Linguistic Regularities in Sparse and Explicit Word Representations

Omer Levy* and **Yoav Goldberg**
Computer Science Department
Bar-Ilan University
Ramat-Gan, Israel
{omerlevy,yoav.goldberg}@gmail.com

Abstract

Recent work has shown that neural-embedded word representations capture many relational similarities, which can be recovered by means of vector arithmetic in the embedded space. We show that Mikolov et al.'s method of first adding and subtracting word vectors, and then searching for a word similar to the result, is equivalent to searching for a word that maximizes a linear combination of three pairwise word similarities. Based on this observation, we suggest an improved method of recovering relational similarities, improving the state-of-the-art results on two recent word-analogy datasets. Moreover, we demonstrate that analogy recovery is not restricted to neural word embeddings, and that a similar amount of relational similarities can be recovered from traditional distributional word representations.

1 Introduction

Deep learning methods for language processing owe much of their success to neural network language models, in which words are represented as dense real-valued vectors in $\mathbb{R}^d$. Such representations are referred to as distributed word representations or *word embeddings*, as they embed an entire vocabulary into a relatively low-dimensional linear space, whose dimensions are latent continuous features. The embedded word vectors are trained over large collections of text using variants of neural networks (Bengio et al., 2003; Collobert and Weston, 2008; Mnih and Hinton, 2008; Mikolov et al., 2011; Mikolov et al., 2013b). The word embeddings are designed to capture what Turney (2006) calls *attributional similarities* between vocabulary items: words that appear in similar contexts will be close to each other in the projected space. The effect is grouping of words that share semantic ("dog cat cow", "eat devour") or syntactic ("cars hats days", "emptied carried danced") properties, and are shown to be effective as features for various NLP tasks (Turian et al., 2010; Collobert et al., 2011; Socher et al., 2011; Al-Rfou et al., 2013). We refer to such word representations as *neural embeddings* or just *embeddings*.

Recently, Mikolov et al. (2013c) demonstrated that the embeddings created by a recursive neural network (RNN) encode not only attributional similarities between words, but also similarities between *pairs of words*. Such similarities are referred to as *linguistic regularities* by Mikolov et al. and as *relational similarities* by Turney (2006). They capture, for example, the *gender* relation exhibited by the pairs "man:woman", "king:queen", the *language-spoken-in* relation in "france:french", "mexico:spanish" and the *past-tense* relation in "capture:captured", "go:went". Remarkably, Mikolov et al. showed that such relations are reflected in vector offsets between word pairs ($apples - apple \approx cars - car$), and that by using simple vector arithmetic one could apply the relation and solve analogy questions of the form "a is to a^* as b is to —" in which the nature of the relation is hidden. Perhaps the most famous example is that the embedded representation of the word *queen* can be roughly recovered from the representations of *king*, *man* and *woman*:

$$queen \approx king - man + woman$$

The recovery of relational similarities using vector arithmetic on RNN-embedded vectors was evaluated on many relations, achieving state-of-the-art results in relational similarity identification tasks

* Supported by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 287923 (EXCITEMENT).

(Mikolov et al., 2013c; Zhila et al., 2013). It was later demonstrated that relational similarities can be recovered in a similar fashion also from embeddings trained with different architectures (Mikolov et al., 2013a; Mikolov et al., 2013b).

This fascinating result raises a question: to what extent are the relational semantic properties a result of the *embedding* process? Experiments in (Mikolov et al., 2013c) show that the RNN-based embeddings are superior to other dense representations, but how crucial is it for a representation to be dense and low-dimensional at all?

An alternative approach to representing words as vectors is the distributional similarity representation, or *bag of contexts*. In this representation, each word is associated with a very high-dimensional but sparse vector capturing the contexts in which the word occurs. We call such vector representations *explicit*, as each dimension directly corresponds to a particular context. These explicit vector-space representations have been extensively studied in the NLP literature (see (Turney and Pantel, 2010; Baroni and Lenci, 2010) and the references therein), and are known to exhibit a large extent of attributional similarity (Pereira et al., 1993; Lin, 1998; Lin and Pantel, 2001; Sahlgren, 2006; Kotlerman et al., 2010).

