Compositional Hyponymy with Positive Operators

Martha Lewis
ILLC, University of Amsterdam
m.a.f.lewis@uva.nl

Abstract

Language is used to describe concepts, and many of these concepts are hierarchical. Moreover, this hierarchy should be compatible with forming phrases and sentences. We use linear-algebraic methods that allow us to encode words as collections of vectors. The representations we use have an ordering, related to subspace inclusion, which we interpret as modelling hierarchical information. The word representations built can be understood within a compositional distributional semantic framework, providing methods for composing words to form phrase and sentence level representations. The resulting representations give competitive results on simple sentence-level entailment datasets.

1 Introduction

Distributional semantics (Harris, 1954; Firth, 1957) is effective and important within the area of computational modelling of language, particularly as regards to synonymy and paraphrasing. Within the field, at least two additional properties are desirable. Firstly, we would like a method by which we can compose vectors to form representations above the word level. Secondly, we would like a notion of lexical entailment, or hyponymy, with which we can capture a sense of the generality of concepts, and the notion of one concept being an instance of another. Furthermore, we would like these two properties to interact nicely with one another, so that the hyponymy relation is not lost when words are composed. In Bankova et al. (2019) the authors provide theory describing a notion of hyponymy that interacts well with compositionality, but do not provide experimental support. In Balkır et al. (2016) the authors suggest a measure of hyponymy based on entropy which also interacts well with compositionality and provide experimental support. A compositional version of the distributional inclusion hypothesis (DIH) (Geffet and Dagan, 2005) is examined in Kartsaklis and Sadrzadeh (2016). In the current paper, we use the framework of Bankova et al. (2019) to build positive operators that represent words. The operators are built using GloVe vectors (Pennington et al., 2014) and information about hyponym-hypernym relationships, which may be sourced either from human-curated resources like WordNet (Miller, 1995), or unsupervised sources, via the use of pattern-based methods. We give two new measures for graded hyponymy that provide a wider range of comparisons than the entropy-derived measure developed in Balkır et al. (2016) or the eigenvalue-related measure of Bankova et al. (2019).

We test our word representations and measures on a range of datasets. Three of the datasets are single-word entailment and have been designed to test directionality (BLESS) (Baroni and Lenci, 2011), detection (WBLESS) (Weeds et al., 2014), and both directionality and direction together (BIBLESS) (Kiela et al., 2015). We also test our models on the compositional dataset of Kartsaklis and Sadrzadeh (2016). This dataset provides a test for entailment at the phrase and sentence level. We find that our model performs fairly well on BLESS and its variants, and very well on the compositional dataset.

2 Background and Related Work

Vector space models of meaning often rely on some form of the distributional hypothesis: that words that occur in similar contexts have similar meanings. However, as well as deriving word meanings, we also need to give meanings to sentences and phrases. This means that we need some method for composing vector representations of words. Commonly-used methods include neural

Proceedings of Recent Advances in Natural Language Processing, pages 638–647,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_075

network methods, as seen in Socher et al. (2013); Bowman et al. (2014), simpler element-wise combination methods (Mitchell and Lapata, 2010), and tensor-based methods (Baroni and Zamparelli, 2010; Coecke et al., 2010; Paperno et al., 2014). Tensor-based methods operate by modelling words of different grammatical types in different vector spaces, and viewing relational words such as verbs and adjectives as linear maps that operate on their arguments. This allows methods from formal semantics to be more easily mapped onto vector space representations, and thereby gives us mechanisms for composing words, in line with their grammatical types, to form phrases and sentences.

We use an extension of the tensor-based approach, based on the methods given in Piedeleu et al. (2015); Bankova et al. (2019); Balkır et al. (2016). We represent nouns as *positive operators*, which can be considered as representing *collections of vectors*. Functional words like adjectives and verbs are now represented as *completely positive maps*, i.e. linear maps which preserve the positivity of their arguments. These can be thought of as linear maps which take valid collections of vectors to valid collections of vectors.

2.1 Related Work

Entailment is an important and thriving area of research within distributional semantics. The PASCAL Recognising Textual Entailment Challenge (Dagan et al., 2006) has attracted a large number of researchers in the area and generated a number of approaches. Previous lines of research on entailment for distributional semantics investigate the development of directed similarity measures which can characterize entailment (Weeds et al., 2004; Kotlerman et al., 2010; Lenci and Benotto, 2012). Geffet and Dagan (2005) introduce a pair of *distributional inclusion hypotheses*, where if a word v entails another word w, then all the typical features of the word v will also occur with the word w. Conversely, if all the typical features of v also occur with w, v is expected to entail w. Clarke (2009) defines a vector lattice for word vectors, and a notion of graded entailment with the properties of a conditional probability. Rimell (2014) explores the limitations of the distributional inclusion hypothesis by examining the properties of those features that are not shared between words. An interesting approach in Kiela et al. (2015) is to incorporate other modes of input into the representation of a word. Measures of entailment are based on the dispersion of a word representation, together with a similarity measure. All of these look at entailment at the word level. Related to the current work are the ideas in Balkır (2014); Balkır et al. (2016). In this work, the authors develop a graded form of entailment based on von Neumann entropy and with links to the distributional inclusion hypotheses developed by Geffet and Dagan (2005). The authors show how entailment at the word level carries through to entailment at the sentence level.

