MDG: Metaphorical sentence Detection and Generation with Masked Metaphor Modeling

Anonymous ACL submission

Abstract

This study tackles literal to metaphorical sentence generation, presenting a framework that can potentially lead to the production of an infinite number of new metaphors. To achieve this goal, we propose a complete workflow that tackles metaphorical sentence classification and metaphor reconstruction. Unlike similar research works regarding metaphor generation, our approach does not require any custom or closed-source model, hence with this work we introduce a complete literal to metaphorical open-source model. The obtained results show that 24%, 31% and 56% of the (originally literal) sentences are turned to metaphorical by changing a single noun, adjective or verb of the sentence, respectively. Human evaluation shows that our constructed metaphors are considered more fluent, creative and metaphorical than figurative statements created by a real person. Furthermore, by using our artificial data to increase the training size of a metaphorical sentence classification dataset, we register an improvement of 3% over the baseline.

1 Introduction

Figurative language is an ambiguous language that often contains mapping of concepts from one domain to another. In order to better understand metaphors and their complexity, as well as the challenges that they can bring to natural language processing tasks, it is important to look at practical examples and at the related core literature works. Consider, for instance, the following metaphorical sentence: *The wheels of Stalin's regime were well-oiled and already turning*, where a political system is viewed in terms of a mechanism which can function, break, have wheels, etc. This association allows us to transfer knowledge from the domain of *mechanisms* to that of *political systems*. Therefore, political systems are thought about in terms of mechanisms, and discussed through the mechanism terminology, leading to multiple metaphorical expressions. This particular view of metaphors is known as Conceptual Metaphor Theory, and it was first introduced by Lakoff and Johnson (1980) in 1980. There are different types of metaphors, such as the **is-a** type (e.g., *That lawyer is a shark*), the **of** type (e.g., *Child of evil*), or **verb**-based (e.g., *He cut me off, yet still I carried his name*).

Some computational approaches among the ones that have been presented in literature have focused on metaphor detection and generation. Detection comprises metaphor *identification* (Steen et al., 2010), where approaches identify metaphor-related words in the text (Fass, 1997; Birke and Sarkar, 2006; Shutova et al., 2010), and *interpretation*, which employs paraphrasing (Tong et al., 2021). Metaphor *generation* concerns the task of creating novel metaphorical sentences, for example by taking literal ones and transforming them in a way that makes them acquire a figurative meaning. This task is useful for poetry generation (Van de Cruys, 2020) or even as a new source to augment datasets used to train metaphor detectors and interpreters.

To the best of the authors' knowledge, there are no studies in literature that try to simultaneously address metaphor detection and generation in an end-to-end setting. Furthermore, all existing metaphor generators (Chakrabarty et al., 2021; Yu and Wan, 2019; Tong et al., 2021; Brooks and Youssef, 2020; Stowe et al., 2021) depend on external and sometimes publicly unavailable systems that go beyond standard fine-tuning procedures. By contrast, we present the first literal-to-metaphorical text-to-text framework, called MDG, that is able to generate novel metaphorical sentences by replacing different types of part-of-speech tokens, not only verbs (Stowe et al., 2021; Chakrabarty et al., 2021; Yu and Wan, 2019), but also nouns and adjectives (examples shown in Fig. 1). Human evaluation showed that metaphors created by MDG were found to be *more fluent, creative and metaphorical* than figurative statements created by a native speaker.

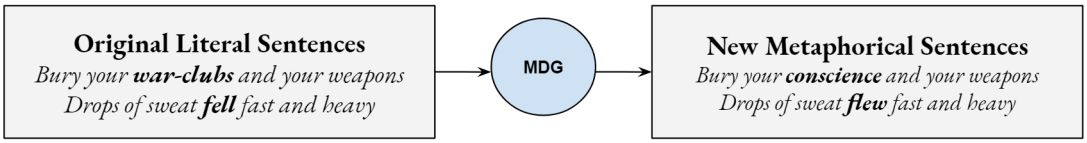

Figure 1: Depiction of literal sentences turned to metaphorical by our framework

2 Related Work

The majority of neural models treat **metaphor identification** as a sequence labelling task, creating an output that consists of a sequence of labels (metaphorical or not) for a sentence or a sequence of input words (Bizzoni and Ghanimifard, 2018; Chen et al., 2020; Dankers et al., 2020; Gao et al., 2018; Gong et al., 2020; Mao et al., 2019; Mykowiecka et al., 2018; Pramanick et al., 2018; Su et al., 2020; Wu et al., 2018). The first sequence labelling approaches usually represented an input sentence as a concatenation of pre-trained word embeddings and generated a context-specific sentence embedding exploiting bidirectional long short-term memory, or *BiLSTM* (Dankers et al., 2020; Gao et al., 2018; Mykowiecka et al., 2018; Pramanick et al., 2018; Bizzoni and Ghanimifard, 2018).

