Revisiting Additive Compositionality: AND, OR, and NOT Operations with Word Embeddings

Anonymous ACL submission

Abstract

It is well-known that typical word embedding methods have the property that the meaning can be composed by adding up the embeddings (additive compositionality). Several theories have been proposed to explain additive compositionality, but the following problems remain: (i) The assumptions of those theories do not hold for the practical word embedding. (ii) Ordinary additive compositionality can be seen as an AND operation of word meanings, but it is not well understood how other operations, such as OR and NOT, can be computed by the embeddings. We address these issues by the idea of frequency-weighted centering at its core. This method bridges the gap between practical word embedding and the assumption of theory about additive compositionality as an answer to (i). This paper also gives a method for taking OR or NOT of the meaning by linear operation of word embedding as an answer to (ii). Moreover, we confirm experimentally that the accuracy of AND operation, i.e., the ordinary additive compositionality, can be improved by our post-processing method (3.5x improvement in top-100 accuracy) and that OR and NOT operations can be performed correctly. We also confirm that the proposed method is effective for BERT.

1 Introduction

Word embedding (Mikolov et al., 2013b; Pennington et al., 2014; Devlin et al., 2019), a fundamental technology in natural language processing, has the property that meaning can be composed by adding up the embeddings. This property is called *additive compositionality*, e.g., $\boldsymbol{v}_{king} \approx \boldsymbol{v}_{royal} + \boldsymbol{v}_{man}$ (Mikolov et al., 2013b). In this paper, we pose two questions about additive compositionality, as described below.

(i) Do the theories of additive compositionality adequately reflect the property of word embeddings in practice? Several theories have been proposed to explain why additive compositionality

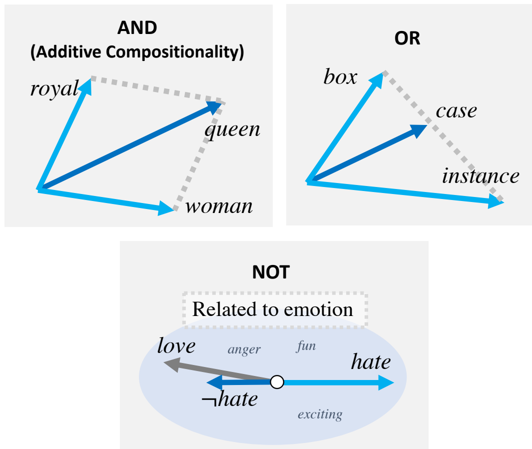

Figure 1: Illustration of AND, OR and NOT operations with word embeddings.

holds (Arora et al., 2016; Gittens et al., 2017; Allen and Hospedales, 2019); however, as we show in this paper, assumptions have been made that do not hold for practical word embedding. Besides, since these theories depend on specific methods such as Skip-Gram (Mikolov et al., 2013a), a unified understanding with other methods such as GloVe (Pennington et al., 2014) and BERT (Devlin et al., 2019) remains a challenge in the field of natural language processing.

(ii) Ordinary additive compositionality corresponds to operations that take logical AND of the word meanings, but how can operations that take the logical OR or NOT be computed by embeddings? OR corresponds to the compositionality of polysemous words (e.g. *case* $\approx$ *box* $\vee$ *instance*), and NOT corresponds to the compositionality of antonyms (e.g. *hate* $\approx \neg$*love*).

In this paper, these two questions are addressed by a simple idea of *frequency-weighted centering* of word vectors. We provide the following theoretical and experimental contributions for the above-mentioned questions.

1. We show that the theory of additive com-

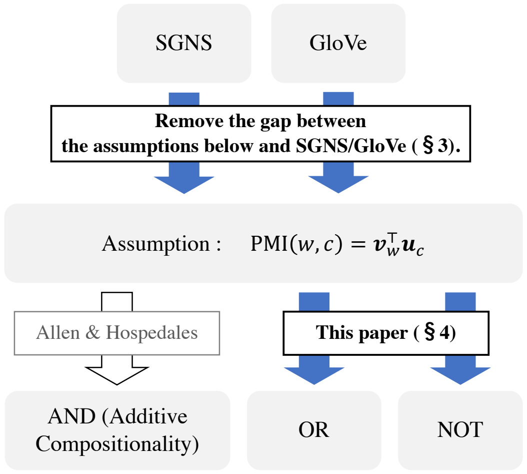

Figure 2: An overview of this paper and the previous research.

positionality (Allen and Hospedales, 2019) can be applied to practical word embeddings when word embeddings are centered using the frequency-weighted average of vocabulary words. To be precise, frequency-weighted centering works so that word embeddings approximately satisfy the assumptions of Allen and Hospedales (2019). This fact holds for both **SGNS** (Skip-Gram with Negative Sampling, one of the variations of Word2Vec by Mikolov et al. 2013b) and **GloVe** (Pennington et al., 2014). This implies that centering allows SGNS and GloVe to be described in an almost unified form. We also point out the similarities between the limited BERT architecture and SGNS, and suggest that frequency-weighted centering may be applicable to BERT as well.

2. Utilizing the results obtained in 1. as a starting point, we extend the theory of ordinary additive compositionality (AND) to compositionality of polysemy (OR) and antonym (NOT) (see Figure 1). OR operation is a frequency-weighted average for a specified subset of vocabulary words. NOT operation is based on a novel *conditional embedding* that is computed by frequency-weighted centering.

3. We experimentally confirm that our theory is correct (§5). The experimental results show that frequency-weighted centering makes additive compositionality, which corresponds to AND operation, holds more accurately (3.5x improvement in top-100 accuracy). We also confirm that this method is effective for BERT (Devlin et al., 2019). We also showed that the proposed formula can successfully compute OR and NOT embeddings.

