* computed by the model are as follows, where subscripts represent the zero-based position in the gold-standard ranking:

$$G = \langle 0.564_0, 0.291_1, 0.086_2, 0.488_3, 0.103_4 \rangle \tag{15}$$

$$G' = \langle 0.367_0, 0.163_1, 0.041_2, 0.020_3, 0.014_4 \rangle \tag{16}$$

First of all, the values in *G* and *G'* are transformed into integers through a mapping function *R*. In this step, and unlike the original formulation of the measure, we can decide to make RankDCG take into consideration (a) rank ordering only or (b) both rank ordering and association strength. In particular, the function *R* assigns a rank-based number to every score in option (a) and rescales the scores from 5 to 1,000 (i.e. min-max normalization) in option (b). In the case of (a), after arranging the elements of *G* and *G'* in descending order, the top-rank element in each vector is mapped to the highest value, and then every

[7] The original RankDCG code can be found in https://github.com/dkaterenchuk/ranking_measures.

following distinct element is mapped to a value decreased by one (except with tie scores), until the last element corresponds to 1. Therefore, the function *R* is applied to *G* and *G'* according to these mappings, returning the *D* and *D'* vectors, respectively:

$$D = \langle 5_0, 3_1, 1_2, 4_3, 2_4 \rangle \tag{17}$$

$$D' = \langle 5_0, 4_1, 3_2, 2_3, 1_4 \rangle \tag{18}$$

In the case of (b), the function *R* rescales the scores in *G* and *G'*, returning the following vectors:

$$D = \langle 1000_0, 431_1, 5_2, 841_3, 40_4 \rangle \tag{19}$$

$$D' = \langle 1000_0, 424_1, 81_2, 21_3, 5_4 \rangle \tag{20}$$

For the sake of brevity and clarity, suppose that we opt for (a) in our example. In the next step, the function R_{rev} is applied to D_{rev} and D'_{rev} to reverse the order of the elements:

$$D_{rev} = \langle 2_0, 4_1, 1_2, 3_3, 5_4 \rangle \tag{21}$$

$$D'_{rev} = \langle 1_0, 2_1, 3_2, 4_3, 5_4 \rangle \tag{22}$$

In RankDCG, the *DCG* component is computed by Equation 23.

$$DCG''[k] = \sum_{i=1}^{k} \frac{E'[i]}{D'_{rev}[i]} \tag{23}$$

In this case, the vector *E'* is constructed in two steps. First, the elements in the D_{rev} vector are arranged in descending order, but their subscript values are retained:

$$E = \langle 5_4, 4_1, 3_3, 2_0, 1_2 \rangle \tag{24}$$

Second, the elements in D'_{rev} are rearranged according to the order of the subscripts in *E*:

$$E' = \langle 5_4, 2_1, 4_3, 1_0, 3_2 \rangle \tag{25}$$

As a result, the *DCG"* vector for our example is as follows:

$$DCG'' = \langle 5, 6, 7.33, 7.58, 8.18 \rangle \tag{26}$$

Finally, $DCG''[q]$ should be normalized from 0 to 1 to create a meaningful and consistent lower bound (Equation 27), where $max(DCG''[q])$ is computed using the perfect-case ordering, i.e. $D = D'$, and $min(DCG''[q])$ is computed using the worst-case ordering, i.e. $D = D'_{rev}$.

$$RankDCG = \frac{DCG''[q] - min(DCG''[q])}{max(DCG''[q]) - min(DCG''[q])} \tag{27}$$

In our example, where the value of $DCG''[q]$ is 8.18, the final result is computed as follows:

$$max(DCG'') = \langle 5, 7, 8, 8.5, 8.7 \rangle \tag{28}$$

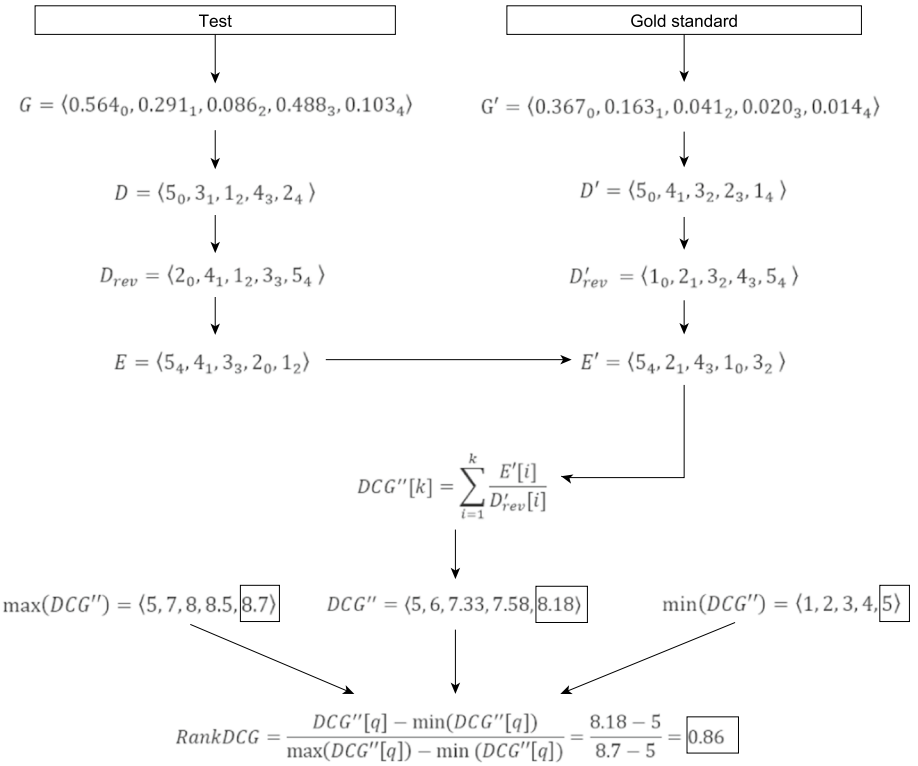

Fig. 2 Description of RankDCG: an example

Table 2 Sample of a group of word pairs

Cue	Target	Score
Accident	Car	0.358
Accident	Crash	0.128
Accident	Pain	0.020
Accident	Danger	0.014

$$min(DCG'') = \langle 1, 2, 3, 4, 5 \rangle \tag{29}$$

$$RankDCG = \frac{8.18 - 5}{8.7 - 5} = 0.86 \tag{30}$$

In contrast, if we had taken into consideration both rank ordering and association strength in *G* and *G'*, the RankDCG coefficient would have been 0.93. In both cases, the closer to 1 the coefficient, the better the performance of the model. To conclude, Fig. 2 illustrates the whole process of RankDCG.

Moreover, another difference concerning the state of the art lies in the method of evaluation. Apart from applying the above measures to a whole list of word pairs, we also

performed independent comparisons of score rankings for multiple groups of pairs. In this context, we define "group" as a set of cue-target word pairs that share the same cue, as illustrated in Table 2.

This approach is motivated by the fact that participants in free-association experiments are usually asked to produce only a single associate for each word, but the databases show the aggregated results of many participants, so free associations do not provide an absolute index of strength but a relative index. Indeed, Nelson et al. (1998) exemplified this limitation as follows:

> Knowing that the response "read" is produced by 43% of the participants to the cue BOOK does not tell us how strong this response is in any absolute sense; it tells us only that this response is stronger than "study" which was produced by 5.5% of the participants. Unfortunately, free association norms like relatedness ratings provide only ordinal measures of strength of association but, as far as we know, there are no known measures of absolute strength.

Therefore, for a group-based evaluation, the RankDCG score of the model is calculated with Equation 31, where k is the number of groups in the test dataset Q, and $RankDCG_{G_j}$ is the RankDCG score corresponding to the group G_j, which should be part of Q.

$$AvgRankDCG = \frac{\sum_j^k RankDCG_{G_j} | G_j \in Q}{k} \tag{31}$$

3.3 Computational implementation

WALE has been computationally implemented as a web interface, developed in C# with ASP.NET 4.0, where the user can explore WALE-1 and WALE-2 by computing the associative strength of the word pairs in any of the ten gold-standard benchmarks for word similarity and relatedness (Faruqui and Dyer 2014).[8] Indeed, this application also allows researchers to conduct experiments with their datasets. Moreover, providing that the pairs of words are accompanied with reference scores (e.g. the ratings of human annotators), researchers can evaluate the effectiveness of the model through Spearman's and Pearson's correlation coefficients as well as RankDCG, taking into consideration only rank ordering or also the associative strength.