Numerous BiLSTM systems take advantage of both contextualised and pre-trained embeddings in the classification layer (Mao et al., 2019; Swarnkar and Singh, 2018). In particular, the *Di-LSTM* Contrast system (Swarnkar and Singh, 2018) encodes the left and right side context of a target word through forward and backward LSTMs. The classification is based on a concatenation of the target word representation and its difference with the encoded context (Tong et al., 2021). Mao et al. (2019) combined GloVe (Pennington et al., 2014) and BiLSTM hidden states for sequence labelling. Static embeddings like GloVe (Pennington et al., 2014) do not change with the context once been learned. Despite their efficiency, the static nature of these embeddings makes it difficult to cope with the *polysemy* problem (crucial when dealing with metaphors), since the meaning of a polysemous word depends on its context (Wang et al., 2020).

To deal with the problem of polysemy, a number of approaches have been recently proposed to learn the representation of words among their context. For example, in the following two sentences: "Apple sells phones" and "I eat an apple", dynamic embeddings (Wang et al., 2020) will represent "apple" differently according to the context, while static embeddings can not distinguish the semantic difference between the two references of "apple". Dynamic embeddings extracted from pre-trained language models (Devlin et al., 2019; McCann et al., 2018; Peters et al., 2018; Radford and Sutskever, 2018) have demonstrated dramatic superiority over their static predecessors in various NLP tasks, and also in metaphor detection and generation approaches.

Recent related work (Chen et al., 2020; Dankers et al., 2020; Gong et al., 2020) adopts a fine-tuning approach, employing pre-trained contextual language models such as Bidirectional Encoder Representations from Transformers (Devlin et al., 2019) (BERT (Devlin et al., 2019) and RoBERTa (Zhuang et al., 2021)), and taking advantage of the aforementioned dynamic embeddings (Wang et al., 2020). For example, Dankers et al. (2020) fine-tuned a BERT (Devlin et al., 2019) model, which gets a discourse fragment as input. Hierarchical attention computes both token and sentence level attention (Kobayashi et al., 2020) after the encoded layers, leading to better results compared to those obtained by applying general attention to all tokens.

Metaphor generation methods used in literature are usually based on obtaining novel figurative sentences either by replacing verbs contained in literal phrases (Chakrabarty et al., 2021; Yu and Wan, 2019; Stowe et al., 2021), or exploiting syntactic patterns that discriminate between creative metaphorical expressions and non-metaphorical ones (Brooks and Youssef, 2020). Table 1 presents these works, but we also describe them in detail below. Chakrabarty et al. (2021) generated novel metaphoric sentences by taking literal expressions and replacing relevant verbs. Furthermore, new metaphors are obtained by transforming metaphorical sentences from the Gutenberg Poetry corpus (Jacobs, 2018) into their literal version, through masked language modeling (Song et al., 2019), and then using a sequence to sequence model finetuned on this parallel data to generate new figurative expressions. Yu and Wan

Table 1: Comparisons between MDG for metaphor generation and related works' frameworks. Each column indicates whether the related approach provides methods respectively for masked language modeling, metaphor reconstruction, and/or extraction. The *Self-sufficiency* column indicates whether the related works' approaches can function relying only on public architectures, or whether they need customized models' implementations. As it is possible to see, MDG is the only one that addresses each one of the different topics highlighted in the columns simultaneously, and which is also totally self-sufficient, relying only on Transformers models and architectures.

Related Work	MLM	Reconstruction	Extraction	Self-sufficiency
Chakrabarty et al. (2021)	✓	✓		
Yu and Wan (2019)		✓	✓	
Brooks and Youssef (2020)		✓		
Stowe et al. (2021)		✓		
MDG	✓	✓	✓	✓

(2019) employed a neural approach to extract the metaphorical verbs from the sentences along with their metaphorical senses in an unsupervised way. Then, the same neural approach is exploited to train a neural language model from Wikipedia corpus. The novel metaphors are obtained by conveying the assigned metaphorical senses through a decoding algorithm. Stowe et al. (2021) obtained new metaphorical sentences by replacing relevant verbs in literal expressions and encoding conceptual mappings (FrameNetbased embeddings - *CM-LEX*, and a custom seq-to-seq model - *CM-BART*) between cognitive domains. Brooks and Youssef (2020) trained an unsupervised LSTM model and used an inherent inference engine to create new metaphors. The novelty of these new metaphors is ensured by checking that none of the generated sentences match the training data, and that the identified syntactic patterns of metaphors were not present in the non-metaphorical data.

MDG does not focus only on verbs nor does employ language-specific syntactic patterns. It does not depend on models that need to be trained with pairs of metaphorical and literal sentences and it does not need any external system, such as *COMET* in Chakrabarty et al. (2021). Furthermore, it fully relies on publicly available Transformers-based language models, that do not require particular customizations, other than being fine-tuned on the right data. The similarities of MDG with the above mentioned studies comprise masked language modeling (Song et al., 2019), which is also employed by Chakrabarty et al. (2021), and reconstruction, used to identify specific words inside the sentences and to replace them with alternative ones, turning them into metaphors.