2 Preliminaries: Word Embedding

In this section, we pointed out the similarities between BERT and SGNS, and briefly introduce some properties of popular word embedding methods. In the next section, we point out the gap between these properties and the assumption of the theory of additive compositionality (Allen and Hospedales, 2019), and propose a method to resolve it.

Word embedding methods based on co-occurrence information between words, such as SGNS and GloVe, are used across a wide range of fields such as information retrieval and recommendation systems (Roy et al., 2018; Grover and Leskovec, 2016; Grbovic et al., 2015).

Furthermore, BERT (Devlin et al., 2019), which has attracted attention in recent years, obtains word embeddings by predicting a word from its context, and can be regarded as an extension of SGNS. Consider a one-layer BERT model pre-trained by masked LM only. If the attention weight of a [MASK] token is one-hot vector, BERT predicts [MASK] from one context word and can be regarded as a Skip-gram model (Mikolov et al., 2013b). Thus, since BERT can be regarded as a generalization of Skip-gram, methods based on Skip-gram theory may be applicable to BERT.

SGNS and GloVe (and maybe BERT) encode the co-occurrence information of words. Levy and Goldberg (2014) showed that optimally trained SGNS embedding satisfies

$$\log \frac{p(w,c)}{p(w)q(c)} - \log k = \boldsymbol{v}_w^\top \boldsymbol{u}_c, \qquad (1)$$

where p is the word distribution of corpus, q is the distribution of negative samples, k is the number of negative samples per co-occurring word pair (w, c), $\boldsymbol{v}_w$ is the embedding of a target word w, and $\boldsymbol{u}_c$ is the embedding of a context word c. GloVe (Pennington et al., 2014) takes a direct approach to factorize the co-occurrence matrix, and the optimally learned embedding satisfies

$$\log p(w,c) = \boldsymbol{v}_w^\top \boldsymbol{u}_c + a_w + b_c - \log Z, \qquad (2)$$

where a_w, b_c are bias terms and Z is a normalization constant. In the following, we assume that SGNS and GloVe satisfy (1) and (2), respectively.

3 Structure Common to SGNS and GloVe

In this section, we show that when SGNS and GloVe are centered using the frequency-weighted average, they share a common structure.

Allen and Hospedales (2019) explained additive compositionality with the assumption

$$\mathrm{PMI}(w,c) = \boldsymbol{v}_w^\top \boldsymbol{u}_c, \tag{3}$$

where $\mathrm{PMI}(w,c)$ is the pointwise mutual information (PMI) between w and c

$$\mathrm{PMI}(w,c) := \log \frac{p(w,c)}{p(w)p(c)}. \tag{4}$$

However, neither SGNS nor GloVe satisfies the assumption (3), as explained in §3.1. If we can modify the word embeddings so that they satisfy assumption (3), then we can expect additive compositionality to more accurately hold.

In this section, we show a post-processing method for this modification, which can be applied to both SGNS and GloVe.

3.1 Error Terms in (3)

Rearranging the formulas of the word embedding assumptions (1) and (2), we have

$$\begin{aligned}
\textbf{SGNS} \quad & \mathrm{PMI}(w,c) \\
& = \boldsymbol{v}_w^\top \boldsymbol{u}_c + \log \frac{q(c)}{p(c)} + \log k, && (5) \\
\textbf{GloVe} \quad & \mathrm{PMI}(w,c) \\
& = \boldsymbol{v}_w^\top \boldsymbol{u}_c + (a_w - \log p(w)) \\
& \quad + (b_c - \log p(c)) - \log Z. && (6)
\end{aligned}$$

Clearly, they differ from (3). Allen and Hospedales (2019) ignores the second and subsequent terms on the right-hand side of (5) and (6); these ignored terms are considered as error terms in (3). The experiments in this paper show that the error terms are not so small as to be negligible (§5.1).

3.2 Frequency-weighted Centering

First, we show that (3) can be derived by centering the SGNS/GloVe embedding in a form that includes some error terms. Let the frequency-weighted average of word embeddings be $\bar{\boldsymbol{v}} = \sum_w p(w)\boldsymbol{v}_w$, $\bar{\boldsymbol{u}} = \sum_c p(c)\boldsymbol{u}_c$, and the centered word embeddings be $\tilde{\boldsymbol{v}}_w = \boldsymbol{v}_w - \bar{\boldsymbol{v}}$, $\tilde{\boldsymbol{u}}_c = \boldsymbol{u}_c - \bar{\boldsymbol{u}}$.

Theorem 1. When the embedding of SGNS and GloVe satisfies (1) and (2), respectively, the following equality holds:

$$\mathrm{PMI}(w,c) = \tilde{\boldsymbol{v}}_w^\top \tilde{\boldsymbol{u}}_c + \bar{\epsilon} - \epsilon_w - \epsilon_c, \tag{7}$$

where the error terms are defined, with KL-divergence, as $\epsilon_w = D_{\mathrm{KL}}(p(\cdot)\|p(\cdot|w))$, $\epsilon_c = D_{\mathrm{KL}}(p(\cdot)\|p(\cdot|c))$, and $\bar{\epsilon} = \sum_w p(w)\epsilon_w$.

Proof. See Appendix A. □

The following proposition shows that the error terms are negligible when $|\mathrm{PMI}(w,c)| \ll 1$.

Proposition 2. Let $\Delta = \max_{w,c} |\mathrm{PMI}(w,c)|$. For sufficiently small Δ, $\epsilon_w = O(\Delta^2)$, $\bar{\epsilon} = O(\Delta^2)$.