In this study, we show that similarly to the neural embedding space, the explicit vector space also encodes a vast amount of relational similarity which can be recovered in a similar fashion, suggesting the explicit vector space representation as a competitive baseline for further work on neural embeddings. Moreover, this result implies that the neural embedding process is not discovering novel patterns, but rather is doing a remarkable job at preserving the patterns inherent in the word-context co-occurrence matrix.

A key insight of this work is that the vector arithmetic method can be decomposed into a linear combination of three pairwise similarities (Section 3). While mathematically equivalent, we find that thinking about the method in terms of the decomposed formulation is much less puzzling, and provides a better intuition on why we would expect the method to perform well on the analogy recovery task. Furthermore, the decomposed form leads us to suggest a modified optimization objective (Section 6), which outperforms the state-of-the-art at recovering relational similarities under both representations.

2 Explicit Vector Space Representation

We adopt the traditional word representation used in the distributional similarity literature (Turney and Pantel, 2010). Each word is associated with a sparse vector capturing the contexts in which it occurs. We call this representation *explicit*, as each dimension corresponds to a particular context.

For a vocabulary V and a set of contexts C, the result is a $|V|\times|C|$ sparse matrix S in which S_{ij} corresponds to the strength of the association between word i and context j. The association strength between a word $w \in V$ and a context $c \in C$ can take many forms. We chose to use the popular *positive pointwise mutual information* (PPMI) metric:

$$S_{ij} = PPMI(w_i, c_j)$$

$$PPMI(w,c) = \begin{cases} 0 & PMI(w,c) < 0 \\ PMI(w,c) & otherwise \end{cases}$$

$$PMI(w,c) = \log \frac{P(w,c)}{P(w)P(c)} = \log \frac{freq(w,c)|corpus|}{freq(w)freq(c)}$$

where $|corpus|$ is the number of items in the corpus, $freq(w,c)$ is the number of times word w appeared in context c in the corpus, and $freq(w)$, $freq(c)$ are the corpus frequencies of the word and the context respectively.

The use of PMI in distributional similarity models was introduced by Church and Hanks (1990) and widely adopted (Dagan et al., 1994; Turney, 2001). The PPMI variant dates back to at least (Niwa and Nitta, 1994), and was demonstrated to perform very well in Bullinaria and Levy (2007).

In this work, we take the linear contexts in which words appear. We consider each word surrounding the target word w in a window of 2 to each side as a context, distinguishing between different sequential positions. For example, in the sentence `a b c d e` the contexts of the word `c` are $\texttt{a}^{-2}$, $\texttt{b}^{-1}$, $\texttt{d}^{+1}$ and $\texttt{e}^{+2}$. Each vector's dimenstion is thus $|C| \approx 4|V|$. Empirically, the number of non-zero dimensions for vocabulary items in our corpus ranges between 3 (for some rare tokens) and 474,234 (for the word "and"), with a mean of 1595 and a median of 415.

Another popular choice of context is the syntactic relations the word participates in (Lin, 1998; Padó and Lapata, 2007; Levy and Goldberg, 2014). In this paper, we chose the sequential context as it is compatible with the information available to the state-of-the-art neural embedding method we are comparing against.

3 Analogies and Vector Arithmetic

Mikolov et al. demonstrated that vector space representations encode various relational similarities, which can be recovered using vector arithmetic and used to solve word-analogy tasks.

3.1 Analogy Questions

In a word-analogy task we are given two pairs of words that share a relation (e.g. "man:woman", "king:queen"). The identity of the fourth word ("queen") is hidden, and we need to infer it based on the other three (e.g. answering the question: "*man* is to *woman* as *king* is to — ?"). In the rest of this paper, we will refer to the four words as a:a^*, b:b^*. Note that the type of the relation is not explicitly provided in the question, and solving the question correctly (by a human) involves first inferring the relation, and then applying it to the third word (b).

3.2 Vector Arithmetic

Mikolov et al. showed that relations between words are reflected to a large extent in the offsets between their vector embeddings ($queen - king \approx woman - man$), and thus the vector of the hidden word b^* will be similar to the vector $b - a + a^*$, suggesting that the analogy question can be solved by optimizing:

$$\arg\max_{b^* \in V} \left(sim\left(b^*, b - a + a^*\right)\right)$$

where V is the vocabulary excluding the question words b, a and a^*, and sim is a similarity measure. Specifically, they used the cosine similarity measure, defined as:

$$\cos\left(u, v\right) = \frac{u \cdot v}{\|u\|\|v\|}$$

resulting in:

$$\arg\max_{b^* \in V} \left(\cos\left(b^*, b - a + a^*\right)\right) \quad (1)$$

Since cosine is inverse to the angle, high cosine similarity (close to 1) means that the vectors share a very similar direction. Note that this metric normalizes (and thus ignores) the vectors' lengths, unlike the Euclidean distance between them. For reasons that will be clear later, we refer to (1) as the 3COSADD method.

