Article:

Combining Pre-trained Word Embeddings and Linguistic Features for Sequential Metaphor Identification

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

We tackle the problem of identifying metaphors in text, treated as a sequence tagging task. The pre-trained word embeddings GloVe, ELMo and BERT have individually shown good performance on sequential metaphor identification. These embeddings are generated by different models, training targets and corpora, thus encoding different semantic and syntactic information. We show that leveraging GloVe, ELMo and feature-based BERT based on a multi-channel CNN and a Bidirectional LSTM model can significantly outperform any single word embedding method and the combination of the two embeddings. Incorporating linguistic features into our model can further improve model performance, yielding state-of-the-art performance on three public metaphor datasets. We also provide in-depth analysis on the effectiveness of leveraging multiple word embeddings, including analysing the spatial distribution of different embedding methods for metaphors and literals, and showing how well the embeddings complement each other in different genres and parts of speech.

1 Introduction

Linguistically, metaphor is defined as a figurative expression that uses one or several lexical units to represent a different meaning in the context, where there is a semantic contrast between the contextual and basic meanings of the lexical units (Pragglejaz, 2007). Metaphors are widely found in corpora, causing issues for natural language processing (NLP) tasks, such as sentiment analysis (Ghosh et al., 2015) and machine translation (Mao et al., 2018). Traditionally, metaphors are identified from dependent word-pairs, e.g. adjective-noun and verb-noun pairs (Turney et al., 2011; Neuman et al., 2013; Rei et al., 2017). There is an increasing trend to attempt metaphor identification

as an end-to-end sequence tagging task (Wu et al., 2018; Gao et al., 2018; Mao et al., 2019), as the settings of traditional approaches are not practical in real-world applications. In contrast, a sequential metaphor identification model can predict the metaphoricity of all words in a sentence, as a sequence labelling task, without needing to locate word-pairs beforehand.

Pre-trained word embeddings have been shown to provide significant improvements to the state of the art for a wide range of NLP tasks (Young et al., 2018). Currently, there are many choices of pre-trained word embeddings, such as GLoVe (Pennington et al., 2014), ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). These word embeddings are based on different training targets, training sets, and model frameworks. E.g., GloVe is trained via token co-occurrence distributions, whose vectors are independent of the context of downstream tasks. ELMo is a BiLSTM based context dependent representation, whose training target is training Language Models (LM) from two directions. BERT is also context dependent, and is trained to predict masked words and next sentences via the Bidirectional Transformer (Vaswani et al., 2017). All the aforementioned pre-trained word embeddings have presented strong performance on a variety of NLP tasks. Our hypothesis is that different word embeddings may capture different semantics and syntax knowledge, and thus metaphor identification models may benefit from their combination by taking advantage of their complementarity.

The motivation of this paper is to investigate the utilities of different pre-trained word embeddings, e.g., GloVe, ELMo, feature-based BERT and their combination (GEB) for metaphor identification. In particular, if we can combine existing word embeddings to take advantage of their respective strengths, then we do not have to retrain a supermodel on a multi-genre corpus for

every technological breakthrough in word embedding. To facilitate our investigation, we propose a neural architecture consisting of a multi-channel CNN layer and a BiLSTM layer, where each CNN channel takes one embedding as input. Combining CNN and BiLSTM also allows the model to capture local and long-term dependent information of an input sequence.

We evaluate on three publicly available datasets, namely VUA (Steen et al., 2010), MOH-X (Mohammad et al., 2016) and TroFi (Birke and Sarkar, 2006) datasets. Experiments show that our GEB model yields competitive performance for sequential metaphor identification compared to the strongest baseline. Incorporating linguistic features (i.e., PoS and word abstractness (Rabinovich et al., 2018)) into the GEB model achieves further gains, yielding the state-of-the-art performance for sequential metaphor identification. In terms of the effectiveness of different embeddings, using the combination model GEB yields better results than using each embedding individually or the combinations of any two of them. In addition, it was found that GloVe, ELMo and BERT have different strengths in modelling metaphors from different genres and in different parts of speech (PoS), because they are trained on different corpora with different algorithms. This explains why the combination outperforms the individuals.

The contribution of this paper can be summarised as follows: (1) we propose a 3-channel CNN based model[1] that can incorporate diverse embeddings and features for sequential metaphor identification, achieving state-of-the-art performance; (2) we investigate the utilities of the combinations of GloVe, ELMo and feature based BERT across 24 layers, showing their complementarity in terms of different genres and PoS; (3) we provide insight into the factors determining which combinations of embeddings that are likely to have good performance on a downstream task, e.g., maintaining a balance between general and task-specific features.

2 Related Work

Traditionally, metaphors are identified from dependent word-pairs with various feature engineering methods. Psychological features such as abstractness and concreteness have been widely applied in verb-noun and adjective-noun metaphor

---

[1]Our code will be released after blind review.

identification tasks (Turney et al., 2011; Neuman et al., 2013; Assaf et al., 2013; Tsvetkov et al., 2014). Their hypothesis is that metaphors normally are concrete, associated with an abstract dependent word. Shutova et al. (2016, 2017) and Rei et al. (2017) classified metaphors by modelling word distribution features with supervised and unsupervised models. Mao et al. (2018) extended the word-pair approaches by identifying metaphors from full sentences, taking the full context of a sentence into account using input and output vectors of word2vec (Mikolov et al., 2013). However, these methods are based on a condition that the position of a target word whose metaphoricity to be identified in a sentence has been annotated in advance. Such approaches are inconvenient in real world applications.