4 Experiments

We conducted a suite of experiments to examine the performance of WALE with different types of word associations. Following (Faruqui and Dyer 2014), we employed ten gold-standard benchmarks that have been widely used to prove the effectiveness of word vectors: RG (Rubenstein and Goodenough 1965), MC (Miller and Charles 1991), WS-ALL (Finkelstein et al. 2001), YP (Yang and Powers 2006), WS-SIM, WS-REL (Agirre et al. 2009), MTurk-287 (Radinsky et al. 2011), MTurk-771 (Halawi et al. 2012), MEN (Bruni

[8] WALE is freely accessible from the FunGramKB website: http://www.fungramkb.com/nlp.aspx.

et al. 2012), and RW (Luong et al. 2013).[9] These datasets are oriented to word similarity (i.e. RG, MC, WS-SIM, and RW) and word relatedness (i.e. WS-ALL, YP, WS-REL, MTurk-287, MTurk-771, and MEN), where the latter can contain syntagmatically and paradigmatically related words. RG, MC, WS-SIM, and WS-REL contain only nouns and YP only verbs, whereas MTurk-287, RW, WS-ALL, MTurk-771, and MEN include all kinds of words, although nouns predominate. Finally, whereas datasets such as MC, RG, and WS-ALL contain very frequent words, RW has a more diverse set of words in terms of frequencies, having the largest number of rare words.

It should be noted that the words in the above datasets may or may not be associates. For this reason, we also experimented with University of South Florida Free Association Norms (FAN),[10] which contains pairs of words where cue and target are meaningfully associated, although they may or may not be semantically related. It should be recalled that the traditional way to collect word-association norms in psycholinguistic research is to present a word to several people (i.e. the stimulus) and ask them to express the first word that comes to their minds upon receiving the stimulus (i.e. the response). FAN (Nelson et al. 1998) contains 63,619 cue-target word pairs that have been normed, where we make use of the Forward Cue-to-Target Strength score. The word-association norms resulted from an experiment in which more than 6,000 participants, who produced nearly three-quarters of a million responses to 5019 stimulus words, were involved in a discrete association task. In particular, participants were asked to write the first word that came to mind that was meaningfully connected or strongly associated with a given word. The great majority of the stimulus words are nouns, but adjectives, verbs and other POS can also be found. There was not a well-designed purpose in the choice of these stimulus words. It is noteworthy to mention that there are other collections of word association norms, such as Edinburgh Associative Thesaurus (EAT)[11] and SWOW-EN.[12] However, we chose to focus only on FAN because the methodology of a given resource undoubtedly affects the type of responses that participants can generate. In particular, whereas participants in SWOW-EN were asked to respond with the first three words that came to mind in the broadest possible sense, and those in EAT were asked to write down for each cue the first word they could think of as quickly as possible, participants in FAN were asked to write down the first word that came to mind that was "meaningfully related or strongly associated to the presented cue word".

The goal of our experiments was to assess the significance of several factors using the above test datasets, such as the word-embedding technique (i.e. Word2Vec, Glove, and FastText), the model for the projection of distinct word-embedding matrices (i.e. single or double vector-space model, that is, WALE-1 or WALE-2, respectively), the degree of integration of external and internal language models (i.e. the parameters α and β in WALE, respectively), the evaluation measure (i.e. Spearman's and Pearson's correlation coefficients and RankDCG), and the dataset size. To conduct these experiments, we had to make $X^{WV'}$, $X^{GV'}$, $X^{FT'}$ and $X^{WN'}$ share the same vocabulary, i.e. 18,475 lemmas, so we also had to reduce the size of the above datasets to include only valid words. Moreover, for group-based evaluation, all pairs in FAN that (a) could not be grouped around a common cue or (b) had the same score with other pairs in the same group were further discarded. As we

[9] These datasets were downloaded from https://github.com/mfaruqui/word-vector-demo/tree/master/data.
[10] http://w3.usf.edu/FreeAssociation/.
[11] http://rali.iro.umontreal.ca/rali/?q=en/Textual%20Resources/EAT.
[12] https://smallworldofwords.org.

Table 3 Size of test datasets

Dataset	Original	Test	Coverage (%)
MC	30	30	100
YP	130	47	36.15
RG	65	65	100
MTurk-287	287	76	26.48
WS-SIM	203	192	94.58
WS-REL	252	233	92.46
RW	2034	276	13.57
WS-ALL	353	328	92.92
MTurk-771	771	769	99.74
MEN	3000	1592	53.07
FAN	63,619	17,204	27.04

Table 4 Evaluation with Spearman's correlation coefficient (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
MC	.84 (.4–.6)	.84 (.8–.2)	.80 (.2–.8)	.83 (.8–.2)	**.85 (.9–.1)**	**.85 (.9–.1)**
YP	**.77 (0–1)**	**.77 (0–1)**	**.77 (0–1)**	**.77 (0–1)**	**.77 (0–1)**	**.77 (0–1)**
RG	.81 (.4–.6)	.83 (.6–.4)	.84 (.2–.8)	.86 (.8–.2)	.86 (.8–.2)	**.87 (.8–.2)**
MTurk-287	.76 (.6–.4)	.78 (.7–.3)	.75 (.2–.8)	.75 (.7–.3)	**.85 (1–0)**	**.85 (.9–.1)**
WS-SIM	.78 (.4–.6)	.80 (.7–.3)	.78 (.2–.8)	.79 (.6–.4)	.84 (.8–.2)	**.85 (.8–.2)**
WS-REL	.62 (.7–.3)	.64 (.8–.2)	.64 (.3–.7)	.65 (.8–.2)	.73 (.9–.1)	**.74 (.9–.1)**
RW	.56 (.5–.5)	.57 (.9–.1)	.52 (.2–.8)	.52 (.7–.3)	**.60 (.9–.1)**	**.60 (.9–.1)**
WS-ALL	.70 (.4–.6)	.72 (.8–.2)	.70 (.2–.8)	.72 (.7–.3)	.78 (.8–.2)	**.79 (.9–.1)**
MTurk-771	.70 (.4–.6)	.72 (.8–.2)	.73 (.2–.8)	.75 (.8–.2)	.75 (.9–.1)	**.77 (.8–.2)**
MEN	.78 (.5–.5)	.78 (.8–.2)	.76 (.3–.7)	.77 (.8–.2)	**.84 (1–0)**	**.84 (.9–.1)**
FAN	.32 (.4–.6)	.32 (.8–.2)	.30 (.2–.8)	.30 (.7–.3)	**.36 (.7–.3)**	**.36 (.8–.2)**

aim to compare the pairs of words within a given group, each pair should be unique in the score for that group. Table 3 shows the size of each test dataset.

5 Results

First, we evaluated WALE with Word2Vec, Glove, and FastText and with all test datasets. Tables 4, 5, 6, and 7 show the results returned by Spearman's correlation coefficient, Pearson's correlation coefficient, RankDCG' (only rank ordering), and RankDCG" (rank ordering together with association strength), respectively. The values within round brackets refer to the weighting factors of the parameters α and β in WALE (Equation 2 and Equation 3), where α represents the factor for the corpus-derived embeddings and β is the factor for the WordNet-derived embeddings.

Second, we conducted a group-based evaluation with FAN. Tables 8 and 9 show the results with averaged RankDCG' and averaged RankDCG", respectively.

