Contextualized Translations of Phrasal Verbs with Distributional Compositional Semantics and Monolingual Corpora

Pablo Gamallo
CiTIUS*
University of Santiago de Compostela
pablo.gamallo@usc.es

Susana Sotelo
CiTIUS
University of Santiago de Compostela
susana.sotelo.docio@usc.es

José Ramom Pichel
Imaxin|Software
jramompichel@imaxin.com

Mikel Artetxe
IXA Group Universidad del Pais Vasco
Euskal Herriko Unibersitatea
mikel.artetxe@ehu.eus

This article describes a compositional distributional method to generate contextualized senses of words and identify their appropriate translations in the target language using monolingual corpora. Word translation is modeled in the same way as contextualization of word meaning, but in a bilingual vector space. The contextualization of meaning is carried out by means of distributional composition within a structured vector space with syntactic dependencies, and the bilingual space is created by means of transfer rules and a bilingual dictionary. A phrase in the source language, consisting of a head *and a* dependent*, is translated into the target language by selecting both the nearest neighbor of the head given the dependent, and the nearest neighbor of the dependent given the head. This process is expanded to larger phrases by means of incremental composition. Experiments were performed on English and Spanish monolingual corpora in order to translate phrasal verbs in context. A new bilingual data set to evaluate strategies aimed at translating phrasal verbs in restricted syntactic domains has been created and released.*

* Centro de Investigación en Tecnoloxías Intelixentes.

Submission received: 17 June 2018; revised version received: 22 February 2019; accepted for publication: 15 March 2019.

https://doi.org/10.1162/COLI_a_00353

1. Introduction

Compositional models in distributional semantics combine word vectors to yield new compositional vectors that represent the meaning of composite expressions. Some compositional approaches use syntactically enriched vector models, assuming a structured vector space in which word contexts are defined on the basis of dependency relations (Erk and Padó 2008; Thater, Fürstenau, and Pinkal 2010). In those approaches, the compositional vectors correspond to the meaning of words in context and the syntax-based combination of vectors enables words to be disambiguated as a process of contextualization (Weir et al. 2016). Similarly, the compositional process applied to a bilingual vector space should also enable translating polysemous words in context in an appropriate way.

Table 1 shows an example. Given the expression *catch a ball*, the sense of *catch* combined with *ball* is similar to *grab*, and can be translated into Spanish by *coger*. By contrast, this verb has a similar meaning to *contract* when combined with *disease* in the expression *catch a disease*, and its more appropriate translation into Spanish is now *contraer*. On the other hand, the sense of *ball* when combined with *catch* designates a spherical object and its translation into Spanish is *pelota*. However, the meaning of *ball* refers to a dancing event when it is combined with *attend* in *attend a ball*, being translated now into Spanish by *baile*. Both sense disambiguation and language translation are sensitive to the compositional construction of new meanings (Brown et al. 1991). In a bilingual distributional framework, we call "contextualized translation" the construction of compositional vectors for the expressions in the target language that are similar to the compositional vectors of the expressions in the source language. The target expression with the most similar compositional vector to the vector of the source expression will be considered as its most likely (contextualized) translation. This task was known in machine translation as **target word selection**, namely, the task of deciding which target language word is the most appropriate equivalent of a source language word in context (Dagan 1991).

In a monolingual vector space, we propose a compositional model based on that described in Erk and Padó (2008) and Erk et al. (2010). When two words, *catch* and *ball*, are related by a syntactic dependency, for instance *dobj* (direct object), we actually perform two different combinations: on the one hand, we combine the vector of the head word, noted catch, with the selectional preferences, noted ball$^{d\downarrow}$, imposed by the dependent word *ball* on the head *catch*, in order to obtain a new compositional vector: catch$_{dobj\uparrow}$. This is the contextualized sense of the head *catch* given *ball* in relation *dobj*, which would be close to the meaning of *grab* and not to that of *contract*. On the other hand, the vector of the dependent word, ball, is combined with the selectional preferences, noted catch$^{h\uparrow}$, imposed by the head *catch* on *ball*, so as to obtain a new

Table 1
Similar words (second column) and translations into Spanish (third column) of *catch* and *ball* in context.

catch a ball	*grab*	*coger* (spa)
catch a disease	*contract*	*contraer* (spa)
catch a ball	*spherical object*	*pelota* (spa)
attend a ball	*dancing event*	*baile* (spa)

compositional vector: $\text{ball}_{dobj\downarrow}$. This is the contextualized sense of *ball* given *catch* in the same relation *dobj*, which should denote a spherical object and not a dancing event. So, when two words are syntactically dependent, the compositional process builds two new vectors: one for the head expression and another one for the dependent one. In this approach, the vector space is structured with syntactic dependencies, and word senses are contextualized as words are combined with each other through their dependencies. This compositional strategy is useful to identify paraphrases, that is, similar composite expressions in the same language. The key element of such a syntax-sensitive vector space, which is high dimensional and sparse, is the concept of *selectional preferences* (defined later in Section 3).

Our main contribution is to adapt this syntax-based compositional process to a bilingual model. The contextualized translation of a given composite expression in the source language is performed by searching its nearest-neighbor vectors, among a set of candidates, in the target language, after having been contextualized as described here. This could be seen as a first step toward the definition of a compositional strategy for machine translation. Another important contribution of our work is the creation of an evaluation data set consisting of 1,119 Spanish translations of English sentences containing phrasal verbs.

The objective of the article is to define a bilingual vector space to perform contextualized translations of phrasal verbs, on the basis of a distributional compositional model enriched with syntactic information. What we call **contextualized translation** is actually a sort of unsupervised compositional-based machine translation. However, we prefer keeping the term **contextualization** because the compositional strategy we use is the same as that required for generating contextualized senses in the same language. Preliminary ideas underlying this method have been reported in Gamallo (2017c).

This article is organized as follows. Some related work is addressed in the next section (2). Then, Section 3 describes the compositional distributional model we follow. Next, Section 4 introduces the bilingual word space, and Section 5 defines our contextualized translation strategy. Experiments on translation of phrasal verbs are performed in Section 6 and, finally, relevant conclusions are addressed in Section 7.

2. Related Work

Our approach relies on three strategies: a compositional method to build vectors representing the contextualized sense of composite expressions (Subsection 2.1); a way of building a bilingual vector space using monolingual corpora (Subsection 2.2); and a strategy to propose contextualized translations with bilingual and compositional vectors (Subsection 2.3).

2.1 Compositional Vectors

In the last decade, different compositional models have been proposed and most of them use bag-of-words as basic representations of word contexts in the vector space. The most intuitive approach, reported in Mitchell and Lapata (2008, 2009, 2010), consists of combining vectors of two related words with arithmetic operations: component-wise addition and multiplication. Mitchell and Lapata (2009, 2010) describe weighted additive models giving more weight to some constituents—for instance, to the head word in a verb-noun expression, as the whole construction is closer to the verb than to the noun. Other weighted additive models are described in Guevara (2010) and Zanzotto

et al. (2010). These models only define composition operations for two syntactically related words. Their main drawback is that they do not propose a more systematic model covering all types of semantic composition. More precisely, they do not focus on the function–argument relationship underlying compositionality in categorial grammar (CG) approaches—that is, they do not provide a linguistic combination with the elegant mechanism expressed by the principle of compositionality, where words interact with each other according to their syntactic categories (Montague 1970).

Other approaches develop robust models of meaning composition inspired by CG approaches. They learn the meaning of functional words (e.g., verbs and adjectives) from corpus-based occurrences by making use of regression predictive modeling (Baroni and Zamparelli 2010; Baroni 2013; Krishnamurthy and Mitchell 2013; Baroni, Bernardi, and Zamparelli 2014). In our proposal, by contrast, compositional functions are associated not with functional words, but with syntactic dependencies. Besides, they are not learned using regression techniques, but are just modeled as basic arithmetic operations on vectors as in Mitchell and Lapata (2008) and Weir et al. (2016). Arithmetic operations are easy to implement and produce high-quality compositional vectors, which makes them suited to practical applications (Baroni, Bernardi, and Zamparelli 2014).

There are other compositional methods still relying on CG that use tensor products (Coecke, Sadrzadeh, and Clark 2010; Grefenstette et al. 2011). Two problems can arise with tensor products. First, they lead to a problem of information scalability, because tensor representations grow exponentially as the phrases lengthen (Turney 2013). Second, tensor products do not seem to perform as well as basic arithmetic operations (e.g., multiplication) as was reported in Mitchell and Lapata (2010).

There are also studies making use of neural-based approaches (or deep learning strategies) to deal with word contextualization. In Peters et al. (2018), unlike traditional word type embeddings, each token is assigned a representation that is a function of the entire input sentence. In particular, the authors use vectors derived from a bidirectional long short-term memory network (LSTM) that is trained with a coupled language model in order to build contextualized vectors. Melamud, Goldberger, and Dagan (2016) also make use of bidirectional LSTM for efficiently learning a generic context embedding function. Devlin et al. (2018) make use of masked language models to enable pre-trained deep bidirectional representations. In a similar way, McCann et al. (2017) use a deep LSTM encoder from an attentional sequence-to-sequence model trained for machine translation to contextualize word vectors. However, in these four studies, word contextualization is not defined by means of syntax-based compositional functions, as they do not consider the syntactic functions of the constituent words.