Proof. See Appendix B. □

3.3 Discussion

Interpretation Theorem 1 suggests that the centered SGNS and GloVe can be described in roughly the same form. In other words, we can say that (3) is a structure essentially common to SGNS and GloVe if properly centered.

Relation to experimental results The experiments described below confirm that the error in assumption (3) is greatly reduced by the frequency-weighted centering (§5.1), which supports the theory in §3.2. Furthermore, we have confirmed that the accuracy of additive compositionality is improved by frequency-weighted centering, as we expected. These improved word vectors are applicable to various downstream tasks.

Comparison with All-but-The-Top Mu and Viswanath (2018) suggested that uniform centering $\boldsymbol{v}_w \leftarrow \boldsymbol{v}_w - \sum_c \boldsymbol{v}_c/|V|$ is useful as a post-processing method to get high-performance word embeddings. This method is described as correcting the embeddings so that they satisfy isotropy, a property that the RAND-WALK model (Arora et al., 2016) should satisfy. However, its argument is not complete because the theoretical basis for RAND-WALK's high performance on each downstream task is not clearly stated. On the other hand, since our method is designed with the goal of satisfying the assumption of the theory of additive compositionality (Allen and Hospedales, 2019), there is a direct connection between our theory and the experimental results.

4 Logical Operations with Word Embeddings

In this section, we point out that ordinary additive compositionality is an AND-like operation, and show that other operations, such as OR and NOT, can also be computed by embeddings. We adopt assumption (3) in this section as well as Allen and Hospedales (2019); embeddings satisfying (3) can be obtained by simple post-processing of SGNS and GloVe (see §3).

4.1 AND Operation

Allen and Hospedales (2019) showed that when the PMI factorization structure (3) is strictly satisfied, a semantic AND composite such as *queen* $=$ *royal* $\wedge$ *woman* corresponds to vector additivity such as the following formula:

$$\boldsymbol{v}_{\textit{royal}} = \boldsymbol{v}_{\textit{royal}} + \boldsymbol{v}_{\textit{woman}}. \tag{8}$$

In this section, we outline the proof of Allen and Hospedales (2019).

4.1.1 Formulation with Co-occurrence Probability

Let $w = w_1 \wedge w_2 \wedge \cdots \wedge w_s$. Let us assume, for example, that *queen* meaning appearing is the multiplication of the probabilities of *royal* and *woman* meaning appearing. Generalizing this, we formulate AND-like compositionality as follows:

$$\forall c \in V, \quad p(w|c) = p(w_1|c) \cdots p(w_s|c), \tag{9}$$

$$p(w) = p(w_1) \cdots p(w_s) \tag{10}$$

4.1.2 Computation on Embedding Space

From the above formulation, additive compositionality is proved.

Theorem 3 (Allen and Hospedales 2019)**.** When $w, w_1, \ldots, w_s$ satisfy (3), (9) and (10),

$$\boldsymbol{v}_w = \sum_{i=1}^{s} \boldsymbol{v}_{w_i}. \tag{11}$$

Proof. Dividing (9) by (10) and taking the logarithm, we get $\mathrm{PMI}(w, c) = \mathrm{PMI}(w_1, c) + \cdots + \mathrm{PMI}(w_s, c)$. From (3), $\boldsymbol{v}_w^\top \boldsymbol{u}_c = \boldsymbol{v}_{w_1}^\top \boldsymbol{u}_c + \cdots + \boldsymbol{v}_{w_s}^\top \boldsymbol{u}_c$. Since $c \in V$ is arbitrary, (11) follows. □

4.2 OR Operation

As mentioned in §1, in addition to AND operation, OR operation can also be considered. In this section, we show that OR operation corresponds to the frequency-weighted average of the embeddings for a set of words.

4.2.1 Formulation with Co-occurrence Probability

OR operation is denoted by operator $\vee$. Let w be the OR word of $w_1, w_2, \ldots, w_s$, i.e. $w = w_1 \vee w_2 \vee \cdots \vee w_n$. For example, *case* $\approx$ *box* $\vee$ *instance*. The probability of occurrence of w in each context c can be formulated as the sum of the probabilities of occurrence of $w_1, \ldots, w_s$:

$$\forall c \in V, \quad p(w|c) = p(w_1|c) + \cdots + p(w_s|c). \tag{12}$$

Taking *case* $\approx$ *box* $\vee$ *instance* as an example, the co-occurrence probability of *case* is the sum of the co-occurrence probabilities of *box* and *instance* in each context c. This formulation clearly holds for the polysemous word w composed of imaginary words $w_1, \ldots, w_s$. From (12), we get $p(w) = \sum_{i=1}^{s} p(w_i)$.

4.2.2 Computation on Embedding Space

On the basis of the above simple formulation, we give a method to perform OR operation on the embeddings.

Theorem 4 (OR formula)**.** We assume that $w, w_1, w_2, \ldots, w_s$ satisfy (3) and (12). When $|\mathrm{PMI}(w, c)| \ll 1$, word embeddings satisfy

$$\boldsymbol{v}_w \approx \sum_{i=1}^{s} \frac{p(w_i)}{p(w)} \boldsymbol{v}_{w_i}. \tag{13}$$

Proof. See Appendix C. □

OR formula (13) suggests that the embedding of *case* approximates the sum of the embeddings of *box* and *instance*, weighted by their probability of occurrence in the corpus. Note that the OR formula is invariant to the translation of the origin, so it is valid to some extent for SGNS and GloVe without the frequency-weighted centering.

4.2.3 Discussion

Relation to experimental results On the real data, $|\mathrm{PMI}(w, c)| \ll 1$ does not strictly hold, but we confirmed that the OR formula holds well in the experiment in §5.3.

