Idiomatic Expression Identification using Semantic Compatibility

Ziheng Zeng and **Suma Bhat**
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Champaign, IL USA
{zzeng13, spbhat2}@illinois.edu

Abstract

Idiomatic expressions are an integral part of natural language and constantly being added to a language. Owing to their non-compositionality and their ability to take on a figurative or literal meaning depending on the sentential context, they have been a classical challenge for NLP systems. To address this challenge, we study the task of detecting whether a sentence has an idiomatic expression and localizing it when it occurs in a figurative sense. Prior research for this task has studied specific classes of idiomatic expressions offering limited views of their generalizability to new idioms. We propose a multi-stage neural architecture with attention flow as a solution. The network effectively fuses contextual and lexical information at different levels using word and sub-word representations. Empirical evaluations on three of the largest benchmark datasets with idiomatic expressions of varied syntactic patterns and degrees of non-compositionality show that our proposed model achieves new state-of-the-art results. A salient feature of the model is its ability to identify idioms unseen during training with gains from 1.4% to 30.8% over competitive baselines on the largest dataset.

1 Introduction

Idiomatic expressions (IEs) are a special class of multi-word expressions (MWEs) that typically occur as collocations and exhibit *semantic* non-compositionality (a.k.a. semantic idiomaticity), where the meaning of the expression is not derivable from its parts (Baldwin and Kim, 2010). In terms of occurrence, IEs are individually rare, but collectively frequent in and constantly added to natural language across different genres (Moon et al., 1998). Additionally, they are known to enhance fluency and used to convey ideas succinctly when used in everyday language (Baldwin and Kim, 2010; Moon et al., 1998).

Classically regarded as a ''pain in the neck'' to idiom-unaware NLP applications (Sag et al., 2002) these phrases are challenging for reasons including their non-compositionality (semantic idiomaticity), besides taking a figurative or literal meaning depending on the context (semantic ambiguity), as shown by the example in Table 1. Borrowing the terminology from Haagsma et al. (2020), we call these phrases *potentially idiomatic expressions (PIEs)* to account for the contextual semantic ambiguity. Indeed, prior work has identified the challenges that PIEs pose to many NLP applications, such as machine translation (Fadaee et al., 2018; Salton et al., 2014), paraphrase generation (Ganitkevitch et al., 2013), and sentiment analysis (Liu et al., 2017; Biddle et al., 2020). Accordingly, making applications idiom-aware, either by identifying them before or during the task, has been found to be effective (Korkontzelos and Manandhar, 2010; Nivre and Nilsson, 2004; Nasr et al., 2015). This study proposes a novel architecture that detects the presence of a PIE. When found, its span in a given sentence is localized and returning the phrase if it is used figuratively (i.e., used as an IE); otherwise an empty string is returned indicating that the phrase is used literally (see Table 1). Such a network can serve as a preprocessing step for broad-coverage downstream NLP applications because we consider the ability to detect IEs to be a first step towards their accurate processing. This is the *idiomatic expression identification* problem, which is the MWE identification problem defined by Baldwin and Kim (2010) limited to MWEs with semantic idiomaticity.

Despite being well-studied in the current literature as idiom type and token classification (e.g., Fazly et al., 2009; Feldman and Peng, 2013; Salton et al., 2016; Taslimipoor et al., 2018; Peng et al., 2014; Liu and Hwa, 2019), previous methods are limited for various reasons. They rely on knowing the PIEs being classified and hence their exact

Transactions of the Association for Computational Linguistics, vol. 9, pp. 1546–1562, 2021. https://doi.org/10.1162/tacl_a_00442
Action Editor: Nitin Madnani. Submission batch: 3/2021; Revision batch: 7/2021; Published 12/2021.

Input	Tom said many bad things about Jane *behind her back*. (Figurative)
	He took one from an armchair and put it *behind her back*. (Literal)
Output	behind her back
	<CLS> <SEP>

Table 1: Example input and output for DISC framework. When a potentially idiomatic expression (PIE; italicized) is used idiomatically, the PIE is identified and extracted as the output; otherwise the model outputs the start (<CLS>) and end (<SEP>) tokens to indicate that it is used literally.

positions, or focus on specific syntactic patterns (e.g., verb-noun compounds or verbal MWEs), thereby calling into question their use in more realistic scenarios with unseen PIEs (a likely event, given the prolific nature of PIEs). Additionally, without a cross-type (type-aware) evaluation, where the PIE types from the train and test splits are segregated (Fothergill and Baldwin, 2012; Taslimipoor et al., 2018, the true generalizability of these methods to unseen idioms cannot be inferred. For instance, a model could be classifying by memorizing known PIEs or their tendencies to occur exclusively as figurative or literal expressions.

In contrast, this study aims to identify IEs in general (i.e., *without* posing constraints on the PIE type) in a more realistic setting where new idioms may occur, by proposing the i**D**entifier of **I**diomatic expressions via **S**emantic **C**ompatibility (DISC) that performs detection and localization jointly. The novelty is that we perform the task without an explicit mention of the identity or the position of the PIE. As a result, the task is more challenging than the previously explored idiom token classification.

An effective solution to this task calls for the ability to relate the meaning of its component words with each other (e.g., Baldwin, 2005; McCarthy et al., 2007) as well as with the context (Liu and Hwa, 2019). This aligns with the widely upheld psycholinguistic findings on human processing of a phrase's figurative meaning-in comparison with its literal interpretation (Bobrow and Bell, 1973). Toward this end, we rely on the contextualized representation of a PIE (accounting both for its internal and contextual properties), hypothesizing that a figurative expression's contextualized representation should be different from that of its literal counterpart. We refer to this as its *semantic compatibility* (SC)—if a PIE is semantically compatible with its context, then it is literal; if not, it is figurative. The idea of SC also captures the distinction between literal word combinations and idioms, in terms of the semantics encoded by both (Jaeger, 1999) and the related property of *selectional preference* (Wilks, 1975)—the tendency for a word to semantically select or constrain which other words may appear in its association (Katz and Fodor, 1963) successfully used for processing metaphors (Shutova et al., 2013) and word sense disambiguation (Stevenson and Wilks, 2001). We capture SC by effectively fusing information from the input tokens' contextualized and literal word representations to then localize the span of PIE used figuratively. Here we leverage the idea of attention flow previously studied in a machine comprehension setting (Seo et al., 2017).

Our main contributions in this work are:

A **novel IE identification model, DISC,** that uses attention flow to fuse lexical semantic information at different levels and discern the SC of a PIE. Taking only a sentence as input and using only word and POS representations, it simultaneously performs detection and localization of the PIEs used figuratively. To the best of our knowledge, this is the first such study on this task.

Realistic evaluation: We include two novel aspects in our evaluation methodology. First, we consider a new and stringent performance measure for subsequence identification; the identification is successful if and only if every word in the exact IE subsequence is identified. Second, we consider type-aware evaluation so as to highlight a model's generalizability to unseen PIEs regardless of syntactic pattern.

Competitive performance: Using benchmark datasets with a variety of PIEs, we show DISC[1] compares favorably with strong baselines on PIEs seen during training. Particularly noteworthy is its identification accuracy on *unseen* PIEs, which is 1.4% to 11.1% higher than the best baseline.

