Scalable Learning and Scoring of Compositional Phenomena

Abstract

Effective compositional vector space models of meaning promise new solutions to numerous stubborn language understanding tasks. However, a challenge to learning such models is the lack of large amounts of direct supervision for semantic relatedness. This paper proposes using automatically-extracted paraphrase information as a novel source of supervision when training these compositional models, along with a mechanism that can score phrase compositionality using contextual information. Experimental results indicate that these multiple sources of information can be used to learn partial semantic supervision that matches previous techniques in intrinsic evaluation tasks. Our approaches are also evaluated for their impact on a machine translation system where we show improvements in translation quality, demonstrating that compositionality in interpretation correlates with compositionality in translation.

1 Introduction

Using contextual information to encode lexical semantic information through vector space models of meaning has been shown to be empirically effective across a wide variety of tasks (Turian et al., 2010; Turney and Pantel, 2010; Mikolov et al., 2013). However, while lexical semantic properties may be well captured with contextual information, learning representations of larger objects is more difficult (Sahlgren, 2006; Collobert et al., 2011, *inter alia*), motivating research in models that *combine* word representations to infer the semantics of longer units like phrases or sentences (Mitchell and Lapata, 2010; Baroni and Zamparelli, 2010; Socher et al., 2013). These models attempt to circumvent the obvious computational and statistical issues that arise from estimating multiword representations directly: there are far more units that need to be handled and the occurrences of such units in training corpora diminish rapidly. They are learned in a data-driven manner, but most rely on small, human-annotated datasets for supervision, which can limit their broader applicability. We hypothesize that the equivalent knowledge can be obtained through monolingual paraphrases that have been extracted using word alignments and an intermediate language. Confirming this hypothesis would allow the rapid development of compositional models in a large number of languages.

These models also impose the assumption that longer units like phrases are **compositional**, i.e., a phrase's meaning can be understood from the literal meaning of its parts. However, countless examples that run contrary to the assumption exist, and handling these **non-compositional** phrases has been problematic and of long-standing interest in the community (Lin, 1999; Sag et al., 2002). (Non-) Compositionality detection as a problem is particularly relevant for downstream tasks like machine translation (MT) or information retrieval, where a literal interpretation of a non-compositional phrase could have comical (or disastrous) consequences. Scoring this phenomenon can provide vital information to other language processing systems on whether a multiword unit should be treated semantically as a single entity or not, and we posit that

this behavior is strongly dependent on the context in which the expression occurs.

In this work we propose a robust, scalable framework that learns compositional functions and scores relative phrase (non-)compositionality. We make three contributions: first, a novel way to learn compositional functions for part-of-speech pairs that uses supervision from an automatically-extracted list of paraphrases (§2.1). Second, a context-dependent scoring model that scores the relative compositionality of a phrase (McCarthy et al., 2003) by computing the likelihood of its context given its paraphrase-learned representation (§3.2). And third, an evaluation of the impact of compositionality knowledge in an end-to-end MT setup. Our experiments (§4) reveal that paraphrase information can learn compositional functions and phrase vectors that yield roughly equivalent correlations to human judgments compared to previous approaches, and can consistently improve translations produced by an English–Spanish translation system.

2   Learning Compositional Functions

Our goal is to learn a function $\mathbf{f}(\mathbf{x}, \mathbf{y})$ that maps N-dimensional vector representations of phrase constituents $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{N \times 1}$ to an N-dimensional vector representation of the phrase[1], i.e., the *composed* representation. We assume the existence of word-level vector representations for every word in our vocabulary of size V. Compositionality is modeled as a bilinear map, and two classes of linear models with different levels of parametrization are proposed. Unlike previous work (Baroni and Zamparelli, 2010; Socher et al., 2012; Grefenstette et al., 2013, *inter alia*) where the functions are word-specific, our compositional functions operate on part-of-speech (POS) tag pairs, which facilitates learning by drastically reducing the number of parameters, and only requires a shallow syntactic parse of the input.