An alternative to 3COSADD is to require that the *direction* of transformation be conserved:

$$\arg\max_{b^* \in V} \left(\cos\left(b^* - b, a^* - a\right)\right) \quad (2)$$

This basically means that $b^* - b$ shares the same direction with $a^* - a$, ignoring the distances. We refer to this method as PAIRDIRECTION. Though it was not mentioned in the paper, Mikolov et al. (2013c) used PAIRDIRECTION for solving the semantic analogies of the SemEval task, and 3COSADD for solving the syntactic analogies.[1]

3.3 Reinterpreting Vector Arithmetic

In Mikolov et al.'s experiments, all word-vectors were normalized to unit length. Under such normalization, the $\arg\max$ in (1) is mathematically equivalent to (derived using basic algebra):

$$\arg\max_{b^* \in V} \left(\cos\left(b^*, b\right) - \cos\left(b^*, a\right) + \cos\left(b^*, a^*\right)\right) \quad (3)$$

This means that solving analogy questions with vector arithmetic is mathematically equivalent to seeking a word (b^*) which is similar to b and a^* but is different from a. Relational similarity is thus expressed as a sum of attributional similarities. While (1) and (3) are equal, we find the intuition as to why (3) ought to find analogies clearer.

4 Empirical Setup

We derive explicit and neural-embedded vector representations, and compare their capacities to recover relational similarities using objectives 3COSADD (eq. 3) and PAIRDIRECTION (eq. 2).

Underlying Corpus and Preprocessing Previous reported results on the word analogy tasks using vector arithmetics were obtained using proprietary corpora. To make our experiments reproducible, we selected an open and widely accessible corpus – the English Wikipedia. We extracted all sentences from article bodies (excluding titles, infoboxes, captions, etc) and filtered non-alphanumeric tokens, allowing mid-token symbols as apostrophes, hyphens, commas, and periods. All the text was lowercased. Duplicates and sentences with less than 5 tokens were then removed. Overall, we retained a corpus of about 1.5 billion tokens, in 77.5 million sentences.

Word Representations To create contexts for both embedding and sparse representation, we used a window of two tokens to each side (5-grams, in total), ignoring words that appeared less

[1] This was confirmed both by our independent trials and by corresponding with the authors.

than 100 times in the corpus. The filtered vocabulary contained 189,533 terms.[2]

The explicit vector representations were created as described in Section 2. The neural embeddings were created using the `word2vec` software[3] accompanying (Mikolov et al., 2013b). We embedded the vocabulary into a 600 dimensional space, using the state-of-the-art skip-gram architecture, the negative-training approach with 15 negative samples (NEG-15), and sub-sampling of frequent words with a parameter of 10^{-5}. The parameter settings follow (Mikolov et al., 2013b).

4.1 Evaluation Conditions

We evaluate the different word representations using the three datasets used in previous work. Two of them (MSR and GOOGLE) contain analogy questions, while the third (SEMEVAL) requires ranking of candidate word pairs according to their relational similarity to a set of supplied word pairs.

Open Vocabulary The open vocabulary datasets (MSR and GOOGLE) present questions of the form "a is to a^* as b is to b^*", where b^* is hidden, and must be guessed from the entire vocabulary. Performance on these datasets is measured by micro-averaged accuracy.

The **MSR** dataset[4] (Mikolov et al., 2013c) contains 8000 analogy questions. The relations portrayed by these questions are morpho-syntactic, and can be categorized according to parts of speech – adjectives, nouns and verbs. Adjective relations include comparative and superlative (*good* is to *best* as *smart* is to *smartest*). Noun relations include single and plural, possessive and non-possessive (*dog* is to *dog's* as *cat* is to *cat's*). Verb relations are tense modifications (*work* is to *worked* as *accept* is to *accepted*).

The **GOOGLE** dataset[5] (Mikolov et al., 2013a) contains 19544 questions. It covers 14 relation types, 7 of which are semantic in nature and 7 are morpho-syntactic (enumerated in Section 8). The dataset was created by manually constructing example word-pairs of each relation, and providing all the pairs of word-pairs (within each relation type) as analogy questions.