[1]The implementation of DISC is available at https://github.com/zzeng13/DISC.

2 Related Work

We provide a unified view of the diverse terminologies and tasks studied in prior works that define the scope of our study.

MWEs, IEs and Metaphors. We first introduce the relation between the three related concepts, namely, MWE, IE, and metaphor, in order to present a clearer picture of the scope of our work. According to Baldwin and Kim (2010) and Constant et al. (2017), MWEs (e.g., *bus driver* and *good morning*) satisfy the properties of outstanding collocation and contain multiple words. IEs are a special type of MWE that also exhibit non-compositionality at the semantic level. This has generally been considered to be the key distinguishing property between idioms (IEs) and MWEs in general, although the boundary between IEs and non-idiom MWEs is not clearly defined. (Baldwin and Kim, 2010; Fadaee et al., 2018; Liu et al., 2017; Biddle et al., 2020). Metaphors are a form of figurative speech used to make an implicit comparison at an attribute level between two things seemingly unrelated on the surface. By definition, certain MWEs and IEs use metaphorical figuration (e.g., *couch potato* and *behind the scenes*). However, not all metaphors are IEs because metaphors are not required to possess any of the properties of IEs—that is, the components of a metaphor need *not* co-occur frequently (metaphors can be uniquely created by anyone), metaphors can be direct and plain comparisons and thus are not semantically non-compositional, and they need not have multiple words (e.g., *titanium* in the sentence ''I am titanium'').

PIE and MWE Processing. Current literature considers idiom type classification and idiom token classification (Cook et al., 2008; Liu and Hwa, 2019; Liu, 2019) as two idiom-related tasks. Idiom type classification decides if a phrase could be used as an idiom without specifically considering its context. Several works (e.g., Fazly and Stevenson, 2006; Shutova et al., 2010) have studied the distinguishing properties of idioms from other literal phrases, especially that of non-compositionality (Westerståhl, 2002; Tabossi et al., 2008; 2009; Reddy et al., 2011; Cordeiro et al., 2016).

In contrast, idiom token classification (Fazly et al., 2009; Feldman and Peng, 2013; Peng and Feldman, 2016; Salton et al., 2016; Taslimipoor et al., 2018; Peng et al., 2014; Liu and Hwa, 2019) determines whether a given PIE is used literally or figuratively in a sentence. Prior work has used per-idiom classifiers that are completely non-scalable to be practical (Liu and Hwa, 2017), required the position of the PIEs in the sentence (e.g., Liu and Hwa, 2019), and focused only on specific PIE patterns, such as verb-noun compounds (Taslimipoor et al., 2018). Overall, available research for this task only disambiguates a given phrase. In contrast, we do not assume any knowledge of the PIE being detected; given a sentence, we detect whether there is a PIE and disambiguate its use.

PIEs being special types of MWEs, our task is related to *MWE extraction* and *MWE identification* (Baldwin and Kim, 2010). As with idioms, MWE extraction takes a text corpus as input and produces a list of *new* MWEs (e.g., Fazly et al., 2009; Evert and Krenn, 2001; Pearce, 2001; Schone and Jurafsky, 2001). MWE identification takes a text corpus as input and locates *all* occurrences of MWEs in the text at the token level, differentiating between their figurative and literal use (Baldwin, 2005; Katz and Giesbrecht, 2006; Hashimoto et al., 2006; Blunsom, 2007; Sporleder and Li, 2009; Fazly et al., 2009; Savary et al., 2017); the identified MWEs may or may not be known beforehand. Constant et al. (2017) group main MWE-related tasks into *MWE discovery* and *MWE identification*: MWE discovery is identical to MWE extraction, while the MWE identification here, different from Baldwin and Kim's definition, identifies only *known* MWEs. Our task is identical to Baldwin and Kim's (2010) MWE identification and Savary et al.'s (2017) verbal MWE identification while focusing only on PIEs, and we aim to both detect the presence of PIEs and localize IE positions (boundaries), regardless of whether the PIEs were previously seen or not. Besides, like idiom type classification and MWE extraction, our approach also works for identifying new idiomatic expressions.

Approaches to MWE identification fall into two broad types. (1) A tree-based approach by first constructing a syntactic tree of the sentence and then traversing a selective set of candidate subsequences (at a node) to identify idioms (Liu et al., 2017). However, since the construction of a syntactic tree is itself affected by the presence of idioms (Nasr et al., 2015; Green et al., 2013), the nodes may not correspond to an entire idiomatic

expression, which in turn can affect even a perfect classifier's ability to identify idioms precisely. (2) Framing the problem as a sequence labeling problem for token-level idiomatic/literal labeling, similar to prior work (Jang et al., 2015; Mao et al., 2019; Gong et al., 2020; Kumar and Sharma, 2020; Su et al., 2020) on metaphor detection that label each token as a metaphor or a non-metaphor and Schneider and Smith's (2015) approach to MWE identification by tagging tokens from a MWE with the same supersense tag. This tagging approach provides finer control over subsequence extraction and is unrestricted by factors that could impact a tree-based approach, and does not require the traversal of all possible subsequences in search of the candidate phrases. Our approach is similar to this in spirit but focused on PIEs. In particular, Schneider and Smith (2015) aim to tag all MWEs while making no distinction for the non-compositional phrases, whereas our work aims to only identify IEs from sentences containing PIEs.

Semantic Compatibility. Exploiting SC for processing idioms has been considered in rather restricted settings, where the identity of a PIE (and hence its position) is known. For instance, Liu and Hwa (2019) used SC to classify a given phrase in its context as literal/idiomatic. A corpus of annotated phrases was used to train a linear classification layer to discriminate between phrases' contextualized and literal embeddings. Peng and Feldman (2016) directly check the compatibility between the word embeddings of a PIE with the embeddings of its context words to perform the literal/idiomatic classification. Jang et al. (2015) used SC and the global discourse context to detect the figurative use of a small list of candidate metaphor words. Gong et al. (2017) treated the phrase's respective context as vector spaces and modeled the distance of the phrase from the vector space as an index of SC. We extend these prior efforts to identify both the presence and the position of an IE using only a sentence as input without knowing the PIE.

3 Method

In line with studies on MWE identification mentioned above, we frame the identification of idiomatic subsequences as a token-level tagging problem, where we perform literal/idiomatic classification for every token in the sentence. A simple post-processing step finally extracts the PIE subsequence used in the idiomatic sense.

Task Definition. Given an input sentence $S = w_1, w_2, \ldots, w_L$, where w_i for $i \in [1, L]$ are the tokenized units and L is the number of tokens in S, the task is to label the individual token w_i with a label $c_i \in \{\mathsf{idiomatic}, \mathsf{literal}\}$ so that the final output is a sequence of classifications $C = c_1, c_2, \ldots, c_L$. For a correct prediction, the phrase $w_{i:j}$ in S is idiomatic and the corresponding $c_{i:j}$ are classified into the 'idiom' class, while the rest are the 'literal' class; or the phrase $w_{i:j}$ in S is literal and the corresponding $c_{1:L}$ are all classified into the 'literal' class.