2.1   Supervision through Paraphrases

The Paraphrase Database (Ganitkevitch et al., 2013, PPDB) is a collection of ranked monolingual paraphrases that have been extracted from word-aligned

---

[1]A phrase is defined as any contiguous sequence of words of length 2 or greater, and does not have to adhere to constituents in a phrase structure grammar. We discuss handling phrases longer than 2 words in §2.5.

parallel corpora using the bilingual pivot method (Bannard and Callison-Burch, 2005). The underlying assumption is that if two strings in the same language align to the same string in another language, then the strings in the original language share the same meaning. Paraphrases are ranked by their word alignment scores, and in this work we use the preselected SMALL portion of PPDB as our training data.

The training data is filtered to provide only two-to-one word paraphrase mappings, and the multi-word portion of the paraphrase is subsequently POS-tagged. Table 1 provides a breakdown of such paraphrases by their POS pair type. Given the lack of context when tagging, it is likely that the POS tagger yields the most probable tag for words and not the most probable tag given the context. Furthermore, even the higher quality portions of PPDB yield paraphrases of ranging quality, ranging from non-trivial mappings such as *young people → youth*, to redundant ones like *the ceasefire → ceasefire*. However, PPDB-like resources are more easily available than human-annotated resources (in multiple languages too, Ganitkevitch and Callison-Burch (2014)), so it is imperative that methods which learn compositional functions from such sources handle noisy supervision adequately.

POS Pair	Size
DT-NN	10,982
NN-NN	4781
JJ-NN	3924
VB-VB	2021
RB-JJ	1640
X-X	8548

Table 1: Number of paraphrase examples per POS pair type out of the two-to-one word paraphrases in the SMALL version of PPDB, where POS pair types are marked with tags from the Penn Treebank tagset. We distinguish between the five most common POS pair types, and group the remaining pairs into the generic 'X-X' category.

2.2   Concatenation Models

Given paraphrase training examples that provide semantic mappings between two-word and one-word phrases, our first class of models is a generalization

of the additive models introduced in Mitchell and Lapata (2008):

$$\mathbf{f}(\mathbf{x}, \mathbf{y}) = \mathbf{W}[\mathbf{x}; \mathbf{y}] \qquad (1)$$

where the notation $[\mathbf{x}; \mathbf{y}]$ represents a vertical (row-wise) concatenation of two vectors; namely, the concatenation that results in a $2N \times 1$-sized vector. In addition to the $N \times V$ parameters for the word vector representations that are provided *a priori*, this model introduces $N \times 2N \times T$ parameters, where T is the number of POS-tag pairs we consider.

Mitchell and Lapata (2008) significantly simplify parameter estimation by assuming a certain structure for the parameter matrix $\mathbf{W}$, which is necessary given the limited human-annotated data they use. For example, by assuming a block-diagonal structure, we get a scaled element-wise addition model $f_i(x_i, y_i) = \alpha_i x_i + \beta_i y_i$. Although not strictly in this category due to the non-linearities involved, neural network-based compositional models (Socher et al., 2012; Hermann and Blunsom, 2013) can be viewed as concatenation models, although the order of concatenation and matrix multiplication is switched. However, these models introduce more than $V \times N^2$ parameters.

2.3 Tensor Models

The second class of models leverages pairwise multiplicative interactions between the components of the two word vectors:

$$\mathbf{f}(\mathbf{x}, \mathbf{y}) = (\mathbf{W} \times_3 \mathbf{y}) \times_2 \mathbf{x} \qquad (2)$$

where $\times_n$ corresponds to a tensor contraction along the nth mode of the tensor $\mathbf{W}$. In this case, we first compute a contraction (tensor-vector product) along the third mode between $\mathbf{W}$ and $\mathbf{y}$ corresponding to interactions with the second word vector of a two-word phrase and results in a matrix, which is then multiplied along its second mode (corresponding to traditional matrix multiplication on the right) by $\mathbf{x}$. The final result is an $N \times 1$ vector. This model introduces $N \times N \times N \times T$ parameters.

A special case is the element-wise multiplicative model (Mitchell and Lapata, 2008), where the only non-zero values in $\mathbf{W}$ are along the tensor diagonal. Operating at the vocabulary level, the model of Baroni and Zamparelli (2010) has interesting parallels to our tensor model. They focus on adjective-noun relationships and learn a specific matrix for every adjective in their dataset; in our case, the specific matrix for each adjective has a particular form, namely that it can be factorized into the product of a tensor and a vector; the tensor corresponds to the actual adjective-noun combiner function, and the vector corresponds to specific lexical information that the adjective carries. Our model generalizes to other POS pairs: for example, multiplying the tensor that represents determiner-noun combinations along the second mode with the vector for "the" results in a matrix that represents the semantic operation of definiteness. Learning these parameters jointly makes more sense than separately learning definiteness for each determiner.