Further information regarding recent advances and approaches in metaphor detection, processing and generation, can be found in Tong et al. (2021).

3 The MDG Framework

MDG consists of different steps, from metaphor detection to metaphor generation. Figure 2 depicts the proposed workflow. *Metaphorical sentence classification* is the task of classifying a sentence as metaphorical or literal. We train text classifiers on datasets comprising metaphorical and literal sentences. The classifier yields a probability from 0 (literal) to 1 (metaphorical) and only the correctly-predicted metaphorical sentences (true positives) are passed to the next step of the framework. *Reconstruction*, which follows, comprises two procedures: *extraction*, for the detection of the location of the metaphor within a metaphorical sentence, and *masked metaphor modeling* for the prediction of masked metaphors within metaphorical sentences. After fine-tuning a masked language model for the task, by masking metaphors, we can then apply it on literal, instead of metaphorical sentences, in order to turn a literal sentence into a metaphorical one.

4 Empirical Evaluation

4.1 Metaphor Datasets

The three most common datasets used for tasks related to metaphoricity are MOH-X, TroFi, and TroFi-X (Table 2). We describe each one below.

MOH-X (Mohammad et al., 2016) is derived from the subset of the MOH dataset that was used by Shutova et al. (2016). Mohammad et al. (2016) annotated different verbs for metaphoricity. They extracted verbs that had between three and ten senses in WordNet (Mao et al., 2018) along with their glosses. The verbs were annotated for metaphoricity with the help of crowd-sourcing. Ten annotators were recruited to assess each sentence, and only those verbs that were annotated as positive

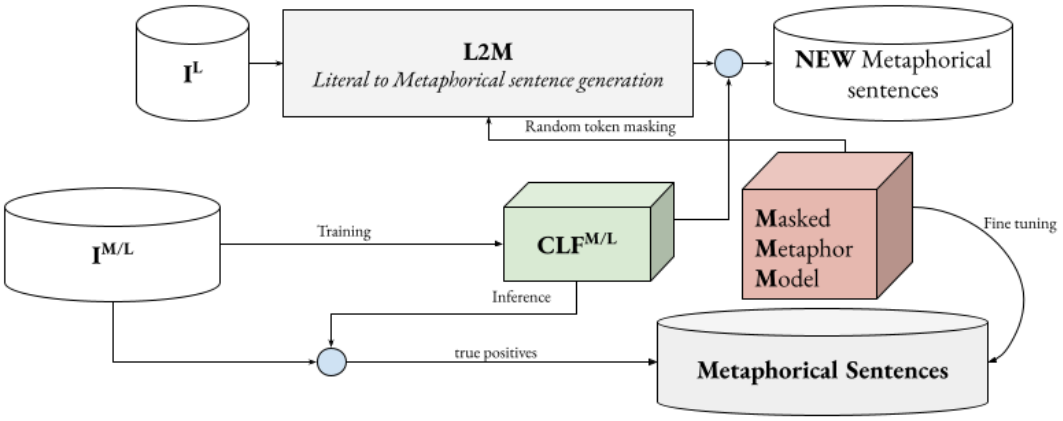

Figure 2: Visual representation of MDG. The input consists of a set of unlabelled sentences (I^L) and a set of sentences that are labelled as metaphorical or literal ($I^{M/L}$). The latter is used to train a classifier ($CLF^{M/L}$), which, after being fine-tuned on masked metaphorical-token modeling (MMM), is used to substitute randomly masked tokens of the I^L sentences in order to turn them into metaphorical. $CLF^{M/L}$ filters out any sentences that failed to become metaphorical and the remaining ones are returned by the system.

for metaphoricity by at least 70% of the annotators were selected in the end. The final dataset consisted of 647 verb-noun pairs: 316 metaphorical, and 331 literal.

TroFi contains feature lists consisting of the stemmed nouns and verbs in a sentence, with target or seed words. It is named after TroFi (Trope Finder), a nearly unsupervised clustering method for separating literal and non-literal usages of verbs (Birke and Sarkar, 2006). For example, given the target verb *pour*, TroFi is able to cluster the sentence *Custom demands that cognac be poured from a freshly opened bottle* as literal, and the sentence *Salsavand rap music pour out of the windows* as nonliteral. The target set is built using the '88-'89 Wall Street Journal Corpus [1] tagged using the Ratnaparkhi (2002) tagger and the Bangalore and Joshi (1999) SuperTagger . The final dataset consisted of 3,737 sentences.

TroFi-X is an alternative version of **TroFi**. It contains 1,444 sentences annotated not only with metaphorical verbs, but also with metaphorical nouns, pronouns and adjectives.