Overview of Proposed Approach. The overall workflow and model architecture of DISC are illustrated in Figure 1. The model can be roughly divided into three distinct phases: (1) the embedding phase, (2) the attention phase, and (3) the prediction phase. In the embedding phase, the input sequence S is tokenized and both the contextualized and static word embeddings are generated and supplemented with character-level information. Furthermore, POS tag embeddings of the input tokens are generated to provide syntactic information. In the attention phase, an attention flow layer combines the POS tag embeddings with the static word embeddings, yielding an enhanced literal representation for every word. Then, a second attention flow layer fuses the contextualized and the enriched literal representations by attending to the rich features of each token in the tokenized input sequence. Finally, the prediction phase further encodes the sequence of feature vectors and performs token-level literal/idiomatic classification to produce the predicted sequence C.

Embedding Phase. Here the input sentence S is tokenized in two ways—one for the pre-trained language model and the other for the pre-trained static word embedding layer—resulting in two tokenized sequences T^c and T^s, such that $|T^c| = M$ and $|T^s| = N$. Since the two tokenizers are not necessarily the same, N and M may be unequal.

Next, T^c is fed to a pre-trained language model to produce a sequence of contextualized word embeddings, $E^{con} \in \mathbb{R}^{M \times D_{con}}$, where D_{con} is the embedding vector dimension. A pre-trained word embedding layer takes T^s to produce a sequence of static word embeddings, $E^s \in \mathbb{R}^{N \times D_s}$, where

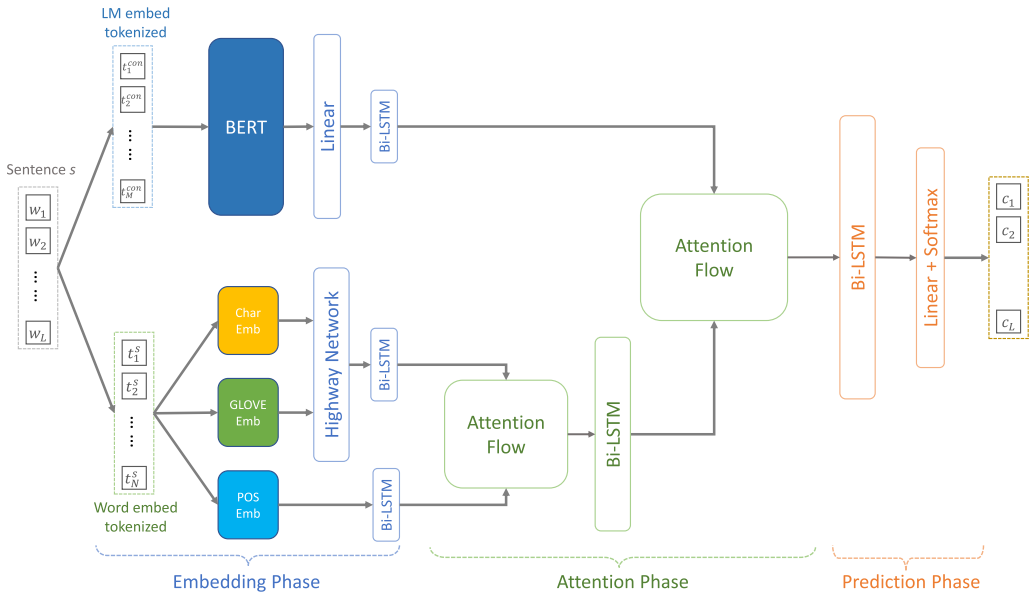

Figure 1: Overview of the DISC framework.

D_s is the embedding vector dimension. The contextualized embeddings capture the semantic content of the phrases within the specific context, while the static word embeddings capture the compositional meaning of the phrases, both of which allow the model to check SC.

Additionally, informed by the finding that character-level information alleviates the problem of morphological variability in idiom detection (Liu et al., 2017), character sequences $C \in \mathbb{R}^{N \times W_t}$ are generated from T^s, and their character-level embeddings, $E^{char} \in \mathbb{R}^{N \times D_{char}}$ obtained using a 1-D Convolutional Neural Network (CNN) followed by a max-pooling layer over the maximum width of the tokens, W_t. Then, E^{char} and E^s are combined via a two-layer highway network (Srivastava et al., 2015) which yields $\hat{E}^s \in \mathbb{R}^{N \times (D_{char}+D_s)}$.

Lastly, to capture shallow syntactic information, a POS embedding layer generates a sequence of POS tags for T^s and a simple linear embedding layer produces a sequence of POS tag embeddings, $E^{pos} \in \mathbb{R}^{N \times D_{pos}}$, where D_{pos} is the POS embedding vector dimension.

In effect, the embedding layer encodes four levels of information: character-level, phrase-internal and implicit context (static word embedding), phrase-external and explicit context (contextual embedding), and shallow syntactic information (POS tag).

To perform an initial feature extraction from the raw embeddings and unify the different embedding vector dimensions, we apply a Bidirectional LSTM (BiLSTM) layer for each embedding sequence resulting in $\tilde{E^{con}} \in \mathbb{R}^{M \times D_{emb}}$, $\tilde{E^s} \in \mathbb{R}^{N \times D_{emb}}$, and $\tilde{E^{pos}} \in \mathbb{R}^{N \times D_{emb}}$, where $D_{embed}/2$ is the hidden dimension of the BiLSTM layers.

Attention Phase. The attention phase mainly consists of two attention flow layers. In its native application (i.e., reading comprehension), the attention flow layer linked and fused information from the context word sequence and the query word sequence (Seo et al., 2017), producing query-aware vector representations of the context words while propagating the word embeddings from the previous layer. Analogously, for our task, the attention flow layer fuses information from the two embedding sequences encoding different kinds of information. More specifically, given two sequences $S^a \in \mathbb{R}^{L \times D}$ and $S^b \in \mathbb{R}^{K \times D}$ of lengths L and K, the attention flow layer computes $H \in \mathbb{R}^{L \times K}$ using, $H_{ij} = W_0^\top \left[S^a_{:i}; S^b_{:j}; S^a_{:i} \circ S^b_{:j}\right]$, where H_{ij} is the attended, merged embedding of the i-th token in S^a and the j-th token in S^b, W_0 is a trainable weight matrix, $S^a_{:i}$ is the i-th column of S^a, $S^b_{:j}$ is the j-th column of S^b, $[;]$ is vector concatenation, and $\circ$ is the Hadamard product. Next, the attentions are computed from both S^a-to-S^b and S^b-to-S^a. The S^a-to-S^b attended representation is computed as $\tilde{S^b}_{:i} = \sum_j a_{ij} S^b_{:j}$, where $a_i = \text{softmax}\,(H_{i:})$; $a_i \in \mathbb{R}^K$ and $\sum a_{ij} = 1$; $\tilde{S^b} \in \mathbb{R}^{2D \times L}$. The S^b-to-S^a attended representation

is computed as $\tilde{S^a}_{:i} = \sum_i b_i S^a_{:i}$, where $b = \text{softmax}(\max_{col}(H))$, $b \in \mathbb{R}^L$, and $\tilde{S^a} \in \mathbb{R}^{2D \times L}$. Finally, the attention flow layer outputs a combined vector $U \in \mathbb{R}^{8D \times L}$, where $U_{:i} = [S^a_{:i}; \tilde{S}^b_{:i}; S^a_{:i} \circ \tilde{S}^b_{:i}; S^a_{:i} \circ \tilde{S}^a_{:i}]$.