2.4 Parameters

The parameters $\mathbf{W}$ in Eq. 1 and 2 can be estimated through standard linear regression techniques in conjunction with the data presented in §2.1. These methods also provide a natural way regularize $\mathbf{W}$ via ℓ_2 (ridge regression) or ℓ_1 (LASSO) regularization. Parameters for the ℓ_1-regularized concatenation model for select POS pairs are displayed in Fig. 1.[2] The heat-maps display the relative magnitude of parameters (darker: higher), with white cells indicating zero values. It is evident that the parameters learned from PPDB indicate a notion of linguistic headedness, namely that for particular POS pairs, the semantic information is primarily contained in the right word, but for others such as the noun-noun combination, the contribution seems equal.

2.5 Longer Phrases

The proposed models operate on pairs of words at a time. To handle phrases of length greater than two, we greedily construct a left-branching tree that eventually dictates the application of the learned bilinear maps.[3] At each internal tree node, we consider POS tags of its children: if the right child is a noun, and the left child is either a noun, adjective, or determiner, then the internal node is marked as a noun,

---

[2]Parameters learned with ℓ_2 regularization yield too many non-zero values, making visualization less informative.

[3]We also tried constructing right-branching trees, but found that performance was never as good as the left-branching ones.

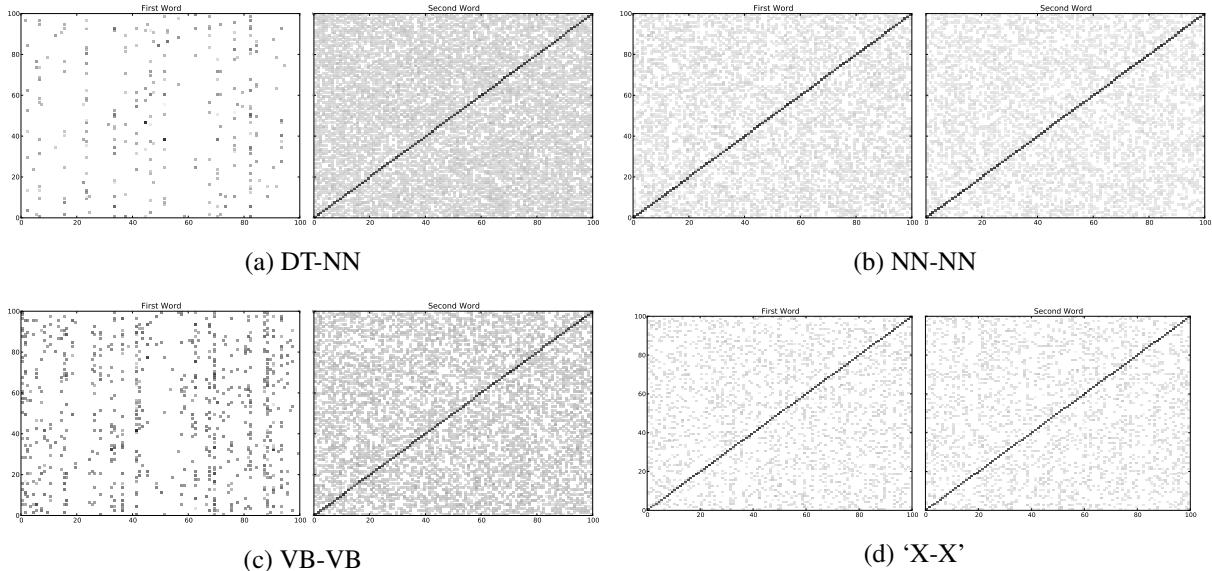|	
(a) DT-NN	(b) NN-NN
(c) VB-VB	(d) 'X-X'

Figure 1: Parameter heat-maps for specific POS pair compositional functions. Higher values are darker, and zero values are white. Certain phrasal relationships exhibit headedness.

otherwise we mark it with a generic 'X' tag. At the end of the procedure, unattached nodes (words) are attached at the highest point in the tree.