Comparison with previous work Arora et al. (2018) obtained the same formula by assuming the random walk of the context vector (RAND-WALK model), but the proof in this paper does not require that assumption.

4.3 Conditional Embedding and NOT Operation

With assumption (3), we derive not only AND and OR operations but also NOT operation. In this section, we formulate the NOT operation using the concept of *conditional embedding*, word embedding that expresses the local relationship between words in a small set of words A. We derive that the conditional embedding of the antonym is proportional to *minus* of the conditional embedding of the original word.

4.3.1 Formulation with Co-occurrence Probability

In contrast to human senses, antonyms have the property of being dissimilar and similar at the same time (Cruse, 1986; Willners, 2001), e.g., *hate* and *love* have opposite meanings, but both of them are related to emotion. For this reason, antonyms tend to appear in similar contexts, and thus their word embeddings trained by the method based on the distributional hypothesis (Harris, 1954; Firth, 1957) exhibit a high similarity[1]. Therefore, antonyms are quite related to synonyms, which makes it difficult to understand how antonyms are embedded. In this section, we dispense with the mystery of antonyms by formulating them in a way that takes their similarity into account.

Let us take the following example: the opposite of *mother* is *father* in the "parent" category, but *daughter* in the "parent-child relationship" category. In this way, when considering antonyms, one needs to specify a category that corresponds to the similarity portion of the antonyms. In this paper, a category is represented by a set of words A. From the intuition that the antonym $\neg w$ of word $w \in A$ corresponds to the complement of w in A when viewed in a small word set A, the co-occurrence probability of NOT word can be formulated by the following conditional probability:

$$\begin{aligned} &p(W = \neg w \mid W \in A, c) \\ &\qquad = p(W \in A \setminus \{w\} \mid W \in A, c), \end{aligned} \tag{14}$$

where word W denotes a random variable and $p(\cdot|\cdot)$ denotes the conditional probability. Because the event $W \in A$ appears in the conditioning for the probability of (14), we need embeddings conditioned on A instead of the whole vocabulary. In this paper, we refer to this embedding as *conditional embedding on A*.

4.3.2 Conditional Embedding

From the analogy to (3), we consider the equality to be satisfied by the conditional embedding $\boldsymbol{v}_{w|A}$ of the word w in set A as follows:

$$\begin{aligned} &p(W = w \mid W \in A, c) \\ &\qquad = p(W = w \mid W \in A) \exp(\boldsymbol{v}_{w|A}^\top \boldsymbol{u}_c). \end{aligned} \tag{15}$$

From the following Theorem 5, we can see that conditional embedding can be approximated by frequency-weighted centering on a subset A.

Theorem 5. When embeddings satisfy (3),

$$\boldsymbol{v}_{w|A} \approx \boldsymbol{v}_w - \boldsymbol{v}_A, \tag{16}$$

where $p(A) = p(W \in A) = \sum_{w \in A} p(w)$ and $\boldsymbol{v}_A = \sum_{w\in A} \frac{p(w)}{p(A)} \boldsymbol{v}_w$.

Proof. See Appendix D. □

Theorem 5 allows us to explain the common practice of centering, although typically unweighted, on a particular set of words (e.g., implicit centering in PCA visualization of the word embedding of country-capital) as "elaboration" of the interrelationships between words in the set of words.

4.3.3 Computation on Embedding Space

On the basis of the above formulation, we derive a method for computing NOT with word embeddings.

Theorem 6 (NOT formula)**.** Assuming that the words w and $\neg w|A$ satisfy (15), we have

$$\boldsymbol{v}_{\neg w|A} \approx -\frac{p(W = w \mid W \in A)}{1 - p(W = w \mid W \in A)} \boldsymbol{v}_{w|A}. \tag{17}$$

Proof. See Appendix E. □

From this formula, we can see that the conditional embedding of the NOT word of w is the vector in the negative direction of the conditional embedding of the original word w.

5 Experiments

In order to keep the description concise, the detailed experimental setup is described in Appendix F.

5.1 Centering and PMI Factorization

In this section, we experimentally confirm that (3) holds more accurately if we perform frequency-weighted centering (§3.2). To show that the accuracy of PMI factorization formula $\mathrm{PMI}(w, c) = \boldsymbol{v}_w^\top \boldsymbol{u}_c$ is improved by centering the embeddings, we observed the distribution of the error $e_{wc} = \mathrm{PMI}(w, c) - \boldsymbol{v}_w^\top \boldsymbol{u}_c$ in several experimental settings.

[1]The cosine similarity between *hate* and *love* is about 0.5 for vectors obtained by SGNS (without centering).

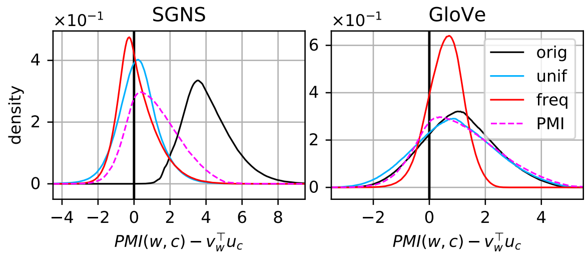

Figure 3: The distribution of the error e_{wc} is plotted for the word pairs (w, c) that co-occur more than once. The distribution of $\mathrm{PMI}(w, c)$ itself is shown as purple dashed line for reference of the error order.

