End-to-End Sequential Metaphor Identification Inspired by Linguistic Theories

Rui Mao[1], **Chenghua Lin**[2], and **Frank Guerin**[1]

[1]Department of Computing Science, University of Aberdeen, AB24 3UE, UK
[1]{r03rm16, f.guerin}@abdn.ac.uk

[2]Department of Computer Science, University of Sheffield, S1 4DP, UK
[2]c.lin@sheffield.ac.uk

Abstract

End-to-end training with Deep Neural Networks (DNN) is a currently popular method for metaphor identification. However, standard sequence tagging models do not explicitly take advantage of linguistic theories of metaphor identification. We experiment with two DNN models which are inspired by two human metaphor identification procedures. By testing on three public datasets, we find that our models achieve state-of-the-art performance in end-to-end metaphor identification.

1 Introduction

Metaphoric expressions are common in everyday language, attracting attention from both linguists and psycho-linguists (Wilks, 1975; Glucksberg, 2003; Group, 2007; Holyoak and Stamenković, 2018). Computationally, metaphor identification is a task that detects metaphors in texts. Traditional approaches, such as phrase-level metaphor identification, detect metaphors with word pairs (Tsvetkov et al., 2014; Shutova et al., 2016; Rei et al., 2017), where a target word whose metaphoricity is to be identified is given in advance. However, such target words are not highlighted in real world text data; a newer approach is sequential metaphor identification, where the metaphoricity of a target word is identified without knowing its position in a sentence. Therefore, it is more readily applied to support Natural Language Processing tasks.

The most recent approaches (Wu et al., 2018; Gao et al., 2018) treat this as a sequence tagging task: the classified labels are only conditioned on BiLSTM (Graves and Schmidhuber, 2005) hidden states of target words. This approach is not tailor-made for metaphors; it is the same procedure to that used in other sequence tagging tasks, such as Part-of-Speech (PoS) tagging (Plank et al., 2016) and Named Entity Recognition (NER) (Lample et al., 2016). However, we have available linguistic theories of metaphor identification, which have not yet been exploited with Deep Neural Network (DNN) models. We hypothesise that by exploiting linguistic theories of metaphor identification in the design of a DNN architecture, the model performance can be further improved.

Linguistic theories suggest that a metaphor is identified by noticing a semantic contrast between a target word and its context. This is the basis of Selectional Preference Violation (SPV) (Wilks, 1975, 1978). E.g., in the sentence *my car drinks gasoline* (Wilks, 1978), 'drinks' is identified as metaphoric, because 'drinks' is unusual in the context of 'car' and 'gasoline'; a car cannot drink, nor is gasoline drinkable. Formally, a label is predicted, conditioned on a target word and its context. An alternative approach by Group (2007) and Steen et al. (2010) is the Metaphor Identification Procedure (MIP): a metaphor is identified if the literal meaning of a word contrasts with the meaning that word takes in this context. E.g., in *my car drinks gasoline*, the contextual meaning of 'drink' is 'consuming too much', which contrasts with its literal meaning of 'taking a liquid into the mouth'[1]. Formally, a label is predicted, conditioned on literal and contextual meanings. Fundamentally, the two models are similar, as both MIP and SPV analyse the relations between metaphors and their contexts, but with different procedures.

We propose two end-to-end metaphor identification models[2], detecting metaphors based on MIP and SPV, respectively. The experimental re-

[1]https://en.oxforddictionaries.com/definition/drink

[2]Our code is available at: https://github.com/RuiMao1988/Sequential-Metaphor-Identification

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3888–3898
Florence, Italy, July 28 - August 2, 2019.

sults show that both of our models perform better than the state-of-the-art baseline (Gao et al., 2018) across three benchmark datasets. In particular, our MIP based model with a simple DNN architecture, outperforms the baseline with an average of 2.2% improvement in F1 score, whereas the SPV based model with a novel multi-head contextual attention mechanism achieves an even higher gain of 2.5% against the baseline.

The contribution of our work can be summarized as follows: (1) To the best of our knowledge, we are the first to explore using linguistic theories (MIP and SPV) to directly inform the design of Deep Neural Networks (DNN) for end-to-end sequential metaphor identification; (2) Our first DNN model is based on MIP, which encapsulates the idea that a metaphor is classified by the contrast between its contextual and literal meanings. The second model is inspired by SPV, in which we propose a novel window-based contextual attentive method, allowing the model to attend to important fragments of BiLSTM hidden states and hence better capture the context of text; (3) We conducted extensive experiments on three public datasets for end-to-end metaphor identification, where both of our models outperform the state-of-the-art DNN models.

2 Related Work

Metaphor identification is a linguistic metaphor processing task that identifies metaphors in textual data, which is different from conceptual metaphor processing that maps concepts between source and target domains (Shutova, 2016), based on Conceptual Metaphor Theory (Lakoff and Johnson, 1980). In linguistic metaphor processing a metaphor is identified when the contextual meaning of a word contrasts with its literal meaning (summarised as MIP by Group (2007) and Steen et al. (2010)). Many metaphor dataset annotations were guided by MIP, e.g., VU Amsterdam Metaphor Corpus (Steen et al., 2010), and a verbal metaphor dataset, formed by Mohammad et al. (2016). Another hypothesis for linguistic metaphor identification, SPV, was proposed by Wilks (1975, 1978) who argued that a metaphoric word could violate selectional preferences of an agent. E.g., 'drinks' violates selectional preferences of the agent of 'car' in the sentence, *my car drinks gasoline*. Ortony (1979) further claimed that metaphoric words, phrases and sentences are contextually anomalous. There are also other relevant theories, e.g., semantic constraints (Katz, 1964) and expectations (Schank, 1975). However, Wilks and Fass (1992) found that these theories are mostly very similar.