After the tree is constructed, we can compute the overall phrasal representation in a bottom-up manner, guided by the labels of leaf and internal nodes. We note that the emphasis of this work is not to compute sentence-level representations. This goal has been explored in recent research (Le and Mikolov, 2014; Kalchbrenner et al., 2014), and combining our models with methods presented therein for sentence-level representations is relatively straightforward and we leave it for future work.

3 Scoring for Compositionality

The concatenation and tensor models compute an N-dimensional vector representation for a multi-word phrase by assuming the meaning of the phrase can be expressed in terms of the meaning of its constituents. This assumption holds true to varying degrees; while it clearly holds for "large amount" and breaks down for "cloud nine", it is partially valid for phrases such as "zebra crossing" or "crash course". In line with previous work, we assume a **compositionality continuum** (McCarthy et al., 2003), but further conjecture that a phrase's level of compositionality is dependent on the specific con-

text in which it occurs, motivating a context-based approach (§3.2) which scores compositionality by computing the likelihoods of surrounding context words given a phrase representation. The effect of context is directly measured through a comparison with context-independent methods from prior work (Bannard et al., 2003; Reddy et al., 2011)

It is important to note that most prior work on compositionality scoring assumes access to *both* word and phrase vector representations (for select phrases that will be evaluated) *a priori*. The latter are distinct from representations that are computed from learned compositional functions as they are extracted directly from the corpus. Our aim is to develop compositional models that are applicable in downstream tasks, and thus assuming pre-existing phrase vectors is unreasonable.[4] For phrases, we only rely on representations computed from our learned compositional functions.

3.1 A Type-based Model

Given vector representations for the constituent words in a phrase and the phrase itself, the idea behind the type-based model is to compute similarities

---

[4] If these phrase representations were easy to extract from corpora, that would obviate the need to learn compositional functions.

between the constituent word representations and the phrasal representation and average the similarities across the constituents. If the contexts in which a constituent word occurs, as dictated by its vector representation, are very different from the contexts of the composed phrase, as indicated by cosine similarity between the representations, then the phrase is likely to be non-compositional. Assuming unit-normalized word vectors $\mathbf{x}, \mathbf{y}$ and phrase vector $\mathbf{z} = \mathbf{f}(\mathbf{x}, \mathbf{y})$ computed from one of the learned models in §2:

$$g(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \alpha(\mathbf{x} \cdot \mathbf{z}) + (1 - \alpha)(\mathbf{y} \cdot \mathbf{z}) \quad (3)$$

where α is a hyperparameter that controls the contribution of individual constituents. This model leverages the average statistics computed over the training corpora, as encapsulated in the word and phrase vectors, to detect compositionality, and is the primary way compositionality has been evaluated previously (Reddy et al., 2011; Kiela and Clark, 2013).

3.2 Using Context

Eq. 3 scores phrases for compositionality regardless of the context that these phrases occur in. However, phrases such as "big fish" or "heavy metal" may occur in both compositional and non-compositional situations, depending on the nature and topic of the texts they occur in.[5] Here, we propose a context-driven model for compositionality detection, inspired by the skip-gram model for learning word representations (Mikolov et al., 2013). The intuition is simple: if a phrase is compositional, it should be sufficiently predictive of the context words around it; otherwise, it is acting in a non-compositional manner. Thus, we would like to compute the likelihood of the context given a phrasal representation:

$$h(\mathbf{z}, \mathbf{C}) = \prod_{i=1}^{l} P(\mathbf{C}^i | \mathbf{z})$$
$$= \prod_{i=1}^{l} \frac{e^{\mathbf{z} \cdot \mathbf{C}^i}}{\sum_{j=1}^{V} e^{\mathbf{z} \cdot \mathbf{C}^j}} \quad (4)$$

where l is the number of context words considered (window size), $\mathbf{C}$ is an $N \times l$ matrix containing the

l N-dimensional context representations[6] concatenated column-wise, and $\mathbf{C}^i$ indexes the ith column of $\mathbf{C}$. In practice, we compute the log-likelihood averaged over the context words or the perplexity instead of the actual likelihood.