5 Evaluation measures

For the classification task, we employed Accuracy (i.e. the fraction of instances that were correctly classified), Precision (i.e., the number of instances that were correctly predicted as metaphorical to the number of instances that were predicted as metaphorical), Recall (i.e., the number of instances correctly predicted as metaphorical to the number of instances that should have been predicted as metaphorical) and F1 (i.e., the harmonic mean of Precision and Recall). For the reconstruction task, we employed Accuracy (i.e., the ratio of sentences that are correctly reconstructed/generated).

[1] https://catalog.ldc.upenn.edu/LDC2000T43

Table 2: Statistics of all the datasets employed in this work. All datasets comprise English sentences. Size is measured in sentences and POS shows the part of speech of the metaphor.

Name	Size	POS
MOH-X	646	Noun/Verb
TROFI	3,737	Verb
TROFI-X	1,444	Noun/Verb/Adjective

5.1 Methods

For the task of metaphorical sentence classification, we employed Naive Bayes (Rish, 2001), Random Forests (Fratello and Tagliaferri, 2019), KNN (Guo et al., 2003), SVM (Evgeniou and Pontil, 2001), Logistic Regression (Peng et al., 2002), MLP (Marius et al., 2009), BERT (Devlin et al., 2019), and XLM-R (Conneau et al., 2020) models.

For the task of metaphor reconstruction, we used data where the location of the metaphor in the sentence is known. We then employed BART and T5 by removing the metaphorical token (known) and asking the models to reconstruct the original sentence. Also, we employed BERT and XLM-R, by masking the metaphorical token and then performing in effect masked metaphor modeling (MMM)

	MOH-X				TROFI				TROFI-X			
	P	R	F1	Ac	P	R	F1	Ac	P	R	F1	Ac
BERT	73.91	80.55	79.45	76.92	39.82	91.21	55.44	41.98	66.50	63.08	65.58	69.17
BERT(FT)	88.11	**94.07**	**91.86**	**91.74**	72.00	64.24	67.77	72.08	70.18	69.85	70.53	74.91
XLM-R	63.53	72.22	69.33	64.62	58.58	59.46	64.94	74.60	58.98	60.00	63.93	69.66
XLM-R(FT)	**88.57**	86.11	87.32	86.15	**95.86**	**93.92**	**94.88**	**95.99**	**92.42**	**93.85**	**93.13**	**93.79**
NB	73.00	12.33	17.64	65.22	72.73	11.27	19.51	64.71	77.11	80.23	78.54	78.17
RF	61.90	81.25	70.27	65.62	67.50	38.03	48.65	69.52	71.22	87.31	78.66	79.43
KNN	65.00	81.25	72.22	68.75	63.29	70.42	66.67	73.26	78.58	81.26	80.09	80.19
SVM	66.67	37.50	48.00	59.38	69.11	76.26	72.43	74.32	77.15	65.38	70.62	77.39
LR	87.50	87.50	87.50	87.50	84.51	84.51	84.51	88.24	77.00	81.00	79.00	79.00
MLP	67.00	71.00	69.00	69.00	55.00	44.00	49.00	58.00	71.00	63.00	67.00	73.00

Table 3: All tested Classifiers and their respective results for the three metaphorical data sources. *P, R, F1 and Ac* stand respectively for Precision, Recall, F1 Score and Accuracy. Along with *BERT* and *XLM-R* (base and fine-tuned), we have, in order: *NB* - Naive Bayes; *RF* - Random Forest; *KNN* - K-nearest Neighbours; *SVM* - Support Vector Machine; *LR* - Logistic Regression; *MLP* - Multi-Layer Perceptron Neural Network.

to reconstruct the sentence.

5.2 Experimental Results

Table 3 provides the results of the metaphorical sentence classifiers (see Section 5.1) on the three metaphorical data sources (see Section 4.1). XLM-R (fine-tuned) has the best Precision in all datasets. BERT (fine-tuned) achieves the best Recall on MOH-X, leading also to the best Accuracy and F1. Overall, BERT and XLM-R (fine-tuned) yield the best results. Naive Bayes, Random Forests, KNN, SVM and MLP performed much lower. However, it is worth noting that Logistic Regression, despite its simple nature, performed surprisingly well.

Table 4 presents the accuracy in metaphor reconstruction on the metaphorical sentences that have been correctly classified as metaphorical (the green box in the middle, in Fig. 2) by the best-performing fine-tuned BERT and XLM-R (see Table 3). We employed T5 and BART, as well as two masked language models, BERT and XLM-R (Alfaro et al., 2019; Goyal et al., 2021), which have been fine-tuned by masking (known) metaphorical tokens of the metaphorical sentences. We refer to this process as Masked Metaphor Modeling (MMM; the red box on the right of Fig. 2). MMM with BERT was applied only on sentences correctly classified as metaphorical by BERT while MMM with XLM-R was applied on sentences correctly classified by XLM-R. T5 and BART were applied on both and results are shown in respective columns 4. In MOH-X, the accuracy scores for *nouns* and *verbs* show the percentage of correctly reconstructed metaphorical tokens (respectively nouns or verbs) inside the sentences, by the different reconstruction models. TroFi sentences comprise only verb metaphors while TroFi-X sentences comprise three metaphorical tokens each; the first two, *T1* and *T2*, can be any part-of-speech tokens, while *V* can only be verb metaphors.[2]