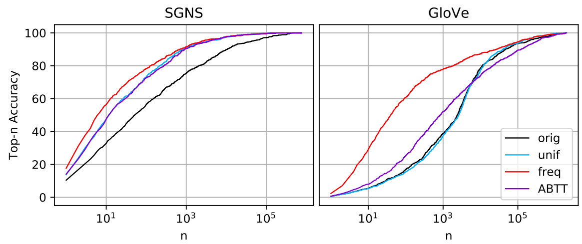

Figure 4: Top-n accuracy of rank (word-to-phrase). Upper left is better.

Embeddings We use 300-dimensional embeddings trained by SGNS and GloVe with text8 corpus[2]. The results are compared between the following three set of embeddings.

- orig : original embeddings.
- freq : embeddings with frequency-weighted centering.
- unif : embeddings with uniform centering[3] (Mu and Viswanath, 2018), i.e., $\boldsymbol{v}_w \leftarrow \boldsymbol{v}_w - \sum_{w'} \boldsymbol{v}_{w'}/|V|$.

Results We plot the histogram of e_{wc} (Figure 3). We see that the magnitude of error e_{wc} of frequency-weighted centering (freq) is small and $\mathrm{PMI}(w, c) = \boldsymbol{v}_w^\top \boldsymbol{u}_c$ holds more accurately *regardless of whether the method is SGNS or GloVe*. It is worth noting that frequency-weighted centering (freq) and uniform centering (unif) have substantially different results, which is non-trivial.

5.2 Assessing Accuracy of AND Formula

From Theorem 1, Proposition 2 and Allen and Hospedales (2019), frequency-weighted centering is expected to result in stronger additive compositionality (§3). In this section, we experimentally confirm that additive compositionality holds more accurately by frequency-weighted centering. The experiments are for three types of additive compositionality: word-to-sentence, word-to-phrase, and word-to-word. We also experimentally see that the same result holds for BERT as well as SGNS and GloVe.

Embeddings We use 300-dimensional embeddings trained by SGNS and GloVe with Wikipedia[4]. For BERT embeddings, we used the first layer[5], which corresponds to the target vector of the skip-gram[6]. We compare four types of embeddings: orig , unif , freq and All-but-the-Top (Mu and Viswanath, 2018) (ABTT , for SGNS and GloVe), a post-processing method for embeddings that incorporates uniform centering.

5.2.1 Word-to-sentence compositionality

We evaluate sentence vectors by simply adding word vectors using semantic textual similarity task (Agirre et al., 2012). If additive compositionality holds more accurately, it is expected that sentence vectors are more accurate and scores increase.

Results The results are shown in Table 1. As we can see, our proposed method freq consistently performs the best. All-but-the-Top is also a post-processing method for correcting the embeddings (Arora et al., 2016), but ours is better in terms of additive compositionality.

5.2.2 Word-to-phrase compositinality

We evaluate how strongly word-to-phrase additive compositionality holds by learning phrase vectors.

Preprocessing of Corpus We train phrase vectors by treating multiple words as single word, i.e.*card game* → *card_game*. Only phrases with high compositionality included in Farahmand et al. (2015); Ramisch et al. (2016); Reddy et al. (2011) were used[7].

[2] http://mattmahoney.net/dc/textdata.html

[3] This is a standard post-processing of word embedding (Mu and Viswanath, 2018).

[4] https://dumps.wikimedia.org/

[5] In the word-to-sentence experiment, the results for the final layer are also included (§G.1); in the word-to-word experiment, only the results for the first layer are included because there is no point in contextualizing the word embedding.

[6] https://huggingface.co/bert-base-uncased

[7] These datasets include human ratings of the compositionality of phrases. Since words with weak compositionality are not suitable for the additive compositionality experiment, only

	SGNS				GloVe				BERT		
STSx	orig	freq	unif	ABTT	orig	freq	unif	ABTT	orig	freq	unif
12	0.53	**0.55**	0.53	0.52	0.32	**0.37**	0.32	0.35	0.52	**0.53**	0.49
13	0.59	**0.64**	0.59	0.57	0.37	**0.48**	0.37	0.43	0.49	**0.52**	0.47
14	0.60	**0.68**	0.60	0.59	0.38	**0.52**	0.38	0.44	0.57	**0.62**	0.54
15	0.62	**0.70**	0.62	0.60	0.44	**0.55**	0.44	0.48	0.60	**0.68**	0.57
16	0.55	**0.65**	0.55	0.53	0.33	**0.51**	0.33	0.38	0.60	**0.69**	0.57

Table 1: Results of semantic textual similarity tasks.

	orig	freq	unif	ABTT
SGNS	0.028	0.071	0.072	**0.074**
GloVe	0.067	**0.078**	0.065	0.057
BERT	0.036	**0.062**	0.044	—

Table 2: MRR of rank (word-to-word).

Evaluation We calculated the cosine similarities between $\boldsymbol{v}_{word1} + \boldsymbol{v}_{word2}$ and all the $\boldsymbol{v}_w, w \in V$, and how many words had a cosine similarity greater than or equal to the cosine similarity between $\boldsymbol{v}_{word1\_word2}$ and $\boldsymbol{v}_{word1} + \boldsymbol{v}_{word2}$; this number is simply denoted as *rank*[8].

Results The top-n accuracy of rank is shown in Figure 4. The overall results show that centering, especially with frequency weights, improves the accuracy for additive compositionality. For SGNS, the top-10 accuracy improves by 1.7 times, and for GloVe, the top-100 accuracy improves by 3.5 times. Moreover, the results for GloVe are significantly different between uniform and frequency-weighted centering, which is consistent with the results in §5.1.

5.2.3 Word-to-word compositinality

We evaluate the additive compositionality of a word from words such as *royal+woman=queen*.

