Multilingual Extraction and Categorization of Lexical Collocations with Graph-aware Transformers

Luis Espinosa-Anke[†] **Alexander Shvets**[♡] **Alireza Mohammadshahi**[♢♠]
James Henderson[♢] **Leo Wanner**[♣♡]
[†]CardiffNLP (Cardiff University) - AMPLYFI [♡]TALN Group, Universitat Pompeu Fabra
[♢]Idiap Research Institute [♠]EPFL [♣]ICREA
espinosa-ankel@cardiff.ac.uk
{alexander.shvets,leo.wanner}@upf.edu
{alireza.mohammadshahi,james.henderson}@idiap.ch

Abstract

Recognizing and categorizing lexical collocations in context is useful for language learning, dictionary compilation and downstream NLP. However, it is a challenging task due to the varying degrees of frozenness lexical collocations exhibit. In this paper, we put forward a sequence tagging BERT-based model enhanced with a graph-aware transformer architecture, which we evaluate on the task of collocation recognition in context. Our results suggest that explicitly encoding syntactic dependencies in the model architecture is helpful, and provide insights on differences in collocation typification in English, Spanish and French.[1]

1 Introduction

Native speech is idiosyncratic. Of special prominence are syntactically-bound restricted binary co-occurrences of lexical items, in which one of the items conditions the selection of the other item. Consider a CNN sports headline from 02/15/2021:

> Rafael Nadal eases into Australian Open quarterfinals, remains on course for record-breaking grand slam (cnn.com).

In this short headline, we see already three of such co-occurrences: *ease* [*into*] *quarterfinals*, *remain* [*on*] *course*, and *record-breaking grand slam*. *Quarterfinals* conditions the selection of [*to*] *ease* [*into*], *course* of *remain* [*on*], and *grand slam* of *record-breaking*. The idiosyncrasy of these co-occurrences becomes obvious when we look at them from a multilingual angle. Thus, in French, instead of the literal translation of *ease* [*into*], we would use *se qualifier* 'qualify [oneself]', in Spanish, *remain* [*on*] will be translated as *seguir* [*en*] 'continue in', and in Italian *record-breaking* will be *da record*, lit. 'of record' – while the translation of *quarterfinals*, *course*, and *grand slam* will be literal. In lexicology, such binary co-occurrences are referred to as *collocations* (Hausmann, 1985; Cowie, 1994; Mel'čuk, 1995; Kilgarriff, 2006), with the conditioning item called the *base* and the conditioned item the *collocate*. Collocations in this sense are of high relevance to second language learning, lexicography and NLP alike, and constitute a challenge for computational models because of their heterogeneity in terms of idiosyncrasy and degree of semantic composition (Mel'čuk, 1995).

Research in NLP has already addressed a number of collocation-related tasks, in particular: **(1)** collocation error detection, categorization, and correction in writings of second language learners (Ferraro et al., 2011; Wanner et al., 2013; Ferraro et al., 2014; Rodríguez-Fernández et al., 2015); **(2)** creation of collocation-enriched lexical resources (Espinosa-Anke et al., 2016; Maru et al., 2019; Di Fabio et al., 2019); **(3)** use of knowledge on collocations in downstream NLP tasks, among them, e.g., machine translation (Seretan, 2014), word sense disambiguation (Maru et al., 2019), natural language generation (Wanner and Bateman, 1990), or semantic role labeling (Scozzafava et al., 2020); **(4)** probes involving collocations for understanding to which extent language models are able to identify non-compositional meanings (Shwartz and Dagan, 2019; Garcia et al., 2021); and **(5)** detection and categorization of collocations with respect to their semantics (Wanner et al., 2006; Espinosa Anke et al., 2019; Levine et al., 2020; Espinosa-Anke et al., 2021). It is this last task which is the focus of this paper.

In general, collocation identification and categorization tend to be treated as two disjoint tasks. Most of the research deals only with collocation identification (Smadja, 1993; Lin, 1999; Pecina and Schlesinger, 2006; Bouma, 2009; Dinu et al., 2014; Levine et al., 2020). Some works deal with the categorization of manually precompiled lists

[1]Data and code are available at https://github.com/TalnUPF/graph-aware-collocation-recognition.

Proceedings of the 11th Joint Conference on Lexical and Computational Semantics, pages 89 - 100

of collocations, either in isolation (Wanner, 2004; Wanner et al., 2006; Espinosa Anke et al., 2019) or with their original sentence-level contextual information (Wanner et al., 2017). Only a few works in the early phase of the neural network era of NLP address the problem of collocation identification and semantic categorization as a joint task in monolingual settings (Rodríguez-Fernández et al., 2015; Espinosa-Anke et al., 2016). Accordingly, the performance of the models put forward in these works is still rather low. In this paper, we propose a sequence tagging framework for simultaneous collocation identification and categorization, with respect to the taxonomy of *lexical functions* (LFs) (Mel'čuk, 1996). The proposed framework is based on mono- and multilingual BERT-based sequence taggers, which are enhanced by a Graph-aware Transformer (Mohammadshahi and Henderson, 2020, 2021a) in order to ensure that the specific syntactic dependencies between the base and the collocate are taken into account. The sequence taggers are executed as part of a multitask learning setup, which is complemented by a sentence classification task, which predicts the occurrence of an instance of a specific LF instance in the sentence under consideration. Our results for English, French and Spanish show the flexibility of our framework and shed light on the multilingual idiosyncrasies of collocations.

2 Background on Collocations

Although widely used in lexicology in the sense defined above, the term *collocation* is ambiguous in linguistics. As introduced by Firth (1957), it refers to common word co-occurrences in discourse in general. Thus, *cast* and *vote*, *strong* and *tea*, but also *public* and *sector*, *night* and *porter*, *supermarket* and *price* form collocations in English in Firth's sense. In computational linguistics, Firth's definition has been taken up, e.g., by (Church and Hanks, 1989; Lin, 1999; Evert, 2007; Pecina, 2008; Bouma, 2009; Dinu et al., 2014; Levine et al., 2020). To avoid confusion between the two different senses, Krenn (2000) proposed to use the narrower term *lexical collocation* to refer to restricted binary lexical item co-occurrences. In what follows, we will use this term to refer to the definition underlying our work.

Lexical collocations can be typified with respect to the meaning of the collocate and the syntactic structure formed by the base and the collocate.

relation	example	LF label
intense	$heavy_C \sim smoker_B$	Magn
minor	$occasional_C \sim smoker_B$	AntiMagn
genuine	$legitimate_C \sim demand_B$	Ver
non-genuine	$illegitimate_C \sim demand_B$	AntiVer
Increase.existence	$temperature_B \sim rise_C$	IncepPredPlus
End.existence	$fire_B \sim go\ out_C$	FinFunc0
A0.Come.to.effect	$avalanche_B \sim strike_C$	Fact0
A0/A1.Cause.existence	$raise_C \sim hope_B$	CausFunc0
A0/A1.Cause.function	$start_C \sim engine_B$	CausFact0
Cause.decrease	$relieve_C \sim tension_B$	CausPredMinus
A0/A1.Cause.involvement	$raise_C\ hope_B$ [*in*]	CausFunc1
Emit.sound	$elephant_B \sim trumpet_C$	Son
A0/A1.act	$lend_C \sim support_B$	Oper1
A0/A1.begin.act	$gain_C \sim impression_B$	IncepOper1
A0.end.act	$withdraw_C \sim support_B$	FinOper1
A0/A1.Act.acc.expectation	$prove_C \sim accusation_B$	Real1
A2.Act.acc.expectation	$enjoy_C \sim support_B$	Real2
A2.Act.x.expectation	$betray_C \sim trust_B$	AntiReal2

Table 1: LF relations used in this paper. 'A_i' refer to AMR argument labels (Banarescu et al., 2013).