3.3 Machine Translation

Modern phrase-based translation systems are faced with a large number of possible segmentations of a source-language sentence during decoding, with all segmentations considered equally likely (Koehn et al., 2003). Thus, it would be helpful to provide guidance on more likely segmentations, as dictated by the compositionality scores of the phrases extracted from a sentence. A low compositionality score would ideally force the decoder to consider the entire phrase as a translation unit, due to its unique semantic characteristics. Correspondingly, a high score informs the decoder that it is safe to rely on word-level translations of the phrasal constituents.

We integrate our compositionality-scoring models into a translation pipeline by first extracting permissible phrases from a source-language sentence and scoring each phrase for compositionality. This score is then added as an additional feature in the system. Empirically, we can also determine if correlating highly with human judgments on compositionality can guide an MT system to provide better translations.

4 Evaluation

Our experiments had three aims: first, demonstrate that the compositional functions learned using paraphrase supervision compute semantically meaningful results for compositional phrases by evaluating on a phrase similarity task (§4.1); second, verify the hypothesis that compositionality is context-dependent by comparing a type-based and token-based approach on a compound noun evaluation task (§4.2); and third, determine if the compositionality-scoring models based on learned representations improve the translations produced by a state-of-the-art phrase-based MT system (§4.3).

The word vectors used in all of our experiments were produced by word2vec[7] using the skip-gram
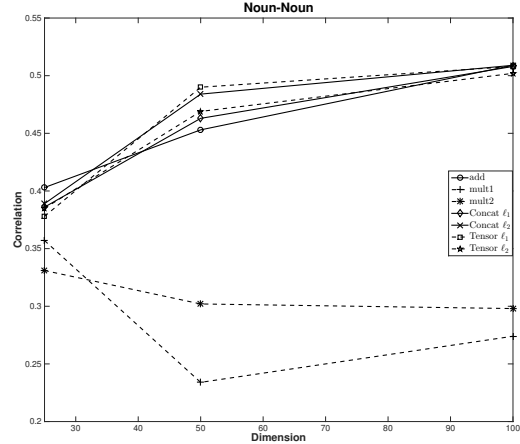
---

[5] In fact, human annotators have access to such context when making compositionality judgments.

[6] As explained in Goldberg and Levy (2014), the context representations are distinct from the word representations.

[7] http://code.google.com/p/word2vec
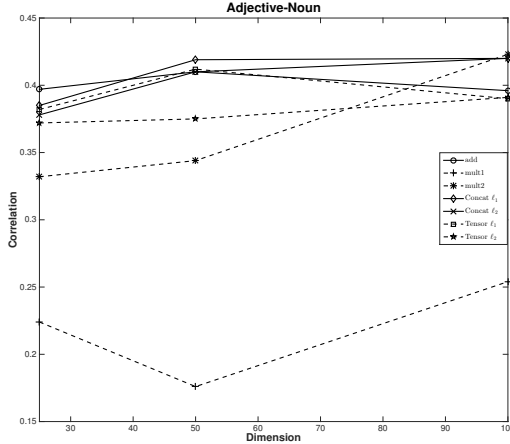
(a) JJ-NN

(b) NN-NN

Figure 2: Spearman's ρ correlation with respect to human judgments for the adjective-noun and noun-noun phrase similarity tasks.

model with 20 negative samples, a context window size of 10, a minimum token count of 3, and sub-sampling of frequent words with a parameter of 10^{-5}. We extracted corpus statistics for `word2vec` using the AFP portion of the English Gigaword[8], which consists of 887.5 million tokens.

4.1 Phrasal Similarity

For the phrase similarity task we first compare our concatenation and tensor models learned using ℓ_1 and ℓ_2 regularization to three baselines:

- ADD: $\mathbf{f}(\mathbf{x}, \mathbf{y}) = \mathbf{x} + \mathbf{y}$
- MULT1: $f_i(x_i, y_i) = x_i y_i$
- MULT2: $f_i(x_i, y_i) = \alpha_i x_i y_i$

Other additive models from previous work (Mitchell and Lapata, 2010; Zanzotto et al., 2010; Blacoe and Lapata, 2012) that impose varying amounts of structural assumptions on the semantic interactions between word representations e.g., $f_i(x_i, y_i) = \alpha_i x_i + \beta_i y_i$ or $\mathbf{f}(\mathbf{x}, \mathbf{y}) = \alpha \mathbf{x} + \beta \mathbf{y}$ are subsumed by our concatenation model. The regularization strength hyperparameter for ℓ_1 and ℓ_2 regularization was selected using 5-fold cross-validation on the PPDB training data.