In terms of computational metaphor identification, feature-engineering has been widely discussed (Leong et al., 2018). Unigrams, imageability, concreteness, abstractness, word embedding and semantic classes are features, commonly employed by supervised machine learning (Turney et al., 2011; Assaf et al., 2013; Tsvetkov et al., 2014; Klebanov et al., 2016), deep learning (Rei et al., 2017; Gutierrez et al., 2017; Bizzoni and Ghanimifard, 2018) and unsupervised leaning (Shutova et al., 2016; Mao et al., 2018) approaches.

Recently, metaphor identification has been treated as a sequence tagging task. Wu et al. (2018) proposed a model based on word2vec (Mikolov et al., 2013), PoS tags and word clusters, which were encoded by a Convolutional Neural Network (CNN) and BiLSTM. The encoded information was directly fed into a softmax classifier. This model performed best on the NAACL-2018 Metaphor Shared Task (Leong et al., 2018) with an ensemble learning strategy. Gao et al. (2018) proposed a model that concatenated GloVe (Pennington et al., 2014) and ELMo (Peters et al., 2018) representations which were then encoded by BiLSTM. Hidden states of the BiLSTM were classified by a softmax classifier. These sequential metaphor identification models classify labels, conditioned on encoder hidden states. However, we expect that explicit modelling of interactions between either contextual and literal meanings (MIP) or target words and their contexts (SPV) may further boost performance.

3 Methodology

Here we detail our two models, inspired by MIP and SPV respectively, and systematically compare the differences between them.

3.1 MIP based model

Our first model (Figure 1) is built upon MIP: a metaphor is classified by the contrast between a word's contextual and literal meanings. To facilitate the classifier in making this comparison we concatenate the contextual meaning representation with the literal meaning representation.

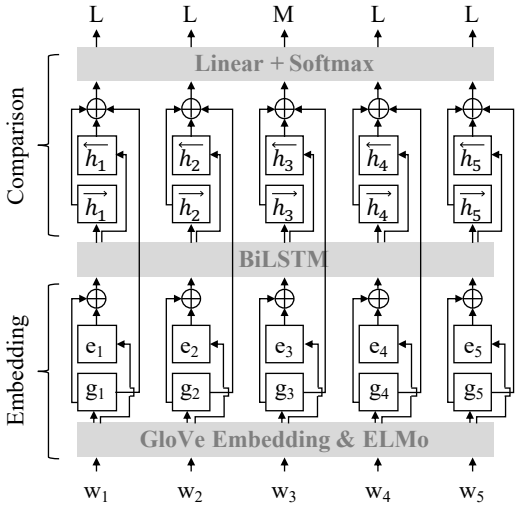

Figure 1: RNN_HG model framework based on MIP. $\oplus$ denotes concatenating tensors along the last dimension.

RNN_HG (Recurrent Neural Network_Hidden-GloVe) Humans infer the contextual meanings of a word conditioned on its context. We use BiLSTM hidden states as our contextual meaning representations, where the hidden state of a word is encoded by its forward and backward contexts and itself (Graves and Schmidhuber, 2005). Pre-trained GloVe (Pennington et al., 2014) is considered as our literal meaning representation, as words have been embedded with their most common senses (trained on Web crawled data[3]). The most common senses are likely literal, as literals occur more than metaphors in typical corpora (Cameron, 2003; Martin, 2006; Steen et al., 2010; Shutova, 2016). The comparison of literal and contextual can be seen at the top of Figure 1, comparison stage; the GloVe embedding (literal) from below joins the hidden state from the BiLSTM (contextual). The probability of a label prediction ($\hat{y}$) for a target word at position t is conditioned on contextual and literal meaning representations of the target word

$$p(\hat{y}_t|h_t, g_t) = \sigma(w^\top[h_t; g_t] + b) \quad (1)$$

where σ is softmax function. h is a BiLSTM hidden state. g is GloVe embedding. w is trained parameters. b is bias. $[;]$ denotes concatenating tensors along the last dimension. Similar to Gao et al. (2018), we use GloVe and ELMo (Embeddings from Language Models) as input features for the BiLSTM. The recommended way of using ELMo is to concatenate ELMo (e) with GloVe (g), e.g., $[g_t; e_t]$ (Peters et al., 2018). Thus, the BiLSTM hidden state h_t is

$$h_t = f_{BiLSTM}([g_t; e_t], \overrightarrow{h}_{t-1}, \overleftarrow{h}_{t+1}). \quad (2)$$

[3]Note that our results are likely to improve if the pre-trained GloVe is trained on a cleaner set of purely literal data.