Other pieces of work make use of deep learning strategies to build compositional vectors, such as recursive neural network models (Socher et al. 2012; Hashimoto and Tsuruoka 2015), which share with our model the idea that in the composition of two words both words modify each other's meaning. Similarly, the deep recursive neural network reported in Irsoy and Cardie (2014) considers the structural representation of a phrase (e.g., a parse tree) so as to recursively generate parent representations in a bottom–up fashion, by combining tokens to produce representations for phrases. However, the opaque embeddings built by means of neural-based strategies cannot be easily adapted to our compositional method since it requires a transparent and syntax-sensitive vector space made of lexico-syntactic contexts.

So far, all the cited works represent vector contexts by means of window-based techniques. However, there are a few studies using vector spaces structured with syntactic information as in our approach. Thater, Fürstenau, and Pinkal (2010) distinguish

between *first-order* and *second-order* vectors in order to allow two syntactically incompatible vectors to be combined. This work is inspired by that described in Erk and Padó (2008) and Erk et al. (2010), in which second order (or indirect) vectors represent selectional preferences and each word combination gives rise to two contextualized word senses. More recently, Weir et al. (2016) describe a similar approach where the meaning of a sentence is represented by the contextualized senses of its constituent words. The main difference is the type of context they use to build word vectors. Each word occurrence is modeled by what they call *anchored packed dependency tree*, which is a dependency-based graph that captures the full sentential context of the word. The main drawback of this context approach is its critical tendency to build very sparse word representations. In the deep learning paradigm, special attention should be given to a syntax-sensitive compositional version of CBOW algorithm, which is called C-PHRASE (Pham et al. 2015).

Our proposal is an attempt to merge the main ideas of these syntax-sensitive models (i.e., to consider two word senses per combination and to use the concept of selectional preferences) in order to apply them to contextualized translation.

2.2 Cross-Lingual Word Similarity from Monolingual Corpora

The method proposed in this article also relies on techniques to build bilingual vectors from monolingual corpora. Most approaches to extract translation equivalents from monolingual corpora define the contextual distribution of a word by considering bilingual pairs of **seed words**. In most cases, seed words are provided by external bilingual dictionaries (Fung and McKeown 1997; Fung and Yee 1998; Rapp 1999; Chiao and Zweigenbaum 2002; Shao and Ng 2004; Saralegi, Vicente, and Gurrutxaga 2008; Gamallo 2007; Gamallo and Pichel 2008; Yu and Tsujii 2009a; Ismail and Manandhar 2010; Rubino and Linarés 2011; Tamura, Watanabe, and Sumita 2012; Aker, Paramita, and Gaizauskas 2013; Ansari et al. 2014). So, a word in the target language is a translation candidate of a word in the source language if it tends to co-occur with the pairs of words from the seed words. A slightly different strategy is reported in Wijaya et al. (2017), where the learning task is modeled as a matrix completion problem with source words in the columns and target words in the rows. More precisely, starting from some observed translations (e.g., from existing bilingual dictionaries), the method infers missing translations in the matrix using matrix factorization with a Bayesian Personalized Ranking.

A very similar but different task is cross-lingual hypernymy detection, which determines whether a word in one language (e.g., *vehicle*) is a hypernym of a word in another language (e.g., *coche* [car] in Spanish). Upadhyay et al. (2018) describe an unsupervised approach for cross-lingual hypernymy detection, which learns sparse, bilingual word embeddings based on dependency contexts. Neural-based strategies also have been used to learn translation equivalents from word embeddings (Mikolov, Le, and Sutskever 2013; Artetxe, Labaka, and Agirre 2016, 2018). They learn a linear mapping between embeddings in two languages, which minimizes the distances between equivalences listed in a bilingual dictionary. Artetxe, Labaka, and Agirre (2017) provide very good results using small lists of seed words. Mapped embeddings are used to train unsupervised machine translation systems, which leverage automatic generation of parallel data by back-translating with a backward model operating in the other direction, and the denoising effect of a language model trained on the target side (Artetxe et al. 2017; Lample et al. 2018).

Unlike most approaches to extract word translations from monolingual corpora, which are based on windowing techniques without syntactic information, we will use a method that relies on dependency-based contexts. A significant number of papers report that contexts based on syntactic dependencies outperform window-based strategies in bilingual extraction (Gamallo and Pichel 2008; Yu and Tsujii 2009b; Andrade, Matsuzaki, and Tsujii 2011; Hazem and Morin 2014).

2.3 Compositional Translation of Composite Expressions

Most approaches to unsupervised compositional translation of phrases, multiwords, and composite terms consist of decomposing the source term into atomic components, translating these components into the target language, and recomposing the translated components into target terms (Tanaka and Baldwin 2003; Grefenstette 1999; Delpech et al. 2012; Morin and Daille 2012). The simplest strategy assumes that the translation of a compound may be obtained by translating each component individually thanks to a general dictionary, by generating all the combinations of word positions, and then filtering the translated expressions using either the target corpus or the Web, as in Grefenstette (1999).

This strategy is limited to the subset of compound expressions that share the same compositional property in both the source and target languages, and it is also limited by the coverage of the translation dictionary (Morin and Daille 2012). Several problems can arise, namely, fertile translations in which the target expression has more words than the source term (e.g., the English word *estrogen-sensitive* is translated in Spanish by *sensible a los estrógenos*), and collocations that can be translated by just one word (for instance, the English expression *go for a walk* is translated in Spanish by the word *pasear*). Our translation approach also follows the decomposing strategy but, unlike the works cited earlier, the source expression will be compared against a very large list of candidates, including single words and composite expressions with different morphological and syntactic properties. The use of syntax-based transfer rules helps us enlarge the list of candidates and makes the model fully compositional.

Finally, concerning neural machine translation (NMT), it is worth noting that it does not decompose the source sentence in a compositional way as in our approach. Instead, NMT *encodes* the source sentence using recurrent neural networks (RNN) and then *decodes* it to generate the target sentence word-by-word. At each generation step, the decoder has access not just to a single word from the source sentence, but to the contextual representations of every word in the source sentence (e.g., Cho et al. 2014). RNN encoder-decoder architecture captures both semantic and syntactic structures of phrases and permits a sequence-to-sequence prediction where the length of the input and output sequences may vary. This makes it possible to deal elegantly with cases of fertile and non-compositional translations. However, unlike our unsupervised approach, standard NMT is a supervised strategy, as it relies on parallel corpora.

3. Compositional Distributional Semantics

In this section, we describe first how word senses are contextualized by making use of selectional preferences (Section 3.1). Then, we describe how compositional vectors are created by combining head-dependent words (Section 3.2). Finally, this process is generalized and extended to a dependency tree (Section 3.3).

3.1 Contextualized Senses and Selectional Preferences

Our model uses vector representations for words (or lemmas) based on syntactic contexts. Syntactic contexts are derived from binary dependencies, which can be found in a corpus analyzed with a dependency-based parser. Let's suppose the composite expression *"catch a ball"* was found in a corpus and is analyzed as follows:

$$(dobj,\ catch,\ ball)$$

It states that the noun *ball* (dependent word) is related to the head verb *catch* by means of the relation *dobj* (direct object). A dependency is then a triple consisting of a relation, a head, and a dependent word. From this dependency, two complementary word contexts are extracted:

$$< dobj_{\uparrow}, catch >,\ < dobj_{\downarrow},\ ball >$$

The oriented relation $dobj_{\uparrow}$ means that the head word *catch* is expecting a dependent word in relation *dobj*, and $dobj_{\downarrow}$ means that the dependent noun *ball* is searching for the head verb in the same relation. This representation is similar to that used for distinguishing traditional selectional preferences[1] from *inverse* selectional preferences (Erk and Padó 2008).

Consider now that we want to build the contextualized senses of *catch* and *ball* in the composite expression, *catch a ball*. Let us start with the distributional vectors of the two related words. In a structured space, the vector of a word represents all the (lexico-)syntactic contexts with regard to which the target word is either a head or a dependent. Given that the vector of a noun is defined in a different syntactic space from the vector of a verb, it is not possible to find common contexts shared by the two vectors. In fact, in a structured (or typed) vector space they are incompatible vectors and cannot be combined (Kober et al. 2017). Our structured vector space is a distributional semantic model where words are represented as lemma–part of speech (PoS) pairs and dimensions are lexico-syntactic positions. For instance, the word *ball* (represented in our structured space as the lemma–PoS pair $< ball,\ NOUN >$ is assigned a corpus-based frequency in context $< dobj_{\uparrow},\ catch >$ (direct object of the verb *catch*)).[2] This lexico-syntactic context is only used to define noun vectors as neither verbs nor adjectives can be direct objects of verb *catch*.

To make the composition of two dependent words compatible, we propose to combine word vectors with their selectional preferences as shown in Figure 1. Selectional preferences (or indirect vectors) are formally defined in the following paragraphs.

To compose the sense of *catch* given *ball* in the *dobj* syntactic relation, we build both:

- the selectional preferences imposed by the dependent noun on the head verb in that syntactic position: $\text{ball}^{d\downarrow}$,
- the selectional preferences imposed by the head verb on the dependent noun in the same syntactic position: $\text{catch}^{h\uparrow}$.

1 Selectional preferences or constraints are the tendency for a word to semantically select or constrain which other words may appear in a direct syntactic relation with it (Resnik 1996).