We evaluated the phrase compositionality models on the adjective-noun and noun-noun phrase simi-

larity tasks compiled by Mitchell and Lapata (2010), using the same evaluation scheme as in the original work.[9] Spearman's ρ between phrasal similarities derived from our compositional functions and the human annotators (computed individually per annotator and then averaged across all annotators) was the evaluation measure.

Figure 2 presents the correlation results for the two POS pair types as a function of the dimensionality N of the representations for the concatenation models (and additive baselines) and tensor models (and multiplicative baselines). The concatenation models seem more effective than the tensor models in the adjective-noun case and give roughly the same performance on the noun-noun dataset, which is consistent with previous work (Guevara, 2011). Since the concatenation model involve fewer parameters, we use it as the compositional model for subsequent experiments. The absolute results are also consistent with state-of-the-art results on this dataset[10] (Blacoe and Lapata, 2012), indicating that paraphrase information is an excellent source of information for learning compositional functions and

is a reasonable alternative to human-annotated training sets. For reference, the inter-annotator agreements are 0.52 for the adjective-noun evaluation and 0.51 for the noun-noun one. The unweighted additive baseline is surprisingly very strong on the noun-noun set, so we also compare against it on subsequent experiments.

4.2 Compositionality

Setup	ADD	ρ Concat. ℓ_1	Concat. ℓ_2
CosSim $\alpha = 0.25$		0.43	0.46
CosSim $\alpha = 0.5$	0.41	0.42	**0.47**
CosSim $\alpha = 0.75$		0.41	0.43
Context $l = 4$	**0.55**	**0.58**	0.58
Context $l = 6$	0.54	0.57	**0.59**
Context $l = 8$	0.53	0.58	0.59

Table 2: Correlation between model judgments on phrase compositionality and human judgments, measured by Spearman's ρ. Context-dependent and concatenation models do better.

To evaluate the compositionality-scoring models, we used the compound noun compositionality dataset introduced in Reddy et al. (2011). This dataset consists of 2670 annotations of 90 compound-noun phrases of varying compositionality with scores ranging from 0 to 5 provided by 30 annotators. It also contains three to five example sentences of these phrases that were shown to the annotators, which we make use of in our context-dependent model. Consistent with the original work, Spearman's ρ is computed on the averaged compositionality score for a phrase across all the annotators that scored that phrase (which varies per phrase). For computing the compositional functions, we evaluate three of the best performing setups from §4.1: the ℓ_1 and ℓ_2-regularized concatenation models, and the simple additive baseline.

For the context-independent model, we select the hyperparameter α in Eq. 3 from the values $\{0.25, 0.5, 0.75\}$. For the context-dependent model, we vary the context window size l by selecting from the values $\{4, 6, 8\}$, and evaluate the concatenation model with both regularizers as well as the simple additive model. Table 2 presents Spearman's ρ for these setups. Note that for the simple additive

model with unit-normalized word vectors, the cosine similarity score is independent of the hyperparameter α. In all cases, the context-dependent models outperform the context-independent ones. Furthermore, the concatenation models are consistently better than the additive compositional model.

4.3 Machine Translation

We built an English-Spanish MT system using the CDEC decoder (Dyer et al., 2010) for the entire training pipeline (word alignments, phrase extraction, feature weight tuning, and decoding). Corpora from the WMT 2012 evaluation[11] was used to build the translation and language models, and for tuning (on `news-test2010`) and evaluation (on `news-test2011`), with scoring done using BLEU (Papineni et al., 2002). The baseline is a hierarchical phrase-based system (Chiang, 2007) with a 4-gram language model. For features, each translation rule is decorated with two lexical and phrasal features corresponding to the forward $(e|f)$ and backward $(f|e)$ conditional log frequencies, along with the log joint frequency (e, f), the log frequency of the source phrase (f), and whether the phrase pair or the source phrase is a singleton. Weights for the language model, glue rule, and word penalty are also tuned. This setup (Baseline) achieves scores *en par* with the WMT results.