The *two* attention flow layers serve different purposes. The *first* one fuses the static word embeddings and the POS tag embeddings resulting in token representations that encode information from a given word's POS and that of its neighbors. The POS information is useful because different idioms often follow common syntactic structures (e.g., verb-noun idioms), which can be used to recognize idioms unseen in the training data based on their similarity in syntactic structures (and thus aid generalizability). In all, the first attention flow layer yields enriched static embeddings that more effectively capture the literal representation of the input sequence. The *second* attention flow layer combines the contextualized and literal embeddings so that the resulting representation encodes the SC between the literal and contextualized representations of the PIEs. This is informed by prior findings that the SC between the static and the contextualized representation of a phrase is a good indicator of its idiomatic usage (Liu and Hwa, 2019). In addition, this attention flow layer permits working with contextualized and static embedding sequences of differing lengths using model-appropriate tokenizers for the pre-trained language model and the word embedding layer without having to explicitly map the tokens from the different tokenizers.

Prediction Phase. The prediction phase consists of a single BiLSTM layer and a linear layer. The BiLSTM layer further processes and encodes the rich representations from the attention phase. The linear layer that follows uses a log softmax function to predict the probability of each token over the five target classes idiomatic, literal, start, end, and padding. This architecture is inspired by the RNN-HG model from (Mao et al., 2019) with the difference that our BiLSTM has only one layer. During training, the token-level negative log-likelihood loss is computed and backpropagated to update the model parameters.

Implementation Details. In our implementation, the tokenizer for the language model uses the WordPiece algorithm (Schuster and Nakajima, 2012) prominently used in BERT (Devlin et al., 2019), whereas the static word embedding layer used Python's Natural Language Toolkit (NLTK) (Loper and Bird, 2002).

The pre-trained language model is the uncased base BERT from Huggingface's Transformers package (Wolf et al., 2020) with an embedding dimension of $D_{con} = 768$. The pre-trained word embedding layer is the cased Common Crawl version of GloVE, which has a vocabulary of 2.2 M words and the embedding vectors are of dimension $D_s = 300$ (Pennington et al., 2014). Both the BERT and GloVE models are frozen during training. We use NLTK's POS tagger for the POS tags.

For the character embedding layer, the input embedding dimension is 64 and the number of CNN output channels is $D_{char} = 64$. The highway network has two layers. The POS tag embedding is of dimension $D_{pos} = 64$. All the BiLSTM layers have a hidden dimension of 256, and thus $D_{emb} = 512$.

4 Experiments

Datasets. We use the following three of the largest available datasets of idiomatic expressions to evaluate the proposed model alongside other baselines. *MAGPIE* (Haagsma et al., 2020): MAGPIE is a recent, the largest-to-date corpus of PIEs in English. It consists of 1,756 PIEs across different syntactic patterns along with the sentences in which they occur (56,622 annotated data instances with an average of 32.24 instances per PIE), where the sentences are drawn from a diverse set of genres, such as news and science, collected from resources such as the British National Corpus (BNC) (BNC Consortium, 2007). For our experiments, we only considered the complete sentences of up to 50 words in length that contain the unambiguously labelled PIEs (as indicated by the perfect confidence score).

SemEval5B (Korkontzelos et al., 2013): This set has 60 PIEs unrestricted by syntactic pattern appearing in 4,350 sentences from the ukWaC corpus (Baroni et al., 2009). As in MAGPIE, we only consider the sentences with the annotated phrases. *VNC* (Cook et al., 2008): Verb Noun Combinations (VNC) dataset is a popular benchmark dataset that contains expert-curated 53 PIE types that are only verb-noun combinations and around 2,500

Dataset	Split	Size (pct. idiomatic)		# of idioms		Avg. idiom occ		Std. idiom occ	
		Train	Test	Train	Test	Train	Test	Train	Test
MAGPIE	Random	32,162 (76.63%)	4,030 (76.48%)	1,675	1,072	19.2	3.76	24.82	3.65
	Type-aware	32,155 (77.90%)	4,050 (70.54%)	1,411	168	22.79	24.11	29.96	32.05
SemEval5B	Random	1,420 (50.56%)	357 (50.70%)	10	10	142	35.7	51.25	12.69
	Type-aware	1,111 (58.74%)	341 (58.65%)	31	9	35.81	37.89	28.84	30.12
VNC	Random	2,285 (79.52%)	254 (70.47%)	53	50	43.11	5.08	25.89	2.93
	Type-aware	2,191 (79.69%)	348 (71.84%)	47	6	46.62	58	27.99	27.77

Table 2: Statistics of the datasets in our experiments showing the size of training and testing sets, proportion of instances having a figurative PIE (pct. idiomatic), the size of the PIE set (# of idioms), the average number of occurrences per PIE (avg. idiom occ), and standard deviation of the number of occurrences per PIE (std. idiom occ).

sentences containing them either in a figurative or literal sense—all extracted from the BNC. Because VNC does not mark the location of the idiom, we manually labeled them.
Together, the datasets account for a wide variety of PIEs, making this the largest available study on a wide variety of PIE categories.

Baseline Models. We use the following six baseline models for our experiments. We note that because our method is similar to idiom type classification only in its end goal and not in setting, we exclude SOTA models for idiom classification from this comparison, but include the more recent MWE extraction methods.

Gazetteer is a naïve baseline that looks up a PIE in a lexicon. In our experiments, to make the Gazetteer method independent of the algorithm and lexicon, we present the theoretical performance upper bound for any Gazetteer-based algorithm as follows. We assume that the Gazetteer perfectly detects the idiom boundaries in sentences and, in turn, predicts all PIEs to be idiomatic. We point out that since the idiomatic class is the most frequent-class in all of our benchmark datasets, this also turns out to be the *majority-class* baseline for the case of sentence-level, binary idiomatic and literal classification, that is, it predicts every sentence in a dataset to be idiomatic for binary idiom detection.

BERT-LSTM has a simple architecture that combines the pre-trained BERT and a linear layer to perform a binary classification at each token and was used in Kurfalı and Östling (2020) for disambiguating PIEs.

Seq2Seq has an encoder-decoder structure and is commonly used in sequence tagging tasks (Filippova et al., 2015; Malmi et al., 2019; Dong et al., 2019). It first uses the pre-trained BERT to generate contextualized embeddings and then sends them to a BiLSTM encoder-decoder model to tag each token as literal/idiomatic. Although not commonly used in idiom processing tasks, the encoder-decoder framework serves as a simple yet effective baseline for our tagging based idiom identification.

BERT-BiLSTM-CRF (Huang et al., 2015) is an established model for sequence tagging (and the state-of-the-art for name entity recognition in different languages [Huang et al., 2015; Hu and Verberne, 2020]), which uses a BiLSTM to encode the sequence information and then performs sequence tagging with a conditional random field (CRF).