More recent approaches involve specialising word vectors for entailment Vulić and Mrkšić (2018), using non-Euclidean geometries Nickel and Kiela (2017); Nguyen et al. (2017); Le et al. (2019), and using pattern-based hyponymy extraction Roller et al. (2018); Le et al. (2019).

Most approaches, however, provide only word-word hyponymy. To test hyponymy in a compositional setting, we refer to the dataset of Kartsaklis and Sadrzadeh (2016) where a number of sentence and phrase-level hyponymy relationships are built from WordNet (Miller, 1995)

Another approach to detecting lexical entailment is via the identification of certain text patterns which indicate a hyponym-hypernym relationship. Examples are: *y such as x*, *x is a type of y*, which allow us to pick out pairs (x, y) which stand in the relation x $\texttt{is-a}$ y. This approach was first outlined in Hearst (1992) and has been recently used in Roller et al. (2018) to build vectors able to encode the required hierarchical relationships.

In the current paper we provide methods for building word representations as positive operators, using hierarchical information either from human-curated sources such as WordNet, or unsupervised methods such as using Hearst patterns. We will show how these word representations can be composed, and how the hierarchical information percolates to the phrase level. Our contribution is to provide a means of building hierarchically ordered word representations, that can be composed into phrases and sentences. Previous work in this area has either concentrated on word-level hyponymy or phrase-level hyponymy. In this paper we combine the two in one framework.

3 Methods

We model words as collections of vectors, as follows. For a given vector $\vec{v} \in V$,[1] we can 'lift' this vector into the larger space $V \otimes V$, by taking the outer product of the vector with itself. We use the following notation:

$$\bar{v} := \vec{v}\vec{v}^\top \tag{1}$$

When $\vec{v}$ is a unit vector, the resulting matrix $\bar{v}$ is a projection operator. Multiplying another vector $\vec{x}$ by $\bar{v}$ projects $\vec{x}$ onto the one-dimensional subspace spanned by $\vec{v}$. A matrix of the form $\bar{v}$ can be thought of as a collection of just one vector, giving sharp, unambiguous information.

To represent collections of more than one vector, we sum together their matrix representations, resulting in another matrix:

$$\begin{aligned}\{\vec{v}, \vec{w}, \vec{x}\} &\mapsto \bar{v} + \bar{w} + \bar{x} \in V \otimes V \\ &= \vec{v}\vec{v}^\top + \vec{w}\vec{w}^\top + \vec{x}\vec{x}^\top \in V \otimes V\end{aligned}$$

Matrices M built in this way are called *positive operators* and have the following two properties:

- $\forall v \in V. \langle \vec{v}, M\vec{v} \rangle \geq 0$
- M is self-adjoint.

If we additionally impose that M has trace 1, then we can understand M as encoding a probability distribution over $\vec{v} \in V$ (Nielsen and Chuang, 2010). In the present work, we do not impose this condition, instead viewing M as representing a collection of vectors.

The eigenvectors and eigenvalues of M can be thought of as providing a summary of the information contained in M. A matrix of the form $\bar{v} = \vec{v}\vec{v}^\top$ will have one non-zero eigenvalue, corresponding to the normalized eigenvector $\vec{v}/||\vec{v}||$. When multiple vectors have been included in the collection, the matrix M will have more than one non-zero eigenvalue, and these will represent the weights for their corresponding eigenvectors.

We take a kind of extensional stance. We consider words to be modelled as collections of their instances. To model a noun, we can consider the collection of nouns that are hyponyms of that noun, and form the matrix representation corresponding to that collection.

Example 1 (Nouns)**.** Consider the noun *pet*, and suppose we have three types of pet: a pug, a goldfish, and a tabby cat. We give these values in a distributional space spanned by the basis vectors $\{\overrightarrow{furry}, \overrightarrow{domestic}, \overrightarrow{working}, \overrightarrow{aquatic}\}$ as follows:

	pug	*goldfish*	*tabby*
furry	3	0	5
domestic	4	5	5
working	0	0	0
aquatic	0	6	0

We form the representation of the noun *pet* by summing over the matrix representations of each vector:

$$\begin{aligned}[\![pet]\!] &= \overline{pug} + \overline{goldfish} + \overline{tabby} \\ &= \overrightarrow{pug}\,\overrightarrow{pug}^\top + \overrightarrow{gfish}\,\overrightarrow{gfish}^\top + \overrightarrow{tabby}\,\overrightarrow{tabby}^\top \\ &= \begin{pmatrix} 34 & 37 & 0 & 0 \\ 37 & 66 & 0 & 30 \\ 0 & 0 & 0 & 0 \\ 0 & 30 & 0 & 36 \end{pmatrix}\end{aligned}$$

