University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR)

ISSN: 2714-3627

A Journal of the Faculty of Computing, University of Ibadan, Ibadan, Nigeria

Volume 16 No. 1, January 2026

journals.ui.edu.ng/uijslictr
http://uijslictr.org.ng/
uijslictr@gmail.com

University of Ibadan
Journal of Science and Logics in ICT Research

A Natural Language Processing (NLP) Model for Metaphor Detection and Interpretation: A Case Study of Use of English Passages in UTME

[1]Oladejo E. A., [2]Aboderin O. A., [3]Amire K. S. and [4]Oladejo B. F.

[1,2,3,4]*Department of Computer Science and Artificial Intelligence, University of Ibadan, Ibadan, Oyo State, Nigeria*

[1]*ezekieloladejo01@gmail.com*, [2]*aolakunleabayomi@gmail.com*, [3]*koladeamire@gmail.com*, [4]*oladejo.bolanle@dlc.ui.edu.ng*

Abstract

Metaphors play a fundamental role in language comprehension. They convey abstract concepts through vivid imagery and analogy, and can help students to understand written texts. In Natural Language Processing (NLP), the accurate detection and interpretation of metaphors pose significant challenges because of their complexity and contextual variability. This study developed an NLP model for metaphor detection and interpretation, using sentences from the 'Use of English' passages in Unified Tertiary Matriculation Examination (UTME) past questions as a case study. The approach involved training a transformer-based RoBERTa model on the Vrije Universiteit Amsterdam metaphor corpus (VUA-20), and fine-tuning it on a dataset built from UTME comprehension passages. Contextual embeddings and Word Sense Disambiguation (WSD) were used to interpret metaphorical meanings. The results showed promising performance in metaphor detection, with precision, recall, F1 score and accuracy values that indicated the effectiveness of the model on both datasets. The interpretation step also produced literal meanings for detected metaphors, which can aid language comprehension in an educational context. The study confirmed that transformer-based NLP models can be adapted to specific domains for metaphor detection and analysis.

Keywords: *Metaphors, Natural Language Processing, RoBERTa model, Transformer-based model, Contextual embeddings, Word sense disambiguation*

1. Introduction

According to the Oxford Learner's Dictionaries, a metaphor is a word or phrase used to describe somebody or something else in a way that is different from its normal use. In educational contexts, metaphors are important in shaping the learning process. They help to connect new information with existing knowledge, and so support learning. They are also a part of speech and communication, since they shape opinions about the world and the phenomena around it [1].

The Unified Tertiary Matriculation Examination (UTME) is the major examination used for university admissions in Nigeria. The Joint Admissions and Matriculation Board (JAMB), which organises the examination, reported in 2022 that only 378,639 of the 1,761,338 candidates who sat for the 2022 UTME scored 200 and above [2]. That figure represents only 21.51% of the candidates. Yearly reports over the years show that this is not a new occurrence.

The Use of English subject is the only compulsory subject for all UTME candidates. The format places much emphasis on language comprehension, using passages that test the analytical and inferential abilities of the candidates. In many cases, the message of a passage depends on metaphorical expressions. Failure to recognise these metaphors can result in misinterpretation of the passage, and so in incorrect responses to comprehension questions.

The detection of metaphors can provide useful information about a given text. It has been shown that, when metaphor-annotated data is limited, generating contextualised word embeddings from pre-trained transformer models captures more long-range dependencies between words than statistical methods can capture [3]. *Mao et al.* [4] applied this idea using a pre-trained RoBERTa model, a variant of BERT (Bidirectional Encoder

Oladejo, E. A., Aboderin, O. A., Amire, K. S. and Oladejo, B. F. (2026). An NLP Model for Metaphor Detection and Interpretation: A Case Study of Use of English Passages in UTME. *University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR)*, Vol. 16 No. 1, pp. 221 - 228

Representations from Transformers), to build a token-level metaphor paraphrase dataset. Deep contextualised word embeddings, which capture context and the nuances of language, have also been used to improve metaphor detection [5]. Pre-trained word embeddings such as GloVe, ELMo and BERT have each performed well on sequential metaphor recognition [4], and language model pre-training has produced significant gains in metaphor processing [3].