RNN-MHCA (Mao et al., 2019) is a recent state-of-the-art model for metaphor detection on the benchmark VUA dataset that uses GloVe and ELMo embeddings with a multi-head contextual attention.

IlliniMET (Gong et al., 2020) is one of the most recent models for metaphor detection, achieving state-of-the-art performance on VUA (Steen et al., 2010) and the TOEFL (Beigman Klebanov et al., 2018) dataset. It uses RoBERTa and a set of linguistic features to perform token level metaphor tagging.

Experimental Setup. For a fair comparison across the models, we use a pre-trained BERT model in place of the linear embedding layers, ELMo, and RoBERTa model respectively in the last three baselines. The pre-trained BERT model is also frozen for all the baseline model and DISC. Owing to a lack of a good fine-tuning strategy

that fits all baselines, we leave to future work exploring improved performance via end-to-end BERT fine-tuning.

In order to test the models' ability to identify unseen idioms, each dataset was split into train and test set in two ways: *random* and *type-aware*. In the random split, the sentences are randomly divided and the same PIE can appear in both sets, whereas in the type-aware split, the idioms in the test set and the train set do not overlap. For MAGPIE and SemEval5B, we use their respective random/type-aware and train/test splits. For VNC, to create the type-aware split, we randomly split the idiom types by a 90/10 ratio, leaving 47 idiom types in train set and 6 idiom types in test set. For every dataset split, we trained every model for 600 epochs with a batch size of 64, an initial learning rate of $1e-4$, using the Adam optimizer.

The checkpoints with the best test set performance during training are recorded later in the result tables. For models with BiLSTMs, we used the same specifications as in our model with a hidden dimension of 256 and a single layer, except for BiLSTM-CRF, where we used a stacked two-layer BiLSTM. For the linear layers, we set a dropout rate of 0.2 during training. For Seq2Seq, we used a teacher forcing ratio of 0.7 during training and brute force search during inference. The same pre-trained BERT model from Huggingface's Transformers package was used as a frozen embedding layer in all models. All the other hyperparameters were in their default values.

All training and testing were done on a single machine with an Intel Core i9-9900K processor and a single NVIDIA GeForce RTX 2080 Ti graphics card.

Evaluation Metrics. We use two metrics to evaluate the performance of the models. (1) *Classification F1 score* (**F1**) measures the binary idiom detection performance at the sequence level with the presence of idioms being the positive class. (2) *Sequence accuracy* (**SA**) computes the idiom identification performance at the sentence level, where a sequence is considered as being classified correctly if and only if all its tokens are tagged correctly. We point out that the performance in terms of F1 score is essentially analogous to the performance of the idiom token classification task (see Section 2), the primary difference being whether the idiom is specified or not. Because SA is stricter than F1, we regard it to be the most relevant metric for idiom detection and span localization. Here we consider SA to be the primary evaluation metric with F1 providing additional performance references.

5 Results and Analyses

IE Identification Performance. A comparative evaluation of the models on the MAGPIE, SemEval5B, and VNC datasets is shown in Table 3.

Overall, DISC is the best performing model among all baseline models. Specifically, DISC and RNN-MHCA show competitive results in all random split settings, however, DISC has stronger performance on type-aware settings, indicating that the SC check enables DISC to recognize the non-compositionality of idioms permitting it to generalize better to idioms unseen in the training set. Therefore, while RNN-MHCA might be as good as DISC when it comes to identifying (and potentially memorizing) known idioms, DISC is more capable of identifying unseen idioms since it better leverages the SC property of idioms in addition to memorization.

In the random setting, DISC performs on par with RNN-MHCA and BERT-BiLSTM-CRF in terms of F1 and SA for MAGPIE while outperforming all baselines using the other datasets. It is notable that even with the ability to perfectly localize PIEs, Gazetteer has a low SA compared to the other top-performing models due to its inability to use the context to determine if the PIE is used idiomatically. In the type-aware setting, the F1 of DISC is comparable to that of RNN-MHCA and BERT-BiLSTM. However, in terms of SA, DISC outperforms all models across all datasets. We also observe that for all datasets, achieving high F1 scores is much easier than achieving high SA. This is especially salient in the MAGPIE type-aware split where all the models achieve similar F1s, whereas DISC outperforms the others in terms of SA by margins ranging from 7% to 30.8% absolute points. Moreover, Gazetteer is unable to perform PIE localization at all in this setting on accounts of its being limited to the instances available in an idiom lexicon.

For MAGPIE random split, it is notable that all the models (including Gazetteer with its majority-class prediction) achieve at least 86% F1 score. For MAGPIE type-aware split, DISC

Data Split	Model	Magpie		SemEval5B		VNC	
		F1	SA	F1	SA	F1	SA
Random	Gazetteer	86.67	76.47	67.29	50.70	82.68	70.47
	BERT	87.16	37.10	92.51	76.47	93.09	50.00
	Seq2Seq	92.70	83.21	94.41	*94.12	95.21	86.61
	BERT-BiLSTM-CRF	94.22	***87.71**	93.29	92.44	95.45	85.03
	RNN-MHCA	**95.51**	*86.82	*94.94	93.56	*96.15	91.33
	IlliniMET	86.54	37.97	92.59	78.15	93.55	59.45
	DISC	95.02	*87.47	***95.80**	***95.23**	***96.97**	**93.31**
Type-aware	Gazetteer	82.73	0.00	**73.94**	0.00	83.61	0.00
	BERT	86.27	39.70	73.37	35.19	86.85	50.86
	Seq2Seq	83.81	63.42	50.35	44.28	88.80	73.56
	BERT-BiLSTM-CRF	80.47	61.78	57.82	44.57	83.30	65.52
	RNN-MHCA	86.34	61.42	56.25	42.23	*88.74	79.02
	IlliniMET	83.58	39.68	69.49	41.94	87.97	54.60
	DISC	**87.78**	**70.47**	58.82	**55.71**	***89.02**	**80.46**

Table 3: Performance of models on the MAGPIE, SemEval5B, and VNC Dataset as evaluated by Classification F1 score (F1;%) and Sequence Accuracy (SA;%); best performances are boldfaced; performances marked with asterisks are comparable in their differences are not statistically significant at $p = 0.05$ using bootstrapped samples that are estimated 10^5 times.

is decisively the best performing model with absolute gains of at least 7.1% in SA and at least 1.4% in F1. For SemEval5B type-aware split, DISC is the best performing model in terms of SA with gains of at least 11.1%. Note that in terms of F1, although BERT outperforms DISC by 14.6%, Gazetteer outperforms all methods. We believe this is due to a combination of factors, including the insufficiency of the training instances (there were only 1,111 instances) and the number of idioms (there were only 31 unique idioms in the train set), and the distributional dissimilarity between the train and test sets with respect to the semantic and the syntactic properties of the PIEs in the SemEval5B dataset, for example, unlike VNC where both the train and test idioms were verb-noun constructions, SemEval5B idioms are of more diverse syntactic structures, yet SemEval5B has fewer training instances and total number of idioms. However, DISC outperforms BERT by 20.5% in SA, which shows that DISC has the best idiom identification ability. For VNC, DISC and RNN-MHCA perform competitively in all evaluation metrics in both random- and type-aware settings. In terms of SA, DISC has a $> 1\%$ gain over RNN-MHCA in both random and type-aware settings.