Sequential metaphor identification takes one step further because the metaphoricity of all words in a sentence can be identified without knowing the position of a target word in advance. Wu et al. (2018) employed a one-channel CNN and BiLSTM based framework for sequential metaphor identification. Their model encodes concatenated word2vec, word2vec cluster labels and PoS labels. Then, the hidden states of BiLSMT are classified by a softmax classifier. This model with an ensemble learning strategy achieved the best performance on the NAACL-2018 Metaphor Shared Task (Leong et al., 2018). Gao et al. (2018) used a vanilla BiLSTM based sequence tagging model. The concatenation of GloVe and ELMo is encoded by BiLSTM and then classified by a softmax classifier to predict a sequence of metaphoricity labels. Mao et al. (2019) proposed two linguistics informed models. The first Metaphor Identification Procedure (MIP) (Pragglejaz, 2007) inspired model emphasised the contrast between contextual and literal meanings of a target word. The second Selectional Preference Violation (SPV) theory (Wilks, 1975, 1978) inspired model highlighted the contrast between a metaphor and its context. Both models outperform the models of Wu et al. (2018) and Gao et al. (2018), while the SPV informed model is the state-of-the-art.

While Gao et al. (2018) and Mao et al. (2019) have used more than one word embeddings for metaphor identification, the number of embeddings was limited at two, and more importantly they did not explore how and why combining pretrained embeddings is effective for the task. We

seek to answer this question by providing systematic evaluation on the efficiency of a variety of embedding combinations as well as their complementarity, leading to interesting findings relating to genre and PoS types of the corpus for training word embedding models. In addition, linguistic features such as abstractness have only been used in traditional machine learning based verb-noun and adjective-noun pair metaphor identification (Turney et al., 2011; Neuman et al., 2013; Tsvetkov et al., 2014). We examine the use of abstractness and PoS features for sequential metaphor identification.

3 Methodology

We propose a CNN-BiLSTM framework for sequential metaphor identification, partially inspired by the works of Kim (2014) and Wu et al. (2018). One of the key differences between our model and that of Wu et al. (2018) is that we use a multi-channel CNN layer to incorporate different word embedding features, whereas Wu et al. simply used a single-channel CNN on word2vec. Note that using a multi-channel CNN component allows us to combine different pre-trained word embeddings in a simple way. Additionally, CNN has shown its effectiveness on a variety of sequence tagging tasks (Poria et al., 2016; Chen et al., 2017; Xu et al., 2018), but these works have only explored using a single-channel CNN and a single word embedding. We give the technical details of our model in the following sections.

3.1 Framework

Figure 1 shows the overall architecture of our model. We apply a CNN operation on the input which has three channels for leveraging different embeddings. The pre-trained contextual embeddings ELMo and BERT have the same dimensionality, which is 1024. However, GloVe only has 300 dimension, so we project GloVe into 1024 dimensions with a linear layer to align the dimensionality of all the input embeddings.

$$\mathbf{M}_G = W^\phi \cdot \text{GloVe} + b^\phi \tag{1}$$

Here W^ϕ is the parameter matrix of the linear layer, and b^ϕ is its bias vector. We then stack the input embedding matrices (i.e., the projected GloVe, ELMo and BERT) which is used as input for the subsequent feature map calculation.

$$\mathbf{I} = \begin{bmatrix} \mathbf{M}_G & \mathbf{M}_E & \mathbf{M}_B \end{bmatrix} \tag{2}$$

Note that we pad zero vectors for $\mathbf{I}$ to make the row dimensionality of the feature map matrix the same as the length of the input sequence.

An alternative version of our model also incorporates external linguistic features for model learning. In our study, we experimented with PoS and word abstractness ($Abst$) features, which are represented in one-hot encoding and real numbers ($\in [0, 1]$), respectively. We concatenate ($\oplus$) the linguistic feature representations with Glove

$$\mathbf{M}_{GPA} = \text{GloVe} \oplus \text{PoS} \oplus \text{Abst}. \tag{3}$$

$\mathbf{M}_{GPA}$ is then projected to 1024 dimension via a linear layer (Eq. 1) to make an alternative version of $\mathbf{M}_G$. Finally, the resulting representation is combined with ELMo and BERT in a similar way (Eq. 2) as previously described.

Given a 3-channel embedding input ($\mathbf{I}$), a feature map (F) is calculated as follows:

$$F_{i,w} = \sum_c \sum_w \sum_d \mathbf{I}_{(c,i+w-1,d)} K_{(c,w,d)}, \tag{4}$$

where $F_{i,w}$ is the new representation of the ith word of an input sentence obtained by applying CNN operations on input embeddings for the corresponding word with kernel K and window size w. c, d are channel size ($c = 3$) and the dimensionality of input word embeddings ($d = 1024$), respectively. In order to capture different local contextual information, we use CNN kernels with four types of window sizes $w = \{2, 3, 4, 5\}$ and for each window size 100 kernels. This results in four different feature maps, each of which corresponds to a particular window size. Finally, we concatenate all the feature maps together