This study used these ideas in the context of UTME comprehension passages. The aim was to apply word embeddings and a transformer-based language model for metaphor recognition and interpretation in the passages used in the examination.

2. Related Works

Metaphors are common in natural language. They are used to convey abstract ideas and emotions by drawing parallels between concepts that are not obviously related. Unlike literal expressions, metaphors invite the reader to infer meaning from shared knowledge and experience, and they have a strong influence on both literature and everyday communication [1]. In recent years, the use of computational methods and NLP has produced interest in automatic detection and interpretation of metaphors [6]. Several studies have argued that proper handling of metaphors is needed if NLP systems are to capture the full complexity of human language [7].

Mao *et al.* [8] integrated linguistic metaphor theories into deep neural network architectures. The approach was based on the Metaphor Identification Procedure (MIP) and Selectional Preference Violation (SPV). Two models were introduced: RNN_HG, which is MIP-based, and RNN_MHCA, which is SPV-based. The two models used different neural network architectures and embeddings to capture the literal and contextual meanings of words. Reliance on specific linguistic theories and complex network architectures, however, limited the scalability of the models to different metaphor detection tasks.

Liu *et al.* [3] proposed a feature-based method that used pre-trained BERT and XLNet models to generate contextualised word embeddings. The method combined transformer-based embeddings, part-of-speech (POS) tags and a Bidirectional Long Short-Term Memory (Bi-LSTM) network for classification. The combination allowed the model to capture both complex semantic information and long-range dependencies between words. The method involved multiple steps, however, which made it harder to interpret and optimise.

Aggarwal and Singh [5] presented an end-to-end method using deep contextualised word embeddings, bidirectional LSTM and multi-head attention. The method worked directly on raw text and removed the need for hand-crafted features. Pre-training on large unannotated corpora produced contextualised word representations from models such as BERT and ELMo. Evaluations on TroFi and MOH-X showed that the BERT-based variant outperformed the ELMo variant and the existing baselines. The method, however, had difficulty with some types of metaphors, particularly personification, because of the difficulty in separating people from objects without extra contextual information such as POS or named-entity tags.

Mao *et al*. [4] treated metaphor detection as a sequence tagging task. The study showed that GloVe, ELMo and BERT each performed well, and that combining several embeddings was even more effective. A multi-channel CNN was combined with a BiLSTM to incorporate the embeddings and linguistic features. The model was evaluated on the VUA, MOH-X and TroFi datasets, and it outperformed the baselines in [9], [7] and [8]. The gain over the SPV model was small because of the simpler encoder, and the output-side BERT layers did not separate metaphors from literals as cleanly as expected, which suggested that further work on architecture and layer selection was needed.

Choi *et al.* [10] introduced MelBERT, a model based on MIP and SPV that extends RoBERTa. The MelBERT architecture is based on RoBERTa and is given embeddings for token, position and segment, which are passed to the transformer encoder. The model has two key layers: the MIP layer and the SPV layer. The MIP layer takes two embeddings of the target word, one contextualised and one isolated, and tries to find the difference in meaning between the two. The SPV layer takes input from the sentence encoder and captures the gap between the contextual embedding of a word and the meaning of the word in a specific context. The model focuses on the semantic gap between target words and their contexts, and so may not capture metaphors that operate at multiple levels of meaning.

Babieno *et al.* [11] built on the MelBERT architecture of Choi et al. [10]. The model also uses RoBERTa and incorporates SPV and MIP, but adds a new input feature: the dictionary definition of the target word. The use of target-word definitions in training, however, may limit how well the model generalises to tasks that do not provide such annotations.

