Final Report: Metaphor Detection

Benjamin Cassel Emmanuel Adebayo Rideep Moran Yousif Hag Ahmed
Department of Computer Science
Indiana University - Purdue University, Indianapolis
{bccassel, eadebayo, ridmoran, yohagahm}@iu.edu

Abstract

Metaphors play a crucial role in language, enriching expressions and conveying nuanced meanings. In this study, we explore the task of predicting the metaphorical usage of words, leveraging diverse machine learning models. Specifically, we employ a Random Forest Decision Tree, a one-layer neural network, and two BERT-based models to discern whether a word is used metaphorically or not. Our approach involves feature engineering and representation learning to capture subtle linguistic nuances indicative of metaphorical language. We conduct comprehensive experiments on benchmark datasets, evaluating the models' performance in terms of precision, recall, and F1-score. The results highlight the strengths and limitations of each model, shedding light on the varying effectiveness of traditional machine learning and state-of-the-art deep learning approaches for metaphor prediction. Our findings contribute valuable insights to the intersection of natural language processing and metaphor analysis. As metaphor comprehension is a challenging linguistic task, our comparative study aids in understanding the strengths and weaknesses of different models, paving the way for future advancements in metaphor identification and interpretation.

Keywords: Metaphorical, language, linguistic, Random Forest, Neural Network, BERT-based Models, Interpretation.

Introduction

This paper navigates the intricate landscape of metaphor prediction, emphasizing data preprocessing, accuracy metrics, and a diverse model ensemble—Random Forest, one-hidden-layer neural network, and two BERT-based transformer models (roBERTa and deBERTa). Specifically, the models are provided two important data points: a "target" word, for which the model must determine whether its usage is either literal or metaphorical, and a short amount of text, in which the model is given the context of the usage of the word. Data preprocessing forms the foundation, leading to a nuanced evaluation using accuracy metrics.

Data Preprocessing

In our approach, the data preprocessing phase is vitally important, and for the Random Forest Classifier, this involved the transformation of text data into a numerical format that a machine learning model can interpret. We employed the TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique to convert the text data into a feature matrix. The TF-IDF technique was chosen for its efficacy in reflecting the importance of words in relation to the dataset's entirety, thereby capturing the essence of metaphorical markers within our corpus.

Our decision to prioritize data preprocessing for the one-hidden-layer neural network finds inspiration in the paper titled "A corpus-based description of metaphorical marking patterns in scientific and popular business discourse." (1) This study extensively surveys various articles, shedding light on the distinctions in the usage of metaphorical markers—words intricately connected to metaphors. These markers span a diverse array, including explicit markers, intensifiers, hedges, downtoners, semantic metalanguage, mimetic terms, symbolism terms, superordinate terms, copular similes, clausal similes, perceptual processes, misperception processes, cognitive processes, verbal processes, orthography, modals, conditionals, and more. The exploration of these markers guides our data preprocessing approach, aligning our neural network with the nuanced intricacies of metaphorical language.

Each example underwent vectorization, encapsulating twenty-four distinct features. The initial element signifies the metaphor ID, encoded numerically. The subsequent seven elements—"road," "candle," "light," "spice," "ride," "train," "boat"—are represented in binary (1 for present, 0

for absent).

Following this, the vector incorporates eight elements denoting the frequency of selected metaphorical markers. These markers encompass orthography (!), hedges and downtoners (as), cognitive processes (believe), intensifiers (just), copular similes (like), modal and verbal processes (could), intensifiers (really), and modals (would). Closing the feature set are eight elements representing the positional mapping of eight parts of speech in the English language. This comprehensive feature vectorization captures both the presence of metaphor-related elements and the nuanced frequencies, providing a rich input for subsequent modeling.

The preprocessing phase for the BERT-based models involved the use of the tokenizer specific to each architecture - the tokenizer converts the raw text into tokens in the format necessary to be read by the transformer layers the models use to learn the context surrounding the words used. Part of this preprocessing also involved including the target word with the example text so that the model starts building context around the target word early in the training process. While both of these models have different architecture, they both rely on Byte-Level Byte-Pair Encoding as their tokenization process.

Models

The Random Forest classifier was selected for its robustness and versatility. This ensemble learning method operates by constructing a multitude of decision trees at training time and outputting the mode of the classes (classification) of the individual trees. It mitigates overfitting, a common pitfall in decision tree models, by averaging the results over a forest of independently constructed trees, leading to improved generalization and accuracy. The default hyperparameters of the Random Forest in our initial experiments were set in accordance with the scikit-learn library's standards, we chose the criterion of 'gini' for the quality of splits because it is faster to compute as it doesn't require calculating logarithmic functions, 'auto' for the number of features considered when looking for the best split, and using bootstrapping samples when building trees. The number of trees, however, was a hyperparameter we actively tuned to discern its impact on model performance. After several iterations, a forest of 250 trees was chosen as it provided a balanced trade-off between computational efficiency and predictive performance, as evidenced by a thorough analysis of accuracy, precision, recall, and F1 score metrics.