Each of the matrices $\overline{pug}$, $\overline{goldfish}$, and $\overline{tabby}$ has just one non-zero eigenvalue, which is $||\overrightarrow{v}||$, and corresponds to the normalised eigenvector $\overrightarrow{v}/||\overrightarrow{v}||$, for $\overrightarrow{v} = \overrightarrow{pug}$, $\overrightarrow{goldfish}$, and $\overrightarrow{tabby}$ respectively.

The matrix $[\![pet]\!]$, however, has three non-zero eigenvalues of 100.52, 35.21, and 0.25, each corresponding to a combination of the basis vectors $\overrightarrow{furry}, \overrightarrow{domestic}, \overrightarrow{aquatic}$. The basis vector $\overrightarrow{working}$ has an eigenvalue of 0, indicating that $[\![pet]\!]$ is orthogonal to the vector $\overrightarrow{working}$.

3.1 Ordering Positive Operators

The set of positive operators on a vector space has an ordering introduced by Löwner (1934). For positive operators A and B, we define:

$$A \sqsubseteq B \iff B - A \text{ is positive}$$

In Bankova et al. (2019) the authors introduce a notion of graded hyponymy. The hyponymy relation may be true up to some error term, as follows. If $A \sqsubseteq B$, then $B - A = D$, where D is some positive operator. If this does not hold, it is possible to add in some error term E so that $A \sqsubseteq B + E$. This is viewed as saying that A entails B to the extent E. We wish to find the smallest such error term.

In Bankova et al. (2019), the error term was of the form $(1 - k)A$ and the scalar $k \in [0, 1]$ gave a graded notion of hyponymy. The effect of this

[1] Throughout the paper, we assume that vector spaces are $\mathbb{R}^n$

scalar is to reduce the size of A until it 'fits inside' B, giving a notion of graded hyponymy that says that A is a k-hyponym of B, $A \sqsubseteq_k B$ if $B - kA$ is positive.

One of the drawbacks of this measure is that if the space spanned by eigenvectors of A, called $Span(A)$, is not a subspace of $Span(B)$, then the value of k must be 0. We therefore consider two new measures, which we now describe. If $B - A$ is not positive, it is possible to make it positive by adding in a positive operator constructed in the following manner.

1. Firstly diagonalize $B - A$, resulting in a real-valued matrix, since $B - A$ is real symmetric.

2. Construct a matrix E by setting all positive entries of $B - A$ to 0 and changing the sign of all negative eigenvalues.

Then $B - A + E$ will give us a positive matrix. This E is our error term. The size of E is bounded above by the size of A, since certainly $B - A + A$ is positive. We propose two different measures related to this error term that give us a grading for hyponymy.

The first measure is

$$k_{BA} = \frac{\sum_i \lambda_i}{\sum_i |\lambda_i|} \tag{2}$$

where λ_i is the ith eigenvalue of $B - A$ and $|\cdot|$ indicates absolute value. This measures the proportions of positive and negative eigenvalues in the expression $B - A$. If all eigenvalues are negative, $k_{BA} = -1$, and if all are positive, $k_{BA} = 1$. This measure is symmetric in the sense that $k_{BA} = -k_{AB}$.

Secondly, we propose

$$k_E = 1 - \frac{||E||}{||A||} \tag{3}$$

where $||\cdot||$ denotes the Frobenius norm. This measures the size of the error term as a proportion of the size of A. Since $A = E$ in the worst case, this measure ranges from 0 when $E = A$ to 1 when $E = 0$.

Example 2 (Full Hyponymy)**.** Recall the example

$$[\![\mathit{pet}]\!] = \overline{\mathit{pug}} + \overline{\mathit{goldfish}} + \overline{\mathit{tabby}}$$

To determine whether a goldfish is a pet, we calculate:

$$[\![\mathit{pet}]\!] - \overline{\mathit{goldfish}} = \overline{\mathit{pug}} + \overline{\mathit{tabby}}$$

Now, since $\overline{\mathit{pug}}$ and $\overline{\mathit{tabby}}$ are both positive, and positivity is preserved under addition, we know that $[\![\mathit{pet}]\!] - \overline{\mathit{goldfish}}$ is also positive. Therefore, under either of our graded measures, the extent to which a goldfish is a pet is 1.