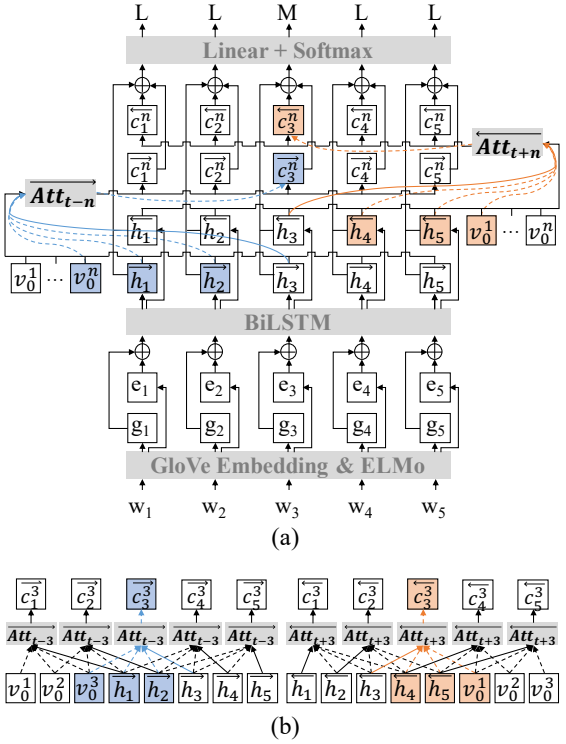

Figure 2: (a) RNN_MHCA model framework based on SPV. $Att_{t\pm n}$ denotes attention mechanisms on a window of n context words. The blue and orange nodes and lines denote examples of computing $\overrightarrow{c}^3_3$ by a query of $\overrightarrow{h}_3$ and its context v^3_0 (padding zero vectors), $\overrightarrow{h}_1$, $\overrightarrow{h}_2$, and computing $\overleftarrow{c}^3_3$ by a query of $\overleftarrow{h}_3$ and its context $\overleftarrow{h}_4$, $\overleftarrow{h}_5$, and v^1_0, respectively. (b) Attentive context representations with a window size of 3. Solid lines are queries. Dashed lines are their contexts (keys and values).

3.2 SPV based model

The intuition behind SPV is that metaphoricity is identified by detecting the incongruity between a target word and its context.

RNN_MHCA (Recurrent Neural Network_Multi-Head Contextual Attention) Our second model (Figure 2) compares a target word representation h_t with its context c_t. This is achieved by concatenating these two representations (see top of Figure 2). Target word representation h_t is a BiLSTM hidden state. Context is composed of left-side ($\overrightarrow{c}^n_t$) and right-side ($\overleftarrow{c}^n_t$) attentive context representations, where n is a window size of

context words. We adopt a multi-head contextual attention (MHCA) mechanism to compute c_t^n. The BiLSTM hidden state matrix (H, where $h \in H$) is split into equivalent pieces

$$H = [H^1; H^2; ...; H^{\mathrm{M}}; ...; H^{\mathrm{N}}] \quad (3)$$

$$\overrightarrow{head}_{t-n}^{\mathrm{M}} = \sum_{i=1}^{n} \sigma(\overrightarrow{h}_t^{\mathrm{M}\top} \overrightarrow{h}_{t-i}^{\mathrm{M}}) \overrightarrow{h}_{t-i}^{\mathrm{M}} \quad (4)$$

$$\overrightarrow{c}_t^n = [\overrightarrow{head}_{t-n}^{\mathrm{M}} | \mathrm{M} = 1, 2, ..., \mathrm{N}] \quad (5)$$

$$\overleftarrow{head}_{t+n}^{\mathrm{M}} = \sum_{i=1}^{n} \sigma(\overleftarrow{h}_t^{\mathrm{M}\top} \overleftarrow{h}_{t+i}^{\mathrm{M}}) \overleftarrow{h}_{t+i}^{\mathrm{M}} \quad (6)$$

$$\overleftarrow{c}_t^n = [\overleftarrow{head}_{t+n}^{\mathrm{M}} | \mathrm{M} = 1, 2, ..., \mathrm{N}] \quad (7)$$

$$c_t^n = [\overrightarrow{c}_t^n; \overleftarrow{c}_t^n] \quad (8)$$

where N is the number of heads. Irrelevant context hidden states, $h_j \notin [h_{t\pm1}, h_{t\pm n}]$, are masked out. We apply a window size of n context words, as h_j only encodes words that are out of the window. In computing a context representation, h_j may bring in noise, and it may miss important context information, provided by the close context words, while the distant context information could be memorized by $h_i \in [h_{t\pm1}, h_{t\pm n}]$.

Noticeably, MHCA is similar to dot-product attention (Luong et al., 2015), if $\mathrm{N} = 1$. Using $\mathrm{N} > 1$ heads would attend to different parts of hidden states of context words and recall previous important context information that is forgotten at the current point. Unlike multi-head self-attention (Vaswani et al., 2017) that encodes a target word by its context, MHCA computes the context representation by attending to a target word. The query of MHCA is a hidden state of a target word, while the key and value are hidden states of its context. We do not employ training parameters, non-linear operations or positional encoding in MHCA, because performance is better (compared with MHA in Figure 4) when we model context (via attention) and the target word (via BiLSTM) in the same space (see § 3.3). Besides, extra position encoding is unnecessary in our model, as input sentences have been encoded along with a time sequence by BiLSTM. The probability of a label prediction, given by RNN_MHCA is

$$p(\hat{y}_t | h_t, c_t^n) = \sigma(w^\top [h_t; c_t^n] + b) \quad (9)$$

where a label prediction is conditioned on a hidden state of a target word (h_t) and its attentive context representation (c_t^n). The input feature of word t is also $[g_t; e_t]$. So, h_t is given by Equation 2.

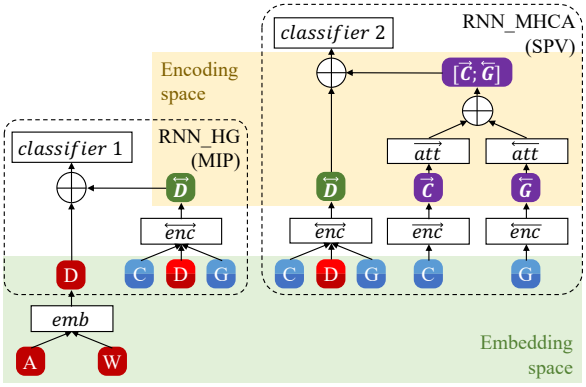

Figure 3: A comparison between RNN_HG and RNN_MHCA. C is car. D is drinks. G is gasoline. A is animal. W is water. *emb* is GloVe embedding. *enc* is BiLSTM encoding. *att* is an attention mechanism. In embedding space, the lighter part of a node is ELMo embedding, while the darker part is GloVe embedding.