MMM with XLM-R is consistently better than that with BERT. This is true also for MOH-X, where BERT outperforms XLM-R for metaphorical sentence classification (see Table 3), which means that XLM-R is better in reconstruction. BART and T5 are also overall better when metaphorical sentence classification has been performed with XLM-R. When focusing on results obtained using XLM-R as the metaphorical sentence classifier, nouns are more accurately reconstructed by BART on MOH-X and TroFi-X (for T2). T5, which achieves a high accuracy in all datasets in verb reconstruction, is better than BART in TroFi and TroFi-X and only slightly worse in MOH-X. When comparing MMM with T5 and BART, the latter two seem to work better across MOH-X and TroFi sentences. MMM models, however, perform better on the first tokens (*T1*) of TroFi-X sentences.

6 Discussion

In the classification step of MDG, we classified sentences as metaphorical or literal. Metaphorical sentences which were correctly classified, then, were used in a reconstruction step. Here, metaphorical tokens (known in the datasets) were masked and recovered through extraction (T5, BART) and Masked Metaphor Modeling (BERT, XLM-R). In a final experiment, which we describe here, we used

[2]The following sentence taken from TroFi-X is given as an example: *Beyond that, conditions on board were so vile that " the sailor was at greater **risk eating** his **meals** aboard than fighting. "*. Here, **risk** is "token 1" (in this case, it is a noun), **meals** is "token 2" (in this case, also a noun), and **eating** is the verb token of the sentence (one of the three metaphorical tokens in each TroFi-X sentence is always a verb).

	MOH-X				TROFI		TROFI-X					
	BERT		XLM-R		BERT	XLM-R	BERT			XLM-R		
	N	V	N	V	V	V	T1	T2	V	T1	T2	V
T5	**80.65**	**93.55**	83.87	96.77	**95.38**	**96.92**	77.14	82.86	88.57	68.57	85.71	**97.14**
BART	64.67	90.32	**84.62**	**96.92**	**95.38**	95.38	64.62	**83.08**	**93.85**	69.73	**87.69**	95.38
MMM(ft)	71.43	48.57	77.42	45.16	74.29	83.87	**85.71**	77.78	58.33	**94.44**	86.11	66.67

Table 4: Accuracy of T5, BART, and two MMMs (BERT, XLM-R) used to reconstruct metaphorical tokens on three datasets. Only sentences classified correctly as metaphorical (by BERT and XLM-R sentence classifiers) are used. Noun (N) and verb (V) accuracy scores indicate the percentage of correctly reconstructed metaphorical nouns and verbs, respectively. TroFi-X sentences comprise three metaphorical tokens each. The first two, *T1* and *T2*, can be of any part-of-speech while *V* is always a verb. The best per column is shown in bold.

MDG to generate new metaphorical sentences, by altering literal sentences. The hypothesis is that if we mask a token from a literal sentence, the prediction of a MMM will be effectively turning the sentence to metaphorical.

As literal sentences we used 2,000 sentences scraped from Wikipedia, related respectively to music (1,000 sentences) and technology (1,000 sentences) topics; and 1,000 sentences scraped from the Gutenberg Poetry Corpus (Jacobs, 2018), which comprises 3,085,117 lines of poetry extracted from hundreds of books. We applied the fine-tuned XLM-R classifier (Table 3) on these sentences and we applied the XLM-R-based MMM (Table 4) only on the sentences that were classified as literal by the aforementioned classifier. The resulted sentences are hypothesised to be metaphorical, hence we re-apply the same XLM-R classifier and we keep only sentences that were classified as metaphorical.

Table 5 presents the ratio of originally literal sentences that have been (automatically) classified as metaphorical, after replacing a randomly selected (literal) noun, verb or adjective with a metaphorical token. Higher ratios are preferred, because they indicate a successful transfer based on the employed classifier. When the token to be replaced by the MMM was a verb, more than 50% of the literal sentences from the Gutenberg Poetry Corpus and 43% of the Wikpedia sentences related to music were turned into metaphorical ones. When the token was an adjective, the ratios dropped to 27% and 31% respectively. The lowest ratios were obtained for nouns, where 24% of the Gutenberg and 22% of the Wikipedia (related to music) sentences were transferred. Wikipedia sentences related to technology had the lowest ratios of all, achieving 29% for verbs but 8% for nouns and 7% for adjectives.[3]

[3]We note that in principle, any number of new metaphorical sentences can be generated given any positive ratio. For example, MDG can be applied on more literal sentences to counter-balance a low ratio.

Table 5: Ratio of literal sentences that were classified as metaphorical, after applying MMM on a verb, noun, or adjective per sentence. XLM-R used in both tasks.