$$F = [F_{w=2} \oplus F_{w=3} \oplus F_{w=4} \oplus F_{w=5}]. \tag{5}$$

We fed F with a Tanh activation function into a BiLSTM layer as input. A BiLSTM is employed for capturing long-term dependencies of F, because the CNN layer only encodes the contextual information of a word in n-grams

$$h_i = \text{BiLSTM}(\text{Tanh}(F_i), \overrightarrow{h}_{i-1}, \overleftarrow{h}_{i+1}), \tag{6}$$

where h is a hidden state of BiLSTM. Finally, the metaphoricity (y_i) for a given word is predicted by a softmax (σ) classifier
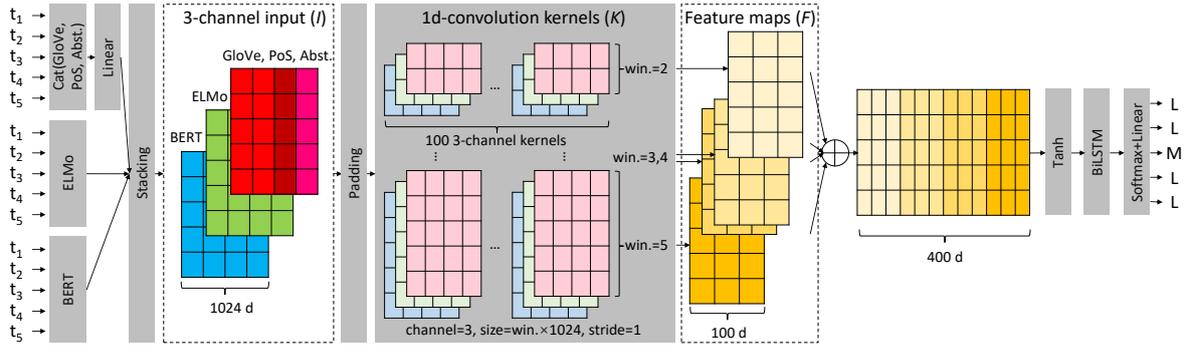
$$p(y_i) = \sigma(W_i^\psi \cdot h_i + b_i^\psi), \tag{7}$$

Figure 1: Model framework. Grey boxes are embedding and computational layers. $\oplus$ denotes concatenating.

where W^{ψ} and b^{ψ} are model parameters and biases estimated by optimising the following objective function based on weighted cross entropy loss.

$$\mathcal{L} = -\sum_i \omega_{y_i} y_i \log(\hat{y}_i) \qquad (8)$$

Following Wu et al. (2018) and Mao et al. (2019), we set the weight (ω) for metaphors and literals to 2 and 1, respectively.

3.2 Features

We select GloVe, ELMo and BERT because they were trained on different corpora with different genres and training targets, which may exhibit distinctive semantic and syntactic characteristics. For instance, GloVe was trained on Common Crawl[2], from billions of web pages (840 billion tokens); ELMo was trained on WMT 2011 News Crawl data[3] (800 million tokens); BERT was trained on Wikipedia[4] (2.5 billion tokens) and BookCorpus[5] (800 million tokens) (Zhu et al., 2015), where scientific articles and novels are included. We employed feature-based BERT rather than fine-tuned BERT because the former allows us to investigate the utility of each BERT layer of a fixed depth neural network. We hypothesise that sequential metaphor identification will benefit from the complementarity of multiple word embeddings.

Apart from embedding features, we also explored linguistic features, i.e., PoS and abstractness of words, in order to examine whether these universal features can further improve our model performance. The abstractness feature set ($\mathcal{V}$) contains 99,954 words with each word assigned with

---

[2] http://commoncrawl.org
[3] http://www.statmt.org/lm-benchmark/
[4] https://dumps.wikimedia.org/
[5] https://yknzhu.wixsite.com/mbweb

Dataset	# Tgt token	% M	# seq	Avg # M/S
VUA_all	205,425	11.6	10,567	3.4
VUA_trn	116,622	11.2	6,323	3.3
VUA_dev	38,628	11.6	1,550	4.0
VUA_tst	50,175	12.4	2,694	3.4
VERB_tst	5,873	30.0	2,694	1.5
MOH-X	647	48.7	647	1.0
TroFi	3,737	57.6	3,737	1.0

Table 1: Dataset statistics (Mao et al., 2019). 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 # M/S is the average number of metaphors per metaphorical sentence.

an abstractness score ranging from 0 to 1 (Rabinovich et al., 2018), i.e., the higher the score, the more abstract the corresponding word. For instance, the abstractness score of *purism* is 0.97 (abstract), while abstractness of *ski* is 0.25 (concrete). For a word w_o which is not listed in $\mathcal{V}$, we predict its abstractness by taking the abstractness score of its most semantically similar word (w_s) in $\mathcal{V}$, where the calculation is based on GloVe (g) using cosine similarity.

$$w_s = \underset{w_i \in \mathcal{V}}{\mathrm{argmax}}\, \mathrm{cosine}(g_{w_o}, g_{w_i})$$
$$Abst_{w_o} = Abst_{w_s} \qquad (9)$$

In case w_o is out of the vocabulary of GloVe, it will be assigned with an abstractness score of 0.5.

4 Experiment

4.1 Dataset

We examine our model on three public datasets, whose statistics are summarised in Table 1.