Tgt. Domain	Models	F1	SA
SemEval5B	RNN-MHCA	81.35	54.72
	DISC	77.70	61.80
VNC	RNN-MHCA	85.01	69.74
	DISC	83.57	72.55

Table 4: The performance of idiom identification in a cross-domain setting where models are trained on MAGPIE random and tested on target domains (Tgt. Domain) SemEval5B random and VNC random. The performance is measured by Classification F1 score (F1;%) and Sequence Accuracy (SA;%).

Idiom identification Cross-domain Performance across Datasets. To check the cross-domain performance of the best performing models (DISC and RNN-MHCA), we train them on the MAGPIE train set (as it contains the largest number of instances) and test on the VNC and the SemEval5B test sets in a random-split setting. As shown in Table 4, both models show a performance drop due to the change of the sentence source between SemEval5B and MAGPIE, and the small overlap between VNC and MAGPIE (only 4 common idioms). RNN-MHCA obtains F1 scores that are

3.6% and 1.44% higher than that of DISC on SemEval5B and VNC respectively, indicating its better ability to detect PIEs in this cross-domain setting. However, DISC is able to detect and locate the idioms more precisely, yielding SA gains of 7.8% and 2.81% over those of RNN-MHCA on SemEval5B and VNC, respectively. We argue that this demonstrates DISC's ability to identify idioms with a higher precision and that DISC's gain in SA outweighs its loss in F1, given that the gain is generally higher than the loss and SA is a more reliable measure of identification performance.

We now evaluate the performance by paying specific attention to one specific property of PIEs that makes them challenging to NLP applications—syntactic flexibility (fixedness) (Constant et al., 2017).

Effect of Idiom Fixedness. We analyze the idiom identification performance with respect to the idiom fixedness levels. Following the definitions given by Sag et al. (2002) for lexicalized phrases, we categorized idioms into three fixedness levels: (1) *fixed* (e.g., *with respect to*)—fully lexicalized with no morphosyntactic variation or internal modification, (2) *semi-fixed* (e.g., *keep up with*)—permit limited lexical variations (*kept up with*) such as inflection and determiner selection, while adhering to strict constraints on word order and composition, and (3) *syntactically flexible* (e.g., *serve someone right*)—largely retain basic word order and permit a wide range of syntactic variability such that the internal words of the idioms are subject to change. The authors (both near-native English speakers, one with linguistics background) manually labeled the PIEs in the MAGPIE test set into these 3 levels by first independently labeling 35 per level. Seeing that the agreement was 91%, all the remaining idioms were labeled by one researcher. We note that the highest level is occupied by verbal MWEs (VMWEs) that are characterized by complex structures, discontinuities, variability, and ambiguity (Savary et al., 2017).

We use this labeled set to compute the DISC performance for each fixedness level in terms of classification F1 and SA. As shown in Table 5, although the fixed idioms obtain the best performance as expected, the performance difference between semi-fixed and syntactically flexible idioms suggests that DISC can reliably detect idioms from different fixedness levels.

Metric	Idiom fixedness level		
	Fixed	Semi-fixed	Syntactically-flexible
SA	93.11	88.25	87.57
F1	94.75	92.10	92.66

Table 5: Performance of DISC on MAGPIE random split dataset as evaluated by Classification F1 score (F1;%) and Sequence Accuracy (SA;%) for each idiom fixedness level.

Error Analysis. Next, we analyze DISC's performance and its errors on the MAGPIE dataset to gain further insights into DISC's idiom identification abilities and its shortcomings.

A closer inspection of the results showed that 65.9% of the 1,071 idiom types from the MAGPIE random split test set have perfect average SA (i.e., 100%), indicating that DISC successfully learned to recognize the SC of the vast majority of the idiom types from the training set.

In order to gain insights related to DISC's ability to memorize the instances of known PIEs to perform identification on the known ones, we analyze the relationship between the average SA and the number of training samples on a per PIE basis in the MAGPIE random split using Pearson correlation. A correlation of 0.1857 with a $p < 0.05$ indicates a weak relationship between the number of training instances and the performance. This, taken together with the strong type-aware performance, suggests that DISC's identification ability relies on more than just memorizing known PIE instances.

Visualizing the attention matrices (matrix H as described in Section 3) for a sample of instances showed that the model attends to only the correct idiom (in relatively shorter sentences) or to many phrases in a sentence (longer or those with literal phrases). In the sentence *but they'd had a thorough look through his life and just to be sure and hit the jackpot entirely by chance*, underlined phrases are those with high attention and *hit the jackpot* was correctly selected as the output.[2] In some instances of incorrect prediction, that is, incompletely identified IE tokens or wrongly predicting the sentence to be literal, we found that the model still attended to the correct phrase. We hypothesize that the two attention flow layers have a hierarchical relation in their functions: The

[2]Owing to space constraints we were unable to present detailed illustrations of attention matrices to make our point.

Case #	Error Type (Pct.)	Sentence with *PIE*	Prediction
1	Alternative (9.7%)	But an *on-the-ball* whisky shop could make a killing with its special ec-label malt scotch at £27.70 a bottle.	make a killing
2	Partial (29.3%)	Dragons can lie for dark centuries brooding over their treasures, bedding down on frozen flames that will never *see the light of day*.	of day
3	Meaningful (4.3%)	Given a method, we can avoid mistaken ideas which, confirmed by the authority of the past, have *taken deep root*, like weeds in men's minds.	weeds in men's minds
4	Literal (8.0%)	If you must jump *out of the loop*, you should use until true to ''pop'' the stack.	out of the loop
5	Missing (42.3%)	We have *friends in high places*, they said.	Empty string
6	Other (6.3%)	With the chips down, we had to *dig down*.	With down

Table 6: Case studies on the DISC's idiom identification. The ground truth PIEs are in italic and colored green in sentences. The Error Type column lists the name of the error and their percentage (Pct.) in parenthesis. The percentage is obtained by manually categorizing 300 incorrect samples.

first attention flow layer, using static word embeddings and their POS tags, identifies candidate phrases that could have idiomatic meanings, and the second attention flow layer, by checking for SC, identifies the idiomatic expression's span if it exists. Hence, accurate span prediction requires the model to (1) attend to the right tokens, (2) generate/extract meaningful token representations (from the attention phase), and then (3) correctly classify the tokens. Based on the fact that the model is attending to the phrases correctly, future studies should improve upon the prediction phase using models that more efficiently leverage the features for improved token classification.

Moreover, we present case studies on the wrongly predicted instances from the MAGPIE type-aware split. Toward this, we randomly sample 25% of incorrectly predicted instances by DISC (300 instances), and group them into 6 case types: (1) *alternative*, (2) *partial*, (3) *meaningful*, (4) *literal*, (5) *missing*, and (6) *other*. These are shown in Table 6 and we discuss them below.