Table 5 Evaluation with Pearson's correlation coefficient (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
MC	.83 (.4–.6)	.83 (.8–.2)	.82 (.2–.8)	.83 (.8–.2)	**.84 (.9–.1)**	**.84 (.9–.1)**
YP	.84 (0–1)	.84 (.2–.8)	.84 (.1–.9)	.84 (.2–.8)	.84 (0–1)	**.85 (.3–.7)**
RG	.81 (.3–.7)	.82 (.7–.3)	.83 (.2–.8)	.84 (.7–.3)	**.86 (.7–.3)**	**.86 (.8–.2)**
MTurk-287	.72 (.5–.5)	.72 (.8–.2)	.74 (.3–.7)	.74 (.8–.2)	**.80 (1–0)**	**.80 (1–0)**
WS-SIM	.78 (.4–.6)	.79 (.8–.2)	.78 (.2–.8)	.79 (.7–.3)	**.84 (.8–.2)**	**.84 (.9–.1)**
WS-REL	.59 (.5–.5)	.60 (.8–.2)	.66 (.3–.7)	.67 (.8–.2)	**.72 (.9–.1)**	**.72 (.9–.1)**
RW	.51 (.4–.6)	.53 (.8–.2)	.49 (.2–.8)	.49 (.7–.3)	.57 (.9–.1)	**.58 (.9–.1)**
WS-ALL	.65 (.5–.5)	.67 (.8–.2)	.69 (.2–.8)	.70 (.8–.2)	**.75 (.9–.1)**	**.75 (.9–.1)**
MTurk-771	.69 (.4–.6)	.70 (.7–.3)	.73 (.2–.8)	.74 (.8–.2)	.73 (.8–.2)	**.75 (.8–.2)**
MEN	.76 (.5–.5)	.76 (.9–.1)	.75 (.3–.7)	.76 (.8–.2)	**.82 (1–0)**	**.82 (1–0)**
FAN	.34 (.4–.6)	.34 (.8–.2)	.31 (.2–.8)	.32 (.7–.3)	**.38 (.7–.3)**	**.38 (.8–.2)**

Table 6 Evaluation with RankDCG' (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
MC	.97 (.3–.7)	.97 (.8–.2)	.95 (.2–.8)	.95 (.8–.2)	**.98 (.9–.1)**	.97 (.9–.1)
YP	.91 (.2–.8)	**.94 (.4–.6)**	.91 (0–1)	.91 (0–1)	.92 (.5–.5)	.93 (.6–.4)
RG	.94 (.1–.9)	.94 (.2–.8)	.94 (.1–.9)	.94 (.1–.9)	.94 (.6–.4)	**.95 (.4–.6)**
MTurk-287	.90 (.2–.8)	**.91 (.3–.7)**	.90 (.6–.4)	.90 (1–0)	.90 (.7–.3)	.90 (.8–.2)
WS-SIM	.92 (.3–.7)	.92 (.8–.2)	.92 (.1–.9)	**.93 (.6–.4)**	**.93 (.6–.4)**	**.93 (.6–.4)**
WS-REL	.88 (.3–.7)	.87 (.6–.4)	.89 (.3–.7)	.89 (.8–.2)	.89 (.6–.4)	**.91 (.8–.2)**
RW	.84 (.3–.7)	.83 (.9–.1)	.84 (.6–.4)	.84 (1–0)	**.85 (1–0)**	**.85 (1–0)**
WS-ALL	.91 (.3–.7)	.91 (.6–.4)	.92 (.2–.8)	.92 (.6–.4)	.92 (.8–.2)	**.93 (.8–.2)**
MTurk-771	**.90 (.3–.7)**	**.90 (.5–.5)**	.89 (.2–.8)	**.90 (.6–.4)**	.87 (.6–.4)	**.90 (.6–.4)**
MEN	.88 (.4–.6)	.88 (.7–.3)	.87 (.2–.8)	.88 (.7–.3)	**.90 (.8–.2)**	**.90 (.9–.1)**
FAN	.53 (.5–.5)	.53 (.8–.2)	.53 (.2–.8)	.53 (.8–.2)	**.56 (.7–.3)**	**.56 (.9–.1)**

Third, we evaluated eleven samples of different sizes extracted from FAN. In particular, we split FAN into five bins of about 3,500 pairs of words and, in turn, the first bin into seven other bins of about 500 pairs of words. From these groupings, we employed RankDCG to evaluate datasets of 503, 999, 1504, 2001, 2494, 3003, 3435, 6,882, 10,324, 13,759 and 17,204 pairs of words. To illustrate, Fig. 3 shows the results with FastText and WALE-2 (0.9–0.1).

Finally, we conducted an experiment that looks much like the first, but with the original 850 dimensions of X^{WN}. To illustrate, Table 10 shows the results with FastText and WALE-2. The scores that are higher or lower than the corresponding ones in Tables 4, 5, 6, and 7 (300 dimensions) have been marked in bold or italics, respectively.

Table 7 Evaluation with RankDCG" (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
MC	**.99 (.3–.7)**	.98 (.4–.6)	.98 (.2–.8)	.98 (.7–.3)	.98 (.9–.1)	.98 (.9–.1)
YP	.96 (.2–.8)	**.98 (.4–.6)**	.93 (.2–.8)	.92 (1–0)	.96 (.5–.5)	**.98 (.6–.4)**
RG	.96 (.1–.9)	.96 (.2–.8)	.96 (.1–.9)	.96 (.5–.5)	.96 (.6–.4)	**.97 (.4–.6)**
MTurk-287	.92 (.2–.8)	**.93 (.3–.7)**	**.93 (.6–.4)**	**.93 (1–0)**	.92 (.3–.7)	.92 (.2–.8)
WS-SIM	.95 (.1–.9)	.95 (.1–.9)	.95 (.1–.9)	**.96 (.6–.4)**	**.96 (.8–.2)**	**.96 (.9–.1)**
WS-REL	.91 (.3–.7)	.91 (.7–.3)	.92 (.3–.7)	.93 (.8–.2)	.92 (.6–.4)	**.94 (.8–.2)**
RW	.88 (.3–.7)	.86 (.9–.1)	**.89 (.6–.4)**	.88 (1–0)	.87 (1–0)	.87 (.9–.1)
WS-ALL	.94 (.1–.9)	.94 (.6–.4)	**.95 (.2–.8)**	**.95 (.6–.4)**	**.95 (.8–.2)**	**.95 (.9–.1)**
MTurk-771	.88 (.3–.7)	.88 (.8–.2)	.88 (.2–.8)	**.89 (.7–.3)**	.86 (.7–.3)	.88 (.7–.3)
MEN	.86 (.8–.2)	.87 (.7–.3)	.85 (.2–.8)	.86 (.7–.3)	**.89 (.6–.4)**	**.89 (.8–.2)**
FAN	.58 (.4–.6)	.59 (.8–.2)	.58 (.2–.8)	.58 (.7–.3)	**.62 (.7–.3)**	.61 (.9–.1)

Table 8 Group-based evaluation of FAN with RankDCG' (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
FAN (groups)	.67 (.4–.6)	.68 (.8–.2)	.69 (.3–.7)	.69 (.9–.1)	.69 (1–0)	**.70 (.9–.1)**

Table 9 Group-based evaluation of FAN with RankDCG" (α-β)

dataset	Word2Vec		GloVe		FastText	
	WALE-1	WALE-2	WALE-1	WALE-2	WALE-1	WALE-2
FAN (groups)	.56 (.4–.6)	.56 (.8–.2)	58 (.4–.6)	.58 (.9–.1)	**.59 (.9–.1)**	**.59 (.9–.1)**

6 Discussion

6.1 Word-embedding techniques and models to integrate word vectors

We can draw some conclusions from analyzing the data in Tables 4, 5, 6, and 7. First, it is important to note that Spearman's and Pearson's correlation coefficients never outperformed RankDCG' and, in turn, RankDCG' only outperformed RankDCG" with MTurk-771 and MEN. This demonstrates that an evaluation conducted on the strength of associations, and not only on the rank ordering of word pairs, contributes to revealing the psychological plausibility of word-association models based on deep neural embeddings. In other words, vector-space models show greater quality and coherence when evaluated with a measure oriented to the associative strength.