We chose the neural network with one hidden layer because of its theoretical versatility as universal function approximator as outlined in the universal approximation theorem. Its simplicity, interpretability and computational efficiency is also another reason. Most importantly, the ease of implementation helps with the constraints of the project timeline. For our first model, the feature vector - comprising twenty-four distinctive elements - was fed into a one-hidden layer neural network. This hidden layer comprised 1000 units, each activated by the sigmoid activation function. Importantly, this choice extended to not only the hidden layer but also the output layer.

The sigmoid activation function, known for its capability to model non-linear relationships, was employed across both layers. This architectural configuration aimed to capture intricate patterns within the data and facilitate the nuanced interpretation of metaphorical language. The resulting neural network leveraged these elements to generate predictions based on the provided feature set.

While the use of a versatile classifier like the Random Forest Algorithm provides a fine baseline to start with, and a neural network allows us to expand on the non-linear relationships between different metaphorical markers within our data, the use of BERT-based model architecture is where we focused on using context-related information in order to make accurate predictions on the metaphorical or literal use of language.

One of the most important aspects of determining whether or not a word is a metaphor is understanding how it can be used, both literally and metaphorically, in different contexts. If a model only ever learns a word being used in a metaphorical sense, it will have trouble determining the literal uses, and vice versa. This is where context becomes important - if the model can learn how the target word is used in conjunction with other words, it can then begin to learn the difference between the metaphorical and literal meaning of the word. This is where BERT-based models shine: their ability to learn word meanings based on context of surrounding words will allow the model to learn deeper meanings behind word usage. We see their use in modern day search engines to allow for more accurate search query results.

The original BERT architecture was trained, us-

ing transformers, on two separate Natural Language Processing (NLP) tasks: Masked Language Modeling, and Next Sentence Prediction. The first objective, Masked Language Modeling, is where the model is given a sentence, with a small percent of its words masked, or hidden. From there, the model must determine what the hidden words are, using the other words in the sentence. The second objective, Next Sentence Prediction, is where a model is given two sentences, and it must determine whether or not the sentences would come sequentially. By optimizing performance in regards to these two objectives, especially Masked Language Modeling, BERT-based models can be trained to learn the context of a word in regards to all the words in the sentence the word belongs to.

The first model we tested, RoBERTa, while based on the original BERT transformer model, was trained without one of the two objectives the original model was trained on: Next Sentence Prediction. The creators of the model focused only on the Masked Language Modeling objective, however that objective was achieved in a slightly different approach: while the base BERT model used a static masking approach where only one permutation of each masked token was provided to the model, RoBERTa used a "dynamic," masking system in that the same token was fed in multiple times, with different masks applied for the model to uncover. On top of this, it was trained with larger batch sizes, for longer times, with longer input sequences to further increase its capabilities post-training. (2)

The second model, DeBERTa, is trained on the same objectives the original BERT model was trained on: Masked Language Modeling and Next Sentence Prediction. However, the architecture of the DeBERTa model is different, as the authors believe that the content of the words and their positions need to be separated from each other before being fed into the transformer blocks. They believed that separation is necessary as the model needs to learn the importance of the context of the words separate from the position of the words, and that combining them makes it difficult for the model to learn which aspect is more important. Thus, the transformer blocks learn context and relative position separately, but in order to assist with ambiguous cases, the absolute position of the wording is fed in before the final layer, the softmax layer, of the model is reached. (3)

We formulated the input as <CLS> Target Word <SEP> CONTEXT <SEP> and had interaction between the target word and context from the start, as our hypothesis was to formulate the problem as an Information retrieval problem of classification after query and content interaction. Interaction from the start can lead to better feature creation which is very difficult to create manually for a task like metaphor detection, which models can do much better.

Results

The random-forest model using 250 trees achieved an accuracy of 75.9%, with a precision of 77.9%, recall of 55.5%, and an F1 score of 53.4%.The average tree had 220 branches, a depth of 78, and 222 leaf nodes.

Number of Trees	Accuracy	Precision	Recall	F1 Score
5	77.0%	72.8%	60.4%	61.4%
10	77.3%	74.8%	60.0%	60.7%
25	75.7%	77.1%	55.0%	52.6%
100	75.0%	77.0%	55.0%	52.0%
250	75.9%	77.9%	55.5%	53.4%
2500	75.1%	75.1%	54.0%	50.8%

Table 1: Performance metrics for Random Forest models with varying numbers of trees.