Li *et al*. [12] introduced FrameBERT, a BERT-based model for conceptual metaphor detection. FrameNet embeddings were introduced into the training, which the authors stated was the first such use in deep learning for metaphor identification [12]. The design also uses MIP and SPV. FrameBERT has two components: a sentence encoder that uses RoBERTa to produce contextualised embeddings for each word in an input sequence, and a concept encoder that fine-tunes a RoBERTa model on FrameNet to classify frame labels. The hidden states from the two encoders are then used to predict the frame of the target word. Frame Elements (FEs) and Lexical Units (LUs) from FrameNet were not explored in the work, although they may add useful information for metaphor detection.

3. Methodology

The Machine Learning Development Life Cycle (MLDLC) guided the development of the metaphor detection model. The key stages of the cycle, namely Data Collection, Data Preprocessing, Model Design, Model Training and Model Evaluation, are shown in Figure 1. The pipeline used in this work is illustrated in Figure 2.

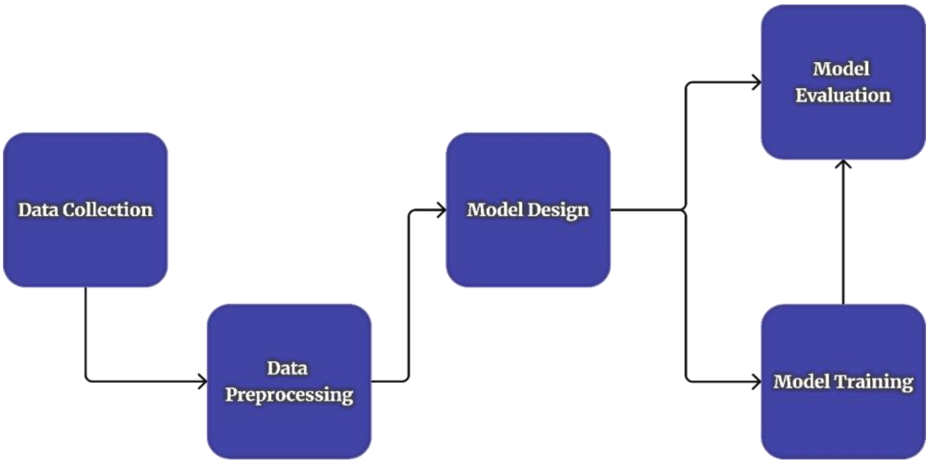

Figure 1: Machine Learning Development Life Cycle (MLDLC)