3.3 Model comparison

Figure 3 gives an overview of the two models and how they process the example of 'drinks' in the sequence *car drinks gasoline*. We use different coloured nodes to indicate that words are distant from each other in vector space. E.g., red 'drinks' (D) is distant from blue 'car' (C) and 'gasoline' (G), because they are from non-literally related domains (Shutova et al., 2016; Mao et al., 2018). Note that there is no external knowledge base for domain knowledge. 'Drinks' (D) is distant because of the statistics of the corpus; it occurs in contexts relating to humans and other animals consuming liquids such as water.

Our MIP based RNN_HG model is on the left. In the leftmost part of the figure, we have the literal embedding of 'drinks' (D), which is embedded by words in the domains of 'animal' (A) and 'water' (W). To the right of this, the green 'drinks' ($\overleftrightarrow{D}$) captures the meaning of 'drinks' in context via BiLSTM encoding; it is encoded by 'car' (C), 'gasoline' (G) and itself (D). These two different vectors for 'drinks' are concatenated. Classifier 1 (RNN_HG) learns to recognise if the two vectors represent similar meanings (indicating literal) or different meanings (indicating metaphor), which is $p(\hat{y}_t | h_t, g_t)$ in Equation 1. In the case illustrated, the meaning of 'drinks' (green $\overleftrightarrow{D}$) from the encoding is very different from its word embedding meaning (red D).

The right part of Figure 3 is our SPV based RNN_MHCA model. Blue 'car' (C) and 'gasoline' (G) are encoded by themselves from left to right

and right to left, respectively. Purple 'car' ($\overrightarrow{C}$) and 'gasoline' ($\overleftarrow{G}$) are still closer to each other than green 'drinks' ($\overleftrightarrow{D}$) in encoding space, because the green 'drinks' ($\overleftrightarrow{D}$) has a component of literal meaning from red 'drinks' (D). Our attention mechanism does not employ non-linear transformations. Thus, the attentive context ($[\overrightarrow{C}; \overleftarrow{G}]$) does not significantly change its colour from the context word encoding ($\overrightarrow{C}$ and $\overleftarrow{G}$). Classifier 2 (RNN_MHCA) learns to recognise the contrast between green 'drinks' ($\overleftrightarrow{D}$) and its purple context ($[\overrightarrow{C}; \overleftarrow{G}]$), which is $p(\hat{y}_t|h_t, c_t^n)$ in Equation 9.

In RNN_MHCA, we use the BiLSTM green 'drinks' ($\overleftrightarrow{D}$) as the target word representation, rather than the word embedding red 'drinks' (D). This is necessary because it will be concatenated with the purple attentive context representation, in encoding space; we found that performance is better when both meanings are in the encoding space. On the other hand, the RNN_HG does concatenate vectors from two different spaces; this works because they are representations of the same word, rather than word versus context.

In Figure 3, it appears that both models use the same BiLSTM encoded green 'drinks' ($\overleftrightarrow{D}$), however the two models have different objective functions (Equation 1 and 9), therefore the two classifiers backpropagate different errors to the BiLSTM during training. The result is that the two models are actually receiving different hidden states (different green 'drinks' ($\overleftrightarrow{D}$) vectors).

4 Experiment

4.1 Dataset

We adopt three widely used metaphor datasets. Relevant statistics can be viewed in Table 1.

VUA[4] (Steen et al., 2010) VU Amsterdam Metaphor Corpus (VUA) is the largest publicly available metaphor dataset. Every word in the corpus is labeled, guided by MIP. Each sequence contains several metaphors, ranging from 0 to 28. The corpus was used by the NAACL-2018 Metaphor Shared Task. Similar to the task that has all PoS and verb tracks, we also examine our methods on VUA ALL POS and VUA VERB tracks.

MOH-X[5] (Mohammad et al., 2016) Its sample sentences are from WordNet (Fellbaum, 1998). Only a single target verb in each sentence is annotated. The average length of sentences is the shortest of our three datasets.

Dataset	# Tgt token	% M	# Seq	Avg # seq len	Avg # M/S
VUA_all	205,425	11.6	10,567	19.4	3.4
VUA_trn	116,622	11.2	6,323	18.4	3.3
VUA_dev	38,628	11.6	1,550	24.9	4.0
VUA_tst	50,175	12.4	2,694	18.6	3.4
VERB_tst	5,873	30.0	2,694	18.6	1.5
MOH-X	647	48.7	647	8.0	1.0
TroFi	3,737	43.5	3,737	28.3	1.0

Table 1: Dataset statistics. NB: # Tgt token is the number of target tokens whose metaphoricity is to be identified. % M is the percentage of metaphoric tokens among target tokens. # Seq is the number of sequences. Avg # seq len is the average of the number of sequence lengths. Avg # M/S is the average number of metaphors per metaphorical sentence.

TroFi[6] (Birke and Sarkar, 2006) The dataset consists of sentences from the 1987-89 Wall Street Journal Corpus Release 1 (Charniak et al., 2000). The average length of sentences is the longest of our datasets. Each sentence has a single annotated target verb.

4.2 Baselines

CNN+RNN$_{ensmb}$ (Wu et al., 2018) This is the best model at the NAACL-2018 Metaphor Shared Task, which encodes three concatenated input features (word2vec, PoS tags, and word2vec clusters) with CNN and BiLSTM. The label prediction is conditioned on BiLSTM hidden states $p(\hat{y}_t|h_t)$ with a weighted softmax classifier. The performance is further boosted by ensemble learning.