2 For the sake of simplicity, we will continue to represent words not as lemma-PoS pairs, but as simple lemmas.

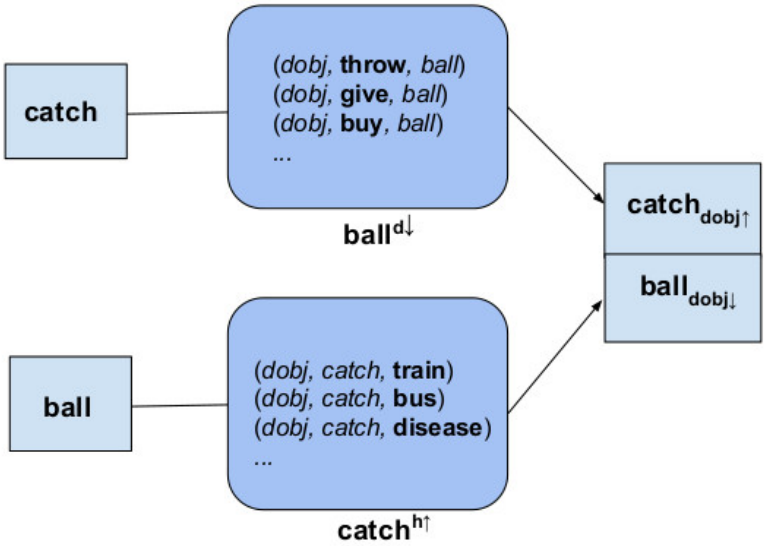

Figure 1
Diagram representation of the selectional preferences, ball$^{d\downarrow}$ and catch$^{h\uparrow}$, which are combined with catch and ball, respectively, to build two contextualized senses in the direct object dependency.

Here, ball$^{d\downarrow}$ and catch$^{h\uparrow}$ are indirect vectors resulting from the following vector component-wise additions:

$$\text{ball}^{d\downarrow} = \sum_{\text{v}\in \boldsymbol{B}} \text{v} \quad (1)$$

$$\text{catch}^{h\uparrow} = \sum_{\text{n}\in \boldsymbol{C}} \text{n} \quad (2)$$

where B is the vector set of those verbs having *ball* as direct object in the corpus, for instance: {**throw**, **give**, **buy**, ...} (see Figure 1). More precisely, given the linguistic context $< dobj_{\downarrow}, ball >$, the indirect vector $\mathbf{ball}^{d\downarrow}$ is obtained by adding the vectors $\{\mathbf{v}|\mathbf{v} \in \boldsymbol{B}\}$ of all verbs co-occurring with the noun *ball* in the *dobj* relation. More intuitively, ball$^{d\downarrow}$ stands for the inverse selectional preferences imposed by *ball* on any verb at the direct object position. Given that this new vector is constituted by verbal contexts, it belongs to the same vector space as verbs, and therefore it can be combined with the word vector of *catch*.

On the other hand, C in Equation (2) represents the vector set of those nouns occurring as direct object of *catch* in the corpus: {**train**, **bus**, **disease**, ...} (see Figure 1). More precisely, given the lexico-syntactic context $< dobj_{\uparrow}, catch >$, the vector $\mathbf{catch}^{h\uparrow}$ is obtained by adding the vectors $\{\mathbf{n}|\mathbf{n} \in \mathbf{C}\}$ of those nouns that occur at the direct object position of the verb *catch*. Indirect vector $\mathbf{catch}^{h\uparrow}$ stands for the selectional preferences imposed by the verb on any noun in the *dobj* relation. Such a new vector is only constituted by nominal contexts, and, therefore, is compatible and might be combined with the word vector of *ball*.

3.2 Dependencies and Composition

Once the selectional preferences have been elaborated, given two words linked by a dependency, two new compositional vectors are created by just multiplying word vectors with their corresponding selectional preferences (indirect vectors). In our approach, composition is driven by binary dependencies. A syntactic dependency consists of two functions. First, head function, $h_{\uparrow}$, combines the vector of the head word, **catch**, with the selectional preferences of the dependent word, $\mathbf{ball}^{d\downarrow}$, by component-wise multiplication. It yields a new vector, $\mathbf{catch}_{dobj\uparrow}$, which represents the contextualized sense of *catch* given *ball* in the *dobj* relation:

$$h_{\uparrow}(dobj, \mathbf{catch}, \mathbf{ball}^{d\downarrow}) \quad = \quad \mathbf{catch} \odot \mathbf{ball}^{d\downarrow} \quad = \quad \mathbf{catch}_{dobj\uparrow} \tag{3}$$

Similarly, dependent function $d_{\downarrow}$ combines the vector of the dependent word, **ball**, with the inverse preferences of the head, $\mathbf{catch}^{h\uparrow}$, by component-wise multiplication, in order to build a new compositional vector, $\mathbf{ball}_{dobj\downarrow}$, which stands for the contextualized sense of *ball* given *catch* in the *dobj* relation:

$$d_{\downarrow}(dobj, \mathbf{catch}^{h\uparrow}, \mathbf{ball}) \quad = \quad \mathbf{ball} \odot \mathbf{catch}^{h\uparrow} \quad = \quad \mathbf{ball}_{dobj\downarrow} \tag{4}$$

Each multiplicative operation results in a compositional vector of a contextualized word. Component-wise multiplication has an intersective effect. The indirect vector restricts the direct vector by assigning frequency 0 to those contextual features that are not shared by both vectors.

3.3 Incremental Composition

Following dependency grammar (Kahane 2003; Hudson 2003), in our approach, semantic composition is driven by syntactic dependencies. They contextualize word senses in an incremental way. The consecutive application of the syntactic dependencies found in an expression is, in fact, the process of building the contextualized sense of all the lexical words constituting the expression. So, the meaning of a complex expression is represented by a contextualized vector for each constituent word rather than by a single vector standing for the entire expression. Figure 2 illustrates the incremental process of building the sense of words by the consecutive application of two dependencies. Given the expression *a girl catches the ball* and its dependency analysis shown on the top of the figure, two compositional processes are carried out by the two dependencies involved in the analysis: *nsubj* and *dobj*. Each dependency is decomposed into two functions: head $h_{\uparrow}$ and dependent $d_{\downarrow}$. As a final result, no single meaning has been constructed for the entire expression *a girl catches the ball*, but we have obtained one contextualized sense per lexical word: $\mathbf{girl}_{nsubj\downarrow}$, $\mathbf{catch}_{nsubj\uparrow+dobj\uparrow}$, and $\mathbf{ball}_{dobj\downarrow}$. This strategy may be considered as an incremental extension of that reported in Erk et al. (2010). The main difference with their approach is that we use *contextualized selectional preferences* at different levels of analysis. By contrast, the work by Erk et al. (2010) is not incremental because selectional preferences are not contextualized. In the *dobj* application of Figure 2, the contextualized selectional preferences imposed by the verb, and noted $\mathbf{catch}^{h\uparrow}_{nsubj\uparrow}$, were created by selecting the contexts of the nouns appearing as direct object of *catch*, which are also part of *girl* after having been contextualized by the verb at the subject position. In

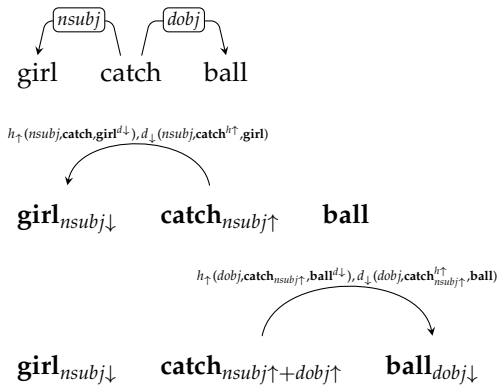

Figure 2
Syntactic analysis of the expression *a girl catches the ball* and left-to-right construction of the word senses.

other terms, the application of the second dependency requires building the selectional preferences imposed by the verbal expression *the girl catches* on the nouns appearing in the direct object position.[3]

It is worth noting that it would also be possible to incrementally apply dependencies following a different order—for example, right-to-left direction. However, in the experiments of Section 6, we will use only the incremental left-to-right order. A more informal and linguistic-based description of the current method is reported in Gamallo (2017c).

4. Bilingual Vectors Extracted from Monolingual Corpora

In a bilingual vector space, vector dimensions correspond to **bilingual contexts**. To build bilingual contexts, we require both a bilingual dictionary and a set of bilingual transfer rules. For example, take first an English–Spanish dictionary providing us with the following lexical correspondences: *station* is translated by *estación* into Spanish, and the English noun *bus* is translated by *autobús*. Second, take the following English–Spanish transfer rules:

$$(nmod, N1, N2) \rightarrow ((nmod, N1, N2), R) \quad (5)$$

$$(nmod, N1, N2) \rightarrow ((nmod/de, N1, N2), R) \quad (6)$$

where the English constructions (source language) are on the left of the transfer rules (5) and (6), while the Spanish constructions (target language) are on the right. Dependency *nmod* stands for the nominal modifier relation and R represents a strong restriction on the dependent word that must be situated to the left of the head one in the target constructions. This restriction will be used in the last translation step to decode the expressions in the target language from the selected candidates. This way, $((nmod, N1, N2), R)$ gives rise to "N1 N2" Spanish expressions, for example, *estado*

3 We do not consider the meaning of determiners, auxiliary verbs, or tense affixes. Quantificational issues associated with them are beyond the scope of this work.

miembro (*member state*), while $((nmod/de, N1, N2), R)$ gives rise to "N1 de N2" expressions, for example *estación de autobuses* (*bus station* or literally *station of buses*). It means that English constructions of type "N2 N1" can be translated by "N1 N2" and "N1 de N2" constructions into Spanish. A transfer rule is thus a bilingual pair of constructions provided with word order restrictions in the target language, and a construction is a dependency without lexical information.