Practical collocations dictionaries such as, e.g., the *Oxford Collocations Dictionary*[2] or the *McMillan Collocations Dictionary*[3], already offer a coarse-grained semantic typification. However, their typification still does not make a distinction between, e.g., *control* and *cut* in co-occurrence with *expenditure* or between *cavernous* and *palatial* in co-occurrence with *room* — distinctions which are essential in the context of both second language learning and NLP. To the best of our knowledge, *lexical Functions* (LFs) (Mel'čuk, 1996) are the most fine-grained taxonomy of lexical collocations.

A lexical function (LF) is defined as a function $f(B)$ that delivers for a base B a set of synonymous collocates that express the meaning of f. LFs are assigned Latin abbreviations as labels; cf., e.g., "Oper1" ("operare" 'perform'): Oper1(*condolences*) = {*convey*, *express*, *extend*}; "Magn" ("magnum" 'big'/'intense'): Magn(*grief*) = {*deep*, *inconsolable*, *great*, ... }. Each LF can also be considered as a specific lexico-semantic relation between the base and the collocate of a collocation in question (Evens, 1988). Table 1 displays the subset of the relations we experiment with, along with their corresponding LF names and illustrative examples.

As seen in Table 1, where pertinent, an LF label also encodes the subcategorization structure of the base+collocate combination; cf., e.g., FinFunc0, Oper1, Real2, etc. Thus, the index '1' in Oper1 encodes the information that the first argument of the base (A0/A1) is realized as grammatical subject and the base itself as object; the '2' in Real2

[2]https://www.freecollocation.com/

[3]https://www.macmillandictionary.com/collocations

encodes the realization of the second argument of the base (A2) as grammatical subject and the base as object; etc. This generic structure translates into a number of *Universal Dependency* (UD) patterns.

3 Related Work

Previous works that consider collocations in a Firthian sense look at word adjacency in terms of n-grams (Smadja, 1993), although most often, statistical measures of co-occurrence are used; cf. Pearce et al. (2002); Pecina and Schlesinger (2006); Pecina (2010); Garcia et al. (2019). Some complement statistical measures by morphological (Krenn and Evert, 2001; Evert and Krenn, 2001) and/or syntactic (Heid and Raab, 1989; Lin, 1999; Seretan and Wehrli, 2006) patterns. In view of the *asymmetrical* nature of the relation between the base and the collocate, e.g., Gries (2013) proposes to investigate "directional measures" as an addition to association measures; Carlini et al. (2014) explicitly encode this asymmetry in terms of NPMI (Bouma, 2009), which is a normalized version of PMI; see also (Garcia et al., 2019). In the collocation classification task, substantial research focused on the identification of *Light Verb Constructions*, which are captured by the Oper- (and partially by the Real-) families of LFs; cf., e.g., (Dras, 1995; Vincze et al., 2013; Kettnerová et al., 2013; Chen et al., 2016; Cordeiro and Candito, 2019; Shwartz and Dagan, 2019), whereas Huang et al. (2009) and Wanner et al. (2017) focus on broad semantic collocation categories. Several works also use LFs as a collocation taxonomy. Thus, Wanner et al. (2006) leverage a vector-based similarity metric on a subset of LFs, whereas Gelbukh and Kolesnikova (2012) explore a suite of classical supervised ML algorithms.

More recently, word embeddings have been successfully applied in unsupervised setups, e.g., Rodríguez Fernández et al. (2016a) use simple vector arithmetic. In supervised setups, we find, first, the "collocate retrieval" approach proposed by Rodríguez Fernández et al. (2016b), who train a linear transformation to go from a "base" to a "collocate" vector space, exploiting regularities in multilingual word embeddings (Mikolov et al., 2013), and second, Espinosa Anke et al. (2019), who train an SVM on a dedicated relation vector space for base and collocate. Embeddings have also been used in multilingual English/Spanish (Rodríguez Fernández et al., 2016b) and English/Portuguese/Spanish

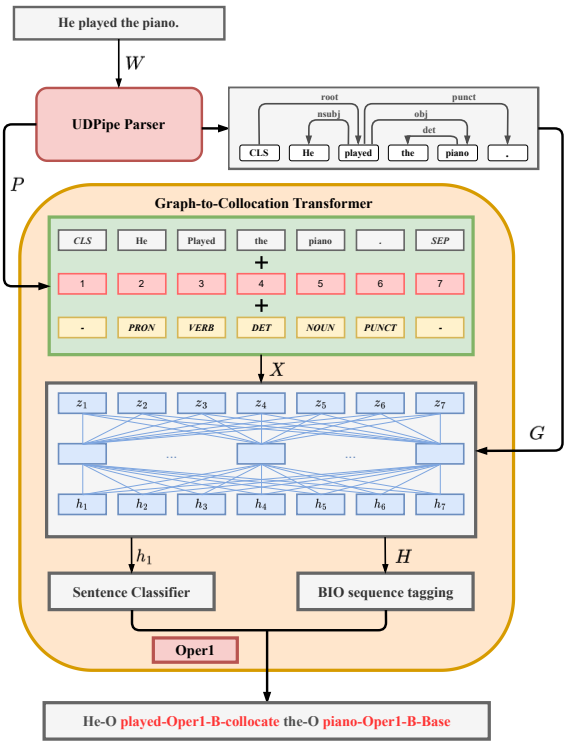


Figure 1: Graph-to-Collocation Transformer, which generates a BIO-tagged sequence given a sentence with, optionally, its parsed tree.

(Garcia et al., 2017) LF classification. While successful, none of these approaches explicitly leveraged in the language model the crucial syntactic dependency information between base and collocate, or considered how sentence-level information could benefit the extraction task – as we do.

4 Graph-to-Collocation Transformer

We propose a Graph-to-Collocation Transformer (G2C-Tr) to: (1) cast collocation identification and classification as a **sequence tagging** problem: as pointed out above, lexical collocations are lexico-semantic relations, and relation extraction has been recently successfully addressed as sequence tagging (Ji et al., 2021); (2) **boost performance** by enabling multitask learning via joint sentence classification and LF-instance BIO tagging; and (3) capture the asymmetric **semantic and syntactic dependency** between the base and the collocate by the use of a modified attention mechanism.