[2]Initial experiments with different window-sizes and cutoffs showed similar trends.

[3]`http://code.google.com/p/word2vec`

[4]`research.microsoft.com/en-us/projects/rnn/`

[5]`code.google.com/p/word2vec/source/browse/trunk/questions-words.txt`

Out-of-vocabulary words[6] were removed from both test sets.

Closed Vocabulary The SEMEVAL dataset contains the collection of 79 semantic relations that appeared in SemEval 2012 Task 2: Measuring Relation Similarity (Jurgens et al., 2012). Each relation is exemplified by a few (usually 3) characteristic word-pairs. Given a set of several dozen target word pairs, which supposedly have the same relation, the task is to rank the target pairs according to the degree in which this relation holds. This can be cast as an analogy question in the following manner: For example, take the *Recipient:Instrument* relation with the prototypical word pairs *king*:*crown* and *police*:*badge*. To measure the degree that a target word pair *wife*:*ring* has the same relation, we form the two analogy questions "*king* is to *crown* as *wife* is to *ring*" and "*police* is to *badge* as *wife* is to *ring*". We calculate the score of each analogy, and average the results. Note that as opposed to the first two test sets, this one does not require searching the entire vocabulary for the most suitable word in the corpus, but rather to rank a list of existing word pairs.

Following previous work, performance on SEMEVAL was measured using accuracy, macro-averaged across all the relations.

5 Preliminary Results

Our first experiment uses 3COSADD (method (3) in Section 3) to measure the prevalence of linguistic regularities within each representation.

Representation	MSR	GOOGLE	SEMEVAL
Embedding	53.98%	62.70%	38.49%
Explicit	29.04%	45.05%	38.54%

Table 1: Performance of **3COSADD** on different tasks with the explicit and neural embedding representations.

The results in Table 1 show that a large amount of relational similarities can be recovered with both representations. In fact, both representations achieve the same accuracy on the SEMEVAL task. However, there is a large performance gap in favor of the neural embedding in the open-vocabulary MSR and GOOGLE tasks.

Next, we run the same experiment with PAIRDIRECTION (method (2) in Section 3).

[6]i.e. words that appeared in English Wikipedia less than 100 times. This removed 882 instances from the MSR dataset and 286 instances from GOOGLE.

Representation	MSR	GOOGLE	SEMEVAL
Embedding	9.26%	14.51%	44.77%
Explicit	0.66%	0.75%	45.19%

Table 2: Performance of **PAIRDIRECTION** on different tasks with the explicit and neural embedding representations.

The results in Table 2 show that the PAIRDIRECTION method is better than 3COSADD on the restricted-vocabulary SEMEVAL task (accuracy jumps from 38% to 45%), but fails at the open-vocabulary questions in GOOGLE and MSR. When the method does work, the numbers for the explicit and embedded representations are again comparable to one another.

Why is PAIRDIRECTION performing so well on the SEMEVAL task, yet so poorly on the others? Recall that the PAIRDIRECTION objective focuses on the similarity of $b^* - b$ and $a^* - a$, but does not take into account the spatial distances between the individual vectors. Relying on direction alone, while ignoring spatial distance, is problematic when considering the entire vocabulary as candidates (as is required in the MSR and GOOGLE tasks). We are likely to find candidates b^* that have the same relation to b as reflected by $a - a^*$ but are not necessarily similar to b. As a concrete example, in *man:woman, king:?*, we are likely to recover feminine entities, but not necessarily royal ones. The SEMEVAL test set, on the other hand, already provides related (and therefore geometrically close) candidates, leaving mainly the direction to reason about.

6 Refining the Objective Function

The 3COSADD objective, as expressed in (3), reveals a "balancing act" between two attractors and one repeller, i.e. two terms that we wish to maximize and one that needs to be minimized:

$$\arg\max_{b^* \in V} \left(\cos\left(b^*, b\right) - \cos\left(b^*, a\right) + \cos\left(b^*, a^*\right)\right)$$

A known property of such linear objectives is that they exhibit a "soft-or" behavior and allow one sufficiently large term to dominate the expression. This behavior is problematic in our setup, because each term reflects a different aspect of similarity, and the different aspects have different scales. For example, *king* is more royal than it is masculine, and will therefore overshadow the gender aspect of the analogy. It is especially true in the case of explicit vector representations, as each aspect of the similarity is manifested by a different set of features with varying sizes and weights.