Example 3 (Graded Hyponymy)**.** Now suppose that we define $[\![\mathit{dog}]\!] = \overline{\mathit{pug}} + \overline{\mathit{collie}}$, with $\overrightarrow{\mathit{pug}}$ and $\overrightarrow{\mathit{collie}}$ defined as below:

	pug	*collie*
furry	3	3
domestic	4	2
working	0	2
aquatic	0	0

Then to determine whether a dog is a pet, we calculate:

$$\begin{aligned}
&[\![\mathit{pet}]\!] - [\![\mathit{dog}]\!] \\
&\quad = \begin{pmatrix} 34 & 37 & 0 & 0 \\ 37 & 66 & 0 & 30 \\ 0 & 0 & 0 & 0 \\ 0 & 30 & 0 & 36 \end{pmatrix} - \begin{pmatrix} 18 & 18 & 6 & 0 \\ 18 & 20 & 4 & 0 \\ 6 & 4 & 4 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \\
&\quad = \begin{pmatrix} 16 & 19 & -6 & 0 \\ 19 & 46 & -4 & 30 \\ -6 & -4 & -4 & 0 \\ 0 & 30 & 0 & 36 \end{pmatrix}
\end{aligned}$$

The eigenvalues of $[\![\mathit{pet}]\!] - [\![\mathit{dog}]\!]$ are 75.38, 24.39, -5.77, 0, i.e., they are *not* all positive. This is because the subspace corresponding to $[\![\mathit{dog}]\!]$ is not a subspace of $[\![\mathit{pet}]\!]$, in particular because $[\![\mathit{dog}]\!]$ is not orthogonal to the basis vector $\overrightarrow{\mathit{working}}$.

However, much of $[\![\mathit{dog}]\!]$ is included in $[\![\mathit{pet}]\!]$. Using our graded measures given in equations (2) and (3), we can calculate that under k_{BA}, dog is a hyponym of pet to the extent 0.89 and under k_E, dog is a hyponym of pet to the extent 0.86.

3.2 Composing Positive Matrices

To compose positive matrices, we combine the methods outlined in Bankova et al. (2019) with the type-lifting methods outlined in Kartsaklis et al. (2012) to lift word representations into a higher-dimensional space. The impact of these methods are that we can define nouns and verbs within the same distributional space, and then lift the verb representations to a space that corresponds to a completely positive map.

We have choices about how to implement this type-lifting. One choice is composition, i.e. matrix multiplication, of the two operators. This latter operation results in a matrix that is no longer self-adjoint, and so Piedeleu (2014) suggests using the non-commutative and non-associative operator $M_2^{\frac{1}{2}} M_1 M_2^{\frac{1}{2}}$ in its place. This operator can be thought of as a kind of subspace projection, where M_1 is projected onto M_2. Piedeleu (2014) also notes that the pointwise multiplication of two positive operators is a completely positive map, giving us another choice for composition.

Following Kartsaklis et al. (2012), this gives us a method for building higher-level operators for verbs from lower-level operators. Firstly, we assume the noun space $N \otimes N$ to be equal to the sentence space $S \otimes S$, and refer to these both as $W \otimes W$. Given a representation of an intransitive verb $[\![verb]\!] \in W \otimes W$, the effect of lifting the verb to a higher order space and then composing with a noun $[\![noun]\!] \in W \otimes W$ is to apply the Frobenius multiplication to $[\![noun]\!] \otimes [\![verb]\!]$.

For intransitive verbs we can therefore combine the noun and the verb via three operations which we call **Mult**, **MMult1** (for matrix multiplication), and **MMult2**:

$$\text{Mult: } [\![n\ verb]\!] = [\![n]\!] \odot [\![verb]\!] \tag{4}$$

$$\text{MMult1: } [\![n\ verb]\!] = [\![n]\!]^{\frac{1}{2}} [\![verb]\!] [\![n]\!]^{\frac{1}{2}} \tag{5}$$

$$\text{MMult2: } [\![n\ verb]\!] = [\![verb]\!]^{\frac{1}{2}} [\![n]\!] [\![verb]\!]^{\frac{1}{2}} \tag{6}$$

For transitive verbs there is one possibility for pointwise multiplication of the operators, since this is both commutative and associative. For the second operation there are a number of composition orders. We will concentrate on two which reflect the difference between composing the verb with the object first and composing with the subject first. We therefore have:

$$\text{Mult: } [\![s\ v\ o]\!] = [\![s]\!] \odot [\![v]\!] \odot [\![o]\!] \tag{7}$$

$$\text{MMultO: } [\![s\ v\ o]\!] = [\![s]\!]^{\frac{1}{2}} [\![o]\!]^{\frac{1}{2}} [\![v]\!] [\![o]\!]^{\frac{1}{2}} [\![s]\!]^{\frac{1}{2}} \tag{8}$$

$$\text{MMultS: } [\![s\ v\ o]\!] = [\![o]\!]^{\frac{1}{2}} [\![s]\!]^{\frac{1}{2}} [\![v]\!] [\![s]\!]^{\frac{1}{2}} [\![o]\!]^{\frac{1}{2}} \tag{9}$$