VUA VU Amsterdam Metaphor Corpus (Steen et al., 2010) is the largest all-word annotated

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**
Wu et al.	60.8	70.0	65.1	-	60.0	76.3	67.2	-	69.2	69.9	69.6	70.0	79.6	78.8	79.2	78.3
Gao et al.	71.6	73.6	72.6	93.1	68.2	71.3	69.7	81.4	79.1	73.5	75.6	77.2	87.7	87.4	87.6	86.9
Mao et al.-SPV	73.0	75.7	<u>74.3</u>	93.8	66.3	75.2	<u>70.5</u>	81.8	77.5	83.1	<u>80.0</u>	79.8	89.8	88.1	<u>88.9</u>	88.0
GEB_{17}	74.9	74.4	74.7*93.9		70.4	72.1	71.2*82.5		78.0	83.1	80.4*80.2		90.7	89.0	89.8*88.4	
PoS+Abst+GEB_{17}	72.5	77.4	**74.9***94.0		68.8	74.5	**71.5***82.4		77.9	83.8	**80.7***80.3		89.3	91.0	**90.2***88.6	

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

metaphor dataset (Fleiss' Kappa = 0.84), whose sentences are sampled from the British National Corpus (Leech, 1992). The dataset contains four genres (i.e., academic, conversation, fiction, and news) and has been used in the NAACL-2018 Metaphor Shared Task (Leong et al., 2018). Following the shared task, we construct two versions of the datasets, i.e., VUA-all-PoS and VUA-verb. VUA-all-PoS considers all words as target words, whereas VUA-verb takes a subset of verbs in VUA as target words. VUA is our main dataset for the later breakdown analysis (§ 5.3) as it contains metaphors from different genres and PoS.

MOH-X The dataset was formed by sampling example sentences from WordNet (Fellbaum, 1998) by Mohammad et al. (2016). A named target verb for each sentence was annotated by 10 participants. In MOH-X, each metaphoricity label has been agreed by at least 70% annotators. There are 453 unique target verbs in MOH-X.

TroFi The dataset was sampled from the 1987-89 Wall Street Journal Corpus Release 1 (Charniak et al., 2000), where target verbs of 3,737 sentences were annotated by Birke and Sarkar (2006). We sample these human annotated sentences of the original TroFi dataset in our tests. There are 50 unique target verbs in TroFi.

For MOH-X and TroFi, we consider the rest of words (non-targets) as literal for the model training, because only a single target word in a sentence is annotated, which is in line with Gao et al. (2018). Our model is evaluated on the target words with 10-fold cross validation on these two datasets.

4.2 Baselines

We compare our model against three strong baselines on the task of sequential metaphor identification. Details of the baselines are given below.

Wu et al. (2018) proposed a CNN+BiLSTM model, fitting word2vec, PoS, word2vec cluster features to sequential metaphor identification tasks

with an ensemble learning strategy.

Gao et al. (2018) adopted a standard sequence tagging model with a BiLSTM encoding layer. They used ELMo and GloVe as input features.

Mao et al. (2019) proposed two models inspired by linguistic theories (MIP and SPV), claiming state-of-the-art performance across three benchmark metaphor datasets using ELMo and GloVe.

4.3 Setup

We adopt 840B GloVe[6], original ELMo[7] and cased BERT large pre-trained word embeddings. PoS tags are parsed by spaCy. We adopt four kernel sizes (2×1024, 3×1024, 4×1024, and 5×1024) with a stride of 1 in the convolution layer. Concatenated feature maps are activated by the Tanh function. The dimension of BiLSTM hidden states is 256×2. We use SGD optimizer with a learning rate of 0.2. Dropout is applied on 3-channel input and BiLSTM hidden states with rates of 0.5 and 0.1, respectively.

5 Results and discussion

5.1 Overall performance

F1 score is our main measure, where metaphor is the positive class. As shown in Table 2, our model GEB_{17} (the subscript 17 denotes the 17th BERT layer) achieves better performance than the state-of-the-art baselines. By incorporating linguistic features, our model PoS+Abst+GEB_{17} yields extra performance gains, outperforming the best baselines on all datasets significantly.

When comparing GEB_{17} against Wu et al. (2018), the baseline model which has the most similar architecture to ours, we see that using multiple CNN channels and word embeddings, especially contextual dependent word embeddings, can substantially improve CNN-BiLSTM model performance by an average of 8.8% in F1 across all

---
[6] http://nlp.stanford.edu/data/glove.840B.300d.zip
[7] https://allennlp.org/elmo

	VB	ADJ	NN	ADV	ALL
GEB_{17}	69.4	57.8	61.3	57.6	72.5
$PoS+GEB_{17}$	69.4	55.3	**63.4**	62.1	72.8
$Abst+GEB_{17}$	69.1	55.2	61.7	59.8	72.3
$PoS+Abst+GEB_{17}$	**70.7**	**58.2**	**63.4**	**62.4**	**73.5**

Table 3: Linguistic feature analysis on VUA-all-PoS development set, measured by F1 score.

datasets. GEB_{17} also outperforms the models of Gao et al. (2018) and Mao et al. (2019) which both utilise GloVe and ELMo, with an average gain of 2.7% and 0.6% in F1, respectively. One may notice that the gain of GEB_{17} over the SPV model (Mao et al., 2019) is relatively small. This is likely because SPV benefits from the use of extra multi-head contextual attention mechanism whereas our encoder is simpler with only BiLSTM. Overall, the experimental results have clearly demonstrated the effectiveness of employing multiple word embeddings for the task of metaphor identification.