The G2C-Tr is implemented as a suite of BERT-based models for joint sentence classification and sequence tagging. The syntactic dependency graph of the sentence is input to a G2C-Tr model through its attention mechanism. Figure 1 illustrates the framework of our model. Given the input sen-

tence $W = (\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N)$, we first use a pre-trained dependency parser $\mathrm{DP}()$ to build the dependency graph G, and Part-of-Speech (PoS) tags $P = (\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_N)$. Due to the fact that each LF is characterized by the PoS of its lexical items and the syntactic dependency between them, this information is of significant importance. Then, G2C-Tr predicts the tagged sequence $Y = (\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N)$ as follows:

$$\begin{cases} P, G = \mathrm{DP}(W) \\ H = \mathrm{Enc}(W, P, G) \\ Y = \mathrm{Dec}(H) \end{cases} \quad (1)$$

where $\mathrm{Enc}()$, $\mathrm{Dec}()$ are the encoder and decoder parts of our model, described below. $H = [\mathbf{h}_1, \ldots, \mathbf{h}_T]$ is the contextualised vector representation, and T is the length of the tokenized sequence. The parameters of $\mathrm{DP}()$ are frozen for training.

4.1 Encoder

To compute the contextualised vector embeddings H, we use a modified version of the Graph-to-Graph Transformer model proposed by Mohammadshahi and Henderson (2021a) to encode both PoS tags (P) and the dependency graph (G). Let us first introduce the encoding mechanism.

4.1.1 Input Embeddings

Given an input sentence (W) with its associated PoS tags (P), the G2C-Tr model first computes the input embeddings ($X = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$). To make it compatible with BERT (Devlin et al., 2019), we append two special tokens, `CLS`, and `SEP` to the start and end of the tokenized sequence, respectively. The input embeddings are calculated as the summation of pre-trained token embeddings of BERT, position embeddings, and PoS tag embeddings (as shown in the green part of Figure 1).

4.1.2 Self-attention Mechanism

Given the input embeddings (X), and a dependency graph (G), we compute the contextualised vector representations (H) using a modified version of the Transformer architecture. The original Transformer model (Vaswani et al., 2017) is composed of several Transformer layers. Each Transformer layer includes a self-attention module and a position-wise feed-forward network. Previous work (Ying et al., 2021; Mohammadshahi and Henderson, 2020, 2021a,b) modified the attention mechanism by adding scalar biases to the attention scores (Ying et al., 2021), or multiplying the query representation with relation vectors (Mohammadshahi and Henderson, 2021a, 2020) to encode graph structures.

Algorithm 1: Build Relation Matrix R

Input: Graph $G = \{(i, j, l)\}, j = 1, .., T$
`/* i,j,l are parent node id, dependent id and label */`
`/* CLS is the root node */`
Output: Relation Matrix R

```
R = zeros(T, T)
for (i, j, l) ∈ G do
    r_{i,j} = k_l
    r_{j,i} = k_l + |G|
/* k_l is the index of label l */
```

Since in collocations, base and collocate are syntactically related and LFs are characterized by specific dependency relations, we modify the attention mechanism of the base transformer model to inject syntactic information. In each Transformer layer, given $Z_n = (\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_T)$ as the output representations of the previous layer, the attention weights are calculated as a Softmax over the attention scores α_{ij}, defined as:

$$\alpha_{ij} = \frac{1}{\sqrt{3d}} \Big[\mathbf{z}_i \boldsymbol{W}^{\boldsymbol{Q}} (\mathbf{z}_j \boldsymbol{W}^{\boldsymbol{K}})^T + \mathbf{z}_i \boldsymbol{W}^{\boldsymbol{Q}} (\mathbf{r}_{ij} \boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{A}})^T + \mathbf{r}_{ij} \boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{A}} (\mathbf{z}_j \boldsymbol{W}^{\boldsymbol{K}})^T \Big] \quad (2)$$

where $\boldsymbol{W}^{\boldsymbol{Q}}, \boldsymbol{W}^{\boldsymbol{K}} \in \mathbb{R}^{d_h \times d}$ are learned query and key parameters. $\boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{A}} \in \mathbb{R}^{2|G|+1 \times d}$ is the graph relation embedding matrix, learned during training, d_h is the dimension of hidden vectors, d is the head dimension of self-attention module, and $|G|$ is the overal number of dependency labels. $\mathbf{r}_{ij}$ is the one-hot vector representing both the relation and direction of syntactic relation between token $\mathbf{x}_i$ and $\mathbf{x}_j$, so $\mathbf{r}_{ij} \boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{A}}$ selects the embedding vector for the appropriate syntactic relation. Algorithm 1 shows the procedure of building relation matrix R. Finally, we also add the graph information to the value computation of the Transformer as:

$$\mathbf{v}_i = \sum_j \frac{\exp(\alpha_{ij})}{\sum_j \exp(\alpha_{ij})} (\mathbf{z}_j \boldsymbol{W}^{\boldsymbol{V}} + \mathbf{r}_{ij} \boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{V}}) \quad (3)$$

where $\frac{\exp(\alpha_{ij})}{\sum_j \exp(\alpha_{ij})}$ is the Softmax for the attention weights, $\boldsymbol{W}^{\boldsymbol{V}} \in \mathbb{R}^{d_h \times d}$ is the learned value matrix, $\boldsymbol{W}^{\boldsymbol{R}}_{\boldsymbol{V}} \in \mathbb{R}^{2|G|+1 \times d}$ is the graph embedding

parameter, and v_i is the output representation of the self-attention mechanism for the token i. To find the output representations (H), we use the same mechanism for position-wise feed-forward layer, and layer normalisation as proposed in Vaswani et al. (2017).
Intuitively, additional terms in Equation 2 (second and third multiplications), and Equation 3 (second addition) add a soft bias toward the syntactic information. The model can still decide to use the injected syntactic information, or just rely on the context information (first terms in both Equation 2 and 3).

4.2 Decoder

BERT-based joint sentence classification and sequence tagging has already been used, e.g., for natural language understanding in the context of question answering and goal-oriented dialogue systems, where it serves for *speaker intent* identification and *semantic frame slot filling* (Chen et al., 2019; Castellucci et al., 2019). In the context of sentence classification, we can specify such a model as:

$$y^i = \text{softmax}\left(\mathbf{W}^i\mathbf{h}_1 + \mathbf{b}^i\right), \quad (4)$$

with i as the index of the sentence that is to be classified, and $\mathbf{h}_1$ as the hidden state of the first pooled special token (`CLS` in the case of BERT). For sequence tagging, this equation is extended such that the sequence $[\mathbf{h}_2, \ldots, \mathbf{h}_T]$ is fed to word-level softmax layers:

$$y^s_n = \text{softmax}\left(\mathbf{W}^i\mathbf{h}_n + \mathbf{b}_n\right), n \in 1 \ldots |W| \quad (5)$$

where $\mathbf{h}_n$ is the hidden state corresponding to w_n. Finally, the joint model combines both architectures and is trained, end-to-end, by minimizing the cross-entropy loss for both tasks.

$$p\left(y^i, y^s|W\right) = p\left(y^i|H\right)\prod_{n=1}^{N} p\left(y^s_n|H\right) \quad (6)$$

5 Experimental setup

5.1 Dataset Construction

We carry out experiments on English, French, and Spanish datasets constructed from manually compiled instances of LFs. For English and French, we start from Fisas et al. (2020). For English, Fisas et al.'s list is enriched by 500 instances of low-resourced LFs in order to obtain a more balanced distribution of samples across different LFs; for French, we work with their original list. To obtain the LF instances for Spanish, we use the English list: for each English LF instance, we retrieve from the web via the multilingual search index *Reverso-Context*[4] its translation equivalents, which are then examined and filtered manually.