RNN_ELMo (Gao et al., 2018) This is a model that uses GloVe and ELMo as features for sequential metaphor identification. GloVe and ELMo are concatenated and encoded by BiLSTM, classified by a softmax classifier, which is also conditioned on BiLSTM hidden states $p(\hat{y}_t|h_t)$. RNN_ELMo is the strongest baseline to our knowledge.

RNN_BERT (Devlin et al., 2018) We introduce feature-based BERT (cased, large) as a baseline, as it has shown strong performance on the NER task, which is also a sequence tagging task. We use the same framework as RNN_ELMo. The inputs are the concatenation of the hidden states of the last four BERT layers, which was recommended

[4] http://ota.ahds.ac.uk/headers/2541.xml

[5] http://saifmohammad.com/WebPages/metaphor.html

[6] http://natlang.cs.sfu.ca/software/trofi.html

Model	VUA ALL POS				VUA VERB				MOH-X (10-fold)				TroFi (10-fold)			
	P	R	F1	Acc	P	R	F1	Acc	P	R	F1	Acc	P	R	F1	Acc
CNN+RNN$_{ensmb}$	60.8	70.0	65.1	-	60.0	76.3	67.2	-	-	-	-	-	-	-	-	-
RNN_ELMo	71.6	73.6	72.6	93.1	68.2	71.3	69.7	81.4	79.1	73.5	75.6	77.2	70.7	71.6	71.1	74.6
RNN_BERT	71.5	71.9	71.7	92.9	66.7	71.5	69.0	80.7	75.1	81.8	78.2	78.1	70.3	67.1	68.7	73.4
RNN_HG_ours	71.8	76.3	74.0*	93.6	69.3	72.3	**70.8***	82.1	79.7	79.8	79.8*	79.7	67.4	77.8	72.2*	74.9
RNN_MHCA_ours	73.0	75.7	**74.3***	93.8	66.3	75.2	70.5*	81.8	77.5	83.1	**80.0***	79.8	68.6	76.8	**72.4***	75.2

Table 2: Model performance. * denotes $p < 0.01$ on a two-tailed t-test, against the best baseline with an underline.

by Devlin et al. (2018). Hyperparameters are fine-tuned on each dataset.

4.3 Setup

The inputs are 300 dimension pre-trained GloVe[7] embeddings, concatenated with 1024 dimension pre-trained ELMo (Peters et al., 2018). We adopt a batch size of 2, 2×256 dimension hidden state BiLSTM, SGD optimiser, and weighted cross entropy loss

$$\mathcal{L} = -\sum_{i} w_{y_i} y_i \log(\hat{y}_i) \tag{10}$$

where y_i is a ground truth label for a word at position i. $\hat{y}_i$ is its prediction. The weight $w_{y_i} = 1$, if y_i is literal, otherwise $w_{y_i} = 2$, which is in line with Wu et al. (2018). In RNN_MHCA, the window size (n) is 3 on VUA and MOH-X, while n is 5 on TroFi. The number of attention heads (N) is 16, which is in line with Vaswani et al. (2017).

Training, development and testing sets of VUA ALL POS are built in line with the NAACL-2018 Metaphor Shared Task (see Table 1). Since the examined models predict labels for all words in a sentence, the outputs have covered the target verbs in VUA VERB. So, we simply evaluate on the verb track without training models separately. As annotations of MOH-X and TroFi datasets only cover target verbs, we consider the remaining words as literal for training, but only evaluate on the target words. We adopt 10-fold cross validation on MOH-X and TroFi datasets, since the sizes of these two datasets are small. Our hyperparameters are tuned on each dataset.

5 Results

F1 score is the main measurement of model performance. Metaphors are positive labels. The accuracy is measured by the number of correct target token predictions over the total number of target tokens. For the VUA ALL POS dataset, we consider all tokens as the target tokens. For the VUA VERB, MOH-X and TroFi, we consider target verbs as target tokens.

As shown in Table 2, our two proposed models are consistently the top two for F1 on the four evaluation tasks, where the improvements against the third best model (F1 with an underline) are statistically significant (two-tailed t-test, $p < 0.01$). RNN_MHCA achieves state-of-the-art performance in VUA ALL POS (F1=74.3%), MOH-X (F1=80.0%) and TroFi (F1=72.4%). RNN_HG performs slightly worse than RNN_MHCA. However, it exceeds RNN_MHCA by 0.3% on the VUA VERB track (F1=70.8%).

Compared with RNN_ELMo, the biggest improvements of RNN_HG and RNN_MHCA appear in MOH-X dataset, gaining 4.2% and 4.4%, respectively. Our models also outperform RNN_BERT by at least 1.6% in MOH-X. In contrast with VUA ALL POS that has an average of 3.4 metaphors (see Table 1) per metaphoric sentence, each metaphoric sentence in MOH-X contains a single metaphor. We observed that in MOH-X most non-target words are literal, so that a metaphor can be better identified by RNN_MHCA via modelling the contrast between the metaphor and its context in a single-metaphor sentence. Furthermore, the average length of MOH-X sentences is the shortest, therefore the context of a target word will be cleaner. MOH-X source sentences are from WordNet sample sentences, where the language is straightforward because the writer designed it to illustrate the meaning of a word, e.g., *Don't abuse the system*. Similarly, the straightforward contexts also help RNN_HG to infer contextual meanings of words. The anomalies that MIP and SPV are designed to detect are very clear in MOH-X, so that our models improve the most against RNN_ELMo. VUA in contrast is more complex (see examples in VUA Breakdown Analysis and Error Analysis below).