4 Experimental Setting

We build representations of words as positive matrices, and selected from a number of alternative embeddings including GloVe vectors (Pennington et al., 2014), FastText (Bojanowski et al., 2017), and distributional vectors built from the concatenation of the UKWaC and Wacky corpora using PPMI and dimensionality reduction, all with 300 dimensions. To select the vector embeddings, we built word matrices as described above and tested them using the word vector evaluation package provided by Faruqui and Dyer (2014). We compared on the Rubinstein and Goodenough (Rubenstein and Goodenough, 1965), WordSim353, (Finkelstein et al., 2002), Miller and Charles (Miller and Charles, 1991), SimLex999 (Hill et al., 2015), SimVerb (Gerz et al., 2016), the Yang and Powers dataset (Yang and Powers, 2006) and the MEN dataset (Bruni et al., 2012). We did not inclue the rare word dataset or the similarity/relatedness splits of WS-353. Word matrices derived from GloVe vectors score best overall when using WordNet-derived hyponyms (table 1). This is also seen in the validation settings of the non-compositional datasets. When using Hearst-derived hyponyms, the generated word matrices perform poorly, although GloVe vectors still score most highly.

Table 1: Performance of word matrices derived from different word embeddings, using WordNet derived hyponyms. Bold highlights the highest value of each row.

	GloVe	Count	FastText
MC	**0.8441**	0.7124	0.8223
MEN	0.4836	0.2861	**0.5090**
RG	**0.8460**	0.6325	0.8387
SIMLEX-999	**0.4426**	0.2638	0.4272
SimVerb	**0.3458**	0.2158	0.3290
WS-353	0.3001	0.1677	**0.3033**
YP-130	**0.5619**	0.3465	0.5166

We compare our WordNet models to a symbolic model called Symb. For this model we mark two words w_1 and w_2 as being in the hyponym-hypernym relationship if w_1 is found in the transitive closure of the hyponyms of w_2.

4.1 Nouns

In order to build positive matrices for nouns, we use information about hypernymy relations. This information can be elicited using human built resources such as WordNet Miller (1995), or using Hearst patterns Hearst (1992). These are patterns like 'y such as x', which give markers for hyponym-hypernym pairs (x, y). To collect the hyponyms of a given word w from WordNet, we traverse the WordNet hierarchy and collect every word w_i in the transitive closure of the hyponymy relation.

For hyponyms generated by Hearst patterns, we use the publicly available dataset described in Roller et al. (2018), and refer the reader to that paper for details of its creation. The dataset consists of a set of word pairs $\mathcal{P} = \{(x_i, y_i)\}_i$ which are in a hyponym-hypernym relationship, together with the count $w(x, y)$ of the number of times that relationship has been seen in the text. As described in Le et al. (2019), the relationships thus extracted are both noisy and sparse, containing cycles and inconsistencies. As one example of this, the dataset contains the pair (*rome*, *european country*). To mitigate these phenomena, we apply a ppmi weighting to the counts. The weighting is as described in Roller et al. (2018), specifically

$$\text{ppmi}(x, y) = \max\left(0, \log \frac{p(x, y)}{p^{-}(x)p^{+}(y)}\right),$$

where, letting $W = \sum_{(x,y)\in\mathcal{P}}$, we have:

$$\begin{aligned} p(x, y) &= w(x, y)/W \\ p^{-}(x) &= \sum_{(x,y)\in\mathcal{P}} w(x, y)/W \\ p^{+}(x) &= \sum_{(y,x)\in\mathcal{P}} w(y, x)/W \end{aligned}$$

These equations give, respectively, the probability that the pair (x, y) is chosen from $\mathcal{P}$, the probability that x appears as a hyponym, and the probability that x appears as a hypernym. This sets the weighting of various unwanted pairs, such as the aforementioned (*rome*, *european country*), to 0.

To collect the set of hyponyms of a noun, we use only those hyponyms with a non-zero ppmi weighting, and take one transitive step. So the set of hyponyms of a given word x is the union of the sets $\{y_i|\text{ppmi}(x, y_i) > 0\}_i$ and $\{z_{ij}|\text{ppmi}(y_i, z_{ij}) > 0\}_{ij}$. We limit to one transitive step to again mitigate the noisiness of the dataset.

4.2 Verbs

WordNet contains verb hyponymy relationships, and therefore we can use similar methods to extract lists of hyponyms. However, we cannot use Hearst patterns to collate verb hyponymy relationships. As a proxy, we represent verbs as collections of their arguments. The intuition behind this is that of the extensional approach in formal semantics, mapped to distributional semantics in Grefenstette and Sadrzadeh (2011). We can think of both nouns and verbs as predicates, and consider the instances that the predicate applies to.

To collect the arguments of the verbs, we use the concatenation of the dependency parsed ukWaC and WaCky corpora (Ferraresi et al., 2008), and collect those arguments that appear at least 300 times in the corpus.