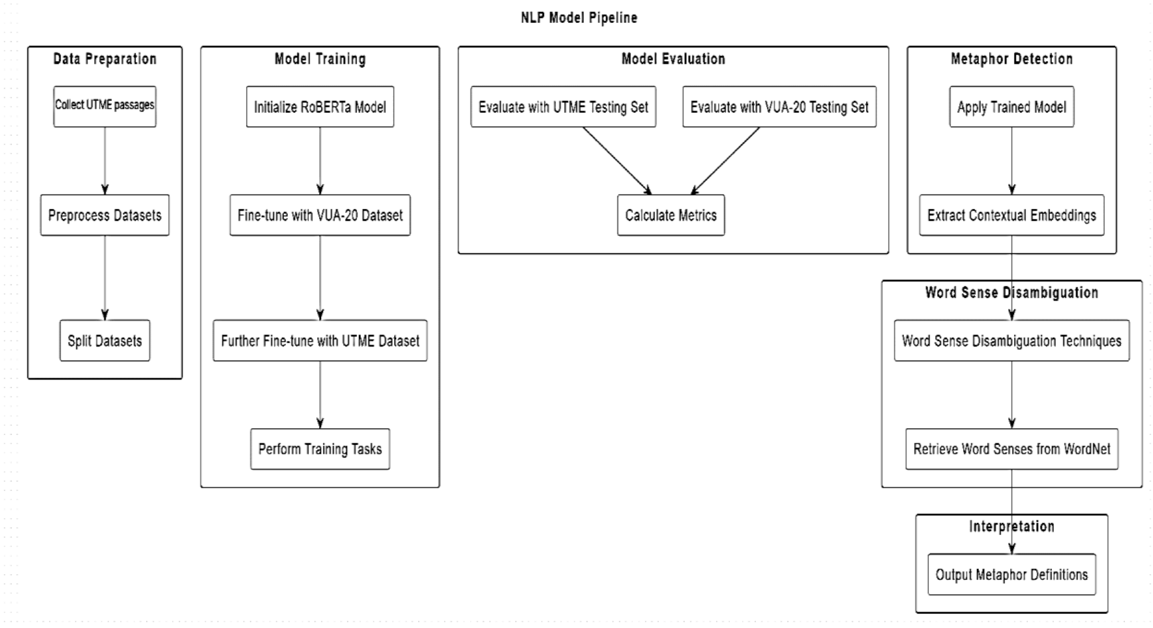

Figure 2: Metaphor Detection Model Methodology Framework

3.1 Data Collection

At the data collection stage, passages from UTME ‘Use of English’ past questions were obtained from publicly available question banks on the internet. The open-source Vrije Universiteit Amsterdam (VUA-20) metaphor corpus was also obtained from the Hugging Face platform.

3.2 Data Preprocessing

In the preprocessing stage, the UTME passages were cleaned and standardised so that the data could be fed into the model. The NLP techniques applied at this stage included tokenisation, removal of stop words, indexing and Part-of-Speech (POS) tagging. The dataset was then annotated using the Metaphor Identification Procedure (MIP) to indicate which words were used metaphorically and which were used literally. POS tagging was carried out with NLTK, a pre-trained POS tagging library, to assign grammatical categories such as noun, verb and adjective to each word in a sentence. The annotated data was then transformed into a format suitable for input into the machine learning model.

3.3 Model Design and Training

The metaphor detection model was based on the Robustly Optimised BERT Pre-training Approach (RoBERTa), which is a transformer-based model. The pre-trained model, named ‘roberta-base’, was obtained from the Hugging Face library and initialised with the appropriate configuration. Box 1 outlines the pseudocode for training the model on the VUA-20 dataset.

Box 1. Pseudocode for training the RoBERTa model on the VUA-20 dataset

```
1. Load pre-trained RoBERTa model
2. Load VUA dataset (metaphor-annotated)
3. Tokenise text with RoBERTa tokenizer
4. Convert tokens to input IDs
5. Prepare metaphor labels per token
6. Split into train / validation / test
7. Define batch size, epochs, optimiser
8. For each epoch:
  a. Init epoch metrics (loss, acc)
  b. Set model to training mode
  c. For each training batch:
    i.   Forward pass
    ii.  Compute loss
    iii. Backward pass: compute
         gradients, update params
    iv.  Update epoch metrics
  d. Set model to evaluation mode
  e. Evaluate on validation set
  f. Print epoch number and metrics
9. Evaluate trained model on test set
10. Save model parameters to disk
```

The processed UTME dataset was then used to fine-tune the model from the previous stage. The pre-trained model was initialised with the same architecture and hyperparameters as those used for pre-training on the large corpus. The pre-trained weights were loaded into the model, as shown in Box 2, and training was carried out using the UTME dataset.