Case 1 is the ''alternative'' case, which is a common 'mis-identification' where DISC only identifies one of the IEs when multiple IEs are present; hence, the model detects the alternative IE to the IE originally labeled as the ground truth. Strictly speaking, this is not a limitation of our method but rather an artifact of the available dataset; all the datasets used in our experiments only label at most one PIE for each sentence even when there may be more than one. Case 2 is the ''partial'' case, which is another common wrong prediction where only a portion of the idiom span is recognized, namely, the boundary of the entire idiom is not precisely localized. Case 3 is the ''meaningful'' case in which DISC identifies figurative expressions instead of the ground truth idiom (and in this sense relates to Case 1 above). As an example, when the ground truth is *taken deep root*, DISC identifies *weeds in men's minds,* which is clearly used metaphorically and so could have been an acceptable answer. Since the idioms are unknown to DISC during test time, we argue that, as in Case 1, the identification is still meaningful, although the detected phrases are not exactly the same as the ground truth. Case 4 is the ''literal'' case in which DISC identifies a PIE that is actually used in the literal sense. Case 5, the ''missing'' case, is the opposite of Case 4, where DISC fails to recognize the presence of an idiom completely and returns only an empty string. Case 6 is the final error type ''other'' in which DISC returns words or phrases that are not meaningful or figurative, nor part of any PIEs.

After categorizing the 300 incorrect instances according to the above definitions, we found that 42.3% of them are of the ''missing'' case and around 43.4% are samples with partially correct predictions or meaningful alternative predictions. Their detailed breakdown is listed in Table 6. Tackling the erroneous cases will be a fruitful future endeavor.

6 Conclusion and Future Work

In this work, we studied how a neural architecture that fuses multiple levels of syntactic and semantic information of words can effectively perform idiomatic expression identification. Compared to

competitive baselines, the proposed model yielded state-of-the-art performance on PIEs that varied with respect to syntactic patterns, degree of compositionality and syntactic flexibility. A salient feature of the model is its ability to generalize to PIEs unseen in the training data.

Although the exploration in this work is limited to IEs, we made no idiom-specific assumptions in the model. Future directions should extend the study to nested and syntactically flexible PIEs (verbal MWEs) and other figurative/literal constructions such as metaphors—categories that were not sufficiently represented in the datasets considered in this study. Other concrete research directions include performing the task in cross- and multi-lingual settings.

Acknowledgments

We thank the anonymous reviewers for their comments on earlier drafts that significantly helped improve this manuscript. This work was supported by the IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR)—a research collaboration as part of the IBM AI Horizons Network.

References

Timothy Baldwin. 2005. Deep lexical acquisition of verb–particle constructions. *Computer Speech & Language*, 19(4):398–414. https://doi.org/10.1016/j.csl.2005.02.004

Timothy Baldwin and Su Nam Kim. 2010. Multiword expressions. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*, pages 267–292. Chapman and Hall/CRC

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226. https://doi.org/10.1007/s10579-009-9081-4

Beata Beigman Klebanov, Chee Wee (Ben) Leong, and Michael Flor. 2018. A corpus of non-native written English annotated for metaphor. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 86–91, New Orleans, Louisiana. Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-2014

Rhys Biddle, Aditya Joshi, Shaowu Liu, Cecile Paris, and Guandong Xu. 2020. Leveraging sentiment distributions to distinguish figurative from literal health reports on Twitter. In *Proceedings of The Web Conference 2020*, pages 1217–1227. https://doi.org/10.1145/3366423.3380198

Philip Blunsom. 2007. *Structured Classification for Multilingual Natural Language Processing*. Ph.D. thesis, University of Melbourne.

BNC Consortium. 2007. British national corpus, XML edition. Oxford Text Archive.

Samuel A. Bobrow and Susan M. Bell. 1973. On catching on to idiomatic expressions. *Memory & Cognition*, 1(3):343–346. https://doi.org/10.3758/BF03198118, PubMed: 24214567

Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke Van Der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu. 2017. Multiword expression processing: A survey. *Computational Linguistics*, 43(4):837–892. https://doi.org/10.1162/COLI_a_00302

Paul Cook, Afsaneh Fazly, and Suzanne Stevenson. 2008. The VNC-tokens dataset. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pages 19–22.

Silvio Cordeiro, Carlos Ramisch, Marco Idiart, and Aline Villavicencio. 2016. Predicting the compositionality of nominal compounds: Giving word embeddings a hard time. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1986–1997. https://doi.org/10.18653/v1/P16-1187

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.

Yue Dong, Zichao Li, Mehdi Rezagholizadeh, and Jackie Chi Kit Cheung. 2019. EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3393–3402, Florence, Italy. Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1331

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. https://doi.org/10.3115/1073012.1073037

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2018. Examining the tip of the iceberg: A data set for idiom translation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Afsaneh Fazly, Paul Cook, and Suzanne Stevenson. 2009. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, 35(1):61–103. https://doi.org/10.1162/coli.08-010-R1-07-048

Afsaneh Fazly and Suzanne Stevenson. 2006. Automatically constructing a lexicon of verb phrase idiomatic combinations. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Anna Feldman and Jing Peng. 2013. Automatic detection of idiomatic clauses. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 435–446. Springer. https://doi.org/10.1007/978-3-642-37247-6_35

Katja Filippova, Enrique Alfonseca, Carlos A. Colmenares, Łukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368. https://doi.org/10.18653/v1/D15-1042

Richard Fothergill and Timothy Baldwin. 2012. Combining resources for MWE-token classification. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics, *SEM 2012, June 7–8, 2012, Montréal, Canada*, pages 100–104. Association for Computational Linguistics.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764.

Hongyu Gong, Suma Bhat, and Pramod Viswanath. 2017. Geometry of compositionality. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Hongyu Gong, Kshitij Gupta, Akriti Jain, and Suma Bhat. 2020. IlliniMET: Illinois system for metaphor detection with contextual and linguistic information. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 146–153. https://doi.org/10.18653/v1/2020.figlang-1.21

Spence Green, Marie-Catherine de Marneffe, and Christopher D. Manning. 2013. Parsing models for identifying multiword expressions. *Computational Linguistics*, 39(1):195–227. https://doi.org/10.1162/COLI_a_00139

Hessel Haagsma, Johan Bos, and Malvina Nissim. 2020. MAGPIE: A large corpus of potentially idiomatic expressions. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 279–287.

Chikara Hashimoto, Satoshi Sato, and Takehito Utsuro. 2006. Japanese idiom recognition: Drawing a line between literal and idiomatic meanings. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 353–360. https://doi.org/10.3115/1273073.1273119

Yuting Hu and Suzan Verberne. 2020. Named entity recognition for Chinese biomedical patents. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 627–637.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *Clinical Orthopaedics and Related Research*, abs/1508.01991.

Leon Jaeger. 1999. *The Nature of Idioms: A Systematic Approach*. Peter Lang Pub Incorporated.

Hyeju Jang, Seungwhan Moon, Yohan Jo, and Carolyn Rosé. 2015. Metaphor detection in discourse. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 384–392, Prague, Czech Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/W15-4650

Graham Katz and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 12–19. https://doi.org/10.3115/1613692.1613696

Jerrold J. Katz and Jerry A. Fodor. 1963. The structure of a semantic theory. *Language*, 39(2):170–210. https://doi.org/10.2307/411200

Ioannis Korkontzelos and Suresh Manandhar. 2010. Can recognising multiword expressions improve shallow parsing? In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 636–644.