Dataset BATS (Gladkova et al., 2016), the dataset for the analogy task, specifies the relationship between the two words: the file `country-capital` contains word pairs such as *bankok*:*thailand* and *beijing*:*china*. For example, by assigning *thailand* to x, *capital* to y, and *bankok* to z, we create a dataset of triplets of words for which $x + y = z$.

Evaluation We use ranks of $\boldsymbol{v}_x + \boldsymbol{v}_y$ and $\boldsymbol{v}_z$ for evaluation. Mean Reciprocal Rank (MRR) is used as the representative value.

Results Table 2 shows the results. One can see that the proposed method freq consistently contributes to the performance improvement of additive compositionality. We can also see that, for GloVe and BERT, freq is superior to the other methods. freq loses to unif in SGNS, but this is related to the lack of significant difference in the structure of embeddings between unif and freq, as can be seen in Figure 3.

5.3 Assessing Accuracy of OR Formula

In this section, we confirm that the OR formula (13) is valid.

Embeddings We use 300-dimensional embeddings trained by SGNS and GloVe with Wikipedia based corpus.

Preprocessing of Corpus We generated 500 artificial polysemous words and learned their embeddings as follows. We constructed artificial polysemous words from two randomly selected words (e.g. *apple*, *banana* $\to$ *apple_OR_banana*), and create a new corpus in which all the selected words are replaced by the artificial polysemous words. Then we concatenate the original corpus with the new corpus and used it to train word embeddings.

Evaluation As in § 5.2, we evaluated the OR formula by the rank of *word1_OR_word2*.

Results The average rank was **1.012** for SGNS and **1.000** for GloVe, surprisingly good performance. Even though the OR formula is an approximation, the precision of the OR formula is high enough that it is almost always able to predict

phrases with a rating of 3 or 4 were used in Farahmand et al. (2015) and only phrases with a rating of 3.0 or higher were used in Reddy et al. (2011) or Ramisch et al. (2016).

[8] We should not simply use the similarity between $\boldsymbol{v}_{word1} + \boldsymbol{v}_{word2}$ and $\boldsymbol{v}_{word1\_word2}$ as the accuracy of additive compositionality. To be precise, if the similarity with $\boldsymbol{v}_{word1\_word2}$ is high and the similarity with other embeddings is low, we can say that additive compositionality is accurate. This paper uses rank as a metric that satisfies this requirement.

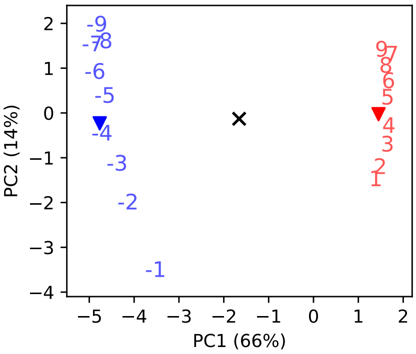

Figure 5: The embeddings of numbers are visualized using PCA. The $\times$ is the origin of the conditional embedding $\boldsymbol{v}_{w|A}$ when $A = \{-9, \ldots, 9\} \setminus \{0\}$, and red triangle and blue triangle are the origins of A for positive numbers and negative numbers, respectively.

correct answer (among 2M words). Since the OR formula is translation-invariant, the result is the same for orig , unif and freq .

5.4 Observation of NOT Formula

The visualization of the embeddings of the numbers from -9 to 9 is shown in Figure 5[9]. Confining attention to $A = \{1, \ldots, 9\}$, 1 and 9 are located in the negative direction each other across the origin (red triangle) in the conditional embedding of A; this confirms the NOT formula (29). On the other hand, if we expand set A to include negative numbers, 1 and 9 are located in a similar direction from the origin (black $\times$). In this case, the positive and negative numbers are on the opposite sides of the origin, which supports the NOT formula again. As you can see, it is important to determine the category in which antonyms are considered, and the NOT formula is able to formulate this fact well.

6 Connection to Previous Work

The summaries of the previous researches on additive compositionality and their relationships to this study are given below.

- Arora et al. (2016, 2018) explained the operations of Analogy and OR by considering a latent variable model. On the other hand, there is a slight gap between the embedding properties suggested by their theory and the properties of the word embeddings used in practice. For example, their theory shows that high-frequency words have large norm, but in actual word embeddings, the norm of medium and low frequency words is large (Schakel and Wilson, 2015), and this is one of the reasons why additive construction works well (Yokoi et al., 2020). Our theory describes a more realistic embedding model.

- Gittens et al. (2017) explained the AND operation with the assumption $p(w) = 1/|V|$ in the Skip-Gram model (Mikolov et al., 2013a). While their theory succeeds in explaining the essential reason for additive compositionality, note that it makes the assumption that all words have the same frequency, an assumption that does not hold in practice. Word frequencies are known to have a skewed distribution (Piantadosi, 2014), and a feature of our theory is that we actively incorporate this non-uniform distribution into our theory (§4).

- Allen and Hospedales (2019) explained additive compositionality (AND) and Analogy starting from assumption (3). Our theory is positioned as contributing to the elaboration of their theory by resolving the problems of arbitrariness of bias terms in GloVe and the $\log k$ shift in SGNS, which they had raised as an issue in their own theory.

7 Conclusion

In this paper, we show that when frequency-weighted centering is performed, SGNS and GloVe share a common structure and additive compositionality becomes more accurate. We also show how to compute OR and NOT operations by word embeddings in addition to the ordinal additive compositionality (AND).

Simple models such as SGNS and GloVe are explained theoretically in this paper, but our theory is not directly applied to more complex models such as BERT (Devlin et al., 2019). We confirmed the effectiveness of our method on BERT, though experimentally. However, we do not know how general the results of the BERT experiment in this paper are. As future work, we aim to interpret BERT theoretically and explain the results of these experiments on BERT, and clarify the generality of the experimental results.