4.3 Building Matrices

Finally, having collected instances of nouns and verbs, we take the vectors $\overrightarrow{w}_i$ corresponding to each of these instances, take the outer product of each with itself, and add these together, i.e.:

$$[\![w]\!] = \sum_i \overrightarrow{w}_i \overrightarrow{w}_i^{\top} \in W \otimes W \qquad (10)$$

We have discarded weighting information. Words which have more instances are both more widely dispersed in terms of their eigenvalues, and also larger in terms of their matrix norm.

4.4 Datasets

We evaluate single word representations on the non-compositional BLESS hyponymy subset (Baroni and Lenci, 2011), WBLESS (Weeds et al., 2014), and BIBLESS (Kiela et al., 2015) datasets. We test our models using the hypernymy suite provided by Roller et al. (2018). The BLESS dataset requires the model to infer the direction of a hypernym pair. All pairs in the model are indeed in the hyponym-hypernym relationship, and the model must identify that this is the case. WBLESS consists of a set of pairs which may be in the hyponym-hypernym relationship, or unrelated. Each pair is assigned a value of 1 or 0 based on whether or not there is a hyponymy relationship. The software provided performs 1000 random iterations in which 2% of the data is used as a validation set to learn a classification threshold, and tests on the remainder of the data. Average accuracy across all iterations is reported. The BIBLESS dataset assigns values of 1, 0, and -1 based on whether the first word is a hyponym of the second, whether there is no relationship, or whether the second is a hyponym of the first. The software firstly tunes a threshold using 2% of the data, identifying whether a pair exhibits hypernymy in either direction. Secondly, the relative comparison of scores determines which direction is predicted. Again, the average accuracy over 1000 iterations is reported.

We further test on the compositional datasets from Kartsaklis and Sadrzadeh (2016). This is a series of three datasets, covering simple intransitive sentences, transitive sentences, and verb phrases. The intransitive verb dataset consists of paired sentences consisting of a subject and a verb. In half the cases the first sentence entails the second, and in the other half of cases, the order of the sentences is reversed. For example, we have:

> summer finish, season end, T
>
> season end, summer finish, F

The first sentence is marked as entailing, whereas the second is marked as not entailing. The dataset is created by selecting nouns and verbs from WordNet that stand in the correct relationship.

To test our models, we build the basic word representations as in equation (10). We then use the compositional methods outlined in section 3.2 to create the sentence representations. We calculate the graded entailment value between the composed sentence representations, and in results report area under ROC curve for comparison with previous literature. In particular, we compare with the best model from Kartsaklis and Sadrzadeh (2016), which uses a metric based on the distributional inclusion hypothesis, together with a tensor-based compositional model.

4.5 Significance Testing

To test significance of our results, we use bootstrapping Efron (1979) to calculate 100 values of the test statistic (either accuracy or AUC) drawn from the distribution implied by the data. We compare with figures from the literature using a one-sample t-test, and compare between models using a paired t-test. We apply the Bonferroni correction to compensate for multiple model comparisons.

5 Results

5.1 BLESS Variants

We present results on variants of the BLESS dataset in terms of accuracy, for comparison with other models, presented in table 2. Our best perfoming model is the WordNet based model with metric k_E. Althoough this model does not outperform the best supervised model (the differences in score are significant), the differences are fairly minimal (0.01 accuracy). Our methods (and those of others) outperform the symbolic baseline for the BLESS dataset. Our WordNet-based model does outperform the earlier model HyperVec with significance. Hearst-pattern based representations do not perform so strongly.

Table 2: Accuracy on the variants of the BLESS dataset. HyperVec figures are from Nguyen et al. (2017), Hearst from Roller et al. (2018), HypeCones from Le et al. (2019), LEAR from Vulić and Mrkšić (2018). Entries tagged with WN use WordNet.

Model	BLESS	WBLESS	BIBLESS
HyperVec - WN	0.92	0.87	0.81
Hearst	0.96	0.87	0.85
HypeCones	0.94	0.90	0.87
LEAR - WN	0.96	0.92	0.88
Symb - WN	0.91	0.93	0.91
k_{BA} - WN	0.95	0.88	0.84
k_E - WN	0.95	0.91	0.87
k_{BA} - Hearst	0.91	0.84	0.76
k_E - Hearst	0.91	0.86	0.80

Table 3: Area under ROC curve on the KS2016 datasets using k_E and WordNet derived hyponyms. For the SV and VO datasets, MMult1 refers to the model described in equation (5) and MMult2 refers to the model described in equation (6). For SVO, MMult1 refers to the model described in equation (8) and MMult2 refers to the model described in equation (9). $*$ indicates statistically significantly higher than the previous best performance Kartsaklis and Sadrzadeh (2016). $+$ indicates significantly higher than the additive baseline.