Second, when analyzing the behaviour of WALE in relation to word-embedding techniques (i.e. Word2Vec, GloVe, and FastText), we realize that Spearman's and Pearson's correlation coefficients return similar results, where the best option with all test datasets

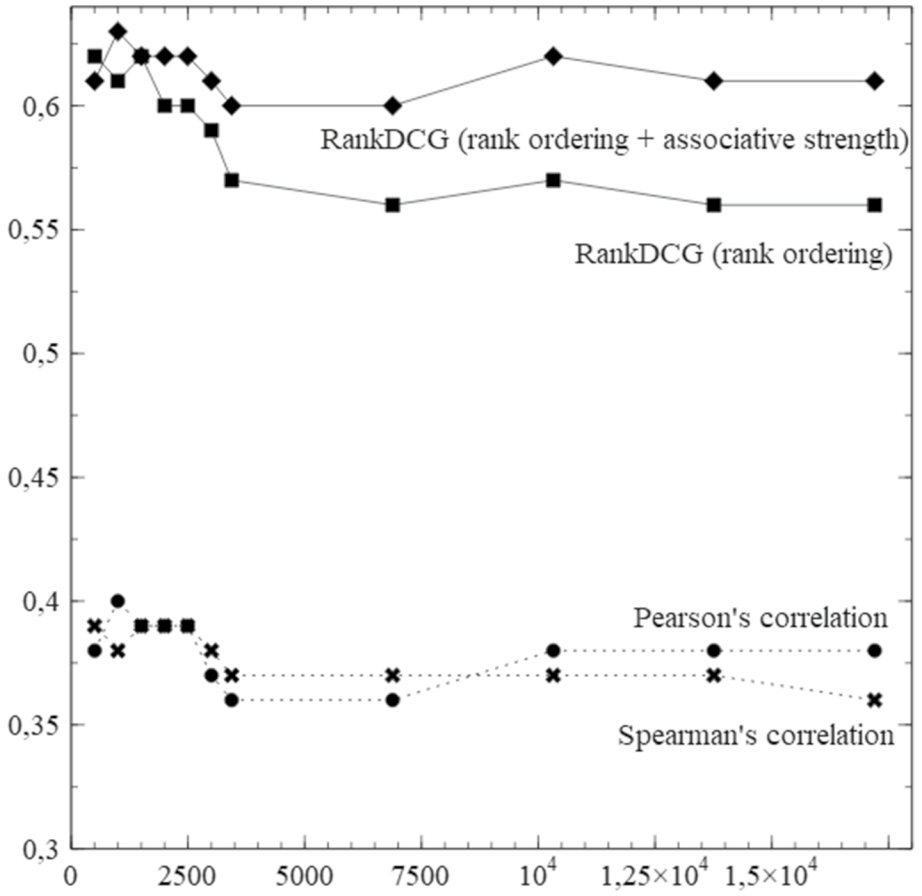

Fig. 3 Evaluation of different-sized samples of FAN with FastText and WALE-2 (0.9-0.1)

Table 10 Evaluation with FastText and WALE-2 (α-β): 850-dimension WNet2Vec

dataset	Spearman	Pearson	RankDCG'	RankDCG"
MC	*.83 (.9–.1)*	.84 (.9–.1)	.97 (.9–.1)	.98 (.6–.4)
YP	*.75 (.1–.9)*	*.84 (.2–.8)*	**.94 (.4–.6)**	.98 (.4–.6)
RG	*.86 (.7–.3)*	.86 (.8–.2)	.95 (.4–.6)	.97 (.4–.6)
MTurk-287	.85 (.8–.2)	.80 (.9–.1)	**.94 (.5–.5)**	**.94 (.7–.3)**
WS-SIM	.85 (.8–.2)	.84 (.9–.1)	**.94 (.8–.2)**	.96 (.8–.2)
WS-REL	*.73 (.9–.1)*	.72 (.9–.1)	.91 (.8–.2)	.94 (.8–.2)
RW	.60 (.9–.1)	*.57 (.9–.1)*	.85 (1–0)	.87 (.9–.1)
WS-ALL	.79 (.8–.2)	.75 (.9–.1)	.93 (.8–.2)	**.96 (.8–.2)**
MTurk-771	.77 (.9–.1)	.75 (.8–.2)	.90 (.6–.4)	.88 (.8–.2)
MEN	.84 (.9–.1)	.82 (1–0)	.90 (.8–.2)	.89 (.8–.2)
FAN	.36 (.8–.2)	.38 (.8–.2)	.56 (.9–.1)	.61 (.9–.1)

is FastText. However, in the case of Word2Vec and GloVe, there is no clear evidence to prove the superiority of one technique over the other. Irrespective of the technique, WALE-1 never outperforms WALE-2, whereas the latter outperforms the former in 28.79% of the ratings with Spearman's correlation and 25.76% with Pearson's correlation. On the other hand, most of the test datasets provide good results with FastText when evaluated with RankDCG' and RankDCG" (i.e. 81.82% and 63.64% of the ratings, respectively), where Word2Vec and GloVe are again much less significant. WALE-1 rarely outperforms WALE-2 (i.e. 4.55% of the ratings with RankDCG' and 7.58% with RankDCG"), but the latter only outperforms the former in 15.16% of the ratings with RankDCG' and 16.67% with RankDCG". In other words, the choice of the WALE model is a determining factor with Spearman's and Pearson's correlation coefficients, but it plays a minor role with RankDCG.

Third, as the parameters of WALE serve to determine the influence of a given type of language model, we notice that each evaluation measure highlights different properties of the vector-space model generated by each technique. For example, in Word2Vec, Spearman's and Pearson's correlation coefficients emphasize the dominant influence of the corpus with WALE-2 (i.e. 90.91% of the ratings with each measure) and that of the semantic network with WALE-1 (i.e. 63.64% with each measure). RankDCG' and RankDCG" also bring to light the influence of the semantic network with WALE-1 (i.e. 90.91% of the ratings with each measure) and that of the corpus with WALE-2 (i.e. 63.64% and 54.55% of the ratings, respectively). In GloVe, all measures give more importance to the semantic network with WALE-1 (i.e. 100% of the ratings with Spearman's and Pearson's correlation coefficients and 81.82% with RankDCG) and to the corpus with WALE-2 (i.e. 90.91% of the ratings with Spearman's and Pearson's correlation coefficients and RankDCG", and 81.82% with RankDCG'). In FastText, the influence of the corpus is greater both in WALE-1 and WALE-2, being more dominant with Spearman's and Pearson's correlation coefficients and RankDCG' (i.e. 90.91% of the ratings) than with RankDCG" (i.e. 81.82%). Therefore, our experiments showed that Word2Vec and GloVe expose the dominant influence of the semantic network through WALE-1 and that of the corpus through WALE-2, whereas the corpus dominates in both WALE models with FastText. This finding is in line with the assumption that internal language models encode mental representations differently compared to external language models. However, unlike previous studies (De Deyne et al. 2015, 2016), we also demonstrate that internal language models do not always perform better than external language models, even with word-similarity datasets.

Finally, the benefit of integrating word-embedding matrices is also evidenced when we take as the baseline the results yielded by a single matrix. On the one hand, the standalone corpus-based model (i.e. 1 and 0 in α and β, respectively) only outperforms hybrid models in 3.03% of the ratings with Pearson's correlation and RankDCG' and 9.09% with Spearman's correlation. It is worthwhile to mention that all these cases only occur when evaluating YP. On the other hand, the standalone WordNet-based model (i.e. 0 and 1 in α and β, respectively) only outperforms hybrid models in 3.03% of the ratings with Spearman's correlation and 6.06% with the remaining measures. In the case of Spearman's and Pearson's correlation coefficients, this occurs when evaluating MTurk-287 and MEN with WALE-1 in FastText. In the case of RankDCG, however, this occurs when evaluating MTurk-287 and RW with WALE-2 in GloVe, as well as the latter with WALE-1 in FastText. Without a doubt, our experiments demonstrate that hybrid language models tend to increase performance when compared against the baseline, as demonstrated in previous studies. However, our research relies on linear compositional functions that allow assessing the relative influence of a given language model in relation to another.

6.2 Group-based evaluation

In group-based evaluation, where RankDCG' always outperforms RankDCG", the best results are obtained again with FastText and WALE-2, and the worst with Word2Vec and WALE-1 (Tables 8 and 9). A comparison with the results derived from the evaluation conducted on the whole list of word pairs (Tables 4, 5, 6, and 7) showed that scores are significantly higher in group-based evaluation with RankDCG' but slightly better in the evaluation of the whole test dataset with RankDCG".

6.3 Size of datasets

As shown in Fig. 3, if we focus on small-sized datasets (i.e. the first seven dots in each line of the graph, which correspond to datasets containing less than 3500 pairs of words), it can be noticed that Spearman's correlation and RankDCG" show a smaller amount of variability than Pearson's correlation and RankDCG', where performance degrades progressively in the latter. On the other hand, if we focus on medium-sized datasets (i.e. the last five dots in each line of the graph, which correspond to the datasets containing over 3500 pairs of words), the pattern of change is very similar for the four measures. In either of the two cases, RankDCG" provides the highest scores.