In another set of experiments, we further explore whether universal linguistic features (e.g., word abstractness and PoS) that are not explicitly encoded by pre-trained word embeddings can further improve deep learning models for sequential metaphor identification. Experimental results show that incorporating word abstractness and PoS features does help boosting model performance but not substantially, where the gain of $PoS+Abst+GEB_{17}$ over GEB_{17} is around 0.3 F1 on average. This is probably because contextual dependent word embeddings have already implicitly learnt linguistic features such as PoS from corpora (Reif et al., 2019), resulting in the explicit incorporation of linguistic features being less effective (see § 5.2.2 for detailed discussion).

We ran ablation experiments to test the utilities of PoS and abstractness with GEB_{17} on VUA-all-PoS development set. As seen in Table 3, the gain of $PoS+GEB_{17}$ over GEB_{17} is small (0.3 F1 on all PoS). Additionally, simply adding abstractness features ($Abst+GEB_{17}$) does not improve the GEB_{17} model performance. This is because the abstractness level is less distinctive between metaphors and literals overall (see Table 4). Although metaphorical verbs, adjectives and nouns have lower abstractness scores than literals in terms of mean, first quartile and median, adverbs hold the opposite. By combining PoS and abstractness features, $PoS+Abst+GEB_{17}$ yields a gain of 1.0 F1 against GEB_{17} on VUA-all-PoS de-

Figure 2: Model performance on VUA-all-PoS development set given by different word embedding features. BERT layer 1 is close to the input side, while layer 24 is close to the output side.

velopment set, where the largest improvement appears in adverbs. This is somewhat unexpected as using each of the linguistic features alone is not effective. A plausible explanation is that guided by explicit PoS features, our model can better distinguish and make use of the more useful abstractness features for metaphor identification.

5.2 Different BERT layer analysis

In this section, we explore which BERT layers are more effective for the metaphor identification task. We focus on BERT rather than ELMo because previous works have already discovered that the most transferable layer of ELMo is the first layer as tested across a number of semantic and syntactic tasks (Liu et al., 2019; Schuster et al., 2019).

Figure 2 shows that both low BERT layers$_{1:4}$ and top layers$_{21:24}$ yield weak results (blue line), while the middle layers$_{9:11}$ are strong. This is somewhat contradictory to the finding of Devlin et al. (2019), where they proposed to use the concatenation of the last four feature-based BERT layers$_{21:24}$ as the input features for the NER task. This implies that which BERT layer works best as a contextualised embedding is task dependent. Although ELMo (orange dash line) significantly outperforms GloVe (purple line), the combination of them (GE, yellow line) still yields better performance. Such a pattern can also be observed in GB$_{13:24}$ combinations (green line) compared with BERT layer$_{13:24}$ (blue line) and GloVe. Finally, GEB$_{17:20}$ (red line) outperform other word embed-

	Verb		Adjective		Noun		Adverb		Overall	
	Meta	**Lite**	**Meta**	**Lite**	**Meta**	**Lite**	**Meta**	**Lite**	**Meta**	**Lite**
Mean	0.26	**0.28**	0.27	**0.34**	0.33	**0.36**	**0.25**	0.22	0.29	**0.30**
Std.	0.18	0.16	0.19	0.22	0.21	0.21	0.21	0.18	0.21	0.19
1st Quart.	0.11	**0.16**	0.13	**0.16**	0.17	**0.19**	**0.13**	0.11	**0.15**	0.14
Median	0.21	**0.29**	0.22	**0.26**	0.28	**0.33**	**0.18**	**0.18**	0.21	**0.24**
3rd Quart.	**0.32**	0.30	0.35	**0.50**	0.48	**0.51**	**0.31**	0.23	**0.43**	**0.43**

Table 4: Abstractness statistics for metaphors and literals.

dings and their combinations across BERT layers, where GEB_{17} is the best.

5.2.1 Probing BERT layer effectiveness via word similarity

We hypothesise that the effectiveness of different contextualised embedding layers for metaphor detection can be probed by measuring how well the embedding can distinguish metaphors and literals that take the same word form. The underlying intuition is that metaphoric meanings are strongly context dependent, and hence the task provides a natural challenging setting for testing the effectiveness of contextualised embeddings for the task.

We conducted the experiment based on the VUA-all-PoS training set[8], which contains 1,516 unique words that have both metaphoric and literal labels. For each of those words, we constructed a metaphor-literal word-pair set. For instance, the word-pair corresponding to "wash" contains a literal "wash" from "I'm supposed to wash up aren't I" (VUA_ID: kbw-fragment04-2697) and a metaphorical "*wash*" from "Do we really *wash down a good meal with claret?*" (VUA_ID: a3c-fragment05-233). In the case where a word associated with a particular label appears in multiple sentences, we randomly sample one sentence for that label for constructing the word-pair. The total number of word-pairs is therefore 1,516.