Model	SV	VO	SVO
KS2016 best	0.84	0.82	0.86
Verb only	0.870^{*}	0.944^{*}	0.908^{*}
Addition	0.941^{*}	0.948^{*}	0.972^{*}
Mult	0.975^{*+}	0.981^{*+}	0.978^{*}
MMult1	0.970^{*+}	0.971^{*+}	0.965^{*}
MMult2	0.967^{*+}	0.969^{*+}	0.971^{*}

Table 4: Area under ROC curve on the KS2016 datasets, using k_{BA} and WordNet derived hyponyms. Refer to Table 3 for explanations.

Model	SV	VO	SVO
KS2016 best	0.84	0.82	0.86
Verb only	0.902^{*}	0.967^{*}	0.931^{*}
Addition	0.970^{*}	0.964^{*}	0.978^{*}
Mult	0.974^{*}	0.984^{*+}	0.982^{*}
MMult1	0.987^{*+}	0.985^{*+}	0.995^{*+}
MMult2	0.987^{*+}	0.985^{*+}	0.995^{*+}

Table 5: Area under ROC curve on the KS2016 datasets, using k_E and Hearst-pattern derived hyponyms. Refer to Table 3 for explanations.

Model	SV	VO	SVO
KS2016 best	0.84	0.82	0.86
Verb only	0.714	0.808	0.716
Addition	0.877^{*}	0.807	0.912^{*}
Mult	0.887^{*}	0.808	0.864
MMult1	0.902^{*+}	0.824^{+}	0.883^{*}
MMult2	0.903^{*+}	0.800	0.877^{*}

Table 6: Area under ROC curve on the KS2016 datasets, using k_{BA} Hearst-pattern derived hyponyms. Refer to Table 3 for explanations.

Model	SV	VO	SVO
KS2016 best	0.84	0.82	0.86
Verb only	0.719	0.819	0.716
Addition	0.880^{*}	0.811	0.909^{*}
Mult	0.867^{*}	0.792	0.843
MMult1	0.909^{*+}	0.842^{*+}	0.930^{*+}
MMult2	0.904^{*+}	0.830^{*+}	0.924^{*+}

5.2 Compositional Datasets

On the KS2016 compositional datasets results are reported in terms of area under ROC curve. Our measures perform particularly well with WordNet derived hypernyms (Tables 3 and 4). This is likely to be due to the fact that both the dataset and our word representations were constructed from WordNet, and hence the high performance is to be expected. More interestingly, the word representations built using unsupervised methods also outperform previous best scores on this dataset, (tables 5 and 6), based on a form of the distributional inclusion hypothesis for tensor-based composition (Kartsaklis and Sadrzadeh, 2016), despite not performing so strongly on the single-word datasets.

6 Discussion and Further Work

We have suggested a mechanism for building the positive operators needed for the theory presented in Bankova et al. (2019), together with two novel measures of graded hyponymy. We tested these representations and measures on a number of well-known datasets, looking at similarity at the word level, hyponymy at the word level and one of which gives hyponymy at the phrase and sentence level. The representations and the measures we have developed perform competitively on these datasets. We have used both human-curated information and unsupervised methods to build the word representations. Unsurprisingly, human-curated information gives better performance.

A nice comparison is with the symbolic model. The fact that our WordNet-based models outperform this baseline shows that the models we propose can provide a 'smoothed' representation that allows hyponymy relationships to be inferred. For example, one of the hyponymy relationships not picked up in WordNet is *(oven, device)*. However, there are a number of other instances such as *(electric appliance, device)* that are similar enough to *oven* that *device* can be understood as including *oven*. What cannot be remedied, however, is where a term has no hyponyms in WordNet. For example, *herbivore* has no hyponyms in WordNet. This means that the WordNet-based representations have no way of forming a wide representation of *herbivore* thatincludes any of its instances.

As well as performing well on single-word hyponymy datasets, the representations we build sit within a compositional framework that allows us to form phrases and sentences and to reason about their entailment relationships. The WordNet-based representations behaved particularly well on this dataset, due to the fact that the dataset is built from WordNet. However, it is still an interesting set of results in that our graded measures interact well with the compositional methods we have proposed. Note that the measures we propose result in high baseline values to beat - i.e. for the verb-only and addition models. Again, this is likely due to the construction of the dataset. The dataset is formed from upwardly-monotone contexts, so computing entailment based only on the verb will still perform extremely well. Again, although this is due to the construction of the dataset, is is interesting to note that the measures and word representations we developed can model this structure so well. Furthermore the Hearst-pattern derived representations also outperformed previous work, indicating that these representations interact nicely with compositionality.

Similarities to our approach can be found in the notion of words being represented as Gaussians (Jameel and Schockaert, 2017; Vilnis and McCallum, 2014). The positive operators we build have the same structure as covariance matrices and, if appropriately normalized, are interpreted as representing a probability distribution over vectors. Representing words as Gaussians does not come with a given mechanism for composing words as we do. Exploring these connections is an area of further work.