In all three lists, the bases and collocates are annotated with PoS and lemmas. As corpora, we use the 2019 Wikipedia dumps. First, we preprocess (removing metadata and markups) and parse the dumps with the UDPipe2.5 parsers.[5] Then, we extract from the parsed dumps sentences that contain LF instances from any of our collocation lists, observing the PoS of the base and collocate and the dependency relation between them. To further filter the remaining erroneous samples in which the base and the collocate items do not form a collocation, an additional manual check is performed.

The validated sentences and the collocations they contain are labeled. As sentence label, the sentence's most frequent LF or the first one in case of a draw is chosen. In practice, this most often means that the label of the only LF instance in the sentence is chosen. For instance, in the case of CausFunc0, in the French dataset, only in 1.63% of the cases its instances appear together with instances of other LFs in a sentence, in the Spanish dataset these are 1.85% and in the English dataset 3.42%. However, it should be noted that this varies from LF to LF and for some of the LFs our labeling strategy might be an oversimplification. The highest percentage of "cohabitation" with instances of other LFs can be observed for Oper1: in the French dataset in 7.19% of the cases, in the Spanish dataset in 14.32% and in the English dataset in 25.61%. A more detailed study is necessary to identify potential correlations between different LFs.[6]

To annotate collocations, we use the BI labels of the BIO sequence annotation schema ('B-<LF>$_b$' and 'I-<LF>$_b$' for the base, 'B-<LF>$_c$', 'I-<LF>$_c$' for the collocate, and 'O' for other tokens) (Figure 1). The BIO annotation facilitates a convenient labeling of multi-word elements, and the separate annotation of the base and collocate

[4] https://context.reverso.net/
[5] https://ufal.mff.cuni.cz/udpipe
[6] We would like to thank an anonymous reviewer for pointing out the relevance of the correlation between LFs.

allows for flawless annotation of cases where they are not adjacent.

For the experiments, the annotated datasets are split into training, development, and test subsets in proportion 80–10–10 in terms of LF-wise unique instances, such that all occurrences of a specific instance, i.e., a specific lexical collocation, appear only in one of the subsets. Sentences with several collocations that belong to different splits are dropped. The distribution of samples per LF and language is shown in Figure 2.

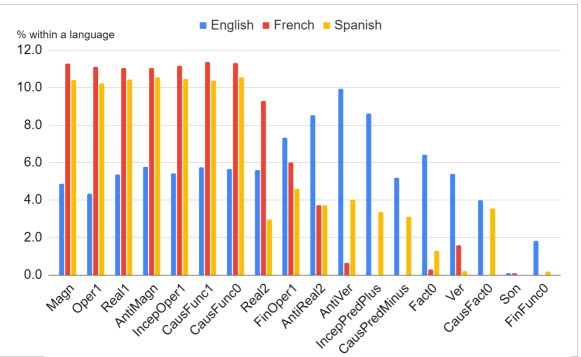


Figure 2: Distribution of examples across lexical functions within a language.

5.2 Experiments

In our experiments, we compare the following architectures:[7]

- Baseline BERT (or similar)-based models (denoted as – in the results tables), specifically BERT-base and large (Devlin et al., 2019), RoBERTa-base and large (Liu et al., 2019); CamemBERT (Martin et al., 2019) and RoBERTa-BNE (Gutiérrez-Fandiño et al., 2021) as monolingual French and Spanish models; and XLM-R for cross-lingual experiments (Conneau et al., 2019).

- Enhanced architectures with the G2C architecture, but without access to the PoS embeddings (G2C (wo) PoS).

- The full model, as depicted in Figure 1, which we refer to as 'G2C'.

In terms of hyperparameter tuning, we fine-tune learning rate and warmup independently for the baseline, G2C(wo)PoS and G2C English models, and fix these values for both French and Spanish. We also use early stopping on the validation set for selecting the best performing models in each configuration.

[7] In all cases, we report only results for the joint architecture, as initial experiments showed a consistent improvement with respect to a sequence tagging-only setup.

6 Results

In what follows, we first present the outcome of the sentence classification and collocation extraction and categorization experiments for the three datasets and then analyze the performance with respect to the individual LFs.

6.1 Sentence classification and collocation extraction results

Tables 2–4 show the performance of various joint models in their original form (marked by '–'), as well as of their G2C(wo)PoS and G2C enhanced variants. We display results on the development ('Dev*') and test sets ('Test*') for the tasks of both sentence classification ('*SentClf') and collocation extraction ('*CollExt'). Sentence classification results are reported in terms of accuracy (there are 18 distinct LF labels), whereas for the collocation extraction task, we report macro F1 over correctly predicted spans. For all experiments, we report average score and standard deviation after three independent runs.

		DevSentClf	DevCollExt	TestSentClf	TestCollExt
$BERT_b$	–	66.86+-5.08	63.21+-1.41	66.04+-1.13	62.95+-3.51
	G2C(wo)PoS	61.72+-2.92	59.90+-1.50	65.18+-1.61	63.61+-1.25
	G2C	64.23+-1.34	62.48+-0.94	**67.25+-0.82**	64.44+-1.12
$BERT_l$	–	66.79+-1.89	65.69+-1.66	63.05+-1.23	61.61+-1.15
	G2C(wo)PoS	67.58+-1.19	66.13+-1.48	66.24+-3.30	64.38+-3.36
	G2C	**70.30+-1.89**	**68.82+-0.86**	64.57+-3.60	62.70+-3.74
$RoBERTa_b$	–	58.09+-0.49	55.93+-1.52	60.96+-1.72	59.20+-3.31
	G2C(wo)PoS	59.89+-1.06	58.05+-0.40	62.51+-0.37	62.17+-0.74
	G2C	59.76+-0.78	58.00+-0.35	62.17+-0.67	61.90+-0.97
$RoBERTa_l$	–	67.47+-2.77	66.97+-1.14	65.55+-0.83	64.79+-3.12
	G2C(wo)PoS	67.40+-3.49	67.97+-4.77	65.95+-2.44	64.84+-1.29
	G2C	61.71+-2.57	59.85+-2.95	65.10+-3.24	**64.98+-2.85**

Table 2: Main results for the English dataset, comparing BERT and RoBERTa, in their base ($_b$) and large ($_l$) variants, and in vanilla (–) and G2C versions.

The results let us conclude, firstly, that the proposed model is considerably more competitive for the task of the compilation of LF-classified collocation resources than competitive baselines. Secondly, incorporating the G2C architecture contributes to an improvement in performance across the board, for all three languages and for most of the models. Thus, for English we see that BERT base sees an improvement of 1 and 2 points in the

sentence classification and sequence labeling results on both the development and test sets, with the improvement on BERT large and RoBERTa base being even more pronounced. RoBERTa large seems to be the model that benefits least from G2C architectures in relative terms, although comparatively, this model is the best performing one on the collocation extraction task on the test set.

With respect to the experiments on French, we can observe that the French camemBERT model does not profit from an enhancement with G2C(wo)PoS; just on the contrary, for the collocation extraction task, performance drops significantly when expanded with either of the G2C variants. This is not the case for XLM-R with its different training variants; its performance is largely maintained in collocation extraction with G2C regimes. The best performance is achieved when XLM-R is enhanced with G2C and trained on both French and English. This also true for the sentence classification task. It is interesting to observe that when trained on English, XLM shows on the development set a higher performance than its extensions for both tasks.