When adding our compositionality score as an additional feature, we also add two binary-valued features: the first indicates if the given translation rule has not been decorated with a compositionality score (either because it consists of non-terminals only or the lexical items in the translation rule are unigrams), and correspondingly the second feature indicates if the translation rule has been scored. Therefore, an appropriate additional baseline would be to mark translation rules with these indicator functions but without the scores, akin to identifying rules with phrases in them (Baseline + SegOn).

Table 3 presents the results of the MT evaluation, comparing the baselines to the context-independent and dependent scoring models. The scores have been averaged over three runs with standard deviation is in parentheses, and bolded results are statistically significant ($p < 0.05$) with respect to the

baseline. Additional discussion of results here.

Setup	BLEU	
	Dev	Test
Baseline	25.23 (0.05)	26.89 (0.13)
Baseline + SegOn	25.15 (0.21)	26.87 (0.19)
ℓ_2 CosSim $\alpha = 0.5$	25.08 (0.03)	**26.99** (0.04)
ADD $l = 4$	24.85 (0.12)	26.82 (0.05)
ℓ_1 $l = 4$	25.02 (0.02)	26.89 (0.14)
ℓ_2 $l = 6$	25.12 (0.22)	**27.26** (0.21)

Table 3: MT results. Discuss more here.

5 Related Work

There has been a large amount of recent work on compositional models that operate on a variety of vector representations. With some exceptions (Mitchell and Lapata, 2008; Mitchell and Lapata, 2010), all of these approaches are lexicalized i.e., parameters (generally in the form of vectors, matrices, or tensors) for *specific* words are learned, which works well for frequently occurring words but breaks down when dealing with compositions of arbitrary word sequences containing infrequent words. The functions are either learned with a neural network architecture (Socher et al., 2013, *inter alia*), which generally require a supervised task, or using linear regression (Baroni and Zamparelli, 2010; Zanzotto et al., 2010), where the directly-extracted phrase representations are also required for supervision. We obtain this information through many-to-one PPDB mappings. Many of these models also require additional syntactic (Socher et al., 2012) or semantic (Hermann and Blunsom, 2013; Grefenstette et al., 2013) resources; on the other hand, our proposed approach only requires a shallow syntactic parse in the form of POS tags. Furthermore, with the exception of Zanzotto et al. (2010), who propose a way to extract compositional function training examples from a dictionary, all of these models require human-annotated data to work, although recent efforts to make these models more practical (Paperno et al., 2014) attempt to reduce the statistically complex and overly-parametrized nature of the models.

Most models that score the relative (non-) compositionality of phrases do so in a context-independent manner. A central idea is to replace phrase constituents with semantically-related words and compute the similarity of the new phrase to the original (Kiela and Clark, 2013; Salehi et al., 2014) or make use of a variety of lexical association measures (Lin, 1999; Pecina and Schlesinger, 2006). Sporleder and Li (2009) however, do make use of the context in a token-based approach, where the context in which a phrase occurs as well as the phrase itself is modeled as a lexical chain, and the cohesion of the chain is measured as an indicator of a phrase's compositionality. To compute cohesion, the authors use a web search engine-based measure, whereas we use a probabilistic model of context given a phrase representation, where the probability is parametrized in terms of the logistic function.

In the context of MT, Zhang et al. (2008) present a Bayesian model that learns non-compositional phrases from a synchronous parse tree of a sentence pair. However, the primary aim of their work is for phrase extraction in MT, and the non-compositional constraints are only applied to make the space of phrase pairs more tractable when bootstrapping their phrasal parser from their word-based parser. In contrast, we score every phrase that is extracted using the standard phrase extraction heuristics (Chiang, 2007), allowing the decoder to make the final decision on the impact of compositionality scores in translation. Thus, our work is more similar to Xiong et al. (2010), who propose maximum entropy classifiers that mark positions between words in a sentence as being a phrase boundary or not, and integrate these scores as additional features in the translation system.