A case in point is the analogy question "*London* is to *England* as *Baghdad* is to — ?", which we answer using:

$$\arg\max_{x \in V} \left(\cos\left(x, en\right) - \cos\left(x, lo\right) + \cos\left(x, ba\right)\right)$$

We seek a word (*Iraq*) which is similar to *England* (both are countries), is similar to *Baghdad* (similar geography/culture) and is dissimilar to *London* (different geography/culture). Maximizing the sum yields an incorrect answer (under both representations): *Mosul*, a large Iraqi city. Looking at the computed similarities in the explicit vector representation, we see that both *Mosul* and *Iraq* are very close to *Baghdad*, and are quite far from *England* and *London*:

(EXP)	↑ England	↓ London	↑ Baghdad	Sum
Mosul	0.031	0.031	0.244	0.244
Iraq	0.049	0.038	0.206	0.217

The same trends appear in the neural embedding vectors, though with different similarity scores:

(EMB)	↑ England	↓ London	↑ Baghdad	Sum
Mosul	0.130	0.141	0.755	0.748
Iraq	0.153	0.130	0.631	0.655

While *Iraq* is much more similar to *England* than *Mosul* is (both being countries), both similarities (0.049 and 0.031 in explicit, 0.130 and 0.153 in embedded) are small and the sums are dominated by the geographic and cultural aspect of the analogy: *Mosul* and *Iraq*'s similarity to *Baghdad* (0.24 and 0.20 in explicit, 0.75 and 0.63 in embedded).

To achieve better balance among the different aspects of similarity, we propose switching from an additive to a multiplicative combination:

$$\arg\max_{b^* \in V} \frac{\cos\left(b^*, b\right)\cos\left(b^*, a^*\right)}{\cos\left(b^*, a\right) + \varepsilon} \qquad (4)$$

($\varepsilon = 0.001$ is used to prevent division by zero)

This is equivalent to taking the logarithm of each term before summation, thus amplifying the differences between small quantities and reducing the differences between larger ones. Using this objective, *Iraq* is scored higher than *Mosul* (0.259 vs 0.236, 0.736 vs 0.691). We refer to objective (4) as 3COSMUL.[7]

[7] 3COSMUL requires that all similarities be non-negative, which trivially holds for explicit representations. With embeddings, we transform cosine similarities to $[0, 1]$ using $(x + 1)/2$ before calculating (4).

7 Main Results

We repeated the experiments, this time using the 3COSMUL method. Table 3 presents the results, showing that the multiplicative objective recovers more relational similarities in both representations. The improvements achieved in the explicit representation are especially dramatic, with an absolute increase of over 20% correctly identified relations in the MSR and GOOGLE datasets.

Objective	Representation	MSR	GOOGLE
3COSADD	Embedding	53.98%	62.70%
	Explicit	29.04%	45.05%
3COSMUL	Embedding	**59.09%**	66.72%
	Explicit	56.83%	**68.24%**

Table 3: Comparison of **3COSADD** and **3COSMUL**.

3COSMUL outperforms the state-of-the-art (3COSADD) on these two datasets. Moreover, the results illustrate that a comparable amount of relational similarities can be recovered with both representations. This suggests that the linguistic regularities apparent in neural embeddings are not a consequence of the embedding process, but rather are well preserved by it.

On SEMEVAL, 3COSMUL preformed on par with 3COSADD , recovering a similar amount of analogies with both explicit and neural representations (38.37% and 38.67%, respectively).

8 Error Analysis

With 3COSMUL, both the explicit vectors and the neural embeddings recover similar amounts of analogies, but are these the same patterns, or perhaps different types of relational similarities?

8.1 Agreement between Representations

Considering the open-vocabulary tasks (MSR and GOOGLE), we count the number of times both representations guessed correctly, both guessed incorrectly, and when one representations leads to the right answer while the other does not (Table 4). While there is a large amount of agreement between the representations, there is also a non-negligible amount of cases in which they complement each other. If we were to run in an oracle setup, in which an answer is considered correct if it is correct in either representation, we would have achieved an accuracy of 71.9% on the MSR dataset and 77.8% on GOOGLE.