Box 2. Pseudocode for fine-tuning the RoBERTa model on the UTME dataset

```
LOAD pretrained RoBERTa model
DEFINE optimiser with learning rate
DEFINE loss function
DEFINE number of epochs
FOR each EPOCH in range(num_epochs):
   SET model to training mode
   INITIALISE epoch metrics
   FOR each BATCH in utme_train_dataloader:
      EXTRACT input IDs and labels
      CLEAR gradients
      PERFORM forward pass
      COMPUTE loss
      PERFORM backward pass
      UPDATE parameters
      UPDATE epoch loss
   COMPUTE average epoch loss
   PRINT average epoch loss
SAVE fine-tuned model
```

3.4 Metaphor Interpretation

To determine the interpretation of metaphorical words in a list of sentences, the trained model was initialised and the list was fed into it. The model returned the contextual embeddings for each word, which captured the semantic and syntactic context within the passage. Words identified as metaphorical were determined based on these embeddings. For each such word, a Word Sense Disambiguation (WSD) model was applied to the embeddings to choose the most appropriate sense of the word in the given context.

4. Results and Discussion

The model was implemented using the Google Colab platform with Python (version 3.10), the PyTorch framework, Hugging Face Transformers, the Natural Language Toolkit (NLTK), WordNet, scikit-learn and NumPy.

4.1 Results

The pre-trained RoBERTa model was first trained on the VUA-20 dataset, which contains annotated examples of metaphors. Training updated the parameters of the model on the dataset so that it adapted to the metaphor detection task. Figure 3 shows a snippet of the training, including the loss computed, while Figure 4 shows the evaluation results after training was completed.

Figure 5 shows the training of the model on the UTME dataset, and Figure 6 shows the evaluation after training was completed.

```
------------------------------------
Writing new batch
Sentence 0: Care Enterprises Inc., a financially troubled nursing home operator, said
33
Prediction:    0.10185    0.89815
Label: 0    !!!WRONG!!! CORRECT LABEL = LIT, PRED = META

Sentence 1: The GAO was also to examine if the law has caused excessive red tape for
5
Prediction:    0.16323    0.83677
Label: 1

Sentence 2: ' Without voodoo, we would drown in our misery.'VERB
5
Prediction:    0.24272    0.75728
Label: 1

Sentence 3: Twenty- eight thousand buildings were leveled in the quake, but a few yea
28
Prediction:    0.81882    0.18118
Label: 0

Loss:0.7351021
------------------------------------
------------------------------------
```

Figure 3: Snippet of training result on the VUA-20 dataset with computed loss

```
[56]2024-04-30 07:27:05,533: device: cuda n_gpu: 1
[56]2024-04-30 07:30:01,230: ***** Running training *****
[56]2024-04-30 07:30:01,232:   Batch size = 32
[56]2024-04-30 07:30:01,236:   Num steps = 15015
[56]2024-04-30 09:41:59,940: [epoch 1] ,lr: 1.5e-05 ,tr_loss: 2429.1549820378423
[56]2024-04-30 09:42:22,063: ***** Running evaluation *****
[56]2024-04-30 09:49:26,569:   acc = 0.8187060731663363
[56]2024-04-30 09:49:26,573:   f1 = 0.5654427645788337
[56]2024-04-30 09:49:26,574:   precision = 0.4960212201591512
[56]2024-04-30 09:49:26,577:   recall = 0.6574585635359116
```

Figure 4: Result of evaluation of training on the VUA-20 dataset

```
--------------------------------------------------------------------------------
Writing new batch
Sentence 0: But human nature can be changed.
Prediction:    0.06036    0.93964
Label: 0        !!!WRONG!!! CORRECT LABEL = LIT, PRED = META
-1
Sentence 1: Then I knew there was no escape    -
Prediction:   Prediction:    0.31029    0.68971
Label: 1
6
Sentence 2: ' A strong opening in platinum coupled with a weak dollar just grabbed gold by the horns and took it on up,'he said VERB
12
Prediction:    0.05352    0.94648
Label: 1

Loss:0.1524478
----------------------------------------------------------------------------------------------------------
```

Figure 5: Snippet of training result on the UTME dataset with computed loss

```
[56]2024-04-30 10:28:19,904: ***** Running evaluation *****
[56]2024-04-30 10:28:34,434:   acc = 0.5308310991957105
[56]2024-04-30 10:28:34,440:   f1 = 0.6049661399548533
[56]2024-04-30 10:28:34,447:   precision = 0.46853146853146854
[56]2024-04-30 10:28:34,453:   recall = 0.8535031847133758
```