	Nouns	Verbs	Adj.
Wikipedia - Music	0.22	0.43	**0.31**
Wikipedia - technology	0.08	0.29	0.07
Gutenberg Poetry Corpus	**0.24**	**0.56**	0.27

To perform a human evaluation of the newly constructed metaphorical sentences, we followed the work of Chakrabarty et al. (2021). In order to assess the quality of any new metaphorical sentences that are created by MDG, we compared them against human-generated ones. First, two hundred metaphorical sentences were selected: 100 were constructed with MDG, starting from sentences that originally came from both Wikipedia and Gutenberg Poetry Corpus data sources, while the other 100 were selected manually by metaphorical datasets. We then asked a linguist to evaluate each sentence. Tokens that were supposedly being used in a figurative way inside the sentences were highlighted (in bold) and sentences were shuffled before the evaluation. For each sentence, four different dimensions were evaluated: fluency, meaning, creativity, and metaphoricity. For each one of these dimensions, a score ranging from 1 (very low) to 5 (very high) had to be assigned based on her personal judgement. Example sentences that were taken from Chakrabarty et al. (2021) were provided to the annotator, in order to clarify the assignment further. Two are shown below:

1. *The scream pierced the night*. Fluency: 4, Meaning: 5, Creativity: 4, Metaphoricity: 4;

2. *The wildfire swept through the forest at an amazing speed*. Fluency: 4, Meaning: 3, Creativity: 5, Metaphoricity: 4

Table 6 shows the human-assigned average scores for both system and human -generated

Table 6: Human evaluation average result scores for system and human generated new metaphorical sentences.

Source	Avg. Fluency	Avg. Meaning	Avg. Creativity	Avg. Metaphoricity
System	**4.00**	3.65	**3.11**	**3.41**
Human	3.96	**4.27**	2.82	3.21

metaphorical sentences, for each one of the four analysed dimensions. MDG received higher scores in three out of four dimensions, namely fluency, creativity and metaphoricity. Since human-generated metaphors are not obtained from prior (e.g., literal) statements, it is reasonable that they are perceived as more meaningful than those constructed through MMM. Therefore, these results are promising and they show the overall effectiveness of our metaphor generation pipeline.

Table 7 shows the three system-generated sentences that obtained the highest-score and the respective three highest-scored human-generated ones, along with their four assigned scores. Although all the six sentences, human and system -generated, got an excellent score in fluency and meaning, MDG creates better metaphors with regards to creativity and metaphoricity. Two MDG-generated sentences out of three got an excellent creativity score with the third one obtaining a score equal to 4, while all human-generated sentences got a creativity score of 4. All three system-generated sentences got a metaphoricity score of 5, while only one of the top human-generated sentences reached this score.[4]

Improving Metaphorical Text Classification

A random sample of the new artificial metaphorical data, which have been produced by MDG starting from literal sentences, have been attached to the TroFiX training set that we used to train the metaphorical sentence XLM-R classifier.[5] We also attached the same number of randomly sampled literal sentences, leading to 428 more training sentences in total (an increase of 37%). Both the artificial metaphorical sentences and the literal ones have been extracted from Wikipedia and the Gutenberg Poetry Corpus. By fine-tuning the XLM-R metaphorical/literal sentence classifier on the increased training set, a percentage increase of all four classification metrics has been registered across TroFiX over the respective scores of Table 3: 3% up in F1 (96.12%), Precision (96.88%) and Recall (95.38%); 2.8% in Accuracy (96.55%).

Metaphor Location Detection

BERT and XLM-R can be used to successfully classify metaphorical sentences (Table 3) and to reconstruct a metaphor through Masked Metaphor Modeling (MMM), with XLM-R achieving even the best reconstruction accuracy in one case (see T1 of TroFi-X in Table 4). As discussed in Section 5.1, however, reconstruction is based on the fact that the information of the location of the metaphor is already known. This is true for the datasets that we used, but we also wanted to assess the ability of the BERT and XLM-R metaphorical sentence classifiers regarding their ability to *detect the exact location of the metaphor*.

We filtered the metaphorical sentences that were correctly classified (true positives) respectively by the fine-tuned BERT and XLM-R sentence classifiers. Then, we used the attention of the CLS token, in order to detect the location of the metaphor. In this study, we employed the fifth attention layer and the second to last (eleventh) head, since this combination yielded the best results in preliminary experiments, but we note that there are 144 possible layer-head combinations that could have also been investigated (Clark et al., 2019; Voita et al., 2019; Rogers et al., 2020). The location of the metaphor, then, is simply considered to be the token of the sentence that received the maximum attention. Table 8 provides the accuracy for this metaphor location detection task, which is the fraction of metaphorical sentences whose metaphor location was correctly detected. XLM-R is consistently better than BERT, while both models perform best in MOH-X and worse in TroFi. Three example MOH-X sentences are shown below with metaphorical tokens in bold and italics, and with XLM-R's attention heatmap in gray shade. In the first sentence, most of the attention was focused on the gold metaphorical verb. In the second, attention was on part of the gold verb

[4]The similarity between the initial literal and the new metaphorical sentences that are constructed was computed with BERTScore (Zhang et al., 2020) and was found to be very high (0.99) for all topics, probably due to the fact that only a single word had to change per sentence.