Ioannis Korkontzelos, Torsten Zesch, Fabio Massimo Zanzotto, and Chris Biemann. 2013. Semeval-2013 task 5: Evaluating phrasal semantics. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 39–47.

Tarun Kumar and Yashvardhan Sharma. 2020. Character aware models with similarity learning for metaphor detection. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 116–125. https://doi.org/10.18653/v1/2020.figlang-1.18

Murathan Kurfalı and Robert Östling. 2020. Disambiguation of potentially idiomatic expressions with contextual embeddings. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 85–94.

Changsheng Liu. 2019. *Toward Robust and Efficient Interpretations of Idiomatic Expressions in Context*. Ph.D. thesis, University of Pittsburgh.

Changsheng Liu and Rebecca Hwa. 2017. Representations of context in recognizing the figurative and literal usages of idioms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Changsheng Liu and Rebecca Hwa. 2019. A generalized idiom usage recognition model based on semantic compatibility. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6738–6745. https://doi.org/10.1609/aaai.v33i01.33016738

Pengfei Liu, Kaiyu Qian, Xipeng Qiu, and Xuan-Jing Huang. 2017. Idiom-aware compositional distributed semantics. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1204–1213.

Edward Loper and Steven Bird. 2002. NLTK: The natural language toolkit. In *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*. https://doi.org/10.3115/1118108.1118117

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. 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 5054–5065, Hong Kong, China. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1510

Rui Mao, Chenghua Lin, and Frank Guerin. 2019. End-to-end sequential metaphor identification inspired by linguistic theories. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3888–3898. https://doi.org/10.18653/v1/P19-1378

Diana McCarthy, Sriram Venkatapathy, and Aravind Joshi. 2007. Detecting compositionality of verb-object combinations using selectional preferences. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 369–379.

Rosamund Moon et al.. 1998. *Fixed Expressions and Idioms in English: A Corpus-Based Approach*. Oxford University Press.

Alexis Nasr, Carlos Ramisch, José Deulofeu, and André Valli. 2015. Joint dependency parsing and multiword expression tokenization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1116–1126, Beijing, China. Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-1108

Joakim Nivre and Jens Nilsson. 2004. Multiword units in syntactic parsing. *Proceedings of Methodologies and Evaluation of Multiword Units in Real-World Applications (MEMURA)*.

Darren Pearce. 2001. Synonymy in collocation extraction. In *Proceedings of the Workshop on WordNet and Other Lexical Resources, Second Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 41–46.

Jing Peng and Anna Feldman. 2016. Automatic idiom recognition with word embeddings. In *Information Management and Big Data - Second Annual International Symposium, SIMBig 2015, Cusco, Peru, September 2–4, 2015, and Third Annual International Symposium, SIMBig 2016, Cusco, Peru, September 1–3, 2016, Revised Selected Papers*, volume 656 of *Communications in Computer and Information Science*, pages 17–29. Springer. https://doi.org/10.1007/978-3-319-55209-5_2

Jing Peng, Anna Feldman, and Ekaterina Vylomova. 2014. Classifying idiomatic and literal expressions using topic models and intensity of emotions. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2019–2027. Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1216

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVE: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. https://doi.org/10.3115/v1/D14-1162

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An empirical study on compositionality in compound nouns. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 210–218. Asian Federation of Natural Language Processing.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for NLP. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 1–15. Springer. https://doi.org/10.1007/3-540-45715-1_1

Giancarlo Salton, Robert Ross, and John Kelleher. 2014. An empirical study of the impact of idioms on phrase based statistical machine translation of English to Brazilian-Portuguese. In *Proceedings of the 3rd Workshop on Hybrid Approaches to Machine Translation (HyTra)*, pages 36–41. Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-1007

Giancarlo Salton, Robert Ross, and John Kelleher. 2016. Idiom token classification using sentential distributed semantics. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 194–204. https://doi.org/10.18653/v1/P16-1019

Agata Savary, Carlos Ramisch, Silvio Ricardo Cordeiro, Federico Sangati, Veronika Vincze, Behrang Qasemi Zadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet. 2017. The PARSEME shared task on automatic identification of verbal multiword expressions. In *The 13th Workshop on Multiword Expression at EACL*, pages 31–47. Association for Computational

Linguistics. https://doi.org/10.18653/v1/W17-1704

Nathan Schneider and Noah A. Smith. 2015. A corpus and model integrating multiword expressions and supersenses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1537–1547, Denver, Colorado. Association for Computational Linguistics. https://doi.org/10.3115/v1/N15-1177

Patrick Schone and Dan Jurafsky. 2001. Is knowledge-free induction of multiword unit dictionary headwords a solved problem? In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and Korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE. https://doi.org/10.1109/ICASSP.2012.6289079

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1002–1010.

Ekaterina Shutova, Simone Teufel, and Anna Korhonen. 2013. Statistical metaphor processing. *Computational Linguistics*, 39(2):301–353. https://doi.org/10.1162/COLI_a_00124

Caroline Sporleder and Linlin Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 754–762. https://doi.org/10.3115/1609067.1609151

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. *CoRR*, abs/1505.00387.

G. J. Steen, A. G. Dorst, J. B. Herrmann, A. A. Kaal, T. Krennmayr, and T. Pasma. 2010. *A method for linguistic metaphor identification. From MIP to MIPVU*. Converging Evidence in Language and Communication Research, number 14, John Benjamins. https://doi.org/10.1075/celcr.14

Mark Stevenson and Yorick Wilks. 2001. The interaction of knowledge sources in word sense disambiguation. *Computational Linguistics*, 27(3):321–349. https://doi.org/10.1162/089120101317066104

Chuandong Su, Fumiyo Fukumoto, Xiaoxi Huang, Jiyi Li, Rongbo Wang, and Zhiqun Chen. 2020. DeepMet: A reading comprehension paradigm for token-level metaphor detection. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 30–39. Association for Computational Linguistics.

Patrizia Tabossi, Rachele Fanari, and Kinou Wolf. 2008. Processing idiomatic expressions: Effects of semantic compositionality. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 34(2):313. https://doi.org/10.1037/0278-7393.34.2.313, PubMed: 18315408

Patrizia Tabossi, Rachele Fanari, and Kinou Wolf. 2009. Why are idioms recognized fast? *Memory & Cognition*, 37(4):529–540. https://doi.org/10.3758/MC.37.4.529, PubMed: 19460959

Shiva Taslimipoor, Omid Rohanian, Ruslan Mitkov, and Afsaneh Fazly. 2018. Identification of multiword expressions: A fresh look at modelling and evaluation. In *Multiword Expressions at Length and in Depth: Extended Papers from the MWE 2017 Workshop*, volume 2, page 299. Language Science Press.

Dag Westerståhl. 2002. On the compositionality of idioms. *Proceedings of LLC8. CSLI Publications*.

Yorick Wilks. 1975. A preferential, pattern-seeking, semantics for natural language inference. *Artificial Intelligence*, 6(1):53–74. https://doi.org/10.1016/0004-3702(75)90016-8

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf,

Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-demos.6