In TroFi the improvements of RNN_HG and RNN_MHCA against RNN_ELMo are small

[7] http://nlp.stanford.edu/data/glove.840B.300d.zip

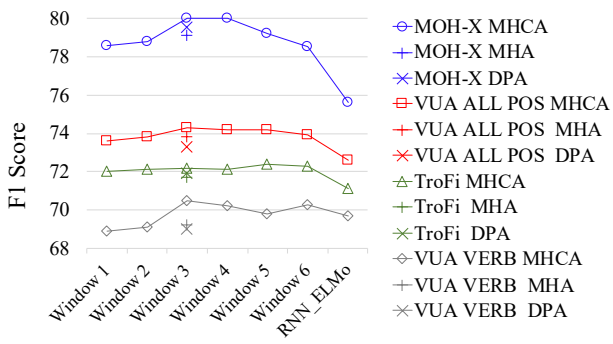

Figure 4: RNN_MHCA performance with different windows and attention mechanisms. MHCA is multi-head (16 heads) context attention. MHA is multi-head (16 heads) attention (Vaswani et al., 2017). DPA is dot-product attention (Luong et al., 2015).

(1.1% and 1.3%). We have observed that many of the non-target words in TroFi are metaphoric (but not labeled), as the sample sentences are from financial news, where word play is common (e.g., VUA news contains the largest percentage of metaphors in Table 4). Our system considers TroFi non-target words as literal without knowing their ground truth labels during training. Additionally, the average length of sequences of TroFi is the longest among the datasets, at 28.3 tokens.

Although RNN_MHCA slightly outperforms RNN_HG, the difference is small. This is because modelling the contrast between contextual and literal meanings of metaphors in MIP is theoretically similar to modelling in SPV (see §1).

Variations of RNN_HG An alternative way of encapsulating contextual and literal meanings in RNN_HG is taking the sum of h_t and g_t ($h_t + g_t$) instead of their concatenations ($[h_t; g_t]$) in Equation 1. Such an idea is inspired by residual connection (He et al., 2016). In this approach, we take 2×150 dimension BiLSTM hidden states so that h_t and g_t are aligned in dimensionality. However, such an approach yields 73.7%, 70.0%, 78.9% and 71.8% F1 scores on VUA ALL POS, VUA VERB, MOH-X and TroFi datasets, which is worse than the concatenation approach (RNN_HG) in Table 2. This is because the concatenation approach highlights the contrast between GloVe and BiLSTM hidden states of metaphors.

Variations of RNN_MHCA We examined the impact of different window sizes and attention mechanisms of RNN_MHCA. All these baselines are fine-tuned on each dataset. Given a window size of 1, bi-directional hidden states of a target

Model	Feature	P	R	F1	Acc.
RNN_BERT	B_l	69.1	72.0	70.5	93.0
RNN_HG	B_l+G	70.3	74.6	72.4	93.4
	E+G	71.0	76.1	**73.5**	93.7
RNN_MHCA	B_l+G	70.5	72.3	71.4	93.2
	E+G	71.3	75.5	73.4	93.6

Table 3: Model performance on VUA ALL POS development set. B_l is BERT large. E is ELMo. G is GloVe.

are concatenated with the left to right hidden state of its left-side word and right to left hidden state of its right-side word ($[\overrightarrow{h}_t; \overleftarrow{h}_t; \overrightarrow{h}_{t-1}; \overleftarrow{h}_{t+1}]$). The context2vec model (Melamud et al., 2016) used $\overrightarrow{h}_{t-1}$ and $\overleftarrow{h}_{t+1}$ as their context representations, with Multilayer Perceptron tuning.

As shown in Figure 4, setting a window size of 3 surpasses other sizes on 3 out of 4 datasets. The attentive context representation with a window size larger than 1 can better represent a context than the hidden states of adjacent words (window = 1). The average length of TroFi sequences is the longest, so that a larger window size, e.g., window = 5, performs better. Given a window size of 3, MHCA outperforms the multi-head attention (Vaswani et al., 2017) which employs training parameters and non-linear operations. This shows that modelling the contrast between a target word and its context in the same space performs better than that in different spaces. MHCA exceeds the dot-prodcut attention (Luong et al., 2015) which demonstrates the utility of multi-heads that attend to different fragments of hidden states. We also examined variations, e.g., an infinite window size and a different number of heads, but the performances did not improve.

Variations of Feature Selection We examine the concatenation of hidden states of the last four BERT large model layers (B_l) instead of ELMo on RNN_HG and RNN_MHCA. Our models with the combination of BERT and GloVe (B_l+G) perform better than the BERT baseline model (RNN_BERT) with B_l on VUA ALL POS development set by at least 2.9% in terms of F1 score (see Table 3). However, the performance, given by B_l+G, is not further improved, compared with the combination of ELMo and GloVe (E+G) on each of our models.

VUA Breakdown Analysis We report the model performance on different types of articles and words based on VUA ALL POS test set. We analyse all four genres and four types of open class words (verbs, adjectives, nouns and adverbs),

Type	Train		Dev		Test		All	
	%T	%M	%T	%M	%T	%M	%T	%M
News	21.8	14.9	23.8	15.5	24.6	15.2	22.9	15.1
Acad.	36.4	11.2	37.3	11.6	27.1	17.3	34.3	12.4
Fict.	23.4	10.7	23.5	10.6	21.9	9.2	23.0	10.4
Conv.	18.3	7.4	15.4	7.2	26.4	7.6	19.8	7.4
Verb	17.9	18.1	18.5	18.7	19.7	19.1	18.5	18.5
Noun	17.6	13.6	17.8	13.5	17.1	15.0	17.5	13.9
Adj	8.3	11.5	8.3	10.7	7.9	13.6	8.2	11.9
Adv	6.0	6.0	5.8	6.9	6.8	7.2	6.1	6.5

Table 4: VUA Statistics on genres and POS. % T denotes the percentage of the category tokens among the total VUA tokens. % M denotes the percentage of the category metaphors among the category tokens.

which is in line with Leong et al. (2018). The verbal statistics in Table 5 are different from VUA VERB in Table 2, as they are different tracks in the Metaphor Shared Task. Not all verbs in VUA ALL POS are included in VUA VERB.