[5]We employed TroFiX for this experiment, since this dataset comprises nouns, verbs and adjectives, similarly to the new artificial data.

Table 7: The three highest-scored human (H) and system (S) -generated metaphors. The latter outperform human-generated ones on average. We show the scores in a 1-5 scale, with 1 denoting the worst and 5 the best, that were assigned to each sentence for Fluency (Fl), Meaning (Mn), Creativity (Cr) and Metaphoricity (Mt). The tokens highlighted in bold are the words that are supposedly being used in a figurative way inside the sentences.

	Metaphorical sentence (metaphor in bold)	**Fl**	**Mn**	**Cr**	**Mt**
S	Day by day his heart within him **grew** more saturated with love and longing	5	5	5	5
S	Through the green lanes of the country, where the tangled barberry-bushes **fluttered** their tufts of crimson berries	5	5	5	5
S	Love the wind among the branches, and the rain-shower and the snow-storm, and the **roaring** of great rivers	5	5	4	5
H	Headlines **scream** of pollution and dwindling natural resources	5	5	4	5
H	Musical creativity really **flowed** inside that family	5	5	4	4
H	This one scandal could very well **sink** his candidacy	5	5	4	4

Table 8: Accuracy of BERT and XLM-R for metaphor location detection across the datasets

	MOH-X	TROFI	TROFI-X
BERT	70.97	47.62	56.67
XLM-R	**77.42**	**57.41**	**63.33**

while in the third it was on the gold noun ('soup') and the (not gold) adjective on the left ('hot').

1. ***He*** ***marched*** into the classroom and announced the exam.

2. I ***wrest led*** with this ***decision*** for years.

3. A *hot* ***soup*** will ***revive*** me.

Ethical Considerations

Our models are fine-tuned on sentence level data obtained from Wikipedia. These do not contain any explicit detail leaking information about any individuals' name, health, negative financial status, racial or ethnic origin, religious or philosophical affiliation or beliefs, sexual orientation, trade union membership, alleged or actual commission of crime. Furthermore, although we use language models trained on data collected from the Web, which have been shown to have issues with bias and abusive language (Sheng et al., 2019; Wallace et al., 2019), the inductive bias of our models should limit inadvertent negative impacts. BART is a conditional language model, which provides more control of the generated output. MDG can help with the generation of metaphorical text, providing resources, for example, to creative writing practitioners. We can not imagine of any dual-use of MDG that could cause ethical problems. Our artificial data and source code are publicly released.[6]

7 Conclusion

We show that transforming literal to metaphorical sentences by using only open-source models is feasible. We propose a complete end-to-end pipeline and a framework (MDG) that tackles several applications related to figurative language, ranging from metaphorical sentence classification, to metaphor location detection, to metaphor reconstruction and generation. The obtained results show that 24%, 31% and 56% of the originally literal sentences get classified as metaphorical after masking and then reconstructing a noun, an adjective or a verb, respectively. What this means is that, potentially, MDG can be used to reach an infinite number of newly reconstructed metaphors. Most importantly, human evaluation performed on a mixed test set of system and human-generated metaphorical sentences shows that we are able to generate metaphors that are considered on average as more fluent, creative and metaphorical than figurative statements created by a real person. Finally, by using our artificial metaphors to increase the training size of a metaphorical sentence classification dataset, we show that the F1 score of an XLM-R metaphorical sentence classifier, fine-tuned on the increased dataset, is improved by 3%. The potential benefit of using a larger-scale version of our artificial dataset, in order to improve metaphorical sentence classification further, will be studied in future work.

[6] link.hidden.for.anonymity

References

Felipe Alfaro, Marta R. Costa-jussà, and José A. R. Fonollosa. 2019. BERT masked language modeling for co-reference resolution. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 76–81, Florence, Italy. Association for Computational Linguistics.

Srinivas Bangalore and Aravind Joshi. 1999. Supertagging: An approach to almost parsing. *Computational Linguistics*, 25:237–265.

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*, pages 329–336, Trento, Italy. 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, New Orleans, Louisiana. Association for Computational Linguistics.

Jennifer Brooks and Abdou Youssef. 2020. Discriminative pattern mining for natural language metaphor generation. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 4276–4283.

Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021. MERMAID: Metaphor generation with symbolism and discriminative decoding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4250–4261, Online. Association for Computational Linguistics.