On the basis of the bilingual dictionary and the transfer rules (5) and (6), we generate four bilingual contexts that would be integrated into the syntax-based dimensions of our English–Spanish vector space:

$$<< nmod_{\uparrow}, station >; < nmod/de_{\uparrow}, estación >> \quad (7)$$

$$<< nmod_{\downarrow}, bus >; < nmod/de_{\downarrow}, autobús >> \quad (8)$$

$$<< nmod_{\uparrow}, station >; < nmod_{\uparrow}, estación >> \quad (9)$$

$$<< nmod_{\downarrow}, bus >; < nmod_{\downarrow}, autobús >> \quad (10)$$

A bilingual context, $< c_s; c_t >$, consists of two monolingual contexts, where c_s is the context in the source language, and c_t represents its translation into the target language. The triple $(w_{en}, < c_{en}; c_{es} >, fr)$ represents the number of times (fr) an English word, w_{en}, co-occurs with the English corresponding context, c_{en}, within a monolingual English text. Given the occurrence of *bus station* in the English corpus, which is an instance of the construction $(nmod, N1, N2)$ occurring 1,394 times in that corpus, we extract two triples: triple (11) below representing the frequency of the English word *bus* in the English context of (7) shown above, and triple (13) codifying the frequency of *station* in the English context of (8). In addition, given the occurrence of *estación de autobuses* in the Spanish corpus, which is an instance of the construction $(nmod/de, N1, N2)$ occurring 765 times, we build two more triples: (12) and (14) shown below.

$$(bus, << nmod_{\uparrow}, station, N >; < nmod/de_{\uparrow}, estación >>, 1394) \quad (11)$$

$$(autobús, << nmod_{\uparrow}, station >; < nmod/de_{\uparrow}, estación >>, 767) \quad (12)$$

$$(station, << nmod_{\downarrow}, bus >; < nmod/de_{\downarrow}, autobús >>, 1394) \quad (13)$$

$$(estación, << nmod_{\downarrow}, bus >; < nmod/de_{\downarrow}, autobús >>, 767) \quad (14)$$

Notice that the other candidate translation, *estación autobuses*, instanciating the construction $(nmod, N1, N2)$ and derived from contexts (9) and (10), is not found in the corpus because it is grammatically odd in Spanish.

The vector space is built with the bilingual contexts described above and their word-context co-occurrences. This is thus a count-based approach characterized by being high dimensional and sparse. In order to reduce sparseness, we apply a technique to filter out not very informative contexts by relevance, as described in Gamallo (2017b). The reducing technique consists of two tasks: First, an association measure (e.g., loglikelihood) is computed between each word and their bilingual contexts and, second, for each word, only the N contexts with highest loglikelihood scores are selected. In this bilingual vector space, given a word in the source language, the nearest neighbors in the target language (in terms of distributional similarity) are, in fact, its most likely translation candidates. A more detailed description of our count-based bilingual model can be found in Gamallo and Pichel (2008) and Gamallo and Bordag (2011).

5. Contextualized Translation in a Bilingual Vector Space

Contextualized translation is the result of combining the method to extract bilingual vectors defined in Section 4, with the compositional distributional approach introduced in Section 3.

Figure 3 depicts the general architecture of the strategy, consisting of two main tasks: extraction from monolingual corpora and contextualized translation in a bilingual distributional space. In the figure, the source language is English (en) and the target is Spanish (es). The extraction module, to the left of the figure, requires monolingual corpora in the source and target languages (English and Spanish). All texts of the corpora are linguistically processed and syntactically analyzed. A bilingual dictionary and transfer rules are also required to define a bilingual distributional model, by making use of the technique described in Section 4. The resulting bilingual model provides all English and Spanish words with a distributional meaning representation out of context. This distributional model is the input of the compositional algorithm used by the translation strategy.

The translation module is illustrated in the right side of Figure 3. It consists of three sub-tasks: (1) generation of the Spanish candidates, (2) building compositional models for the English sentence and Spanish candidates, and (3) selection of the most similar candidate.

(1) Generation of candidates: The input of the system is a sequence in English (en) that is syntactically analyzed. The generation sub-module takes the analyzed sentence and expands it into a set of candidate translations in Spanish (es_1, es_2, ..., es_n), by making use of the bilingual dictionary and bilingual transfer rules.

(2) Compositional meaning: Once the candidates have been generated, the next step is to build the distributional meaning representation of the input sentence (*meaning en*) and the translation candidates: *meaning* es_1, ..., *meaning* es_n. For this purpose, the compositional algorithm described in Section 3 makes use of composition functions operating on the bilingual vector space. The distributional meaning of each sentence stands for the contextualized senses of its constituent words.

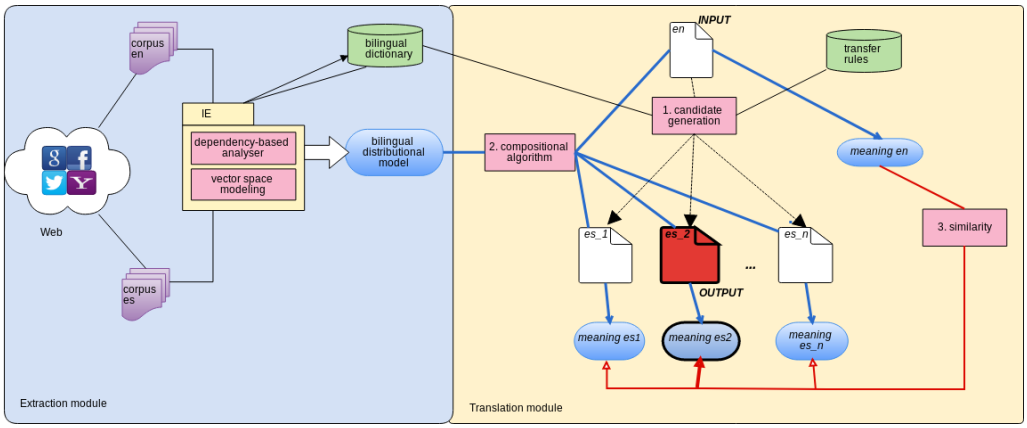

Figure 3
Architecture of the system: extraction and translation modules.

(3) Selection by similarity: Finally, the distributional meanings of the generated candidates are compared pairwise by means of cosine similarity with the English sentence. The generated Spanish sentence associated with the most similar meaning (in bold in the figure) is selected as the best Spanish translation of the English sentence.

It is worth noting that the figure shows a simplified architecture of the translation module, since incrementality across dependencies is not represented. In order to better understand the translation process, the following subsections will help us to explain the three stages of the process from a concrete example: the English expression *coach station*.

5.1 Generation of Candidates

The English expression is syntactically analyzed and then a set of Spanish candidates is generated by using a English–Spanish dictionary and the transfer rules defined above in Equations (5) and (6). Considering the different translations of these two ambiguous words in the dictionary and the two transfer rules, Table 2 shows all 56 possible combinations.

Table 2
56 Spanish candidate translations of "coach station." Only the three in **bold** are acceptable translations. The English *coach* was translated in Spanish by: *bus* (*bus*), *autobús* (*bus*), *autocar* (*bus*), *entrenador* (*trainer*), *instructor* (*instructor*), *preparador* (*trainer*), and *monitor* (*instructor*). And the English *station* was translated by: *estación* (*station*), *canal* (*channel*), *emisora (radio) station*), and *puesto* (*position*). We added the most common English translation of each Spanish word so that readers who do not know Spanish will understand the ambiguity issue.

(*nmod/de*, *estación, bus*), **(*nmod/de*, *estación, autobús*)**,
(*nmod/de*, *estación, autocar*), (*nmod/de*, *estación, entrenador*),
(*nmod/de*, *estación, preparador*), (*nmod/de*, *estación, instructor*),
(*nmod/de*, *estación, monitor*), (*nmod/de*, *canal, bus*),
(*nmod/de*, *canal, autobús*), (*nmod/de*, *canal, autocar*),
(*nmod/de*, *canal, entrenador*), (*nmod/de*, *canal, preparador*),
(*nmod/de*, *canal, instructor*), (*nmod/de*, *canal, monitor*),
(*nmod/de*, *emisora, bus*), (*nmod/de*, *emisora, autobús*),
(*nmod/de*, *emisora, autocar*), (*nmod/de*, *emisora, entrenador*),
(*nmod/de*, *emisora, preparador*), (*nmod/de*, *emisora, instructor*),
(*nmod/de*, *emisora, monitor*), (*nmod/de*, *puesto, bus*),
(*nmod/de*, *puesto, autobús*), (*nmod/de*, *puesto, autocar*),
(*nmod/de*, *puesto, entrenador*), (*nmod/de*, *puesto, preparador*),
(*nmod/de*, *puesto, instructor*), (*nmod/de*, *puesto, monitor*),
(*nmod*, *estación, bus*), (*nmod*, *estación, autobús*),
(*nmod*, *estación, autocar*), (*nmod*, *estación, entrenador*),
(*nmod*, *estación, preparador*), (*nmod*, *estación, instructor*),
(*nmod*, *estación, monitor*), (*nmod*, *canal, bus*), (*nmod*, *canal, autobús*),
(*nmod*, *canal, autocar*), (*nmod*, *canal, entrenador*), (*nmod*, *canal, preparador*), (*nmod*, *canal, instructor*),
(*nmod*, *canal, monitor*), (*nmod*, *emisora, bus*), (*nmod*, *emisora, autobús*),
(*nmod*, *emisora, autocar*), (*nmod*, *emisora, entrenador*),
(*nmod*, *emisora, preparador*), (*nmod*, *emisora, instructor*),
(*nmod*, *emisora, monitor*), (*nmod*, *puesto, bus*), (*nmod*, *puesto, autobús*),
(*nmod*, *puesto, autocar*), (*nmod*, *puesto, entrenador*),
(*nmod*, *puesto, preparador*), (*nmod*, *puesto, instructor*),
(*nmod*, *puesto, monitor*)

5.2 Compositional Meaning

The compositional meaning of the English expression *coach station* corresponds to two compositional vectors, $\mathbf{station}_{nmod\uparrow}$ and $\mathbf{coach}_{nmod\downarrow}$, resulting from the two functions (head and dependent, respectively) derived from the *nmod* relation. Then, the meaning of the 56 Spanish candidates are built following the same procedure, giving rise to 118 compositional vectors. For instance, the two contextualized vectors corresponding to *estación de autobuses*, both derived from the prepositional dependency *nmod/de*, are: $\mathbf{estacion}_{nmod/de\uparrow}$ and $\mathbf{autobus}_{nmod/de\downarrow}$.