6.4 Reduction of dimensionality

The reduction of dimensionality in WNet2Vec did not virtually affect the performance of any model when evaluated by any of the measures with any of the test datasets. For example, in the case of FastText with WALE-2 (Table 10), the 850-dimension word-embedding matrix leads to an improvement and degradation of performance in 11.36% of the ratings in each case, remaining unchanged in 77.28%.

7 Conclusion

During the past few decades, many studies have been published on the topic of word-association assessment, where a variety of techniques have been used from fields such as psychology, linguistics, and NLP. In contrast to most previous studies, this article is not aimed at presenting a new measure of word association (e.g. word relatedness and similarity) but at exploring different ways to integrate existing embeddings to determine the semantic or non-semantic associative strength between words so that correlation with human judgements can be maximized. To this end, we took into consideration several factors, such as the word-embedding technique (i.e. Word2Vec, GloVe, and FastText), the model for the integration of word-embedding matrices (i.e. not only whether to project them into a single or double vector space but also whether to give greater weight to an external or internal language model), the evaluation measure (i.e. Spearman's and Pearson's correlation coefficients and RankDCG), and the dataset size, among others. Several conclusions can be drawn from this research:

(a) FastText has proven to be the best word-embedding technique, probably because embeddings were enriched with sub-word information. However, there is no clear evidence to determine the second-best choice, i.e. Word2Vec or GloVe, whose embeddings were constructed directly from words.
(b) The integration of word-embedding matrices into a double vector space (i.e. WALE-2) always provides optimal results when traditional measures such as Spearman's and Pearson's correlation coefficients are employed. In the case of RankDCG' and RankDCG", the WALE model is not a critical factor, although WALE-2 is also very likely to provide a good result.
(c) The most effective way to integrate external and internal language models (i.e. corpus- and network-based embeddings) through the α and β parameters in WALE is highly conditioned by not only the word-embedding technique but also the evaluation measure. Indeed, our experiments revealed that, regardless of the measure, there is a dominant influence of the semantic network in WALE-1 and the corpus in WALE-2 with Word2Vec and GloVe, but the corpus dominates in both WALE models with FastText.
(d) RankDCG' usually outperforms Spearman's and Pearson's correlation coefficients, and, in turn, RankDCG" usually outperforms RankDCG'. This is true when the whole test dataset is evaluated, regardless of whether or not associative words are semantically related. However, RankDCG' outperforms RankDCG" in group-based evaluation. Moreover, group-based evaluation gives better results than the evaluation of the whole test dataset with RankDCG', where RankDCG" is in the opposite case.
(e) In the light of the previous findings, we can conclude that reliable results can be provided with FastText, WALE-2 and a weight ranging from 0.8 to 1 on the corpus-based embeddings, showing a more pronounced tendency when evaluated with Spearman's and Pearson's correlation coefficients rather than with RankDCG.
(f) RankDCG" is the least sensitive measure to the size of test datasets, mainly when the size is over 2000 pairs of words.
(g) The reduction of dimensionality in the network-based embedding matrix (e.g. WNet-2Vec) did not virtually affect the performance of any model.

Therefore, we demonstrated that:

1. A mathematically simple technique, i.e. the weighted average of the cosine-similarity coefficients derived from independent word embeddings in a double vector-space model, can serve to provide sufficiently successful results from off-the-shelf word embeddings,
2. The weak-knowledge approach based on corpora plays a more critical role than the strong-knowledge approach based on semantic networks in a hybrid model such as WALE, and
3. A measure such as RankDCG" can help researchers discover word-association models that contribute to constructing semantic representations that are more cognitively plausible, as the evaluation is conducted on both rank ordering and the associative strength of word pairs.

Future work will focus on applying our technique to two distinct scenarios: neuropsychology and topic categorization. On the one hand, neuropsychological tests such as the Hayling Sentence Completion Test, where patients complete sentences with the first word that comes to their mind, are liable to bias when examiners assess stimulus-response

associations. Our research can contribute to facilitating the automated scoring of responses. On the other hand, we intend to develop an unsupervised topic-categorization model that relies on the semantic similarity between user-generated text data and a set of pre-defined categories. In this context, our research can contribute to enhancing the embedding-derived meaning representation of both the messages and the topics.

Acknowledgements Financial support for this research has been provided by the Spanish Ministry of Science, Innovation and Universities [grant number RTC 2017-6389-5], the Spanish "Agencia Estatal de Investigación" [grant number PID2020-112827GB-I00 / AEI / 10.13039/501100011033], and the European Union's Horizon 2020 research and innovation program [grant number 101017861: project SMARTLAGOON].

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Agirre E, Alfonseca E, Hall K, Kravalova J, Pasca M, Soroa A (2009) A study on similarity and relatedness using distributional and WordNet-based approaches. In: Proceedings of the 2009 annual conference of the North American chapter of the ACL, pp. 19–27

Agirre E, Soroa A (2009) Personalizing page rank for word sense disambiguation. In: Proceedings of the 12th conference of the European chapter of the ACL, pp. 33–41

Akhtar N, Sufyan Beg MM, Javed H (2019) Topic modelling with fuzzy document representation. In: Singh M, Gupta P, Tyagi V, Flusser J, Ören T, Kashyap R (eds) Advances in computing and data sciences. ICACDS, (2019) Communications in computer and information science, vol 1046. Springer, Singapore, pp 577–587

Artetxe M, Labaka G, Agirre E (2016) Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In: Proceedings of the 2016 conference on empirical methods in natural language processing, pp. 2289-2294

Banerjee S, Pedersen T (2003) Extended gloss overlaps as a measure of semantic relatedness. In: Proceedings of the 18th international joint conference on artificial intelligence, pp. 805-810

Banjade R, Maharjan N, Niraula NB, Rus V, Gautam D (2015) Lemon and tea are not similar: measuring word-to-word similarity by combining different methods. In: Proceedings of the 16th international conference on intelligent text processing and computational linguistics, pp. 335–346

Baroni M, Dinu G, Kruszewski G (2014) Don't count, predict! A systematic comparison of context-counting vs context-predicting semantic vectors. In: Proceedings of the 52nd annual meeting of the ACL, pp. 238-247

Bengio Y, Senécal JS (2003) Quick training of probabilistic neural nets by importance sampling. Proceedings of artificial intelligence statistics 2003:1–9

Bengio Y, Ducharme J, Vincent P, Janvin C (2003) A neural probabilistic language model. J Mach Learn Res 3:1137–1155

Bhatia S (2017) Associative judgment and vector space semantics. Psychol Rev 124(1):1–20