		DevSentClf	DevCollExt	TestSentClf	TestCollExt
camembert Tr: FR	–	66.69+-2.37	62.18+-3.32	54.52+-3.10	51.96+-2.78
	G2C(wo)PoS	64.38+-1.79	38.99+-2.45	50.43+-3.09	30.63+-3.50
	G2C	63.60+-1.33	39.36+-6.38	50.16+-0.46	30.62+-5.24
XLM-r Tr: FR	–	62.22+-2.40	59.30+-5.04	56.38+-3.47	55.23+-3.33
	G2C(wo)PoS	67.08+-4.07	64.32+-6.20	58.41+-3.51	56.97+-2.24
	G2C	64.63+-5.93	61.05+-5.57	56.99+-1.54	55.92+-1.78
XLM-r Tr: EN	–	**67.18+-1.99**	**64.54+-5.65**	54.60+-0.69	52.84+-0.04
	G2C(wo)PoS	65.86+-1.83	64.42+-6.84	54.23+-3.12	50.96+-1.05
	G2C	65.46+-1.49	64.09+-1.03	55.20+-3.62	52.43+-3.77
XLM-r Tr: FR+EN	–	63.07+-2.46	61.59+-1.88	63.35+-2.15	61.32+-1.27
	G2C(wo)PoS	64.40+-0.34	63.88+-1.27	64.95+-0.85	63.55+-0.84
	G2C	62.02+-1.53	61.03+-3.72	**66.48+-1.55**	**64.96+-2.02**

Table 3: Main results for French, comparing the monolingual model CamemBERT with XLM-R variants trained on different slices of the dataset, and G2C(wo)PoS-based extensions.

For Spanish, the performance of the monolingual RoBERTa is in clear contrast to its performance on English. Although it somewhat profits from the G2C enhancement, it seems to underperform compared to XLM-R (which is not the case for English). The reason might be the corpus on which it has been pre-trained (the National Library of Spain corpus) or under-tuning of the set of hyperparameters, which we optimized on the English dataset. We also experiment with XLM-R, trained also only on the Spanish monolingual data (Tr: ES), as well as on the English training set (Tr: EN), and both combined (Tr: ES+EN). Surprisingly enough, XLM-R (stand-alone and G2C+POS-enhanced) performs somewhat better on the test set for both sentence classification and LF-classification when trained on English than when trained on Spanish. In general, the increase in performance provided by the multilingual setting becomes apparent[8], with the G2C model yielding the best results in 3 out of 4 metrics. The best test results of a non-G2C-enhanced model on the collocation extraction task are almost 10 points below the G2Cs models. Moreover, combining both EN and ES training sets into a multilingual language model results in an increase of 6% F1 score. Finally, the differences in the performance of sentence classification and collocation extraction for all three datasets suggest that the predicted sentence label does not always match the label predicted by the BIO-tagger. However, since our primary intention was to use the sentence classifier as an auxiliary task that boosts the performance of the BIO-tagger in a multitask learning setup, we did not analyze the behavior of the sentence classifier and these mismatches in detail.

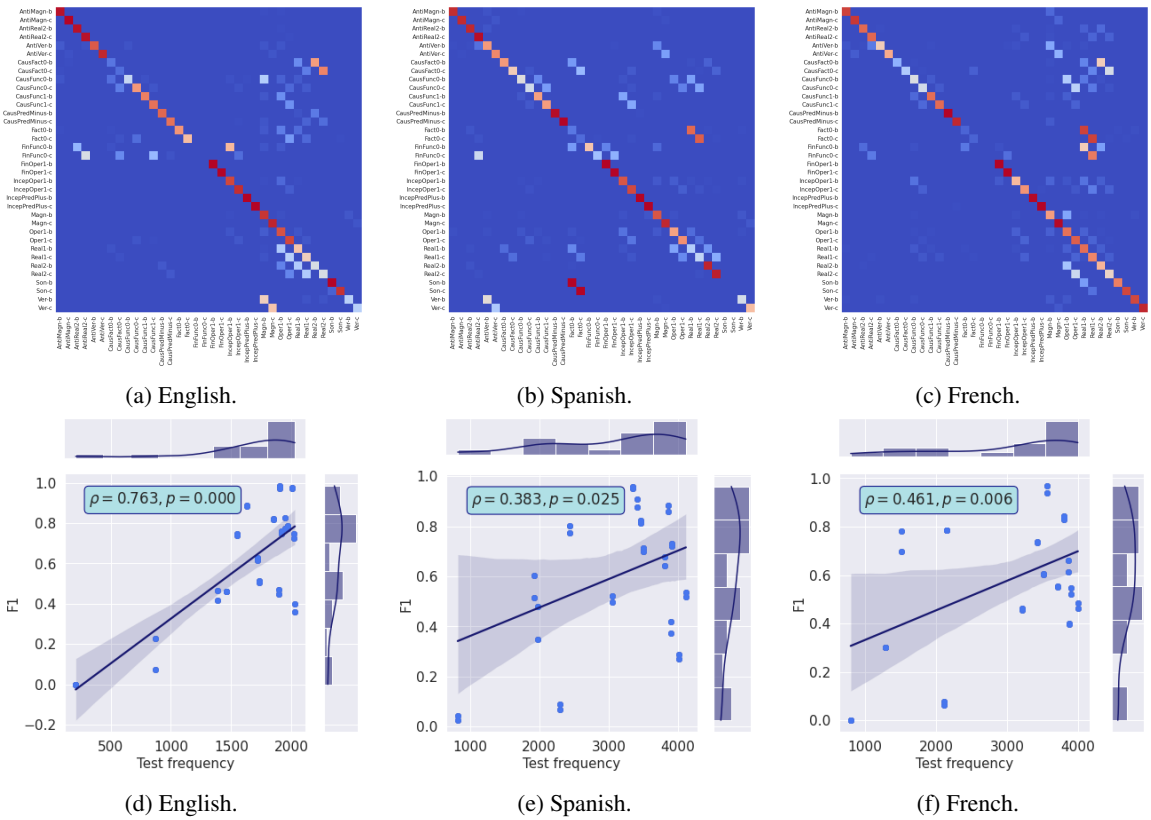
		DevSentClf	DevCollExt	TestSentClf	TestCollExt
RoBERTa$_{es}$ Tr: ES	–	34.42+-0.65	26.65+-1.20	37.90+-0.67	27.94+-0.16
	G2C(wo)PoS	35.62+-1.90	28.42+-2.20	38.60+-1.33	29.73+-2.05
	G2C	37.60+-3.14	31.20+-1.63	40.49+-0.84	31.20+-5.47
XLM-r Tr: ES	–	66.44+-1.02	62.77+-0.01	52.99+-0.29	51.57+-0.12
	G2C(wo)PoS	68.69+-1.96	66.08+-1.95	54.96+-0.35	53.74+-0.42
	G2C	63.96+-5.06	65.32+-2.20	56.42+-0.84	55.07+-0.71
XLM-r Tr: EN	–	65.02+-1.61	63.16+-1.93	60.56+-0.52	56.95+-2.48
	G2C(wo)PoS	63.00+-0.72	62.21+-0.67	58.82+-1.41	57.90+-0.62
	G2C	62.54+-0.45	61.37+-0.48	57.65+-1.81	54.50+-1.57
XLM-r Tr: ES+EN	–	65.91+-0.13	62.73+-0.59	64.26+-1.97	63.37+-0.72
	G2C(wo)PoS	74.18+-1.01	71.20+-0.88	75.42+-0.02	**72.89+-0.07**
	G2C	**74.52+-0.18**	**71.64+-0.01**	**75.55+-0.18**	72.18+-0.92

Table 4: Main results for Spanish, comparing the monolingual model RoBERTa-bne with XLM-R variants trained on different slices of the dataset, and G2C(wo)PoS-based extensions.