	Both Correct	Both Wrong	Embedding Correct	Explicit Correct
MSR	43.97%	28.06%	15.12%	12.85%
GOOGLE	57.12%	22.17%	9.59%	11.12%
ALL	53.58%	23.76%	11.08%	11.59%

Table 4: Agreement between the representations on open-vocabulary tasks.

	Relation	Embedding	Explicit
GOOGLE	capital-common-countries	90.51%	**99.41%**
	capital-world	77.61%	**92.73%**
	city-in-state	56.95%	**64.69%**
	currency	**14.55%**	10.53%
	family (gender inflections)	**76.48%**	60.08%
	gram1-adjective-to-adverb	**24.29%**	14.01%
	gram2-opposite	**37.07%**	28.94%
	gram3-comparative	**86.11%**	77.85%
	gram4-superlative	56.72%	**63.45%**
	gram5-present-participle	63.35%	**65.06%**
	gram6-nationality-adjective	89.37%	**90.56%**
	gram7-past-tense	**65.83%**	48.85%
	gram8-plural (nouns)	72.15%	**76.05%**
	gram9-plural-verbs	**71.15%**	55.75%
MSR	adjectives	45.88%	**56.46%**
	nouns	56.96%	**63.07%**
	verbs	**69.90%**	52.97%

Table 5: Breakdown of relational similarities in each representation by relation type, using 3COSMUL.

8.2 Breakdown by Relation Type

Table 5 presents the amount of analogies discovered in each representation, broken down by relation type. Some trends emerge: the explicit representation is superior in some of the more semantic tasks, especially geography related ones, as well as the ones superlatives and nouns. The neural embedding, however, has the upper hand on most verb inflections, comparatives, and family (gender) relations. Some relations (currency, adjectives-to-adverbs, opposites) pose a challenge to both representations, though are somewhat better handled by the embedded representations. Finally, the nationality-adjectives and present-participles are equally handled by both representations.

8.3 Default-Behavior Errors

The most common error pattern under both representations is that of a "default behavior", in which one central representative word is provided as an answer to many questions of the same type. For example, the word "Fresno" is returned 82 times as an incorrect answer in the *city-in-state* relation in the embedded representation, and the word "daughter" is returned 47 times as an incorrect answer in the *family* relation in the explicit represen-

RELATION	WORD	EMB	EXP
gram7-past-tense	who	0	138
city-in-state	fresno	82	24
gram6-nationality-adjective	slovak	39	39
gram6-nationality-adjective	argentine	37	39
gram6-nationality-adjective	belarusian	37	39
gram8-plural (nouns)	colour	36	35
gram3-comparative	higher	34	35
city-in-state	smith	1	61
gram7-past-tense	and	0	49
gram1-adjective-to-adverb	be	0	47
family (gender inflections)	daughter	8	47
city-in-state	illinois	3	40
currency	currency	5	40
gram1-adjective-to-adverb	and	0	39
gram7-past-tense	enhance	39	20

Table 6: Common default-behavior errors under both representations. EMB / EXP: the number of time the word was returned as an incorrect answer for the given relation under the embedded or explicit representation.

tation. Loosely, "Fresno" is identified by the embedded representation as a prototypical location, while "daughter" is identified by the explicit representation as a prototypical female. Under a definition in which a default behavior error is one in which the same incorrect answer is returned for a particular relation 10 or more times, such errors account for 49% of the errors in the explicit representation, and for 39% of the errors in the embedded representation.

Table 6 lists the 15 most common default errors under both representations. In most default errors the category of the default word is closely related to the analogy question, sharing the category of either the correct answer, or (as in the case of "Fresno") the question word. Notable exceptions are the words "who", "and", "be" and "smith" that are returned as default answers in the explicit representation, and which are very far from the intended relation. It seems that in the explicit representation, some very frequent function words act as "hubs" and confuse the model. In fact, the performance gap between the representations in the *past-tense* and *plural-verb* relations can be attributed specifically to such function-word errors: 23.4% of the mistakes in past-tense relation are due to the explicit representation's default answer of "who" or "and", while 19% of the mistakes in the plural-verb relations are due to default answers of "is/and/that/who".