In Table 5, metaphor identification achieves better performance on academic articles across all the models and genres, where RNN_MHCA yields the highest F1 (79.8%). Intuitively, metaphor identification is easier as the style of English is more formal. E.g., (using underlines for metaphors) *This mixture, heated by recession and high unemployment, inevitably generates a high level of crime.* (VUA ID: as6-fragment01-30). Identifying metaphors in conversation is the hardest for our baselines, probably due to its fragmented language. E.g., *Drawing, oh well!* (VUA ID: kbp-fragment09-4105). However, RNN_HG achieves large improvements against RNN_ELMo (3.8%) and RNN_BERT (3.4%) on conversation. The improvements of our models against RNN_ELMo on news are larger than in TroFi, although source sentences of both datasets are from news. It supports our arguments that the noise of treating non-target words as literals in TroFi negatively impact our models' ability to learn the difference between literals and metaphors. In contrast, all words in VUA news are annotated, so that the advantages of our models are more obvious.

In PoS breakdown analysis, verb metaphors are better identified than others, as verbal metaphors take the largest part among all PoS. RNN_HG achieves the biggest improvement (4.1%) in adverbs against RNN_ELMo, whereas RNN_BERT also presents strong performance. In adjectives, CNN+RNN$_{ensmb}$ surpasses the second best RNN_HG by 2.9%. The use of word embedding clusters, PoS tags and ensemble learning may contribute to identifying adjective metaphors.

	Model	P	R	F1	Acc
Acad.	CNN+RNN$_{ensmb}$	72.5	74.6	73.5	-
	RNN_ELMo	78.2	80.2	79.2	92.8
	RNN_BERT	76.7	76.0	76.4	91.9
	RNN_HG_ours	76.5	83.0	79.6	92.7
	RNN_MHCA_ours	79.6	80.0	**79.8**	93.0
Conv.	CNN+RNN$_{ensmb}$	45.3	71.1	55.3	-
	RNN_ELMo	64.9	63.1	64.0	94.6
	RNN_BERT	64.7	64.2	64.4	94.6
	RNN_HG_ours	63.6	72.5	**67.8**	94.8
	RNN_MHCA_ours	64.0	71.1	67.4	94.8
Fict.	CNN+RNN$_{ensmb}$	48.3	69.2	56.9	-
	RNN_ELMo	61.4	69.1	65.1	93.1
	RNN_BERT	66.5	68.6	67.5	93.9
	RNN_HG_ours	61.8	74.5	67.5	93.4
	RNN_MHCA_ours	64.8	70.9	**67.7**	93.8
News	CNN+RNN$_{ensmb}$	66.4	64.7	65.5	-
	RNN_ELMo	72.7	71.2	71.9	91.6
	RNN_BERT	71.2	72.5	71.8	91.4
	RNN_HG_ours	71.6	76.8	74.1	91.9
	RNN_MHCA_ours	74.8	75.3	**75.0**	92.4
VERB	CNN+RNN$_{ensmb}$	-	-	67.4	-
	RNN_ELMo	68.1	71.9	69.9	-
	RNN_BERT	67.1	72.1	69.5	87.9
	RNN_HG_ours	66.4	75.5	**70.7**	88.0
	RNN_MHCA_ours	66.0	76.0	**70.7**	87.9
ADJ	CNN+RNN$_{ensmb}$	-	-	**65.1**	-
	RNN_ELMo	56.1	60.6	58.3	-
	RNN_BERT	58.1	51.6	54.7	88.3
	RNN_HG_ours	59.2	65.6	62.2	89.1
	RNN_MHCA_ours	61.4	61.7	61.6	89.5
NOUN	CNN+RNN$_{ensmb}$	-	-	62.9	-
	RNN_ELMo	59.9	60.8	60.4	-
	RNN_BERT	63.3	56.8	59.9	88.6
	RNN_HG_ours	60.3	66.8	**63.4**	88.4
	RNN_MHCA_ours	69.1	58.2	63.2	89.8
ADV	CNN+RNN$_{ensmb}$	-	-	58.8	-
	RNN_ELMo	67.2	53.7	59.7	94.8
	RNN_BERT	64.8	61.1	62.9	94.8
	RNN_HG_ours	61.0	66.8	**63.8**	94.5
	RNN_MHCA_ours	66.1	60.7	63.2	94.9

Table 5: Model performance on different types of texts and words in VUA ALL POS.

Error Analysis By comparing our two models, 96.3% of predictions are the same in the VUA ALL POS testing set. For these same predictions, precision, recall, F1 and accuracy are 80.2%, 77.2%, 78.7% and 95.3%, respectively, which is better than each of our models on the full dataset. False negatives are common in sentences with multiple metaphors, e.g., *Or: 'When Cupid shot his dart He shot it at your heart.'* (VUA ID: a5e-fragment06-187), where 10 out of 12 words have true labels as metaphor. However, our models only classify 'heart' as metaphoric in this sentence. Ambiguous contexts are also challenging, e.g., *I'm gonna play with that and see what* (VUA ID: kbd-fragment21-8037), where the referent of

'that' is not in the context, so that 'play with' are also false negatives.

For the samples where our models predict different labels, the main errors of RNN_HG are false negatives, while the main errors of RNN_MHCA are false positives. This is likely due to the fact that some conventional metaphors frequently appear in typical corpora, so that GloVe embeddings of metaphors are not distinct from their contextual meaning encodings. Metaphors may be misclassified as literal by RNN_HG. On the other hand, RNN_MHCA may flag the clash between literals and their contexts, if there are many metaphors in the contexts, so that literal target words may be misclassified as metaphoric.