To probe the effectiveness of different BERT layers, we compute the average cosine similarity between the words for all metaphor-literal word-pairs by taking the embedding from each layer as input. A lower average cosine similarity indicates a BERT layer is more effective in distinguishing metaphors from literals. The result is shown in the red line in Figure 3. For comparison, we also show the metaphor identification performance given by a one-channel CNN+BiLSTM model using different BERT layer as input (blue line in Figure 3).
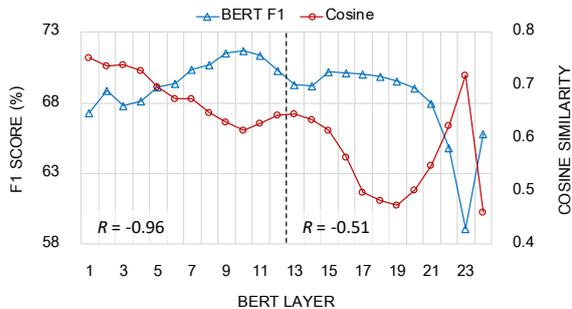

Figure 3: Correlation between model performance on VUA-all-PoS development set and BERT layer effectiveness on the training set. BERT layer effectiveness is represented by the average cosine similarity between metaphor-literal word-pairs (lower is better).

It can be observed from Figure 3 that there exists a negative correlation between the model performance in metaphor identification and how well the BERT embeddings can distinguish metaphors from literals, i.e., in general the lower average cosine similarity, the better model performance. In particular, the correlation is much stronger for the low level BERT layers (1:12; Pearson's $R = -0.96$) than the high level layers (13:24; $R = -0.51$). It can also be observed that the bottom BERT layers (1:4) mostly encoding general features are more effective than the top layers (21:24) which encode LM task-specific features. Thus, layer 24 which is the closest to LM output during BERT pre-training procedure does not yield the highest performance on metaphor identification, although layer 24 has the lowest cosine similarity. The middle layers (9:11), which capture both general and task-specific features, seem to work best with layer 10 giving the best performance. Such an observation is similar to the findings of Liu et al. (2019), who argued that the difference in the transferability across contextualiser layers on downstream tasks is due to the trade-off between embedded general and task-specific features.
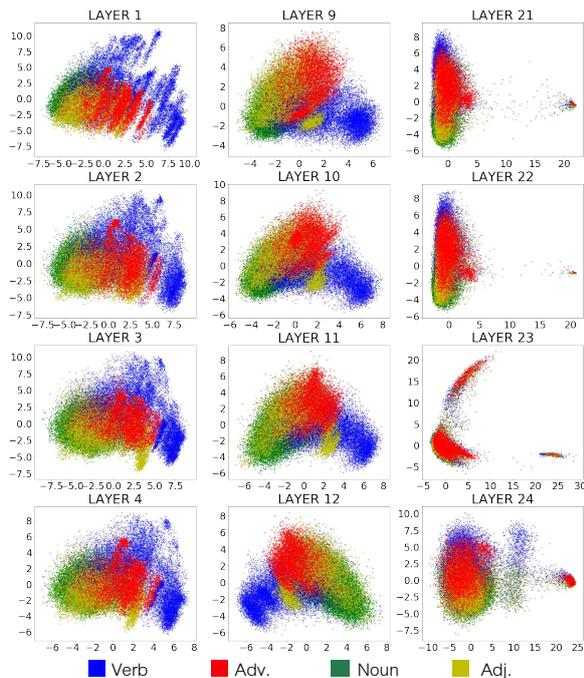
---

[8]The biggest one in our experimental datasets in terms of size.

Figure 4: PCA-based 2D visualisations of BERT on open class words of the VUA-all-PoS training set.

5.2.2 POS

We further visualise the distribution of word embeddings across different BERT layers to analyse how syntactic information is captured. For each BERT layer, we extract the embeddings for the open class words in the VUA-all-PoS training set, and then project the embeddings to a 2D space using Principal Component Analysis (PCA) (Jolliffe, 2011). We show in Figure 4 the distributions of the most representative BERT layers, i.e., 1:4, 9:12, and 21:24. Each point corresponds to a word and the colour indicates the word class.

It can be observed from Figure 4 that the distribution of the top layers (21:24) is highly concentrated in a relatively small region (layer 23 in particular), where words of different PoS classes are heavily overlapped with each other. In contrast, the word distributions of layers 1:4 and 9:12 exhibit a distinctive pattern, where words are more spread out and there are clear boundaries between words of different classes, showing that syntactic features like PoS are captured in those layers. These phenomena also explain why the effectiveness of explicitly incorporating PoS features is small, i.e., recalling the marginal gain of PoS+Abst+GEB$_{17}$ over GEB$_{17}$ discussed in § 5.1. To sum up, which BERT layer is more transferable for metaphor identification appears to depend on the trade-off between the general and task-specific features captured by the layer.

5.3 Effectiveness of combining multiple word embeddings

In order to analyse the semantic and syntactic complementarity between different pre-trained word embeddings, we analyse the performance of each feature based on different genres and PoS. We examine their performance on VUA-all-PoS development set because its sentences were labelled with multiple genre categories, e.g., academic text, fiction, news and conversation. Compared with MOH-X and TroFi whose annotated target words focus on verbs, VUA provides metaphor annotation to all word classes. Here, we examine the model performance on open class words (i.e., verbs, adjectives, adverbs, and nouns) because they consist of the majority of metaphors in the VUA dataset.

It can be seen from Table 5 that ELMo slightly outperforms BERT$_{17}$ on news (0.4%) where BERT$_{17}$ gives better performance on the other three genres with the biggest difference on the fiction category (2.9%), which could be explained by the fact that ELMo was trained on the WMT 2011 News Crawl data whereas BERT was trained on Wikipedia and BookCorpus. The biggest advantage of BERT$_{17}$ against ELMo appearing on fiction also supports such a hypotheses, i.e., the BookCorpus provides good training resources for BERT in the book domain, and hence BERT can better handle the metaphors in fiction. These observations show that that our model can incorporate the strengths of different word embeddings in different domains, so that we do not have to re-train a supermodel on a very large multigenre corpus.