6.2 Lexical Function analysis

To obtain a more detailed picture, we report in Table 5 the results of a run for the best performing models for each language and LF, for both of its collocation elements, the base (_b) and the collocate (_c). While there is certain consistency across LFs and languages, there are also notable cases of discrepancies. For instance, we see that Real2 (as, e.g., *enjoy support*), Ver (as, e.g., *legitimate*

[8]We leave for future work an analysis of whether these results can be fully attributed to multilingual transfer, to having access to more training data, or to a combination of the two.



(a) English. (b) Spanish. (c) French.

(d) English. (e) Spanish. (f) French.

Figure 3: LF analysis visualization. Top row shows confusion matrices for the three languages under study, for all LFs and their corresponding base and collocate label. Bottom row shows scatter plot where we show frequency in the x axis, and F1 score in the y axis, again, for each LF.

demand) and Magn (as, e.g., *heavy smoker*) have been better captured in Spanish than in English and French. This can probably be explained by the number of unique instances of the LFs in our training / test data. For instance, in the case of Magn, the ratio between the total number of instances and the number of the unique number of instances in the English test set is 16.8, while in the Spanish test set it is 31.8. In other words, our Spanish dataset contains less variety to express the meaning of intensification than English and French, and is thus easier to capture. Conversely, the performance on Fact0 (as, e.g., *an avalanche strike(s)*) is much better for English, which is likely due to the limitations of the training dataset: out of the 2,112 occurrences of Fact0 instances in total, [*el*] *avión vuela* 'the airplane flies' is counted 602 times.

Note the overall high figures of the recognition of the Magn and AntiMagn instances, and thus a clear distinction between these antonymic LFs, which is a well-known challenge (Rodríguez Fernández et al., 2016b; Wanner et al., 2017). In the case of AntiVer (as, e.g., *illegitimate demand*), the figures are lower in the case of Spanish, which may again hint at the limitations of the Spanish dataset. For the prediction of the individual collocation items, in general, similar results are obtained for the base and collocate. However, some interesting outliers emerge. For instance, for the Spanish CausFact0 (as, e.g., *start an engine*), the performance for the base elements (in our example, *engine*) is more than twice as high as for the collocate elements (in our example, *start*). We hypothesize that this is because most of the CausFact0 base elements in the Spanish dataset denote artefacts and the model learns to recognize them well. Finally, note that only the Spanish model is able to correctly identify a few FinFunc0 collocations (as, e.g., *fire going out*), possibly due to the fact that Spanish contains less multiword expressions and certainly less phrasal verbs associated with this LF.

To understand whether there are obvious sources of confusion across LFs, and whether we can attribute performance to frequency in the datasets, we plot in Figure 3 confusion matrices, as well as the relationship between results and frequency. In English and French, Oper1 and Real1 are great sources of confusion for Real2, especially when it

	EN			ES			FR		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
AntiMagn_b	90.99	93.15	92.06	85.92	89.46	87.65	86.55	81.78	84.10
AntiMagn_c	90.16	94.39	92.23	82.11	91.72	86.65	85.60	83.55	84.56
AntiReal2_b	77.13	83.19	80.05	66.47	86.39	75.14	83.69	65.71	73.62
AntiReal2_c	83.83	93.19	88.26	70.81	92.10	80.07	79.57	68.40	73.62
AntiVer_b	96.05	83.81	89.51	78.53	46.53	58.44	89.57	45.78	60.59
AntiVer_c	93.52	88.88	91.14	78.81	44.95	57.25	86.90	46.12	60.26
CausFact0_b	25.81	08.26	12.51	62.79	16.39	25.99	66.93	19.47	30.17
CausFact0_c	18.33	06.31	09.39	28.36	7.79	12.22	67.20	19.55	30.29
CausFunc0_b	76.94	30.66	43.85	66.27	38.24	48.49	50.02	32.86	39.66
CausFunc0_c	72.05	34.67	46.81	71.04	42.84	53.44	52.19	32.27	39.88
CausFunc1_b	91.15	75.79	82.76	78.37	70.94	72.05	89.00	79.40	83.93
CausFunc1_c	89.40	77.52	83.04	78.37	71.84	74.96	87.63	78.48	82.80
CausPredMinus_b	88.44	68.09	76.94	82.31	91.81	86.80	78.34	62.86	69.75
CausPredMinus_c	86.97	69.70	77.38	82.57	95.26	88.46	86.97	71.05	78.21
Fact0_b	80.10	45.82	58.30	10.28	6.65	8.07	19.40	3.64	6.13
Fact0_c	73.89	49.14	59.02	10.59	7.26	8.61	26.78	4.63	7.90
FinFunc0_b	0.00	0.00	0.00	10.28	6.65	8.07	0.00	0.00	0.00
FinFunc0_c	0.00	0.00	0.00	36.69	12.36	18.50	0.00	0.00	0.00
FinOper1_b	98.44	99.53	98.98	93.83	99.16	96.42	92.20	95.96	94.04
FinOper1_c	97.44	99.69	98.55	64.52	99.46	96.93	92.20	95.96	94.04
IncepOper1_b	78.54	74.91	76.68	60.40	62.15	61.26	96.30	97.25	96.77
IncepOper1_c	82.10	85.59	83.81	58.47	66.09	62.04	71.41	53.95	61.46
IncepPredPlus_b	95.53	99.10	97.28	87.12	90.50	88.78	71.41	53.95	61.46
IncepPredPlus_c	93.75	98.85	96.24	88.21	92.87	90.48	95.42	90.34	92.81
Magn_b	40.35	85.01	54.72	58.21	82.08	68.05	49.24	63.03	55.27
Magn_c	36.94	97.22	51.90	64.44	83.91	70.94	48.63	63.92	55.23
Oper1_b	38.11	79.47	51.90	41.61	59.48	48.97	34.81	68.95	46.26
Oper1_c	37.11	82.24	51.14	39.06	72.75	50.83	32.85	74.13	45.52
Real1_b	41.22	46.48	43.69	29.13	25.30	27.08	37.55	60.57	46.36
Real1_c	37.11	82.24	51.14	29.16	30.07	29.61	39.02	63.45	48.32
Real2_b	50.82	42.43	46.25	59.61	95.56	73.42	54.64	54.53	54.59
Real2_c	50.66	42.53	46.24	59.86	94.65	73.34	55.67	48.91	52.07
Ver_b	80.97	31.99	45.86	84.16	85.30	84.73	89.17	70.31	78.62
Ver_c	78.52	32.74	46.21	84.16	85.30	84.73	88.72	70.17	78.36

Table 5: Results breakdown per language and per LF, where, for each LF, we list individual results for base and collocate categorization.

comes to categorizing Real2 collocates. However, this is not the case for Spanish. In this context, we need to keep in mind that Real1 and Real2 differ only with respect to their subcategorization pattern (in Real1, it is A0/A1, which is realized grammatical subject, and in Real2, it is A2) and that the semantic difference betweeen Oper and Real is rather fine. Still, for Spanish this difference is captured, while for English and French it is not. This is similar for the distinction between CausFact$_i$ / Oper$_i$ and Real$_i$. Why the confusions are minor for Spanish requires a deeper analysis. We can also see that Magn and Oper bases are often confused in French, but not in English and Spanish. This might be due to parsing and PoS tagging errors. Finally, in the lower part of Figure 3, we see that for English, there is a clear correlation between results and LF frequencies (ρ=0.76), followed by French (ρ=0.46) and, finally, Spanish (ρ=0.38), where we also find highest dispersion across all F1 bins.