[9] The origin of the conditional embedding is calculated with uniform weights instead of the actual frequencies.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393.

Carl Allen and Timothy Hospedales. 2019. Analogies explained: Towards understanding word embeddings. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 223–231.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. A latent variable model approach to PMI-based word embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2018. Linear Algebraic Structure of Word Senses, with Applications to Polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495.

David A. Cruse. 1986. *Lexical Semantics*. Cambridge University Press, Cambridge, UK.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Meghdad Farahmand, Aaron Smith, and Joakim Nivre. 2015. A Multiword Expression Data Set: Annotating Non-Compositionality and Conventionalization for English Noun Compounds. In *Proceedings of the 11th Workshop on Multiword Expressions*, pages 29–33.

J. R. Firth. 1957. A synopsis of linguistic theory 1930-55. *Studies in Linguistic Analysis (special volume of the Philological Society)*, 1952-59:1–32.

Alex Gittens, Dimitris Achlioptas, and Michael W. Mahoney. 2017. Skip-gram - Zipf + Uniform = Vector Additivity. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 69–76.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15.

Mihajlo Grbovic, Vladan Radosavljevic, Nemanja Djuric, Narayan Bhamidipati, Jaikit Savla, Varun Bhagwan, and Doug Sharp. 2015. E-Commerce in Your Inbox: Product Recommendations at Scale. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1809–1818.

Aditya Grover and Jure Leskovec. 2016. Node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 855–864.

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

Omer Levy and Yoav Goldberg. 2014. Neural Word Embedding as Implicit Matrix Factorization. In *Advances in Neural Information Processing Systems*, pages 2177–2185.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR, Workshop Track Proceedings*.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Steven Piantadosi. 2014. Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic bulletin and review*, 21.

Carlos Ramisch, Silvio Cordeiro, Leonardo Zilio, Marco Idiart, and Aline Villavicencio. 2016. How Naked is the Naked Truth? A Multilingual Lexicon of Nominal Compound Compositionality. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 156–161.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An Empirical Study on Compositionality in Compound Nouns. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 210–218.

Dwaipayan Roy, Debasis Ganguly, Sumit Bhatia, Srikanta Bedathur, and Mandar Mitra. 2018. Using Word Embeddings for Information Retrieval: How Collection and Term Normalization Choices Affect Performance. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, page 1835–1838.

Adriaan M. J. Schakel and Benjamin J. Wilson. 2015. Measuring Word Significance using Distributed Representations of Words. *CoRR*, abs/1508.02297.

Caroline Willners. 2001. *Antonyms in Context : A Corpus-Based Semantic Analysis of Swedish Descriptive Adjectives*. Ph.D. thesis, General Linguistics.

Sho Yokoi, Ryo Takahashi, Reina Akama, Jun Suzuki, and Kentaro Inui. 2020. Word rotator's distance. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2944–2960.

Appendices

A Proof of Theorem 1

Proof. For SGNS, let $\zeta_w = 0, \xi_c = \log \frac{q(c)}{p(c)}, \gamma = \log k$; for GloVe, let $\zeta_w = a_w - \log p(w), \xi_c = b_c - \log p(c), \gamma = -\log Z$. Then, from (1), (2), we get

$$\mathrm{PMI}(w,c) = \boldsymbol{v}_w^\top \boldsymbol{u}_c + \zeta_w + \xi_c + \gamma. \quad (18)$$

Multiplying both sides of (18) by $p(w)$ and summing with respect to $w \in V$, we get

$$-\epsilon_c = \bar{\boldsymbol{v}}^\top \boldsymbol{u}_c + \bar{\zeta} + \xi_c + \gamma, \quad (19)$$

where $\bar{\zeta} = \sum_{w \in V} p(w)\zeta_w$. From (18) and (19):

$$\mathrm{PMI}(w,c) = \tilde{\boldsymbol{v}}_w^\top \boldsymbol{u}_c + (\zeta_w - \bar{\zeta}) - \epsilon_c. \quad (20)$$

Multiplying both sides of (20) by $p(c)$ and summing with respect to $c \in V$, we get

$$-\epsilon_w = \tilde{\boldsymbol{v}}_w^\top \bar{\boldsymbol{u}} + (\zeta_w - \bar{\zeta}) - \bar{\epsilon} \quad (21)$$

From (20) and (21), we have

$$\mathrm{PMI}(w,c) = \tilde{\boldsymbol{v}}_w^\top \tilde{\boldsymbol{u}}_c + \bar{\epsilon} - \epsilon_w - \epsilon_c$$

□

B Proof of Proposition 2

Proof. There exists $c_1 > 0$ such that for all $(w,c) \in V^2$,

$$\begin{aligned}&\left|-1 + \frac{p(w,c)}{p(w)p(c)}\right| \\ &\quad = |-1 + \exp(\mathrm{PMI}(w,c))| < c_1 \Delta. \quad (22)\end{aligned}$$

$$\begin{aligned}\epsilon_w &= -\sum_{c \in V} p(c) \log \frac{p(w,c)}{p(w)p(c)} \\ &= -\sum_{c \in V} p(c) \left[\left(-1 + \frac{p(w,c)}{p(w)p(c)}\right) \right. \\ &\qquad \left. + O\left(\left|-1 + \frac{p(w,c)}{p(w)p(c)}\right|^2\right) \right] \\ &= \sum_{c \in V} p(c) - \sum_{c \in V} p(c|w) \\ &\qquad - \sum_{c \in V} p(c) O\left(\left|-1 + \frac{p(w,c)}{p(w)p(c)}\right|^2\right) \\ &= -\sum_{c \in V} p(c) O\left(\left|-1 + \frac{p(w,c)}{p(w)p(c)}\right|^2\right). \quad (23)\end{aligned}$$