It can be observed that contextualised embeddings i.e., ELMo and BERT outperform GloVe by large margins of 5.6% and 6.6% on average, respectively. This is somewhat unsurprising as the task of metaphor identification is strongly context dependent. There is also a clear pattern that the combination of two embeddings works better than individual embeddings, and the combination of three embeddings works best. EB$_{17}$, the best two-embedding combination, outperforms the best single embedding BERT by an average of 1.0% in F1. Using the three embedding combination further boosts the model performance where a gain of

Feature	P	R	F1	Acc	P	R	F1	Acc	P	R	F1	Acc	P	R	F1	Acc
	Academic				Conversation				Fiction				News			
GloVe	65.2	67.5	66.3	92.1	58.4	62.6	60.4	94.1	60.1	55.6	57.8	91.4	69.3	64.9	67.0	90.1
ELMo	65.1	74.1	69.3	92.4	67.6	65.1	66.4	95.2	62.3	68.4	65.2	92.3	72.6	73.4	73.0	91.6
BERT$_{17}$	67.3	71.7	69.4	92.7	70.9	63.0	67.7	95.7	70.3	65.9	68.1	93.5	74.0	71.1	72.6	91.6
GE	66.9	74.6	70.5	92.8	63.3	69.3	66.1	94.9	65.8	65.5	65.7	92.8	73.1	74.5	73.8	91.8
GB$_{17}$	64.7	77.2	70.4	92.5	68.1	67.5	67.8	95.7	70.3	67.6	68.9	93.7	74.3	71.5	72.9	91.7
EB$_{17}$	71.8	72.3	72.0	93.5	69.9	66.3	68.1	95.8	72.9	64.8	68.6	93.6	76.1	70.5	73.2	91.9
GEB$_{17}$	72.7	72.0	**72.3**	93.8	74.0	64.9	**69.1**	95.9	75.9	67.1	**71.2**	94.3	77.7	71.4	**74.4**	92.4
	Verb				Adjective				Noun				Adverb			
GloVe	60.2	57.2	58.7	84.9	54.9	42.2	47.7	90.1	59.1	50.5	54.5	88.6	49.4	49.4	49.4	93.0
ELMo	62.7	70.3	66.3	86.6	46.7	54.9	50.5	88.5	61.5	58.6	60.0	89.5	57.6	51.9	54.6	94.0
BERT$_{17}$	63.3	72.2	67.5	89.9	54.7	49.1	51.8	90.2	66.8	51.7	58.3	90.0	66.7	45.5	54.1	94.7
GE	62.4	68.9	65.5	86.4	56.9	58.7	**57.8**	90.8	62.4	59.9	61.1	90.1	53.7	56.5	55.1	93.6
GB$_{17}$	64.7	69.1	66.8	87.1	58.4	53.8	56.0	90.9	65.0	57.7	61.1	90.1	61.3	49.4	54.7	94.3
EB$_{17}$	66.9	69.0	67.9	87.8	53.7	53.2	53.4	90.1	73.4	49.5	59.1	90.8	63.3	49.4	55.5	94.5
GEB$_{17}$	71.6	67.4	**69.4**	88.9	62.8	53.5	**57.8**	91.6	69.9	54.5	**61.3**	90.7	69.1	49.4	**57.6**	95.0

Table 5: The utilities of different word embeddings and combinations on different genres and PoS of VUA-all-PoS development set.

2.3% over the best performing single embedding BERT is achieved. This clearly shows the benefit of using multiple embeddings for metaphor identification on different genres.

For PoS, similar patterns to the genre analysis can be observed, where incorporating two embeddings gives better performance than each individual embedding and that incorporating three embeddings works best. For adjectives in particular, there is a big improvement of GE over ELMo (7.3%) and GB$_{17}$ over BERT$_{17}$ (4.2%). Overall, the combination of different word embeddings presents the strongest performance across different genres and PoS, which is mainly due to the difference in training algorithms and corpora between the three selected word embeddings.

5.4 Bad combination analysis

We demonstrate the difference in model performance (Δ F1), given by different word embedding setups in Figure 5. As seen, although GEB outperforms most of the single word embedding based models in Figure 5a in terms of F1 score, we also observe that EB surpasses GEB before BERT layer 11 in Figure 5b. An intuitive question is what makes bad combinations across BERT layers. Apart from the trade-off between general and task-specific features (see § 5.2), we try to explain the bad combinations from spatial relations between ELMo and BERT layers here.
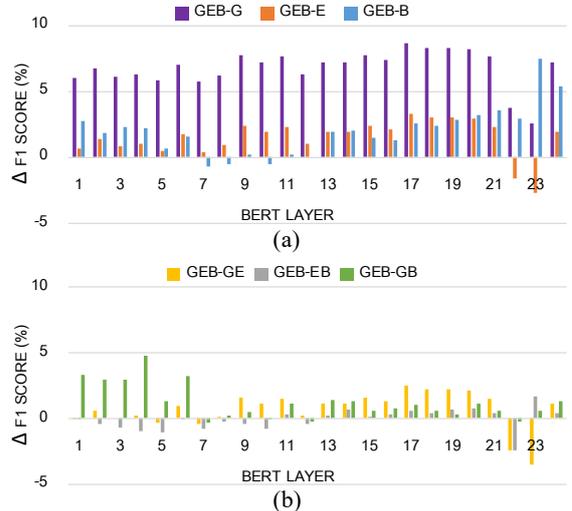

Figure 5: Model performance differences between GEB and other word embedding setups on VUA-all-PoS development set.