6 Conclusion

In this work, we presented two new sources of information for compositionality modeling and scoring, paraphrase information and context. For modeling, we showed that the compositional representations learned using paraphrases performs as well on a phrase similarity task as the average human annotator. For scoring, the importance of context was shown through the comparison of context-independent and dependent models. Improvements by the context-dependent model on an extrinsic machine translation task corroborate the importance of

context. The emphasis of this work was on models that are easily learnable and scalable, and we hope that it encourages further research in making compositional semantic approaches applicable in downstream tasks.

References

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 597–604, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Colin Bannard, Timothy Baldwin, and Alex Lascarides. 2003. A statistical approach to the semantics of verb-particles. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 65–72, Sapporo, Japan, July. 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*, EMNLP '10, pages 1183–1193, Stroudsburg, PA, USA. Association for Computational Linguistics.

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.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228, June.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Chris Dyer, Adam Lopez, Juri Ganitkevitch, Johnathan Weese, Ferhan Ture, Phil Blunsom, Hendra Setiawan, Vladimir Eidelman, and Philip Resnik. 2010. cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models. In *Proceedings of ACL*.

Juri Ganitkevitch and Chris Callison-Burch. 2014. The multilingual paraphrase database. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764, Atlanta, Georgia, June. Association for Computational Linguistics.

Yoav Goldberg and Omer Levy. 2014. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722.

Edward Grefenstette, Georgiana Dinu, Yao-Zhong Zhang, Mehrnoosh Sadrzadeh, and Marco Baroni. 2013. Multi-step regression learning for compositional distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)*.

Emiliano Guevara. 2011. Computing semantic compositionality in distributional semantics. In *Proceedings of the Ninth International Conference on Computational Semantics*, IWCS '11, pages 135–144, Stroudsburg, PA, USA. Association for Computational Linguistics.

Karl Moritz Hermann and Phil Blunsom. 2013. The Role of Syntax in Vector Space Models of Compositional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 894–904, Sofia, Bulgaria, August. Association for Computational Linguistics.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 655–665, Baltimore, Maryland, June. Association for Computational Linguistics.

Douwe Kiela and Stephen Clark. 2013. Detecting Compositionality of Multi-Word Expressions using Nearest Neighbours in Vector Space Models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-13)*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL*.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196. JMLR Workshop and Conference Proceedings.

Dekang Lin. 1999. Automatic identification of non-compositional phrases. In *Proceedings of the 37th Annual Meeting of the Association for Computational*

Linguistics on Computational Linguistics, ACL '99, pages 317–324, Stroudsburg, PA, USA. Association for Computational Linguistics.

Diana McCarthy, Bill Keller, and John Carroll. 2003. Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 73–80, Sapporo, Japan, July. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL-08: HLT*, pages 236–244, Columbus, Ohio.

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

Denis Paperno, Nghia The Pham, and Marco Baroni. 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)*, pages 90–99, Baltimore, Maryland, June. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Pavel Pecina and Pavel Schlesinger. 2006. Combining association measures for collocation extraction. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, COLING-ACL '06, pages 651–658, Stroudsburg, PA, USA. Association for Computational Linguistics.

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, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann A. Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for nlp. In *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '02, pages 1–15, London, UK, UK. Springer-Verlag.

M. 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, Department of Linguistics, Stockholm University.

Bahar Salehi, Paul Cook, and Timothy Baldwin. 2014. Using distributional similarity of multi-way translations to predict multiword expression compositionality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 472–481, Gothenburg, Sweden, April. Association for Computational Linguistics.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic Compositionality Through Recursive Matrix-Vector Spaces. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

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, Seattle, Washington, USA, October. Association for Computational Linguistics.

Caroline Sporleder and Linlin Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 754–762, Athens, Greece, March. 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*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January.

Deyi Xiong, Min Zhang, and Haizhou Li. 2010. Learning translation boundaries for phrase-based decoding. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 136–144, Los Angeles, California, June. Association for Computational Linguistics.

Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar. 2010. Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1263–1271, Beijing, China, August. Coling 2010 Organizing Committee.

Hao Zhang, Chris Quirk, Robert C. Moore, and Daniel Gildea. 2008. Bayesian learning of non-compositional phrases with synchronous parsing. In