5.3 Selection by Similarity

For each binary dependency in the source language, a translation candidate is selected by computing the compositional/contextualized translation measure, *CT*, which selects the most similar expression in the target language by comparing the degree of similarity between heads and dependents in both languages. More precisely, given a composite expression (r, w_1, w_2) in the source language, where r is a dependency linking $w1$ to $w2$, its translation into the target language is computed as follows:

$$CT(r, w_1, w_2) = \underset{(r', w'_1, w'_2) \in \phi}{\arg\max}\ Sim(\mathrm{w}_{1r\uparrow}, \mathrm{w'}_{1r'\uparrow}) + Sim(\mathrm{w}_{2r\downarrow}, \mathrm{w'}_{2r'\downarrow}) \tag{15}$$

where (r', w'_1, w'_2) is any target expression belonging to the set of translation candidates, ϕ. The first *Sim* computes the similarity between the two compositional vectors derived from the head functions. The second one computes the similarity between the vectors derived from the dependent functions. So, the overall similarity between two composite expressions is the basic addition of the similarity scores obtained by comparing their head-based and dependent-based compositional vectors. The resulting translation is thus the expression belonging to ϕ with the highest overall similarity score.

This is a ranked sample of the three most similar candidates with *CT* scores derived from the experiments described in the next section:

$$\begin{aligned}
&Sim(\mathbf{station}_{nmod\uparrow}, \mathbf{estacion}_{nmod/de\uparrow}) + Sim(\mathbf{coach}_{nmod\downarrow}, \mathbf{autobus}_{nmod/de\downarrow}) &&= 0.0912\\
&Sim(\mathbf{station}_{nmod\uparrow}, \mathbf{estacion}_{nmod/de\uparrow}) + Sim(\mathbf{coach}_{nmod\downarrow}, \mathbf{bus}_{nmod/de\downarrow}) &&= 0.0901\\
&Sim(\mathbf{station}_{nmod\uparrow}, \mathbf{estacion}_{nmod/de\uparrow}) + Sim(\mathbf{coach}_{nmod\downarrow}, \mathbf{entrenador}_{nmod/de\downarrow}) &&= 0.0833
\end{aligned} \tag{16}$$

Following our example, the 56 Spanish candidates in Table 2 represent the ϕ set of translation candidates. Only 3 out of 56 are acceptable translations, the rest are unsuitable candidates. Using Equation (15) and the ranked sample (16), the target candidate (*nmod/de*, *estación*, *autobús*) (*estación de autobuses*) is selected as it reaches the highest similarity score to the source expression (*coach station*).

So far, the translation process has been focused on a composite expression constituted by just one binary syntactic dependency. In order to deal with composite expressions with several dependencies (e.g., *the coach station closed*), *CT* measure is computed

for each dependency, while compositional operations are applied in an incremental left-to-right order (as explained in Section 3):

$$\begin{aligned} CT_1(nmod, station, coach) &= (nmod/de, estación, autobús) \\ CT_2(nsubj, close, station) &= (nsubj, cerrar, estación) \end{aligned} \quad (17)$$

where the vectors associated with *station* and *estación* in CT_2 are contextualized, not only by the verb at the subject position, but also by the first combination in CT_1. This gives rise to the following contextualized vectors: $\text{station}_{nsubj\downarrow+nmod\uparrow}$ and $\text{estacion}_{nsubj\downarrow+nmod/de\uparrow}$.

Finally, a very simple decoder takes the *CT* results, dependency by dependency, and builds the lemmatized expression in the target language by taking into account word order information provided by the transfer rules: *estación de autobús cerrar*. In the current version, we only deal with lemmas. In the case of incompatibility between two target words, the decoder adds the CT_i scores obtained by all dependencies in which the incompatible words are involved, and selects the word with the highest global *CT* score. For instance, consider Equation (17) and function CT_2 returning (*nsubj, cerrar, emisora*) instead of (*nsubj, cerrar, estación*). The new Spanish noun *emisora* (*radio station*) is different from *estación* (*bus station*), which has been selected by the first dependency. The two nouns are incompatible as they are assigned to the same syntactic position in the syntactic graph, namely, head of *nmod* and dependent of *nsubj*. In this case, the word with the highest global *CT* value will be *estación* because it reaches high scores in both dependencies and not just in one of them.

6. Experiments

The proposed method for contextualized translation relies on two strategies: compositional distributional semantics and bilingual extraction from monolingual corpora. The syntax-based compositional distributional algorithm described in Section 3 was tested against several monolingual data sets (with intransitive and transitive constructions) and the results of these experiments were reported in Gamallo (2017c). The method to extract bilingual lexicons described in Section 4 participated in the SemEval 2017 Task 10, being the best system using monolingual corpora in the English–Spanish cross-lingual sub-task (Gamallo 2017a).

Concerning contextualized translation, which is the objective of the current work, the most similar task that has been evaluated is cross-lingual semantic textual similarity, which was defined as a shared task at SemEval-2016 Task 1 (Agirre et al. 2016). However, the objective of textual similarity is not to generate a candidate translation but just to provide a degree of similarity between the source and target sentences. The best system in the cross-lingual subtask at SemEval-2016 Task 1 (Brychcín and Svoboda 2016) is very different from the syntax-based strategy we propose in the present work. They translated Spanish sentences to English via Google Translate and, next, made use of the same semantic textual similarity strategy as for the monolingual task. The monolingual task for semantic textual similarity represents the meaning of a sentence using simple linear combination of word vectors, as in the compositional distributional strategy reported in Mikolov, Yih, and Zweig (2013), which is not syntax-based.

Moreover, Task 1 at SemEval-2016 consists of data sets with quite complex and heterogeneous sentences belonging to a large variety of syntactic constructions, which makes it not trivial to treat them through syntax-based compositional approaches. In

Table 3
Sample of English sentences with the *act out* phrasal verb (first column) selected from the data set *PhrasalVerbsToSpanish*. The second column contains the best Spanish translations for each English sentence, and the third one shows the Spanish verbs (lemmas) representing the disambiguated translations of the English phrasal verb.

English sentence	Spanish translations	Spanish verbal phrases
the actors acted out the characters	los actores representaron a los personajes	representar a
the actors act out their performances	los actores interpretan sus representaciones	interpretar
the tired child acted out	el niño cansado se comportó mal, el niño cansado se portó mal	comportar se mal, portar se mal

order to evaluate a syntax-based, contextualized translation system, we require bilingual data sets with simple syntactic constructions, for example, adjective-noun or intransitive and transitive constructions, such as those defined and used for monolingual tasks (Mitchell and Lapata 2008; Grefenstette and Sadrzadeh 2011).

As there is no such bilingual data set with the required characteristics, we created a new resource to evaluate systems aimed at generating contextualized translations in restricted syntactic domains.

6.1 The Data Set

The focus is to create a large number of examples with short and simple constructions, but very ambiguous sentences that require being contextualized in order to be disambiguated. For this purpose, we focused on English sequences containing phrasal verbs, which give rise to very ambiguous expressions. Whereas linguistic ambiguity can be dealt with by means of contextualization, the domain of application is syntactically restricted and, thereby, experiments can be evaluated in an enclosed and controlled setting.

First, an English native translator built a bilingual verbal lexicon with 2,411 different phrasal verbs and 5,761 English–Spanish translations by making use of a great variety of lexicographic resources. Then, she built a list of English (transitive and intransitive) expressions using the most polysemous phrasal verbs of the lexicon. The final data set, called *PhrasalVerbsToSpanish*,[4] consists of 1,119 English sentences with 665 different phrasal verbs, and 1,837 Spanish translations with 1,241 different Spanish verbs (including single and multiword verbs). The 665 English phrasal verbs are highly ambiguous and then have multiple Spanish translations: Their average Spanish translations per verb in the bilingual lexicon is 5.25.

Table 3 is a sample of the data set showing three English sentences with the *act out* phrasal verb. These sentences are in the first column. The Spanish translations for each English sentence are in the second column, and the third column provides lemmatized

4 https://github.com/gamallo/compMT/tree/master/compmtAPI/lib/resources.

Spanish verbs corresponding to the correct translations of the English phrasal verb in context. All examples contain simple constructions: intransitive or transitive constructions merely including noun phrases, verb phrases, adjectives, and prepositional phrases. By contrast, coordination or embedding structures such as relative clauses or completives are not allowed. As distributional-based translation is focused on the meaning of lexical units, grammatical and encyclopedic units such as pronouns, conjunctions, and proper nouns are also not allowed.

The *PhrasalVerbsToSpanish* data set is actually focused on the task of translating the phrasal verb of an English sentence by disambiguating its sense using the meaning of the context words. Thus, contextualization is a key concept in this task. It is worth noting that the bilingual dictionary is used in two different tasks: for constructing this data set and to generate the translation candidates before the contextualized model selects the best one. For constructing the data set, the human translator composed sentences containing the English phrasal verbs included in the dictionary. Concerning the contextualized translation model, both the dictionary and the transfer rules are used to generate candidates that may include phrasal verbs.