6 Conclusion

We proposed two metaphor identification models based on Metaphor Identification Procedure (Group, 2007; Steen et al., 2010) and Selectional Preference Violation (Wilks, 1975, 1978). Our models achieve state-of-the-art performance on three public datasets. The performances of the two models are close in terms of F1 score, as their linguistic fundamentals, MIP and SPV are similar in principle. The breakdown analysis of VUA demonstrates that the improvements of our models derive from the problematic instances for our baselines, e.g., conversation articles and adverb metaphors.

In future work, we will explore ensemble learning. Our error analysis demonstrates that when the predictions of our two models are the same, the prediction is more accurate with high precision, suggesting the idea of combining them. Another interesting direction is to explore combining different semantic similarity measures (Lin et al., 2015) for our task.

Acknowledgments

We thank anonymous reviewers for their comments, which will further influence our next work. We also appreciate Sujie Guo for providing GPU resources. This work is supported by the award made by the UK Engineering and Physical Sciences Research Council (Grant number: EP/P011829/1).

References

Dan Assaf, Yair Neuman, Yohai Cohen, Shlomo Argamon, Newton Howard, Mark Last, Ophir Frieder, and Moshe Koppel. 2013. Why "dark thoughts" aren't really dark: A novel algorithm for metaphor identification. In *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2013 IEEE Symposium on*, pages 60–65. IEEE.

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Yuri Bizzoni and Mehdi Ghanimifard. 2018. Bigrams and BiLSTMs two neural networks for sequential metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, pages 91–101.

Lynne Cameron. 2003. *Metaphor in educational discourse*. A&C Black.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. 2000. Bllip 1987-89 WSJ corpus release 1. *Linguistic Data Consortium, Philadelphia*, 36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Ge Gao, Eunsol Choi, Yejin Choi, and Luke Zettlemoyer. 2018. Neural metaphor detection in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Sam Glucksberg. 2003. The psycholinguistics of metaphor. *Trends in cognitive sciences*, 7(2):92–96.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Pragglejaz Group. 2007. MIP: A method for identifying metaphorically used words in discourse. *Metaphor and symbol*, 22(1):1–39.

E Dario Gutierrez, Guillermo Cecchi, Cheryl Corcoran, and Philip Corlett. 2017. Using automated metaphor identification to aid in detection and prediction of first-episode schizophrenia. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2923–2930.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Keith J Holyoak and Dušan Stamenković. 2018. Metaphor comprehension: A critical review of theories and evidence. *Psychological bulletin*, 144(6):641.

Jerrold J Katz. 1964. Analyticity and contradiction in natural language. In *The Structure of Language: Readings in the Philosophy of Language*. Prentice Hall.

Beata Beigman Klebanov, Chee Wee Leong, E Dario Gutierrez, Ekaterina Shutova, and Michael Flor. 2016. Semantic classifications for detection of verb metaphors. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 101–106.

George Lakoff and Mark Johnson. 1980. *Metaphors we live by*. University of Chicago press.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.

Chee Wee Ben Leong, Beata Beigman Klebanov, and Ekaterina Shutova. 2018. A report on the 2018 VUA metaphor detection shared task. In *Proceedings of the Workshop on Figurative Language Processing*, pages 56–66.

Chenghua Lin, Dong Liu, Wei Pang, and Zhe Wang. 2015. Sherlock: A semi-automatic framework for quiz generation using a hybrid semantic similarity measure. *Cognitive computation*, 7(6):667–679.

Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Rui Mao, Chenghua Lin, and Frank Guerin. 2018. Word embedding and WordNet based metaphor identification and interpretation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1–10.

James H Martin. 2006. A corpus-based analysis of context effects on metaphor comprehension. Technical Report CU-CS-738-94, Boulder: University of Colorado: Computer Science Department.

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

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Saif Mohammad, Ekaterina Shutova, and Peter Turney. 2016. Metaphor as a medium for emotion: An empirical study. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 23–33.

Andrew Ortony. 1979. Some psycholinguistic aspects of metaphor. *Center for the Study of Reading Technical Report; no. 112*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 2227–2237.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *The 54th Annual Meeting of the Association for Computational Linguistics*, page 412.

Marek Rei, Luana Bulat, Douwe Kiela, and Ekaterina Shutova. 2017. Grasping the finer point: A supervised similarity network for metaphor detection. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1537–1546.

Roger C Schank. 1975. The structure of episodes in memory. In *Representation and understanding*, pages 237–272. Elsevier.

Ekaterina Shutova. 2016. Design and evaluation of metaphor processing systems. *Computational Linguistics*.

Ekaterina Shutova, Douwe Kiela, and Jean Maillard. 2016. Black holes and white rabbits: Metaphor identification with visual features. *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 160–170.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. *A method for linguistic metaphor identification: From MIP to MIPVU*, volume 14. John Benjamins Publishing.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 248–258.

Peter D Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 680–690.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Yorick Wilks. 1975. A preferential, pattern-seeking, semantics for natural language inference. *Artificial intelligence*, 6(1):53–74.

Yorick Wilks. 1978. Making preferences more active. *Artificial intelligence*, 11(3):197–223.

Yorick Wilks and Dann Fass. 1992. The preference semantics family. *Computers & Mathematics with Applications*, 23(2-5):205–221.

Chuhan Wu, Fangzhao Wu, Yubo Chen, Sixing Wu, Zhigang Yuan, and Yongfeng Huang. 2018. Neural metaphor detecting with CNN-LSTM model. In *Proceedings of the Workshop on Figurative Language Processing*.