We analyse the combinations of ELMo and different BERT layers, because they are aligned in dimensionality. BERT vectors are rotated to ELMo space with an orthogonal mapping approach (Xing et al., 2015; Conneau et al., 2017).

$$W^* = \underset{W \in O_d(\mathbb{R})}{\mathrm{argmin}} \|WB - E\|_F = UV^\top \quad (10)$$

with $U\Sigma V^\top = \mathrm{SVD}(EB^\top)$, where $O_d(\mathbb{R})$ is 1024×1024 dimension matrices of real numbers. B and E are BERT and ELMo embedding matrices of the VUA-all-PoS training set. W is the learnt linear mapping matrix. SVD is Singular
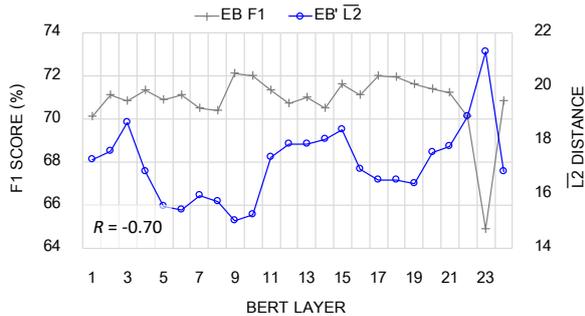
Figure 6: Correlation between EB model performance on VUA-all-PoS development set and EB′ average $L2$ distance on the training set.

Value Decomposition. The rotated BERT (B') is given by WB. We compute the average $L2$ distance ($\overline{L2}$) between E and each layer of B' with

$$\overline{L2}_{EB'} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\sum_{j=1}^{d}(E_{i,j} - B'_{i,j})^2}, \quad (11)$$

where n is the total number of tokens in the training set ($n = 116,622$); i is the position of a token; j is the position of an element of vectors in E and B'. We compare the $\overline{L2}$ with F1 scores on VUA-all-PoS development set.

As seen in Figure 6, there is a negative correlation ($R = -0.70$) between $\overline{L2}$ and model performance[9]. The selected parallel word embeddings are distant from each other in vector space, meaning that the distribution patterns of the two embeddings are likely very different. A model may struggle to combine distant word embeddings, as they present less consistency in representing word semantics and syntax.

6 Conclusion

We propose a model that can incorporate advanced word embedding and linguistic features, achieving state-of-the-art performance on the sequential metaphor identification task. We also examine different combinations of GloVe, ELMo and different BERT layers. Our experiments show that the output-side BERT layers do not distinguish metaphors and literals well, which is different to the finding of Devlin et al. (2019) where the last four layers of BERT yielded strong performance

---

[9]Using non-rotated BERT instead of B' in Eq. 11 also presents a negative correlation with the model performance, whereas the correlation is weaker ($R = -0.53$).

on the NER task. In contrast, the middle layers of BERT perform best for sequential metaphor identification. We also find that different word embeddings can complement each other, because they may embed complementary semantics and syntax, due to training on different corpora with different algorithms. This offers opportunities for leveraging multiple existing pre-trained word embeddings for improving a variety of downstream tasks, which we would like to explore in the future.

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*.

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.

Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2017. A feature-enriched neural model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3960–3966. AAAI Press.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

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*, pages 4171–4186.

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*.

Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. 2015. Semeval-2015 task 11: Sentiment analysis of figurative language in twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 470–478.

Ian Jolliffe. 2011. *Principal Component Analysis*. Springer.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Geoffrey Neil Leech. 1992. 100 million words of English: the British national corpus (BNC). *Language Research*.

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.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1073–1094.

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 1222–1231.

Rui Mao, Chenghua Lin, and Frank Guerin. 2019. End-to-end sequential metaphor identification inspired by linguistic theories. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 3888–3898.

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.

Yair Neuman, Dan Assaf, Yohai Cohen, Mark Last, Shlomo Argamon, Newton Howard, and Ophir Frieder. 2013. Metaphor identification in large texts corpora. *PloS One*, 8(4):e62343.

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*, pages 2227–2237.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2016. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108:42–49.

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

Ella Rabinovich, Benjamin Sznajder, Artem Spector, Ilya Shnayderman, Ranit Aharonov, David Konopnicki, and Noam Slonim. 2018. Learning concept abstractness using weak supervision. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4854–4859.

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.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems*, pages 8592–8600.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1599–1613.

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

Ekaterina Shutova, Lin Sun, Elkin Darío Gutiérrez, Patricia Lichtenstein, and Srini Narayanan. 2017. Multilingual metaphor processing: Experiments with semi-supervised and unsupervised learning. *Computational Linguistics*, 43(1):71–123.

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. Association for Computational Linguistics.

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.

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*.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Double embeddings and CNN-based sequence labeling for aspect extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. 2018. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13(3):55–75.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 19–27.