Xianyang Chen, Chee Wee (Ben) Leong, Michael Flor, and Beata Beigman Klebanov. 2020. Go figure! multi-task transformer-based architecture for metaphor detection using idioms: ETS team in 2020 metaphor shared task. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 235–243, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Verna Dankers, Karan Malhotra, Gaurav Kudva, Volodymyr Medentsiy, and Ekaterina Shutova. 2020. Being neighbourly: Neural metaphor identification in discourse. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 227–234, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Theodoros Evgeniou and Massimiliano Pontil. 2001. Support vector machines: Theory and applications. In *Machine Learning and Its Applications*.

D. Fass. 1997. Processing metonymy and metaphor. In A. Lesgold and V. Patel, editors, *Contemporary Studies in Cognitive Science and Technology*, volume 1. Ablex Publishing Corporation, Greenwich.

Michele Fratello and Roberto Tagliaferri. 2019. Decision trees and random forests. In *Encyclopedia of Bioinformatics and Computational Biology*.

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*, pages 607–613, Brussels, Belgium. Association for Computational Linguistics.

Hongyu Gong, Kshitij Gupta, Akriti Jain, and Suma Bhat. 2020. IlliniMet: Illinois system for metaphor detection with contextual and linguistic information. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 146–153, Online. Association for Computational Linguistics.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 29–33, Online. Association for Computational Linguistics.

Gongde Guo, Hui Wang, David A. Bell, Yaxin Bi, and Kieran R. C. Greer. 2003. Knn model-based approach in classification. In *OTM*.

Arthur Jacobs. 2018. The gutenberg english poetry corpus: Exemplary quantitative narrative analyses. *Frontiers in Digital Humanities*, 5:5.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *EMNLP*.

George Lakoff and Mark Johnson. 1980. *Metaphors we Live by*. University of Chicago Press, Chicago.

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: Long Papers), pages 1222–1231, Melbourne, Australia. Association for Computational Linguistics.

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

Popescu Marius, Valentina Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. 2009. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8:579–588.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2018. Learned in translation: Contextualized word vectors.

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, Berlin, Germany. Association for Computational Linguistics.

Agnieszka Mykowiecka, Aleksander Wawer, and Malgorzata Marciniak. 2018. Detecting figurative word occurrences using recurrent neural networks. In *Proceedings of the Workshop on Figurative Language Processing*, pages 124–127, New Orleans, Louisiana. Association for Computational Linguistics.

Joanne Peng, Kuk Lee, and Gary Ingersoll. 2002. An introduction to logistic regression analysis and reporting. *Journal of Educational Research - J EDUC RES*, 96:3–14.

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 (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. 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 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Malay Pramanick, Ashim Gupta, and Pabitra Mitra. 2018. An LSTM-CRF based approach to token-level metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, pages 67–75, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. In *arxiv*.

Adwait Ratnaparkhi. 2002. A maximum entropy model for part-of-speech tagging. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 133–142.

Irina Rish. 2001. An empirical study of the naïve bayes classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, 3:6.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3407–3412, Hong Kong, China. Association for Computational Linguistics.

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, San Diego, California. Association for Computational Linguistics.

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

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, pages 5926–5936.

G.J. Steen, A.G. Dorst, J.B. Herrmann, A.A. Kaal, T. Krennmayr, and T. Pasma. 2010. *A method for linguistic metaphor identification. From MIP to MIPVU.* Number 14 in Converging Evidence in Language and Communication Research. John Benjamins.

Kevin Stowe, Tuhin Chakrabarty, Nanyun Peng, Smaranda Muresan, and Iryna Gurevych. 2021. Metaphor generation with conceptual mappings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6724–6736, Online. Association for Computational Linguistics.

Chuandong Su, Fumiyo Fukumoto, Xiaoxi Huang, Jiyi Li, Rongbo Wang, and Zhiqun Chen. 2020. DeepMet: A reading comprehension paradigm for token-level metaphor detection. In *Proceedings of the Second*

Workshop on Figurative Language Processing, pages 30–39, Online. Association for Computational Linguistics.

Krishnkant Swarnkar and Anil Kumar Singh. 2018. DiLSTM contrast : A deep neural network for metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, pages 115–120, New Orleans, Louisiana. Association for Computational Linguistics.

Xiaoyu Tong, Ekaterina Shutova, and Martha Lewis. 2021. Recent advances in neural metaphor processing: A linguistic, cognitive and social perspective. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4673–4686, Online. Association for Computational Linguistics.

Tim Van de Cruys. 2020. Automatic poetry generation from prosaic text. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2471–2480, Online. Association for Computational Linguistics.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing NLP. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2153–2162, Hong Kong, China. Association for Computational Linguistics.

Yuxuan Wang, Yutai Hou, Wanxiang Che, and Ting Liu. 2020. From static to dynamic word representations: a survey. *International Journal of Machine Learning and Cybernetics*, 11:1611—-1630.

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*, pages 110–114, New Orleans, Louisiana. Association for Computational Linguistics.

Zhiwei Yu and Xiaojun Wan. 2019. How to avoid sentences spelling boring? towards a neural approach to unsupervised metaphor generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 861–871, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.