6.2 Monolingual Corpora and Linguistic Resources

The extraction module built the bilingual vector space from English and Spanish monolingual corpora. The English corpus consists of 2007–2009 posts of Reddit Comment Corpus, containing about 875 M words.[5] The Spanish corpus corresponds to a 2014 dump file of the Spanish Wikipedia,[6] along with a sample of posts extracted from *MenÉame*.[7] The whole Spanish corpus contains about 480 M word tokens. We decided to use Reddit instead of Wikipedia for English because phrasal verbs are more frequent in informal language such as that used in social forum comments. Notice that the English and Spanish corpora are not comparable.

All texts were linguistically analyzed with LinguaKit (Gamallo et al. 2018), a multilingual suite that also includes the dependency-based parser, DepPattern (Gamallo and Garcia 2018), used to syntactically analyze the two corpora and the input phrases of the translation module. Vectors were built for lexical units occurring more than 100 times in each monolingual corpus.

Concerning the lexical resources, the English–Spanish Collins dictionary,[8] containing 52,463 entries, was merged with our lexicon of phrasal verbs so as to create a new bilingual resource with 57,975 entries. This bilingual dictionary is used for several tasks: to identify English and Spanish phrasal verbs (not only single words) in the monolingual corpora before extraction, to define bilingual distributional contexts in the extraction module, and to generate Spanish candidates in the translation module.

Finally, a set of bilingual transfer rules were manually defined by a linguist. The type of rules chosen to be implemented was determined by the examples of sentences found in the *PhrasalVerbsToSpanish* data set. As they are just transitive and intransitive clauses with no recursive structures and basic nominal modification, most transfer rules required are just duplicated dependencies, as shown in Table 4. The α symbol stands for any English preposition and β represents a Spanish preposition. For the current

5 http://files.pushshift.io/reddit/comments/.
6 http://dumps.wikimedia.org/eswiktionary.
7 https://www.meneame.net/.
8 http://www.collinslanguage.com/.

Table 4
Transfer rules from English to Spanish. V, N, and A stands for verbs, nouns, and adjectives, respectively. Dependency names are inspired by Universal Dependencies (Nivre et al. 2016).

$(nsubj, V, N)$	$\rightarrow$	$((Lnsubj, V, N), LR)$
$(dobj, V, N)$	$\rightarrow$	$((dobj, V, N), RL)$
$(iobj/\alpha, V, N)$	$\rightarrow$	$((iobj/\beta, V, N), RL)$
(cop, V, A)	$\rightarrow$	$((cop, V, A), RL)$
$(amod, N, A)$	$\rightarrow$	$((amod, N, A), RL)$
$(nmod, N1, N2)$	$\rightarrow$	$((nmod, N1, N2), R)$
$(nmod, N1, N2)$	$\rightarrow$	$((nmod/de, N1, N2), R)$
$(nmod/\alpha, N1, N2)$	$\rightarrow$	$((nmod/\beta, N1, N2), R)$

experiments, 13 English prepositions were identified and each one was paired with its three most similar Spanish prepositions, according to distributional similarity. So, each $nmod/\alpha \rightarrow nmod/\beta$ transfer rule was expanded with 13×3 specific rules. In total, 74 specific transfer rules were defined with just verbs, nouns, adjectives, and prepositions. Adverbs and other syntactic categories were not considered for the current experiment.

Transfer rules are provided with four types of word order restrictions: *R* (the dependent word is on the right), *L* (the dependent word is on the left), *RL* (the canonical position of the dependent word is on the right), and *LR* (the cannonical position is on the left). In the last two cases, both positions are allowed but one of them (the non-canonical one) requires more restrictions to be activated.

6.3 Evaluation

Our *Contextualized Translation* (CT) system was evaluated using the *PhrasalVerbsToSpanish* data set as the gold standard. The system selected the most likely translation for each English sentence and then we computed its accuracy. Accuracy is just the result of dividing positive cases by the total size of the data set (1,119 examples). A positive case is defined as follows: The phrasal verb is correctly translated (positive) by checking whether the Spanish verb or phrasal verb in the third column of the data set is also returned by the system. Otherwise, it is considered a negative case.

We also measured four state-of-the-art commercial machine translators, namely DeepL,[9] Google Translator,[10] Bing,[11] and Yandex (all consulted in December 2017).[12] The final evaluation of these systems was done manually because they return inflected verbs that might not match with the verbal lemmas in the third column of the gold standard. So, a manual revision comparing forms with lemmas was required to find positive cases. Additionally, we also implemented some baseline methods. Table 5 shows the accuracy of all evaluated systems as well as a statistical test of significance (last column). The symbols "$\gg$" and "$\ll$" respectively indicate a strong rise and drop with regard to the accuracy of the previous system in the table, being the rise or drop significant for a p-value ≤ 0.005 (paired sample t-test). The symbols "$>$" and "$<$" mean that there is

9 https://www.deepl.com/translator.
10 https://translate.google.com/.
11 https://www.bing.com/translator.
12 https://translate.yandex.com/.

a lighter rise and drop with respect to the previous system, being significant for a p-value ≤ 0.05 and > 0.005. Finally, "$\sim$" indicates that the difference is not statistically significant (p-value > 0.05). Baseline strategies are ordered starting with the lowest accuracy, and commercial translators are ordered from highest to lowest accuracy. The CT system is situated after the best baseline and before the best commercial translator.

Four baseline strategies are on the top of the table. *Dict-first* is based on looking up our bilingual lexicon of phrasal verbs. This method identifies the phrasal verb within the English sentence, looks up the lexicon, and selects the first Spanish translation. The result of this baseline, 0.312 accuracy, is in accordance with the fact that the phrasal verbs occurring in the English sentences of the data set have 5.5 translations/meanings on average, and there are some examples where more than one translation is allowed.

We also tested two more baselines based on non-compositional similarity. *Dict-Nocomp* compares each phrasal verb with just their translation candidates generated with the bilingual lexicon, and the most similar one is selected. The similarity is computed on the same transparent bilingual vector space as the one used by our CT system. *Dict-Nocomp-VecMap* computes the same non-compositional similarity by using word embeddings for each language and a linear mapping between the two vector spaces (Mikolov, Le, and Sutskever 2013). The mapping between embeddings was learned using VecMap (Artetxe, Labaka, and Agirre 2018). These two non-compositional methods returned scores (0.383 and 0.390), significantly improving the accuracy of random dictionary consultation (*Dict-first*) for a p-value ≤ 0.005. However, such an improvement is not too pronounced (less than 8 points over 100). The reason is that non-compositional similarity tends to select the most popular sense/translation, but many examples of the phrasal verbs in the gold standard were created with infrequent meanings. Rare senses are just those that a compositional strategy should try to select in context. It is worth noting that there is no significant difference between the two non-compositional strategies, even though the use of VecMap slightly improves our way of computing vector similarity.

The other baseline is *Dict-Corpus-Based*, which implements the corpus-based strategy described in Grefenstette (1999) by making use of our bilingual dictionary. More precisely, we translated each individual word (including phrasal verbs) of the input English sentence by using the dictionary, and then we generated all the possible well-formed combinations in Spanish. Finally, for each input sentence, we selected the phrasal verb occurring in the most frequent combinations in the Spanish corpus. For instance, let us take the input sentence *the man acts as a manager*, and the following Spanish translations of the constituent words found in our dictionary: *man* is translated by *hombre*, *acts as* by both *servir de* and *hacer de*, and *manager* by both *director* and *gerente*. Then, all the possible syntactic combinations are generated and their frequency is extracted from the syntactically analyzed Spanish corpus:

$$\begin{array}{ll} (nsubj, servir\_de, hombre), 1 & (dobj, servir\_de, director), 0 \\ (nsubj, hacer\_de, hombre), 1 & (dobj, hacer\_de, director), 4 \\ (dobj, servir\_de, gerente), 0 & (dobj, hacer\_de, gerente), 0 \end{array} \quad (18)$$

Finally, the phrasal verb with the highest frequency is selected: *hacer de*, with 5 (4+1) occurrences in total. Notice that this is the correct answer because in the gold reference the human translator also selected *hacer de* as the best choice for the input sentence. The accuracy of this strategy (0.335) is higher than that obtained by the *Dict-First* baseline, even though there is just a slight improvement with a p-value $= 0.03$. On the

other hand, the *Dict-Corpus-Based* strategy is outperformed by the non-compositional baselines in a significant way (p-value $= 0.006$).

As Table 5 shows, CT outperforms both the best baseline (*Dict-Nocomp-VecMap*) and the best commercial translator (DeepL) in a significant way (p-value ≤ 0.005). Concerning the differences between the four commecial systems, all are strongly significant (p-value ≤ 0.005), except that separating Bing from Yandex, which is just significant for a p-value ≤ 0.05. It is worth noting that all accuracy scores are low because the task at stake has a high degree of complexity. All sentences contain very ambiguous phrasal verbs, some of them with very infrequent senses, even though all of them can be disambiguated by considering the meaning of the context words: nominal subjects and/or objects. The improvement of our system with regard to the baseline (from 349 to 571 positives) is perhaps not conclusive, but it shows that the compositional vectors built by the CT system help contextualize in an important number of cases.