The tabulated results highlight the nuanced relationship between the number of trees and the model's predictive capabilities, ultimately guiding our selection towards optimizing the classifier's performance.

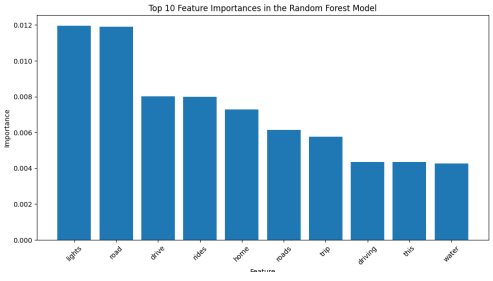

Figure 1: Feature Importance

Figure 1 provides a visualization on the importance factor for each word in the splitting algorithm of the Random Forest Classifier.

Figure 2 provides a visualization of a sample of the structure of the trees.

Figures 3 and 4 provide a visualization of the training loss over the course of 10 epochs across the data. Each one was caught by early-stopping methods, as the evaluation loss began to increase.

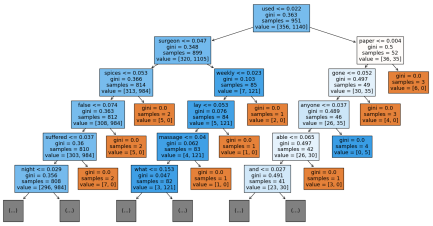

Figure 2: Tree Structure sample

Figure 3: RoBERTa Training Loss

Figure 5 shows that in the end, the BERT-based models had higher accuracy in identifying metaphors than the Random Forest and Neural Network Models, with DeBERTa having a slightly greater set of metrics than RoBERTa.

Conclusion

In this work, we presented a comprehensive approach to metaphor prediction, leveraging the strengths of diverse machine learning models: Random Forest, a one-hidden-layer neural network, and two advanced BERT-based models—RoBERTa and DeBERTa. Our journey began with meticulous data preprocessing, a critical step in transforming textual data into a form suitable for machine learning algorithms. We employed the TF-IDF vectorization technique for the Random Forest Classifier and a detailed vectorization strategy for the neural network, highlighting twenty-four distinct features. These included metaphor IDs, binary representations of selected words, frequencies of metaphorical markers, and positional mapping of parts of speech. This nuanced approach to data preprocessing was pivotal in capturing the subtleties inherent in metaphorical language.

Figure 4: DeBERTa Training Loss

	f1	accuracy	precision	recall
Random Forest	0.52570	0.756700	0.770000	0.550000
Neural Network	0.84000	0.738000	0.750000	0.950000
Roberta	0.95082	0.920635	0.953947	0.947712
Deberta	0.96129	0.936508	0.949045	0.973856

Figure 5: Metrics Comparison for Each Model

Our choice of models was strategic, each offering unique strengths. The Random Forest classifier, known for its robustness and versatility, served as a baseline, demonstrating reasonable accuracy with a carefully chosen number of trees. The one-hidden-layer neural network, with its sigmoid activation function, offered a balance between simplicity and the ability to model non-linear relationships. However, the most significant advancements were observed with the BERT-based models, RoBERTa and DeBERTa. These models excel in contextual understanding, a crucial aspect of metaphor detection. Their transformer architectures and training objectives, focusing on Masked Language Modeling and Next Sentence Prediction, allowed them to discern the nuanced usage of words in various contexts.

Our results underscored the effectiveness of these approaches. The BERT-based models outperformed the Random Forest and Neural Network in accuracy, showcasing their superior capability in handling the complexities of metaphorical language.

In conclusion, through the use of context-based learning, we can train models to achieve significantly high accuracy ratings when it comes to metaphor detection.

References

[1] Sznajder, Hanna Skorczynska, and Jordi Pique-Angordans. "A Corpus-based Description of Metaphorical Marking Patterns in Scientific and Popular Business Discourse." *English for Specific Purposes*, vol. 22, 2005, pp. 172-182.

[2] Sharma, Drishti. “A Gentle Introduction to Roberta.” Analytics Vidhya, 9 Nov. 2022, www.analyticsvidhya.com/blog/2022/10/a-gentle-introduction-to-roberta/.

[3] Uppal, Akshay. “The next Generation of Transformers: Leaving Bert behind with Deberta.” W&B, 7 Aug. 2022, wandb.ai/akshayuppal12/DeBERTa/reports/The-Next-Generation-of-Transformers-Leaving-BERT-Behind-With-DeBERTa–VmlldzoyNDM2NTk2.

The Authors

Figure 6: Yousif Hag Ahmed

Figure 7: Benjamin Cassel

Figure 8: Emmanuel Adebayo

Figure 9: Rideep Moran