8.4 Verb-inflection Errors

A correct solution to the morphological analogy task requires recovering both the correct inflection (requiring syntactic similarity) and the correct base word (requiring semantic similarity). We observe that linguistically, the morphological distinctions and similarities tend to rely on a few common word forms (for example, the "walk:walking" relation is characterized by modals such as "will" appearing before "walk" and never before "walking", and *be* verbs appearing before walking and never before "walk"), while the support for the semantic relations is spread out over many more items. We hypothesize that the morphological distinctions in verbs are much harder to capture than the semantics. Indeed, under both representations, errors in which the selected word has a correct form with an incorrect inflection are over ten times more likely than errors in which the selected word has the correct inflection but an incorrect base form.

9 Interpreting Relational Similarities

The ability to capture relational similarities by performing vector (or similarity) arithmetic is remarkable. In this section, we try and provide intuition as to why it works.

Consider the word "*king*"; it has several aspects, high-level properties that it implies, such as royalty or (male) gender, and its attributional similarity with another word is based on a mixture of those aspects; e.g. *king* is related to *queen* on the royalty and the human axes, and shares the gender and the human aspect with *man*. Relational similarities can be viewed as a composition of attributional similarities, each one reflecting a different aspect. In "*man* is to *woman* as *king* is to *queen*", the two main aspects are gender and royalty. Solving the analogy question involves identifying the relevant aspects, and trying to change one of them while preserving the other.

How are concepts such as gender, royalty, or "cityness" represented in the vector space? While the neural embeddings are mostly opaque, one of the appealing properties of explicit vector representations is our ability to read and understand the vectors' features. For example, *king* is represented in our explicit vector space by 51,409 contexts, of which the top 3 are tut^{+1}, jeongjo^{+1}, adulyadej^{+2} – all names of monarchs. The explicit representation allows us to glimpse at the way different aspects are represented. To do so, we choose a representative pair of words that share an aspect, intersect their vectors, and inspect the highest scoring

Aspect	Examples	Top Features
Female	$woman \odot queen$	estrid^{+1} ketevan^{+1} adeliza^{+1} nzinga^{+1} gunnhild^{+1} impregnate^{-2} hippolyta^{+1}
Royalty	$queen \odot king$	savang^{+1} uncrowned^{-1} pmare^{+1} sisowath^{+1} nzinga^{+1} tupou^{+1} uvea^{+2} majesty^{-1}
Currency	$yen \odot ruble$	devalue^{-2} banknote^{+1} denominated^{+1} billion^{-1} banknotes^{+1} pegged^{+2} coin^{+1}
Country	$germany \odot australia$	emigrates^{-2} 1943-45^{+2} pentathletes^{-2} emigrated^{-2} emigrate^{-2} hong-kong^{-1}
Capital	$berlin \odot canberra$	hotshots^{-1} embassy^{-2} 1925-26^{+2} consulate-general^{+2} meetups^{-2} nunciature^{-2}
Superlative	$sweetest \odot tallest$	freshest^{+2} asia's^{-1} cleveland's^{-2} smartest^{+1} world's^{-1} city's^{-1} america's^{-1}
Height	$taller \odot tallest$	regnans^{-2} skyscraper^{+1} skyscrapers^{+1} 6'4^{+2} windsor's^{-1} smokestacks^{+1} burj^{+2}

Table 7: The top features of each aspect, recovered by pointwise multiplication of words that share that aspect. The result of pointwise multiplication is an "aspect vector" in which the features common to both words, characterizing the relation, receive the highest scores. The feature scores (not shown) correspond to the weight the feature contributes to the cosine similarity between the vectors. The superscript marks the position of the feature relative to the target word.

features in the intersection. Table 7 presents the top (most influential) features of each aspect.

Many of these features are names of people or places, which appear rarely in our corpus (e.g. Adeliza, a historical queen, and Nzinga, a royal family) but are nonetheless highly indicative of the shared concept. The prevalence of rare words stems from PMI, which gives them more weight, and from the fact that words like *woman* and *queen* are closely related (a queen is a woman), and thus have many features in common. Ordering the features of $woman \odot queen$ by prevalence reveals female pronouns ("she", "her") and a long list of common feminine names, reflecting the expected aspect shared by *woman* and *queen*. Word pairs that share more specific aspects, such as capital cities or countries, show features that are characteristic of their shared aspect (e.g. capital cities have *embassies* and *meetups*, while immigration is associated with countries). It is also interesting to observe how the relatively syntactic "superlativity" aspect is captured with many regional possessives ("america's", "asia's", "world's").