Therefore, there exists $c_2 > 0$ such that for all $w \in V$,

$$|\epsilon_w| < \sum_{c \in V} p(c) c_2 c_1^2 \Delta^2 = c_1^2 c_2 \Delta^2. \tag{24}$$

$|\bar{\epsilon}| < c_1 c_2 \Delta^2$ also readily follows. □

C Proof of Theorem 4

Proof. Calculating both sides of (12), we get

$$\begin{aligned} p(w|c) &= p(w) \exp(\mathrm{PMI}(w, c)) \\ &\approx p(w) \left(1 + \mathrm{PMI}(w, c)\right) \\ &= p(w)(1 + \boldsymbol{v}_w^\top \boldsymbol{u}_c), \end{aligned} \tag{25}$$

$$\begin{aligned} \sum_{i=1}^{s} p(w_i|c) &= \sum_{i=1}^{s} p(w_i) \exp(\mathrm{PMI}(w_i, c)) \\ &\approx \sum_{i=1}^{s} p(w_i)(1 + \boldsymbol{v}_{w_i}^\top \boldsymbol{u}_c) \\ &= p(w) \left[1 + \left(\sum_{i=1}^{s} \frac{p(w_i)}{p(w)} \boldsymbol{v}_{w_i} \right)^\top \boldsymbol{u}_c \right]. \end{aligned} \tag{26}$$

(13) follows the fact that for any $c \in V$, (25) $\approx$ (26) . □

D Proof of Theorem 5

Proof. Calculating the left-hand side of (15) using assumption (3) and OR formula (13), we get

$$\begin{aligned} & p(W = w \mid W \in A, c) \\ &= \frac{p(W = w, W \in A \mid c)}{p(W \in A \mid c)} \approx \frac{p(w) \exp(\boldsymbol{v}_w^\top \boldsymbol{u}_c)}{p(A) \exp(\boldsymbol{v}_A^\top \boldsymbol{u}_c)} \\ &= p(W = w \mid W \in A) \exp((\boldsymbol{v}_w - \boldsymbol{v}_A)^\top \boldsymbol{u}_c), \end{aligned} \tag{27}$$

By comparing (15) and (27), we get (16). □

E Proof of Theorem 6

Proof. By using (13), the right-hand side of (14) is rearranged as

$$\begin{aligned} & p(W \in A \setminus \{w\} \mid W \in A, c) \\ &\qquad = \frac{p(W \in A \setminus \{w\} \mid c)}{p(W \in A \mid c)} \\ &\approx \frac{p(A \setminus \{w\})}{p(A)} \exp((\boldsymbol{v}_{A \setminus \{w\}} - \boldsymbol{v}_A)^\top \boldsymbol{u}_c). \end{aligned} \tag{28}$$

Thus $\boldsymbol{v}_{\neg w|A} \approx \boldsymbol{v}_{A \setminus \{w\}} - \boldsymbol{v}_A$, and further calculation yields

$$\begin{aligned} \boldsymbol{v}_{\neg w|A} &\approx \frac{p(A)}{p(A) - p(w)} \left(\boldsymbol{v}_A - \frac{p(w)}{p(A)} \boldsymbol{v}_w \right) - \boldsymbol{v}_A \\ &= -\frac{p(W = w \mid W \in A)}{1 - p(W = w \mid W \in A)} \boldsymbol{v}_{w|A}. \end{aligned} \tag{29}$$

□

F Details of Experiments

The default parameters of the implementation[10] were used for all but the most notable cases.

F.1 Details of §5.1

Corpus text8 corpus[11], from which low-frequency words (< 100) were removed.

Hyperparameters for learning word embeddings We run 100 iterations for 300-dimensional vectors. The size of the context window is 5 words (symmetric context). For SGNS, the number of negative samples k is 15 and subsampling of high-frequency words was disabled. For GloVe, the parameter for the weights of the least-squares method $x_{\max}$ is 100.

Others For freq and unif, $\boldsymbol{u}_c$ is also centered.

F.2 Details of §5.2 and §5.3

Corpus Wikipedia[12](2.1G tokens)

Hyperparameters for learning word embeddings The dimension of the word embeddings is 300 and the size of the context window is 5 words. For SGNS, the number of negative samples k is 15. For GloVe, the parameter for the weights of the least-squares method $x_{\max}$ is 100.

Others In §5.3, the words used to construct artificial polysemous words were those with more than 100 occurrences. In the calculation of rank, *word1* and *word2* were excluded from the search.

F.3 Details of §5.4

Word embeddings GloVe pre-trained with Common Crawl (840G tokens)[13]

[10] https://github.com/tmikolov/word2vec, https://github.com/stanfordnlp/GloVe

[11] http://mattmahoney.net/dc/textdata.html

[12] https://dumps.wikimedia.org/

[13] https://nlp.stanford.edu/projects/glove/

	orig	freq	unif
STS12	**0.350**	0.334	0.335
STS13	0.254	**0.280**	0.277
STS14	0.377	**0.402**	0.391
STS15	0.457	**0.493**	0.486
STS16	0.446	**0.472**	0.455

Table 3: The results of semantic textual similarity using the final layer of BERT.

Others 0 is not used because the sign cannot be defined.

G Additional Experimental Results

G.1 §5.2.1

The results of semantic textual similarity using the final layer of BERT are shown in Table 3. It can be seen that freq is almost consistently the best.