7 Conclusions and Future Work

We have proposed an architecture for joint collocation extraction and lexical function typification by explicitly encoding syntactic dependencies in the attention mechanism. Our experiments show that our proposed architecture drastically improves over its language model-only counterparts, and that joint multilingual training is a promising direction for less resourced languages. For the future, we would like to extend these experiments to other languages and explore zero or few-shot prompt-based methods.

Acknowledgements

Many thanks to Beatriz Fisas, Alba Táboas, and Inmaculada López for their help with the datasets. We would also like to thank the anonymous reviewers for their very helpful comments. The work by Alexander Shvets and Leo Wanner has been supported by the European Commission in the context of the Horizon 2020 Research Program under the grant numbers 825079 and 870930. Alireza Mohammadshahi is supported by the Swiss National Science Foundation (grant number CRSII5-180320).

References

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Roberto Carlini, Joan Codina-Filba, and Leo Wanner. 2014. Improving collocation correction by ranking suggestions using linguistic knowledge. In *Proceedings of the third workshop on NLP for computer-assisted language learning*, pages 1–12.

Giuseppe Castellucci, Valentina Bellomaria, Andrea Favalli, and Raniero Romagnoli. 2019. Multi-lingual intent detection and slot filling in a joint bert-based model. *arXiv preprint arXiv:1907.02884*.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Wei-Te Chen, Claire Bonial, and Martha Palmer. 2016. English light verb construction identification using lexical knowledge. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2375–2381.

Kenneth W. Church and Patrick Hanks. 1989. Word Association Norms, Mutual Information, and Lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 76–83, Vancouver, Canada.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Silvio Ricardo Cordeiro and Marie Candito. 2019. Syntax-based identification of light-verb constructions. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 97–104, Turku, Finland. Linköping University Electronic Press.

Anthony P. Cowie. 1994. Phraseology. In R.E. Asher and J.M.Y. Simpson, editors, *The Encyclopedia of Language and Linguistics, Vol. 6*, pages 3168–3171. Pergamon, Oxford.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Andrea Di Fabio, Simone Conia, and Roberto Navigli. 2019. Verbatlas: a novel large-scale verbal semantic resource and its application to semantic role labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 627–637.

A. Dinu, L.P. Dinu, and I.T. Sorodoc. 2014. Aggregation methods for efficient collocation detection. In *Proceedings of LREC*, pages 4041–4045.

Mark Dras. 1995. Automatic identification of support verbs: A step towards a definition of semantic weight. In *Proceedings of the Eighth Australian Joint Conference on Artificial Intelligence*, pages 451–458.

Luis Espinosa-Anke, Jose Camacho-Collados, Sara Rodríguez Fernández, Horacio Saggion, and Leo Wanner. 2016. Extending wordnet with fine-grained collocational information via supervised distributional learning. In *Proceedings of COLING 2016: Technical Papers. The 26th International Conference on Computational Linguistics; 2016 Dec. 11-16; Osaka (Japan): COLING; 2016. p. 900-10.* COLING.

Luis Espinosa-Anke, Joan Codina-Filbá, and Leo Wanner. 2021. Evaluating language models for the retrieval and categorization of lexical collocations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1406–1417.

Luis Espinosa Anke, Steven Schockaert, and Leo Wanner. 2019. Collocation classification with unsupervised relation vectors. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5765–5772, Florence, Italy. Association for Computational Linguistics.

Martha W. Evens. 1988. *Relational Models of the Lexicon: Representing Knowledge in Semantic Networks*. Cambridge University Press, Cambridge, UK.

Stefan Evert. 2007. Corpora and Collocations. In A. Lüdeling and M. Kytö, editors, *Corpus Linguistics. An International Handbook*. Mouton de Gruyter, Berlin.

Stefan Evert and Brigitte Krenn. 2001. Methods for the qualitative evaluation of lexical association measures. In *Proceedings of the 39th annual meeting of the association for computational linguistics*, pages 188–195.

Gabriela Ferraro, Rogelio Nazar, Margarita Alonso Ramos, and Leo Wanner. 2014. Towards advanced collocation error correction in spanish learner corpora. *Language resources and evaluation*, 48(1):45–64.

Gabriela Ferraro, Rogelio Nazar, and Leo Wanner. 2011. Collocations: A challenge in computer assisted language learning.

John R. Firth. 1957. Modes of Meaning. In J.R. Firth, editor, *Papers in Linguistics, 1934-1951*, pages 190–215. Oxford University Press, Oxford.

Beatriz Fisas, Luis Espinosa Anke, Joan Codina-Filbá, and Leo Wanner. 2020. CollFrEn: Rich bilingual English–French collocation resource. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 1–12, online. Association for Computational Linguistics.

Marcos Garcia, Marcos García Salido, and Margarita Alonso Ramos. 2017. Using bilingual word-embeddings for multilingual collocation extraction. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 21–30.

Marcos Garcia, Marcos García Salido, and Margarita Alonso Ramos. 2019. A comparison of statistical association measures for identifying dependency-based collocations in various languages. In *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, pages 49–59.

Marcos Garcia, Tiago Kramer Vieira, Carolina Scarton, Marco Idiart, and Aline Villavicencio. 2021. Probing

for idiomaticity in vector space models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3551–3564.

Alexander Gelbukh and Olga Kolesnikova. 2012. *Semantic analysis of verbal collocations with lexical functions*, volume 414. Springer.

Stefan Th Gries. 2013. 50-something years of work on collocations: What is or should be next.... *International Journal of Corpus Linguistics*, 18(1):137–166.

Asier Gutiérrez-Fandiño, Jordi Armengol-Estapé, Marc Pàmies, Joan Llop-Palao, Joaquín Silveira-Ocampo, Casimiro Pio Carrino, Aitor Gonzalez-Agirre, Carme Armentano-Oller, Carlos Rodriguez-Penagos, and Marta Villegas. 2021. Spanish language models. *arXiv preprint arXiv:2107.07253*.

Franz Josef Hausmann. 1985. Kollokationen im Deutschen Woerterbuch: ein Beitrag zur Theorie des lexicographischen Biespiels. *Lexikographie und Grammatik*.