Finally, a crucial extension to this whole approach is to be able to model hyponymy, composition, and their interaction in more contexts, for example using the natural logic introduced in Barwise and Cooper (1981).

Acknowledgements

Martha Lewis gratefully acknowledges funding from NWO Veni grant 'Metaphorical Meanings for Artificial Agents'

References

Esma Balkır. 2014. Using density matrices in a compositional distributional model of meaning. Master's thesis, University of Oxford.

Esma Balkır, Mehrnoosh Sadrzadeh, and Bob Coecke. 2016. Distributional sentence entailment using density matrices. In *Topics in Theoretical Computer Science*, pages 1–22. Springer.

Dea Bankova, Bob Coecke, Martha Lewis, and Dan Marsden. 2019. Graded hyponymy for compositional distributional semantics. *Journal of Language Modelling*, 6(2):225–260.

Marco Baroni and Alessandro Lenci. 2011. How we blessed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10. Association for Computational Linguistics.

Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1193. Association for Computational Linguistics.

Jon Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language. In *Philosophy, language, and artificial intelligence*, pages 241–301. Springer.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Samuel R Bowman, Christopher Potts, and Christopher D Manning. 2014. Recursive neural networks can learn logical semantics. *arXiv preprint arXiv:1406.1827*.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Daoud Clarke. 2009. Context-theoretic semantics for natural language: an overview. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 112–119. ACL.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2010. Mathematical foundations for a compositional distributional model of meaning. *arXiv:1003.4394*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer.

B. Efron. 1979. Bootstrap methods: Another look at the jackknife. *Ann. Statist.*, 7(1):1–26.

Manaal Faruqui and Chris Dyer. 2014. Community evaluation and exchange of word vectors at wordvectors.org. In *Proc. of ACL: System Demonstrations*.

Adriano Ferraresi, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can we beat Google*, pages 47–54.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on information systems*, 20(1):116–131.

John R Firth. 1957. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

Maayan Geffet and Ido Dagan. 2005. The distributional inclusion hypotheses and lexical entailment. In *Proceedings of the 43rd Annual Meeting on ACL*, pages 107–114. ACL.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A Large-Scale Evaluation Set of Verb Similarity. In *EMNLP*.

Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1394–1404, Stroudsburg, PA, USA. ACL.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Shoaib Jameel and Steven Schockaert. 2017. Modeling context words as regions: An ordinal regression approach to word embedding. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 123–133.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2016. Distributional inclusion hypothesis for tensor-based composition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2849–2860.

Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. 2012. A unified sentence space for categorical distributional-compositional semantics: Theory and experiments. In *In Proceedings of COLING: Posters*, pages 549–558.

Douwe Kiela, Laura Rimell, Ivan Vulic, and Stephen Clark. 2015. Exploiting image generality for lexical entailment detection. In *Proceedings of the 53rd Annual Meeting of the ACL*. ACL.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(04):359–389.

Matt Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. 2019. Inferring concept hierarchies from text corpora via hyperbolic embeddings. *arXiv preprint arXiv:1902.00913*.

Alessandro Lenci and Giulia Benotto. 2012. Identifying hypernyms in distributional semantic spaces. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, pages 75–79. ACL.

Karl Löwner. 1934. Über monotone Matrixfunktionen. *Mathematische Zeitschrift*, 38(1):177–216.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429.

Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Hierarchical embeddings for hypernymy detection and directionality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 233–243.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in neural information processing systems*, pages 6338–6347.

Michael A Nielsen and Isaac L Chuang. 2010. *Quantum Computation and Quantum Information*. Cambridge University Press.

Denis Paperno, Marco Baroni, et al. 2014. A practical and linguistically-motivated approach to compositional distributional semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 90–99.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Robin Piedeleu. 2014. Ambiguity in categorical models of meaning. Master's thesis, University of Oxford.

Robin Piedeleu, Dimitri Kartsaklis, Bob Coecke, and Mehrnoosh Sadrzadeh. 2015. Open system categorical quantum semantics in natural language processing. *arXiv:1502.00831*.

Laura Rimell. 2014. Distributional lexical entailment by topic coherence. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 511–519.

Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. *arXiv preprint arXiv:1806.03191*.

Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Luke Vilnis and Andrew McCallum. 2014. Word representations via gaussian embedding. *arXiv preprint arXiv:1412.6623*.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1134–1145.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259. Dublin City University and Association for Computational Linguistics.

Julie Weeds, David Weir, and Diana McCarthy. 2004. Characterising measures of lexical distributional similarity. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1015. ACL.

Dongqiang Yang and David M. W. Powers. 2006. Verb similarity on the taxonomy of wordnet. In *In the 3rd International WordNet Conference (GWC-06), Jeju Island, Korea*.