The difficulty of the task is demonstrated by the low values obtained by the unsupervised machine translation system, UNdreaMT (Artetxe et al. 2017) (last row in Table 5). This is a state-of-the-art, unsupervised translation strategy, based on denoising and back-translation, whose embeddings are learned from monolingual corpora. We have trained UNdreaMT using the embeddings mapped with VecMap, and the embeddings were built by applying word2vec (CBOW algorithm, window 5, and 300 dimensions) (Mikolov, Yih, and Zweig 2013) on the same English and Spanish corpora as the ones we used to train our CT method. However, it should be noted that UndreaMT is at a clear disadvantage with respect to all the systems it is compared with: On the one hand, it is a completely unsupervised system that has been trained with very small

Table 5
Accuracy of our system, *Contextualized Translation* (CT), using the data set *PhrasalVerbsToSpanish*, together with the scores obtained with state-of-the-art machine translators: DeepL, Google Translator, Bing, and Yandex (all consulted in December 2017). Four baseline methods, based on looking up our bilingual lexicon of phrasal verbs, are also evaluated along with an unsupervised machine translation system, UNdreaMT (Artetxe et al. 2017). The last column shows the statistical significance test comparing each system with the previous one in the table.

systems	*positive*	*negative*	*accuracy*	*s-test*
Dict-first	349	770	0.312	
Dict-Corpus-Based	375	744	0.335	$>$
Dict-Nocomp	430	689	0.383	$\gg$
Dict-Nocomp-VecMap	437	682	0.390	$\sim$
CT	571	548	0.510	$\gg$
DeepL	501	618	0.447	$\ll$
Google Trans.	410	709	0.366	$\ll$
Bing	326	793	0.291	$\ll$
Yandex	281	838	0.251	$<$
UNdreaMT	12	1,107	0.010	$\ll$

corpora in relation to the commercial translators that use huge amounts of parallel corpora. On the other hand, the generation of word sequences in the target language is not controlled by a bilingual dictionary and syntax-based transfer rules as in the case of CT. We must point out that the commercial translators also do not have access to the bilingual dictionary and the generated candidates, which places them at a disadvantage with respect to CT.

6.4 Error Analysis

In order to analyze the type of errors produced by the evaluated systems, 50 negative examples were randomly selected and manually classified to error categories. Table 6 shows the distribution of the four error types found in the sample. The only negative cases that can be considered as being clearly wrong choices directly derived from the translation system are called "semantically odd," and reach 34% of the total sample. For instance, the CT system wrongly chose the Spanish verb *matar* (*to kill*) to translate *blow away* in *the singer blew away the audience*, instead of *deslumbrar a*. In these cases, the translation module did not select a semantically acceptable translation. By contrast, 32% (called "similar sense") are acceptable cases in CT even if the most acceptable translation, which fits better the collocation requirements, has not been chosen. For example, in *the state acted on the evidence*, the system returned *responder a*, but the human translator preferred another, more appropriate option (*reaccionar ante*), which is semantically similar but seems to be more used in that specific context. It is therefore a stylistic error, less serious than the previous one. We also found a significant number of errors (22%) in CT inherited from the linguistic analyzers, either PoS tagger or dependency parser, which intervene before the construction of the compositional meanings in the semantic step. Finally, the fourth type of error (*preposition*) stands for those cases where the Spanish verb is correct but the preposition is missing, as when the preposition "a" introduces direct objects. In such cases, the presence of the preposition is recommended but not mandatory, for instance: *the worker decided to ask around his colleagues* is translated as *trabajador tantear colega*, instead of *trabajador tantear a colega*. So, it is a stylistic error, like the second one. For CT, this analysis shows that serious errors ("semantically odd") are only one-third of the total, and that there is room for improvement by solving morpho-syntactic problems.

With respect to the other systems evaluated, the number of *semantically odd* cases exceed 50%, except for DeepL, which always tries to find an interpretable solution and, in many cases, semantically approaches the most appropriate target expression. It is worth

Table 6
Error analysis on 50 randomly selected negative examples classified in four types of errors.

systems	*semantically odd*	*similar sense*	*wrong analysis*	*preposition*
CT	34%	32%	22%	12%
Dict-Corpus-Based	52%	40%	–	8%
Dict-Nocomp	66%	28%	–	6%
DeepL	32%	68%	–	–
Google Trans.	58%	42%	–	–
Bing	78%	22%	–	–
Yandex	50%	50%	–	–

noting that the error type called *wrong analysis* only applies to CT because it is the only strategy based on PoS tagging and syntactic parsing. Besides, it is interesting to note that commercial translators do not make mistakes with the preposition *a*. *Dict-Nocomp-VecMap* has not been analyzed, as it is not significantly different from *Dict-Nocomp-Vec*. The two systems with lowest accuracy values, namely, *Dict-First* and *UMdreaMT*, have not been analyzed either.

Our CT system, along with the gold data set, are freely available.[13] A Web demonstrator using our technology to translate short sentences is also provided.[14]

7. Conclusions

One of the main benefits of distributional compositionality is that the systems based on it (try to) solve word-sense disambiguation by modeling the mutual contextualization of words in a compositional way, guided by the syntactic structure. In this article, we claim that it is possible to apply the same procedure on a bilingual vector space to propose contextualized translations.

We have worked with count-based vector spaces because their dimensions are more transparent, more interpretable, and easier to combine in a compositional way than neural network-based models (word embeddings). However, as deep-learning compositional models are emerging in recent years (Cheng, Kartsaklis, and Grefenstette 2014; Cheng and Kartsaklis 2015), they should be studied in order to discover how they might be used for modeling compositional distributional translation.

It is important to point out three important drawbacks of the proposed contextualization method that need to be addressed in the future. First, in the case of collocations such as *save time*, *go mad*, *heavy rain*, the borderline between compositional and non-compositional interpretation is blurred. For these cases, it is not clear whether it is more appropriate to apply either a compositional method of contextualization or simply identify them previously as non-compositional expressions along with their frequency in a corpus. Second, in the case of complex expressions giving rise to deep dependency trees, we may have frequency scarcity problems due to the iterative application of several contextualizations to the same word vector. And third, as transfer rules are manually defined, it makes it complicated to extend the model to more language pairs. These are challenges that we will have to take into account in the future when we extend the approach to all types of linguistic expressions and other language pairs.

In future work, we will address and go into detail about the idea of incremental translation, guided dependency-by-dependency. With the help of incremental translation, we think that unsupervised machine translation, based on monolingual corpora, can be improved. For this purpose, it will be necessary to better define how to generate translation candidates (our ϕ set) at whatever level of composition. Translating dependency-by-dependency with a narrow set of translation candidates and few transfer rules would yield too literal and poor quality translations. To expand the set of candidates, we should consider pseudo-compositional compounds that may be better translated by a single word, as well as **fertile translations**, that is, translations in which the target term has more words than the source one. Moreover, in order to avoid the limitations of generating candidates through a bilingual dictionary, we will

13 https://github.com/gamallo/compMT.
14 http://fegalaz.usc.es/compmt/.

also generate candidates from the context-free bilingual word embeddings learned from monolingual corpora, such as VecMap or similar cross-lingual techniques.

However, if the set of candidates is expanded too much, other problems may arise concerning both precision and computational efficiency. In order to expand candidates in a controlled manner, it would be required to define transfer rules by taking into account complex syntactic alternations at the level of the sentence construction: passive/active, transitive/unaccusative, and so forth. In fact, the translation system should be provided with a rich set of cross-lingual *constructions* (Boas 2010) to define deep syntactic transfer rules and thereby expand the set of candidates in a much more accurate way. By doing this, the translation system would be actually based on a hybrid strategy, relying on deep linguistic knowledge and corpora-based data collected by distributional methods.

Acknowledgments

This work has received financial support from the FBBVA Leonardo program, the DOMINO project (PGC2018-102041-B-I00, MCIU/AEI/FEDER, UE), the Consellería de Cultura, Educación e Ordenación Universitaria (accreditation 2016-2019, ED431G/08) and the European Regional Development Fund (ERDF). Mikel Artetxe has a doctoral grant from the Spanish MECD.

References

Agirre, Eneko, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, CA.

Aker, Ahmet, Monica Paramita, and Robert Gaizauskas. 2013. Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 4–9, Sofia.

Andrade, D., T. Matsuzaki, and J. Tsujii. 2011. Learning the optimal use of dependency-parsing information for finding translations with comparable corpora. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web*, pages 10–18, Portland, OR.

Ansari, Ebrahim, M. H. Sadreddini, Alireza Tabebordbar, and Mehdi Sheikhalishahi. 2014. Combining different seed dictionaries to extract lexicon from comparable corpus. *Indian Journal of Science and Technology*, 7(9):1279–1288.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, pages 2289–2294, Austin, TX.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Volume 1: Long Papers*, pages 451–462, Vancouver.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2018. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, pages 5012–5019, New Orleans, LA.

Artetxe, Mikel, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *CoRR*, abs/1710.11041.

Baroni, Marco. 2013. Composition in distributional semantics. *Language and Linguistics Compass*, 7:511–522.

Baroni, Marco, Raffaella Bernardi, and Roberto Zamparelli. 2014. Frege in space: A program for compositional distributional semantics. *LiLT*, 9:241–346.

Baroni, Marco 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.

Boas, Hans. 2010. *Contrastive Studies in Construction Grammar*. John Benjamins Publishing Company.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1991. Word-sense disambiguation using statistical methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, ACL '91, pages 264–270, Stroudsburg, PA.

Brychcín, Tomáš and Lukáš Svoboda. 2016. UWB at Semeval-2016 task 1: Semantic textual similarity using lexical, syntactic, and semantic information. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 588–594, San Diego, CA.

Cheng, Jianpeng and Dimitri Kartsaklis. 2015. Syntax-aware multi-sense word embeddings for deep compositional models of meaning. In *Conference on Empirical Methods in Natural Language Processing, EMNLP-2015*, pages 1531–1542, Lisbon.