Ulrich Heid and Sybille Raab. 1989. Collocations in multilingual generation. In *Fourth Conference of the European Chapter of the Association for Computational Linguistics*.

Chung-Chi Huang, Kate H. Kao, Chiung-Hui Tseng, and Jason S. Chang. 2009. A thesaurus-based semantic classification of english collocations. *Computational Linguistics and Chinese Language Processing*, 14(3):257–280.

Bin Ji, Shasha Li, Jie Yu, Jun Ma, and Huijun Liu. 2021. Boosting span-based joint entity and relation extraction via squence tagging mechanism. *https://arxiv.org/abs/2105.10080*.

Václava Kettnerová, Markéta Lopatková, Eduard Bejček, Anna Vernerová, and Marie Podobová. 2013. Corpus based identification of czech light verbs. In *Proceedings of the Seventh International Conference Slovko, Natural Language Processing, Corpus Linguistics, E-Learning*, pages 118–128, Lüdenscheid, Germany. RAM Verlag.

Adam Kilgarriff. 2006. Collocationality (And How to Measure it). In *Proceedings of the 12th Euralex International Congress on Lexicography (EURALEX)*, pages 997–1004, Turin, Italy. Springer-Verlag.

Brigitte Krenn. 2000. *The Usual Suspects: Data-Oriented Models for the Identification and Representation of Lexical Collocations*, volume 7 of *Saarbrücken Dissertations in Computational Linguistics and Language Technology*. DFKI and Universität des Saarlandes.

Brigitte Krenn and Stefan Evert. 2001. Can we do better than frequency? a case study on extracting pp-verb collocations. In *Proceedings of the ACL Workshop on Collocations*, pages 39–46.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. 2020. Pmi-masking: Principled masking of correlated spans. *arXiv preprint arxiv:2010.01825*.

Dekang Lin. 1999. Automatic identification of non-compositional phrases. In *Proceedings of the 37th Annual of the Association for Computational Linguistics (ACL)*, pages 317–324.

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. 2019. Robust neural machine translation with joint textual and phonetic embedding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3044–3049, Florence, Italy. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de La Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Marco Maru, Federico Scozzafava, Federico Martelli, and Roberto Navigli. 2019. Syntagnet: challenging supervised word sense disambiguation with lexical-semantic combinations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3525–3531.

Igor A. Mel'čuk. 1995. Phrasemes in Language and Phraseology in Linguistics. In M. Everaert, E.-J. van der Linden, A. Schenk, and R. Schreuder, editors, *Idioms: Structural and Psychological Perspectives*, pages 167–232. Lawrence Erlbaum Associates, Hillsdale.

Igor A. Mel'čuk. 1996. Lexical functions: A tool for the description of lexical relations in the lexicon. In Leo Wanner, editor, *Lexical Functions in Lexicography and Natural Language Processing*, pages 37–102. Benjamins Academic Publishers, Amsterdam.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Alireza Mohammadshahi and James Henderson. 2020. Graph-to-graph transformer for transition-based dependency parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3278–3289, Online. Association for Computational Linguistics.

Alireza Mohammadshahi and James Henderson. 2021a. Recursive Non-Autoregressive Graph-to-Graph Transformer for Dependency Parsing with Iterative Refinement. *Transactions of the Association for Computational Linguistics*, 9:120–138.

Alireza Mohammadshahi and James Henderson. 2021b. Syntax-aware graph-to-graph transformer for semantic role labelling.

Darren Pearce et al. 2002. A comparative evaluation of collocation extraction techniques. In *LREC*.

Pavel Pecina. 2008. A Machine Learning Approach to Multiword Expression Extraction. In *Proceedings of the LREC 2008 Workshop Towards a Shared Task for Multiword Expressions)*, pages 54–57, Marrakech, Morocco.

Pavel Pecina. 2010. Lexical association measures and collocation extraction. *Language resources and evaluation*, 44(1):137–158.

Pavel Pecina and Pavel Schlesinger. 2006. Combining association measures for collocation extraction. In *Proceedings of the COLING/ACL 2006 main conference poster sessions*, pages 651–658.

Sara Rodríguez Fernández, Roberto Carlini, Luis Espinosa-Anke, and Leo Wanner. 2016a. Example-based acquisition of fine-grained collocational resources. In *Calzolari N, Choukri K, Declerck T, Goggi S, Grobelnik M, Maegaard B, Mariani J, Mazo H, Moreno A, Odijk J, Piperidis S, editors. LREC 2016, Tenth International Conference on Language Resources and Evaluation; 2016 May 23-28; Portorož (Slovenia).[Sl]: European Language Resources Association (ELRA); 2016. Session P28, Multiword expressions; p. 2317-22*. ELRA (European Language Resources Association).

Sara Rodríguez-Fernández, Roberto Carlini, and Leo Wanner. 2015. Classification of grammatical collocation errors in the writings of learners of spanish. *Procesamiento del Lenguaje Natural*, 55.

Sara Rodríguez Fernández, Luis Espinosa-Anke, Roberto Carlini, and Leo Wanner. 2016b. Semantics-driven recognition of collocations using word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics; 2016 Aug. 7-12; Berlin (Germany).[place unknown]: ACL; 2016. Vol. 2, Short Papers; p. 499-505*. ACL (Association for Computational Linguistics).

Federico Scozzafava, Marco Maru, Fabrizio Brignone, Giovanni Torrisi, and Roberto Navigli. 2020. Personalized pagerank with syntagmatic information for multilingual word sense disambiguation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–46.

Violeta Seretan. 2014. On collocations and their interaction with parsing and translation. *Informatics*, 1(1):11–31.

Violeta Seretan and Eric Wehrli. 2006. Accurate collocation extraction using a multilingual parser. In *Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the Association for Computational Linguistics*, pages 953–960.

Vered Shwartz and Ido Dagan. 2019. Still a pain in the neck: Evaluating text representations on lexical composition. *Transactions of the Association for Computational Linguistics*, 7:403–419.

Frank Smadja. 1993. Retrieving collocations from text: Xtract. *Computational linguistics*, 19(1):143–178.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Veronika Vincze, István Nagy, and János Zsibrita. 2013. Learning to detect English and Hungarian light verb constructions. *ACM Transactions on Speech and Language Processeing*, 10(2):1–25.

Leo Wanner. 2004. Towards automatic fine-grained semantic classification of verb-noun collocations. *Natural Language Engineering*, 10(2):95–143.

Leo Wanner and John A. Bateman. 1990. A collocational based approach to salience sensitive lexical selection. In *Proceedings of the 5th International Workshop on Natural Language Generation*, Dawson, PA.

Leo Wanner, Bernd Bohnet, and Mark Giereth. 2006. Making sense of collocations. *Computer Speech & Language*, 20(4):609–624.

Leo Wanner, Gabriela Ferraro, and Pol Moreno. 2017. Towards distributional semantics-based classification of collocations for collocation dictionaries. *International Journal of Lexicography*, 30(2):167–186.

Leo Wanner, M Alonso Ramos, Orsolya Vincze, Rogelio Nazar, Gabriela Ferraro, Estela Mosqueira, and Sabela Prieto. 2013. Annotation of collocations in a learner corpus for building a learning environment. *Twenty years of learner corpus research. Looking back, moving ahead*, pages 493–503.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do transformers really perform bad for graph representation?