Bhutada S, Balaram VVSSS, Bulusu VV (2016) Semantic latent dirichlet allocation for automatic topic extraction. J Inf Optim Sci 37(3):449–469
Bizer C, Lehmann J, Kobilarov G, Auer S, Becker C, Cyganiak R, Hellmann S (2009) DBpedia - a crystallization point for the Web of Data. J Web Semant 7(3):154–165
Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022
Bojanowski P, Grave E, Joulin A, Mikolov T (2017) Enriching word vectors with subword information. Trans Assoc Comput Linguist 5:135–146
Bollacker K, Evans C, Paritosh P, Sturge T, 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, pp. 1247–1250
Bollegala D, Alsuhaibani M, Maehara T, Kawarabayashi K (2016) Joint word representation learning using a corpus and a semantic lexicon. In: Proceedings of the 30th AAAI conference on artificial intelligence, pp. 2690–2696
Bruni E, Boleda G, Baroni M, Tran NK (2012) Distributional semantics in technicolor. In: Proceedings of the 50th annual meeting of the ACL, vol. 1, pp. 136–145
Budanitsky A, Hirst G (2001) Semantic distance in WordNet: an experimental, application-oriented evaluation of five measures. In: Proceedings of the 2nd meeting of the North American chapter of the ACL. Workshop on WordNet and other lexical resources, pp. 29–34
Budhkar A, Rudzicz F (2019) Augmenting Word2Vec with latent dirichlet allocation within a clinical application. In: Proceedings of the 2019 conference of the North American chapter of the ACL: Human language technologies, vol. 1, pp. 4095–4099
Camacho-Collados J, Pilehvar MT (2018) From word to sense embeddings: a survey on vector representations of meaning. J Artif Intell Res 63:743–788
Cambria E, Li Y, Xing FZ, Poria S, Kwok K (2020) SenticNet 6: ensemble application of symbolic and subsymbolic AI for sentiment analysis. In: Proceedings of the 29th ACM international conference on information and knowledge management, pp. 105–114
Cambria E, Olsher D, Rajagopal D (2014) SenticNet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis. In: Proceedings of the 28th AAAI conference on artificial intelligence, pp. 1515–1521
Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka ER, Mitchell TM (2010) Toward an architecture for never-ending language learning. In: Proceedings of the 24th AAAI conference on artificial intelligence, pp. 1306–1313
Cattle A, Ma X (2017) Predicting word association strengths. In: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 1283–1288
Chandar S, Lauly S, Larochelle H, Khapra M, Ravindran B, Raykar V, Saha A (2014) An autoencoder approach to learning bilingual word representations. In: Proceedings of the 27th annual conference on advances in neural information processing systems, pp. 1853–1861
Coates JN, Bollegala D (2018) Frustratingly easy meta-embedding – Computing meta-embeddings by averaging source word embeddings. In: Proceedings of the 2018 conference of the North American chapter of the ACL: Human language technologies, pp. 194–198
Collobert R, Weston J (2008) A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on machine learning, pp. 160–167
Dacey M (2019) Association and the mechanisms of priming. J Cognit Sci 20(3):281–321
Dai Z, Yang Z, Yang Y, Carbonell JG, Le QV, Salakhutdinov R (2019) Transformer-XL: attentive language models beyond a fixed-length context. In: Proceedings of the 57th annual meeting of the ACL, pp. 2978–2988
De Deyne S, Navarro DJ, Perfors A, Brysbaert M, Storms G (2019) The 'Small World of Words' English word association norms for over 12,000 cue words. Behav Res Methods 51:987–1006
De Deyne S, Perfors A, Navarro DJ (2016) Predicting human similarity judgments with distributional models: the value of word associations. In: Proceedings of the 26th international conference on computational linguistics, pp. 1861–1870
De Deyne S, Verheyen S, Storms G (2015) The role of corpus size and syntax in deriving lexico-semantic representations for a wide range of concepts. Q J Exp Psychol 68(8):1643–1664
De Souza JVA, Oliveira LES, Gumiel YB, Carvalho DR, Moro CMB (2019) Incorporating multiple feature groups to a siamese neural network for semantic textual similarity task in Portuguese texts. In: Proceedings of the ASSIN 2 shared task: Evaluating semantic textual similarity and textual entailment in Portuguese, XII symposium in information and human language technology, pp. 59–68
Deerwester SC, Dumais ST, Landauer TK, Furnas GW, Harshman RA (1990) Indexing by latent semantic analysis. J Am Soc Inf Sci 41(6):391–407

Demotte P, Senevirathne L, Karunanayake B, Munasinghe U, Ranathunga S (2020) Sentiment analysis of Sinhala news comments using sentence-state LSTM networks. In: Proceedings of the 2020 Moratuwa engineering research conference, pp. 283–288

Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the ACL: Human language technologies, vol. 1, pp. 4171–4186

Du Y, Wu Y, Lan M (2019) Exploring human gender stereotypes with word association test. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, pp. 6133–6143

El Mahdaouy A, El Alaoui SO, Gaussier E (2018) Improving Arabic information retrieval using word embedding similarities. Int J Speech Technol 21:121–136

Erk K (2012) Vector space models of word meaning and phrase meaning: a survey. Lang Linguist Compass 6(10):635–653

Faruqui M, Dyer C (2014) Improving vector space word representations using multilingual correlation. In: Proceedings of the 14th conference of the European chapter of the ACL, pp. 462–471

Faruqui M, Dodge J, Jauhar SK, Dyer C, Hovy E, Smith NA (2015) Retrofitting word vectors to semantic lexicons. In: Proceedings of the 2015 conference of the North American chapter of the ACL: Human language technologies, pp. 1606–1615

Fellbaum C (ed) (1998) WordNet: an electronic lexical database. MIT Press, Cambridge

Finkelstein L, Gabrilovich E, Matias Y, Rivlin E, Solan Z, Wolfman G, Ruppin E (2001) Placing search in context: The concept revisited. In: Proceedings of the 10th international conference on world wide web, pp. 406–414

Firth JR (1957) Papers in linguistics 1934–1951. Oxford University Press, Oxford

Ganitkevitch J, Van Durme B, Callison-Burch C (2013) PPDB: The paraphrase database. In: Proceedings of the 2013 conference of the North American chapter of the ACL: Human language technologies, pp. 758–764

Garimella A, Banea C, Mihalcea R (2017) Demographic-aware word associations. In: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 2285–2295

Gilligan TM, Rafal RD (2019) An opponent process cerebellar asymmetry for regulating word association priming. Cerebellum 18:47–55

Gladkova A, Drozd A (2016) Intrinsic evaluations of word embeddings: What can we do better? In: Proceedings of the 1st workshop on evaluating vector space representations for NLP, pp. 36–42

Goikoetxea J, Soroa A, Agirre E (2015) Random walks and neural network language models on knowledge bases. Proceedings of the 2015 annual conference of the North American chapter of the ACL: Human language technologies, pp. 1434–1439

Goikoetxea J, Agirre E, Soroa A (2016) Single or multiple? Combining word representations independently learned from text and WordNet. In: Proceedings of the 30th AAAI conference on artificial intelligence, pp. 2608–2614

Goldani MH, Momtazi S, Safabakhsh R (2021) Detecting fake news with capsule neural networks. Appl Soft Comput 101(1):1–8

Gomez-Perez JM, Denaux R, Garcia-Silva A (2020) A practical guide to hybrid natural language processing. Springer, Cham

Gong P, Liu J, Yang Y, He H (2020) Towards knowledge enhanced language model for machine reading comprehension. IEEE Access 8:224837–224851

Goodwin TR, Demner-Fushman D (2020) Enhancing question answering by injecting ontological knowledge through regularization. In: Proceedings of Deep Learning Inside Out (DeeLIO): The first workshop on knowledge extraction and integration for deep learning architectures, pp. 56–63

Grave E, Bojanowski P, Gupta P, Joulin A, Mikolov T (2018) Learning word vectors for 157 languages. In: Proceedings of the 11th international conference on language resources and evaluation, pp. 3483–3487

Gross O, Doucet A, Toivonen H (2016) Language-independent multi-document text summarization with document-specific word associations. In: Proceedings of the ACM symposium on applied computing, pp. 853–860

Grover A, Leskovec J (2016) Node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 855–864

Grujić ND, Milovanović VM, (2019) Natural language processing for associative word predictions. In: Proceedings of the 18th international conference on smart technologies, pp. 1–6

Guan J, Huang F, Zhao Z, Zhu X, Huang M (2020) A knowledge-enhanced pretraining model for commonsense story generation. Trans Assoc Comput Linguist 8:93–108

Gunel B, Zhu C, Zeng M, Huang X (2020) Mind the facts: Knowledge-boosted coherent abstractive text summarization. In: Proceedings of the 33rd conference on neural information processing systems, pp. 1–7

Günther F, Dudschig C, Kaup B (2016) Predicting lexical priming effects from distributional semantic similarities: a replication with extension. Front Psychol 7(1646):1–13

Halawi G, Dror G, Gabrilovich E, Koren Y (2012) Large-scale learning of word relatedness with constraints. In: Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1406–1414

Harley TA (2014) The psychology of language: from data to theory. Psychology Press, Hove

Harris ZS (1954) Distributional structure. Word 10(2–3):146–162

Haveliwala TH (2002) Topic-sensitive PageRank. In: Proceedings of the 11th international conference on world wide web, pp. 517–526

Hermann KM, Blunsom P (2013) Multilingual distributed representations without word alignment. In: Proceedings of the 2014 international conference on learning representations, pp. 1–9

Higginbotham G, Munby I, Racine JP (2015) A Japanese word association database of English. Vocab Learn Instr 4(2):1–20