10 Related Work

Relational similarity (and answering analogy questions) was previously tackled using explicit representations. Previous approaches use task-specific information, by either relying on a $(word\text{-}pair, connectives)$ matrix rather than the standard $(word, context)$ matrix (Turney and Littman, 2005; Turney, 2006), or by treating analogy detection as a supervised learning task (Baroni and Lenci, 2009; Jurgens et al., 2012; Turney, 2013). In contrast, the vector arithmetic approach followed here is unsupervised, and works on a generic single-word representation. Even though the training process is oblivious to the task of analogy detection, the resulting representation is able to detect them quite accurately. Turney (2012) assumes a similar setting but with two types of word similarities, and combines them with products and ratios (similar to 3COSMUL) to recover a variety of semantic relations, including analogies.

Arithmetic combination of explicit word vectors is extensively studied in the context of compositional semantics (Mitchell and Lapata, 2010), where a phrase composed of two or more words is represented by a single vector, computed by a function of its component word vectors. Blacoe and Lapata (2012) compare different arithmetic functions across multiple representations (including embeddings) on a range of compositionality benchmarks. To the best of our knowledge such methods of word vector arithmetic have not been explored for recovering relational similarities in explicit representations.

11 Discussion

Mikolov et al. showed how an unsupervised neural network can represent words in a space that "naturally" encodes relational similarities in the form of vector offsets. This study shows that finding analogies through vector arithmetic is actually a form of balancing word similarities, and that, contrary to the recent findings of Baroni et al. (2014), under certain conditions traditional word similarities induced by explicit representations can perform just as well as neural embeddings on this task.

Learning to represent words is a fascinating and important challenge with implications to most current NLP efforts, and neural embeddings in particular are a promising research direction. We believe that to improve these representations we should understand how they work, and hope that the methods and insights provided in this work will help to deepen our grasp of current and future investigations of word representations.

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. In *Proc. of CoNLL 2013*.

Marco Baroni and Alessandro Lenci. 2009. One distributional memory, many semantic spaces. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 1–8, Athens, Greece, March. Association for Computational Linguistics.

Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Dont count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 546–556, Jeju Island, Korea, July. Association for Computational Linguistics.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, pages 160–167.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Ido Dagan, Fernando Pereira, and Lillian Lee. 1994. Similarity-based estimation of word cooccurrence probabilities. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 272–278. Association for Computational Linguistics.

David A Jurgens, Peter D Turney, Saif M Mohammad, and Keith J Holyoak. 2012. Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, pages 356–364. Association for Computational Linguistics.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Dekang Lin and Patrick Pantel. 2001. Dirt: discovery of inference rules from text. In *KDD*, pages 323–328.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98, pages 768–774, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomas Mikolov, Stefan Kombrink, Lukas Burget, JH Cernocky, and Sanjeev Khudanpur. 2011. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5528–5531. IEEE.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, June. Association for Computational Linguistics.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1439.

Andriy Mnih and Geoffrey E Hinton. 2008. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems*, pages 1081–1088.

Yoshiki Niwa and Yoshihiko Nitta. 1994. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *Proceedings of the 15th conference on Computational linguistics-Volume 1*, pages 304–309. Association for Computational Linguistics.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Fernando Pereira, Naftali Tishby, and Lillian Lee. 1993. Distributional clustering of english words. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 183–190. Association for Computational Linguistics.

Magnus Sahlgren. 2006. *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. Ph.D. thesis, Stockholm.

Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161. Association for Computational Linguistics.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394. Association for Computational Linguistics.

Peter D. Turney and Michael L. Littman. 2005. Corpus-based learning of analogies and semantic relations. *Machine Learning*, 60(1-3):251–278.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Peter D. Turney. 2001. Mining the web for synonyms: Pmi-ir versus lsa on toefl. In *Proceedings of the 12th European Conference on Machine Learning*, pages 491–502. Springer-Verlag.

Peter D. Turney. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416.

Peter D. Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, 44:533–585.

Peter D. Turney. 2013. Distributional semantics beyond words: Supervised learning of analogy and paraphrase. *CoRR*, abs/1310.5042.

Alisa Zhila, Wen-tau Yih, Christopher Meek, Geoffrey Zweig, and Tomas Mikolov. 2013. Combining heterogeneous models for measuring relational similarity. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1000–1009, Atlanta, Georgia, June. Association for Computational Linguistics.