Figure 6: Result of evaluation of training on the UTME dataset

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Some weights of RobertaModel were not initialized from the model checkpoint at roberta-base and are n
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and i

Contextual embeddings for 'journey':
tensor([[[-0.1243,  0.1081,  0.0424,  ..., -0.0971, -0.0737, -0.0144],
         [-0.1624,  0.2089, -0.0174,  ..., -0.0835, -0.2671, -0.0092],
         [ 0.0418, -0.0209,  0.0057,  ..., -0.5793, -0.0999, -0.0070],
         [-0.0293,  0.0596,  0.1105,  ..., -0.0049, -0.2756,  0.1299]]],
       grad_fn=<NativeLayerNormBackward0>)

Metaphoric word: journey


Contextual embeddings for 'locked':
tensor([[[-0.1500,  0.1133,  0.0093,  ..., -0.0721, -0.0410, -0.0570],
         [-0.2050, -0.1447,  0.2708,  ..., -0.4023, -0.3127,  0.1740],
         [-0.1859,  0.0435,  0.1243,  ..., -0.0668, -0.1215,  0.0964]]],
       grad_fn=<NativeLayerNormBackward0>)

Metaphoric word: locked


Contextual embeddings for 'in':
tensor([[[-0.0967,  0.0860,  0.0115,  ..., -0.1303, -0.0537, -0.0421],
         [-0.1129, -0.1543,  0.3357,  ..., -0.4323, -0.1958,  0.0719],
         [-0.0085,  0.1500,  0.1908,  ..., -0.0731, -0.1803, -0.1630]]],
       grad_fn=<NativeLayerNormBackward0>)
```

Figure 7: Results of contextual embeddings extracted

For a list of sentences from UTME passages, interpretations of detected metaphors were generated as follows.

The contextual embeddings in Figure 7 captured the semantic information of the words in context, and the literal interpretations of words detected as metaphors are shown in Figure 8.

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Some weights of RobertaModel were not initialized from the model checkpoint at roberta-base and are new
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inf
Input sentence: Life is a journey
Metaphoric word: journey
Best sense: travel upon or across

Input sentence: They were locked in embrace
Metaphoric word: locked
Best sense: keep engaged

Metaphoric word: in
Best sense: currently fashionable

Input sentence: He took his approval
Metaphoric word: took
Best sense: carry out

Input sentence: Okechukwu ate back his words
Metaphoric word: back
Best sense: be in back of

Input sentence: Politics is a dirty game
Metaphoric word: dirty
Best sense: vile; despicable

Metaphoric word: game
Best sense: place a bet on
```

Figure 8: Results showing literal interpretation of words detected as metaphors

Tables 1 and 2 contain the values of the standard metrics, namely F1 score, accuracy, recall and precision, for the two datasets.

Table 1: Quantitative analysis of the metaphor detection model on the VUA-20 dataset

Evaluation Metric	Result
F1 Score	0.56
Accuracy	0.82
Precision	0.50
Recall	0.66

Table 2: Quantitative analysis of the metaphor detection model on the UTME dataset

Evaluation Metric	Result
F1 Score	0.60
Accuracy	0.53
Precision	0.47
Recall	0.85

4.2 Discussion

The model performed reasonably well in detecting metaphors on both the VUA-20 dataset and the UTME dataset. Precision, recall, F1 score and accuracy in Figures 4 and 6 indicated that the model reached moderate levels of accuracy in identifying metaphorical language. The evaluation on the VUA-20 dataset, which is a standard benchmark for metaphor detection, provided a baseline for the performance of the model and confirmed its ability to generalise across diverse metaphorical expressions. Fine-tuning on the UTME dataset adapted the model to the comprehension passages used in the 'Use of English' paper of the UTME, and the performance on the educational texts confirmed that the model can be adapted to this domain. The interpretations generated by the model also provided meaningful explanations for the detected metaphors.