Iacobacci I, Pilehvar MT, Navigli R (2015) Sensembed: Learning sense embeddings for word and relational similarity. In: Proceedings of the 53rd annual meeting of the ACL and the 7th international joint conference on natural language processing, pp. 95–105

Iacobacci I, Pilehvar MT, Navigli R (2016) Embeddings for word sense disambiguation: An evaluation study. In: Proceedings of the 54th annual meeting of the ACL, pp. 897–907

Järvelin K, Kekäläinen J (2000) IR evaluation methods for retrieving highly relevant documents. In: Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval, pp. 41–48

Järvelin K, Kekäläinen J (2002) Cumulated gain-based evaluation of IR techniques. ACM Trans Inf Syst 20(4):422–446

Jiang Y, Bai W, Zhang X, Hu J (2017) Wikipedia-based information content and semantic similarity computation. Inf Process Manag 53(1):248–265

Jiang JJ, Conrath DW (1997) Semantic similarity based on corpus statistics and lexical taxonomy. In: Proceedings of the international conference on research in computational linguistics, pp. 19–33

Jingrui Z, Qinglin W, Yu L, Yuan L (2017) A method of optimizing LDA result purity based on semantic similarity. In: Proceedings of the 32nd youth academic annual conference of Chinese association of automation, pp. 361–365

Jo Y, Alice O (2011) Aspect and sentiment unification model for online review analysis. In: Proceedings of the 4th ACM international conference on web search and web data mining, pp. 815–824

Johansson R, Pina LN (2015) Embedding a semantic network in a word space. In: Proceedings of the 2015 conference of the North American chapter of the ACL: Human language technologies, pp. 1428–1433

Kang B (2018) Collocation and word association: comparing collocation measuring methods. Int J Corpus Linguist 23(1):85–113

Katerenchuk D, Rosenberg A (2016) RankDCG: Rank-ordering evaluation measure. In: Proceedings of the 10th international conference on language resources and evaluation. European Language Resources Association, pp. 3675–3680

Kiela D, Hill F, Clark S (2015) Specializing word embeddings for similarity or relatedness. In: Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 2044–2048

Kober T, Weeds J, Wilkie J, Reffin J, Weir D (2017) One representation per word - Does it make sense for composition? In: Proceedings of the 1st workshop on sense, concept and entity representations and their applications, pp. 79–90

Kulkarni A, Mandhane M, Likhitkar M, Kshirsagar G, Jagdale J, Joshi R (2021) Experimental evaluation of deep learning models for Marathi text classification. https://arxiv.org/pdf/2101.04899.pdf. Accessed 26 February 2021

Leacock C, Chodorow M (1998) Combining local context and WordNet similarity for word sense identification. In: Fellbaum C (ed) WordNet: an electronic lexical database. MIT Press, Cambridge (MA), pp 265–283

Lebret R, Collobert R (2014) Word embeddings through Hellinger PCA. In: Proceedings of the 14th conference of the European chapter of the ACL, pp. 482–490

Lee YY, Ke H, Huang HH, Chen HH (2016) Combining word embedding and lexical database for semantic relatedness measurement. In: Proceedings of the 25th international conference companion on world wide web, pp. 73–74

Lenci A (2018) Distributional models of word meaning. Ann Rev Linguist 4:151–171

Lengerich BJ, Maas AL, Potts C (2017) Retrofitting distributional embeddings to knowledge graphs with functional relations. In: Proceedings of the 27th international conference on computational linguistics, pp. 2423–2436
Lesk M (1986) Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: Proceedings of the 5th annual international conference on systems documentation, pp. 24–26
Levy O, Goldberg Y (2014) Linguistic regularities in sparse and explicit word representations. In: Proceedings of the 18th conference on computational language learning, pp. 171–180
Li Y, Bandar ZA, McLean D (2003) An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans Knowl Data Eng 15(4):871–882
Lin D (1998) An information-theoretic definition of similarity. In: Proceedings of the 15th international conference on machine learning, pp. 296–304
Liu T, Hu Y, Gao J, Sun Y, Yin B (2020a) Zero-shot text classification with semantically extended graph convolutional network. In: Proceedings of the 25th international conference on pattern recognition, pp. 8352–8359
Liu Q, Kusner MJ, Blunsom P (2020b) A survey on contextual embeddings. arXiv:2003.07278. Accessed 15 June 2020
Lund K, Burgess C (1996) Producing high-dimensional semantic spaces from lexical co-occurrence. Behav Res Methods Instr Comput 28(2):203–208
Luong MT, Socher R, Manning CD (2013) Better word representations with recursive neural networks for morphology. In: Proceedings of the 17th conference on computational natural language learning, pp. 104–113
Ma Q, Lee HY (2019) Measuring the vocabulary knowledge of Hong Kong primary school second language learners through word associations: Implications for reading literacy. In: Reynolds B, Teng M (eds) English literacy instruction for Chinese speakers. Palgrave Macmillan, Singapore, pp 35–56
Mandera P, Keuleers E, Brysbaert M (2017) Explaining human performance in psycholinguistic tasks with models of semantic similarity based on prediction and counting: a review and empirical validation. J Mem Lang 92:57–78
Meng Y, Wang G, Liu Q (2019) Multi-layer convolutional neural network model based on prior knowledge of knowledge graph for text classification. In: Proceedings of the IEEE 4th international conference on cloud computing and big data analysis, pp. 618–624
Mihaylov T, Frank A (2018) Knowledgeable reader: enhancing cloze-style reading comprehension with external commonsense knowledge. In: Proceedings of the 56th annual meeting of the ACL, pp. 821–832
Mikolov T, Chen K, Corrado G, Dean J (2013a) Efficient estimation of word representations in vector space. In: Proceedings of the international conference on learning representations workshop track, pp. 1301–3781
Mikolov T, Le QV, Sutskever I (2013b) Exploiting similarities among languages for machine translation. arXiv:1309.4168. Accessed 5 May 2019
Mikolov T, Yih WT, Zweig G (2013c) Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 conference of the North American chapter of the ACL: Human language technologies, pp. 746-751
Miller G, Charles W (1991) Contextual correlates of semantic similarity. Lang Cognit Process 6(1):1–28
Minaee S, Kalchbrenner N, Cambria E, Nikzad N, Chenaghlu M, Gao J (2021) Deep learning based text classification: a comprehensive review. ACM Comput Surv 54(3):1–40
Mnih A, Hinton G (2008) A scalable hierarchical distributed language model. In: Proceedings of the 21st international conference on neural information processing systems, pp. 1081–1088
Morin F, Bengio Y (2005) Hierarchical probabilistic neural network language model. In: Proceedings of the 10th international workshop on artificial intelligence and statistics, pp. 246–252
Mrkšić N, Vulić I, Séaghdha DÓ, Leviant I, Reichart R, Gašić M, Korhonen A, Young S (2017) Semantic specialisation of distributional word vector spaces using monolingual and cross-lingual constraints. Trans Assoc Comput Linguist 5:309–324
Navigli R, Ponzetto SP (2012) BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif Intell 193:217–250
Nelson DL, McEvoy CL, Schreiber TA (1998) The University of South Florida word association, rhyme, and word fragment norms. http://w3.usf.edu/FreeAssociation/Intro.html. Accessed 13 January 2019
Nguyen KA, Walde SS, Vu NT (2016) Integrating distributional lexical contrast into word embeddings for antonym-synonym distinction. In: Proceedings of the 54th annual meeting of the ACL, pp. 454–459

Niraula NB, Gautam D, Banjade R, Maharjan N, Rus V (2015) Combining word representations for measuring word relatedness and similarity. In: Proceedings of the 28th international Florida artificial intelligence research society conference, pp. 199-204

Ostendorff M, Bourgonje P, Berger M, Moreno-Schneider J, Rehm G, Gipp B (2019) Enriching BERT with knowledge graph embeddings for document classification. In: Proceedings of the GermEval 2019 hierarchical text classification workshop, pp. 1–8

Patwardhan S (2003) Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. University of Minnesota, Minneapolis (**PhD thesis**)

Pedersen T, Pakhomov SVS, Patwardhan S, Chute CG (2007) Measures of semantic similarity and relatedness in the biomedical domain. J Biomed Inform 40(3):288–299