Cheng, Jianpeng, Dimitri Kartsaklis, and Edward Grefenstette. 2014. Investigating the role or prior disambiguation in deep-learning compositional models of meaning. In *Proceedings of Learning Semantics Workshop, NIPS-2014*, Montreal.

Chiao, Y.-C. and P. Zweigenbaum. 2002. Looking for candidate translational equivalents in specialized, comparable corpora. In *19th COLING'02*, pages 1–5, Taipei.

Cho, Kyunghyun, Bart van Merriënboer, Çağlar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha.

Coecke, B., M. Sadrzadeh, and S. Clark. 2010. Mathematical foundations for a compositional distributional model of meaning. *Linguistic Analysis*, 36(1-4):345–384.

Dagan, Ido. 1991. Lexical disambiguation: Sources of information and their statistical realization. In *Proceedings of the 29th Annual Meeting on Association for Computational Linguistics*, ACL '91, pages 341–342, Stroudsburg, PA.

Delpech, Estelle, Béatrice Daille, Emmanuel Morin, and Claire Lemaire. 2012. Extraction of domain-specific bilingual lexicon from comparable corpora: Compositional translation and ranking. In *Proceedings of COLING 2012, 24th International Conference on Computational Linguistics*, pages 745–762, Mumbai.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Erk, Katrin and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of EMNLP*, pages 897–906, Honolulu, HI.

Erk, Katrin, Sebastian, Padó, and Ulrike Padó. 2010. A flexible, corpus-driven model of regular and inverse selectional preferences. *Computational Linguistics*, 36(4):723–763.

Fung, Pascale and Kathleen McKeown. 1997. Finding terminology translation from non-parallel corpora. In *5th Annual Workshop on Very Large Corpora*, pages 192–202, Hong Kong.

Fung, Pascale and Lo Yuen Yee. 1998. An IR approach for translating new words from nonparallel, comparable texts. In *COLING'98*, pages 414–420, Montreal.

Gamallo, P., M. Garcia, C. Piñeiro, R. Martinez-Castaño, and J. C. Pichel. 2018. Linguakit: A big data-based multilingual tool for linguistic analysis and information extraction. In *2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 239–244, Valencia.

Gamallo, Pablo. 2007. Learning bilingual lexicons from comparable English and Spanish corpora. In *Machine Translation SUMMIT XI*, pages 191–197, Copenhagen.

Gamallo, Pablo. 2017a. Citius at Semeval-2017 Task 2: Cross-lingual similarity from comparable corpora and dependency-based contexts. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 226–229, Vancouver.

Gamallo, Pablo. 2017b. Comparing explicit and predictive distributional semantic models endowed with syntactic contexts. *Language Resources and Evaluation*, 51(3):727–743.

Gamallo, Pablo. 2017c. The role of syntactic dependencies in compositional distributional semantics. *Corpus Linguistics and Linguistic Theory*.

Gamallo, Pablo and Stefan Bordag. 2011. Is singular value decomposition useful for word similarity extraction? *Language Resources and Evaluation*, 45(2):95–119.

Gamallo, Pablo and Marcos Garcia. 2018. Dependency parsing with finite state transducers and compression rules.

Information Processing & Management, 54:1244–1261.

Gamallo, Pablo and José Ramom Pichel. 2008. In Learning Spanish-Galician translation equivalents using a comparable corpus and a bilingual dictionary. In *International Conference on Intelligent Text Processing and Computational Linguistics*, volume 4919 of Lecture Notes in Computer Science. Springer, pages 423–433.

Grefenstette, Edward and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *Conference on Empirical Methods in Natural Language Processing*, pages 1394–1404, Edinburgh.

Grefenstette, Edward, Mehrnoosh Sadrzadeh, Stephen Clark, Bob Coecke, and Stephen Pulman. 2011. Concrete sentence spaces for compositional distributional models of meaning. In *Proceedings of the Ninth International Conference on Computational Semantics*, IWCS '11, pages 125–134, Oxford.

Grefenstette, Gregory. 1999. The World Wide Web as a resource for example-based machine translation tasks. In *Translating and the Computer 21: Proceedings of the 21st International Conference on Translating and the Computer*, London.

Guevara, Emiliano. 2010. A regression model of adjective-noun compositionality in distributional semantics. In *Proceedings of the 2010 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '10, pages 33–37, Uppsala.

Hashimoto, Kazuma and Yoshimasa Tsuruoka. 2015. Learning embeddings for transitive verb disambiguation by implicit tensor factorization. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 1–11, Beijing.

Hazem, Amir and Emmanuel Morin. 2014. Improving bilingual lexicon extraction from comparable corpora using window-based and syntax-based models. *Lecture Notes in Computer Science*, 8404:310–323.

Hudson, Richard. 2003. The psychological reality of syntactic dependency relations. In *MTT2003*, pages 181–192, Paris.

Irsoy, Ozan and Claire Cardie. 2014. Deep recursive neural networks for compositionality in language. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 2096–2104, Montreal.

Ismail, A. and S. Manandhar. 2010. Bilingual lexicon extraction from comparable corpora using in-domain terms. In *Proceedings of 23rd International Conference on Computational Linguistics*, pages 481–489, Beijing.

Kahane, Sylvain. 2003. Meaning-text theory. In *Dependency and Valency: An International Handbook of Contemporary Research*. Berlin: De Gruyter.

Kober, Thomas, Julie Weeds, Jeremy Reffin, and David J. Weir. 2017. Improving semantic composition with offset inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017 Volume 2: Short Papers*, pages 433–440, Vancouver.

Krishnamurthy, Jayant and Tom Mitchell. 2013. Vector Space Semantic Parsing: A Framework for Compositional Vector Space Models. *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 1–10, Sofia.

Lample, Guillaume, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc Aurelio Ranzato. 2018. Phrase-Based & neural unsupervised machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels.

McCann, Bryan, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. *CoRR*, abs/1708.00107.

Melamud, Oren, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional LSTM. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61, Berlin.

Mikolov, Tomas, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168.

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013. 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, GA.

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

Mitchell, Jeff and Mirella Lapata. 2009. Language models based on semantic composition. In *Proceedings of EMNLP*, pages 430–439, Singapore.

Mitchell, Jeff and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1439.
Montague, Richard. 1970. Universal grammar. *Theoria*, 36:373–398.
Morin, Emmanuel and Béatrice Daille. 2012. Revising the compositional method for terminology acquisition from comparable corpora. In *COLING 2012, 24th International Conference on Computational Linguistics*, pages 1797–1810, Mumbai.
Nivre, Joakim, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666, Portorož.
Peters, Matthew, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, LA.
Pham, Nghia The, Germán Kruszewski, Angeliki Lazaridou, and Marco Baroni. 2015. Jointly optimizing word representations for lexical and sentential tasks with the C-PHRASE model. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, Volume 1: Long Papers*, pages 971–981, Beijing.
Rapp, Reinhard. 1999. Automatic Identification of Word Translations from Unrelated English and German Corpora. In *ACL'99*, pages 519–526, College Park, MD.
Resnik, Philip. 1996. Selectional constraints: An information-theoretic model and its computational realization. *Cognition*, 61:127–159.
Rubino, Raphael and Georges Linarés. 2011. A multi-view approach for term translation spotting. In *CICLing 2011*, pages 29–40, Tokyo.
Saralegi, X., I. San Vicente, and A. Gurrutxaga. 2008. Automatic generation of bilingual lexicons from comparable corpora in a popular science domain. In *LREC 2008 Workshop on Building and Using Comparable Corpora*, pages 27–32, Jeju Island.
Shao, Li and Hwee Tou Ng. 2004. Mining New Word Translations from Comparable Corpora. In *20th International Conference on Computational Linguistics (COLING 2004)*, pages 618–624, Geneva.
Socher, Richard, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 1201–1211, Stroudsburg, PA.
Tamura, Akihiro, Taro Watanabe, and Eiichiro Sumita. 2012. Bilingual lexicon extraction from comparable corpora using label propagation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 24–36, Jeju Island.
Tanaka, Takaaki and Timothy Baldwin. 2003. Noun-noun compound machine translation: A feasibility study on shallow processing. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 17–24, Sapporo.
Thater, Stefan, Hagen Fürstenau, and Manfred Pinkal. 2010. Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 948–957, Stroudsburg, PA.
Turney, Peter D. 2013. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research (JAIR)*, 44:533–585.
Upadhyay, Shyam, Yogarshi Vyas, Marine Carpuat, and Dan Roth. 2018. Robust cross-lingual hypernymy detection using dependency context. *CoRR*, abs/1803.11291.
Weir, David J., Julie Weeds, Jeremy Reffin, and Thomas Kober. 2016. Aligning packed dependency trees: A theory of composition for distributional semantics. *Computational Linguistics*, 42(4):727–761.
Wijaya, Derry Tanti, Brendan Callahan, John Hewitt, Jie Gao, Xiao Ling, Marianna Apidianaki, and Chris Callison-Burch. 2017. Learning translations via matrix completion. In *Proceedings of the 2017*

Conference on Empirical Methods in Natural Language Processing, pages 1452–1463, Copenhagen.

Yu, Kun and Junichi Tsujii. 2009a. Bilingual dictionary extraction from Wikipedia. In *Machine Translation Summit XII*, Ottawa.

Yu, Kun and Junichi Tsujii. 2009b. Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In *NAACL HLT 2009*, pages 121–124, Boulder, CO.

Zanzotto, Fabio Massimo, 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 '10, pages 1263–1271, Beijing.