5. Conclusion

This study developed a metaphor detection and interpretation model based on the RoBERTa transformer. The model identified and interpreted metaphors in textual data with a reasonable degree of accuracy. It used contextual embeddings, the Metaphor Identification Procedure (MIP) and Word Sense Disambiguation (WSD). Through fine-tuning on the VUA-20 dataset and a custom UTME dataset, the model was adapted to the comprehension passages used in the 'Use of English' paper of the UTME and gave promising results on the metaphor detection task.

There are still limitations to the work. The quality and size of the training data, the domain-specific nature of the UTME passages, the accuracy of the interpretation step, and the computational resources required, all affect the performance of the model. Future work can explore data augmentation, since it is difficult to build a sufficiently large annotated dataset for metaphor detection. Alternative fine-tuning strategies for domain-specific data are also worth investigating. A simple user interface or application can be built so that educators and language learners can use the metaphor detection model in practice. Work in these directions can improve the system and also contribute to broader work on natural language understanding.

References

[1] G. Lakoff and M. Johnson, Metaphors We Live By. Chicago, IL, USA: University of Chicago Press, 1980.

[2] A. Mojeed, "JAMB releases 2022 UTME results," Premium Times, May 14, 2022. [Online]. Available: https://www.premiumtimesng.com/news/headlines/529742-breaking-jamb-releases-2022-utme-results.html

[3] J. Liu, N. O'Hara, A. Rubin, R. Draelos, and C. Rudin, "Metaphor detection using contextual word embeddings from transformers," in Proc. 2nd Workshop on Figurative Language Processing, Online, Jul. 2020, pp. 250–255, doi: 10.18653/v1/2020.figlang-1.34.

[4] R. Mao, C. Lin, and F. Guerin, "Combining pre-trained word embeddings and linguistic features for sequential metaphor identification," arXiv preprint arXiv:2104.03285, 2021.

[5] S. Aggarwal and R. Singh, "Metaphor detection using deep contextualised word embeddings," arXiv preprint arXiv:2009.12565, 2020.

[6] E. Shutova, "Design and evaluation of metaphor processing systems," Computational Linguistics, vol. 41, no. 4, pp. 579–623, 2015.

[7] G. Gao, E. Choi, Y. Choi, and L. Zettlemoyer, "Neural metaphor detection in context," in Proc. 2018 Conf. on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, Oct.–Nov. 2018, pp. 607–613.

[8] R. Mao, C. Lin, and F. Guerin, "End-to-end sequential metaphor identification inspired by linguistic theories," in Proc. 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, Jul. 2019, pp. 3888–3898.

[9] Z. Wu, C. Shen, and A. van den Hengel, "Object detection using convolutional neural networks: A review," Computer Vision and Image Understanding, vol. 211, art. no. 103127, 2021, doi: 10.1016/j.cviu.2021.103127.

[10] M. Choi, S. Lee, E. Choi, H. Park, J. Lee, D. Lee, and J. Lee, "MelBERT: Metaphor detection via contextualised late interaction using metaphorical identification theories," in Proc. 2021 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, Jun. 2021, pp. 1763–1773.

[11] M. Babieno, M. Takeshita, D. Radisavljevic, R. Rzepka, and M. Araki, "MIss RoBERTa WiLDe: Metaphor identification using masked language model with Wiktionary lexical definitions," Applied Sciences, vol. 12, no. 4, art. no. 2081, 2022, doi: 10.3390/app12042081.

[12] Y. Li, S. Wang, C. Lin, F. Guerin, and L. Barrault, "FrameBERT: Conceptual metaphor detection with frame embedding learning," in Proc. 17th Conf. of the European Chapter of the Association for Computational Linguistics, Dubrovnik, Croatia, May 2023, pp. 1558–1563.