Pennington J, Socher R, Manning CD (2014) GloVe: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing, pp. 1532–1543

Peters M, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Proceedings of the 2018 conference of the North American chapter of the ACL: Human language technologies, pp. 2227–2237

Phan THV, Do P (2020) BERT+vnKG: using deep learning and knowledge graph to improve Vietnamese question answering system. Int J Adv Comput Sci Appl 11(7):480–487

Pilehvar MT, Camacho-Collados J (2020) Embeddings in natural language processing: theory and advances in vector representation of meaning. Morgan & Claypool Publishers, San Rafael

Pilehvar MT, Collier N (2017) Inducing embeddings for rare and unseen words by leveraging lexical resources. In: Proceedings of the 15th conference of the European chapter of the ACL, pp. 388–393

Playfoot D, Balint T, Pandya V, Parkes A, Peters M, Richards S (2018) Are word association responses really the first words that come to mind? Appl Linguis 39(5):607–624

Poria S, Chaturvedi I, Cambria E, Bisio F (2016) Sentic LDA: improving on LDA with semantic similarity for aspect-based sentiment analysis. In: Proceedings of the 2016 international joint conference on neural networks, pp. 4465–4473

Pylieva H, Chernodub A, Grabar N, Hamon T (2019) RNN embeddings for identifying difficult to understand medical words. In: Proceedings of the 18th BioNLP workshop and shared task, pp. 97–104

Rada R, Mili H, Bicknell E, Blettner M (1989) Development and application of a metric on semantic nets. IEEE Trans Syst Man Cybern 19(1):17–30

Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf. Accessed 30 May 2021

Radinsky K, Agichtein E, Gabrilovich E, Markovitch S (2011) A word at a time: Computing word relatedness using temporal semantic analysis. In: Proceedings of the 20th international conference on world wide web, pp. 337–346

Resnik P (1995) Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th international joint conference on artificial intelligence, pp. 448–453

Reyes-Magaña J, Bel-Enguix G, Sierra G, Gómez-Adorno H (2019) Designing an electronic reverse dictionary based on two word association norms of English language. In: Proceedings of the eLex 2019 conference, pp. 865–880

Riedl M, Biemann C (2017) There's no "Count or Predict" but task-based selection for distributional models. In: Proceedings of the 12th international conference on computational semantics, pp. 1–9

Rieth CA, Huber DE (2017) Comparing different kinds of words and word-word relations to test an habituation model of priming. Cogn Psychol 95:79–104

Rothe S, Schutze H (2015) Autoextend: extending word embeddings to embeddings for synsets and lexemes. In: Proceedings of the 53rd annual meeting of the ACL and the 7th international joint conference on natural language processing, pp. 1793–1803

Ruas T, Grosky W, Aizawa A (2019) Multi-sense embeddings through a word sense disambiguation process. Expert Syst Appl 136:288–303

Rubenstein H, Goodenough J (1965) Contextual correlates of synonymy. Commun ACM 8(10):627–633

Ruder S, Vulic I, Sogaard A (2019) A survey of cross-lingual word embedding models. J Artif Intell Res 65:569–631

Saedi C, Branco A, Rodrigues JA, Silva JR (2018) WordNet embeddings. In: Proceedings of the 3rd workshop on representation learning for NLP, pp. 122–131

Salehi B, Cook P, Baldwin T (2015) A word embedding approach to predicting the compositionality of multiword expressions. In: Proceedings of the 2015 conference of the North American chapter of the ACL: Human language technologies, pp. 977–983

Sap M, Le Bras R, Allaway E, Bhagavatula C, Lourie N, Rashkin H, Roof B, Smith NA, Choi Y (2019) Atomic: an atlas of machine commonsense for if-then reasoning. Proceedings of the AAAI conference on artificial intelligence 33:3027–3035

Seco N, Veale T, Hayes J (2004) An intrinsic information content metric for semantic similarity in WordNet. In: Proceedings of the 16th European conference on artificial intelligence, pp. 1089–1090

Smetanin S, Komarov M (2019) Sentiment analysis of product reviews in Russian using convolutional neural networks. In: Proceedings of the 21st IEEE conference on business informatics, pp. 482–486

Smith SL, Turban DHP, Hamblin S, Hammerla NY (2017) Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In: Proceedings of the 5th international conference on learning representations, pp. 1–10

Speer R, Lowry-Duda J (2017) ConceptNet at SemEval-2017 Task 2: extending word embeddings with multilingual relational knowledge. In: Proceedings of the 11th international workshop on semantic evaluation, pp. 85–89

Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) Line: large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web, pp. 1067–1077

Taylor JR (2012) The mental corpus: how language is represented in the mind. Oxford University Press, Oxford

Tsuboi Y (2014) Neural networks leverage corpus-wide information for part-of-speech tagging. In: Proceedings of the 2014 conference on empirical methods in natural language processing, pp. 938–950

Van Rensbergen B, Storms G, De Deyne S (2015) Examining assortativity in the mental lexicon: evidence from word associations. Psychon Bull Review 22:1717–1724

Vrandecic D, Krotzsch M (2014) Wikidata: a free collaborative knowledge base. Commun ACM 57(10):78–85

Wang Y, Cui L, Zhang Y (2020) How can BERT help lexical semantics tasks? arXiv:1911.02929.pdf. Accessed 27 January 2020

Wang C, Jiang H (2018) Explicit utilization of general knowledge in machine reading comprehension. In: Proceedings of the 57th annual meeting of the ACL, pp. 2263–2272

Wu Z, Palmer M (1994) Verb semantics and lexical selection. In: Proceedings of the 32nd annual meeting of the ACL, pp. 133–138

Xiaosa L, Wenyu W (2016) Word class influence upon L1 and L2 English word association. Chin J Appl Linguist 39(4):440–458

Xu C, Bai Y, Bian J, Gao B, Wang G, Liu X, Liu TY (2014) RC-NET: a general framework for incorporating knowledge into word representations. In: Proceedings of the 23rd ACM international conference on information and knowledge management, pp. 1219–1228

Yang P, Li L, Luo F, Liu T, Sun X (2019a) Enhancing topic-to-essay generation with external commonsense knowledge. In: Proceedings of the 57th annual meeting of the ACL, pp. 2002–2012

Yang D, Powers DMW (2006) Verb similarity on the taxonomy of WordNet. In: Proceedings of the 3rd international WordNet conference, pp. 121–128

Yang X, Tiddi I (2020) Creative storytelling with language models and knowledge graphs. In: Proceedings of the CIKM 2020 workshops co-located with the 29th ACM international conference on information and knowledge management, pp. 1–9

Yang A, Wang Q, Liu J, Liu K, Lyu Y, Wu H, She Q, Li S (2019b) Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In: Proceedings of the 57th annual meeting of the ACL, pp. 2346–2357

Yin W, Schütze H (2016) Learning word meta-embeddings. In: Proceedings of the 54th annual meeting of the ACL, pp. 1351–1360

Yu M, Dredze M (2014) Improving lexical embeddings with semantic knowledge. In: Proceedings of the 52nd annual meeting of the ACL, pp. 545–550

Yu D, Wu Y, Sun J, Ni Z, Li Y, Wu Q, Chen X (2017) Mining hidden interests from Twitter based on word similarity and social relationship for OLAP. Int J Softw Eng Knowl Eng 27(9–10):1567–1578

Zesch T (2010) Study of semantic relatedness of words using collaboratively constructed semantic resources. Technische Universität Darmstadt, Darmstadt (**PhD thesis**)

Zhang F, Gao W, Fang Y, Zhang B (2020) Enhancing short text topic modeling with FastText embeddings. In: Proceedings of the 2020 international conference on big data, artificial intelligence and internet of things engineering, pp. 255–259

Zhang Z, Han X, Liu Z, Jiang X, Sun M, Liu Q (2019) ERNIE: Enhanced language representation with informative entities. In: Proceedings of the 57th annual meeting of the ACL, pp. 1441–1451

Zhang Y, Liu Q, Song L (2018) Sentence-state LSTM for text representation. In: Proceedings of the 56th annual meeting of the ACL, vol. 1, pp. 317–327

Zhou Z, Wang Y, Gu J (2008) A new model of information content for semantic similarity in WordNet. In: Proceedings of the second international conference on future generation communication and networking symposia, pp. 85–89