A Comparative Study of Hyperbole Detection Methods: From Rule-Based Approaches through Deep Learning Models to Large Language Models

Silvia Gargova[1] **Nevena Grigorova**[1] **Ruslan Mitkov**[2]
[1]Big Data for Smart Society Institute (GATE), Bulgaria,
[2]University of Alicante, Spain
{silvia.gargova, nevena.grigorova}@gate-ai.eu
ruslan.mitkov@ua.es

Abstract

We address hyperbole detection as a binary classification task, comparing rule-based methods, fine-tuned transformers (BERT, RoBERTa), and large language models (LLMs) in zero-shot and few-shot prompting (Gemini, LLaMA). Fine-tuned transformers achieved the best overall performance, with RoBERTa attaining an F1-score of 0.82. Rule-based methods performed lower (F1 = 0.58) but remain effective in constrained linguistic contexts. LLMs showed mixed results: zero-shot performance was variable, while few-shot prompting notably improved outcomes, reaching F1-scores up to 0.79 without task-specific training data. We discuss the trade-offs between interpretability, computational cost, and data requirements across methods. Our results highlight the promise of LLMs in low-resource scenarios and suggest future work on hybrid models and broader figurative language tasks.

1 Introduction

Hyperbole, a common figure of speech that involves deliberate exaggeration, plays an important role in natural language communication by conveying emphasis, emotion, and humor. Detecting hyperbole automatically is a challenging yet valuable task for natural language processing (NLP), with applications in sentiment analysis, position detection, sarcasm recognition, and computational humor. Despite its linguistic and practical significance, hyperbole detection remains underexplored compared to related figurative language phenomena such as metaphor, irony, and sarcasm (Zhang and Wan, 2021; Troiano et al., 2018; Zhang et al., 2024).

The first systematic work on hyperbole detection was carried out by Troiano et al. (2018), who introduced HYPO, the first dataset dedicated to exaggeration detection. Their study framed hyperbole detection as a supervised binary classification problem and demonstrated that semantic features, particularly those that capture quantity and quality, two core linguistic dimensions of exaggeration, enabled traditional classifiers such as logistic regression to achieve beyond chance performance. However, these early rule-based and feature-engineered approaches, although interpretable, suffered from limited generalizability and required extensive linguistic knowledge (Chen et al., 2022; Oprea and Magdy, 2019; Eke et al., 2021).

The field progressed with the adoption of deep learning methods, motivated by the need for richer semantic representations. Early neural models such as CNNs and LSTMs provided moderate improvements Ghosh and Veale (2016); Chen et al. (2022), and Kong et al. (2020) demonstrated that deep learners could substantially outperform traditional models. Their introduction of HYPO-cn, a Chinese dataset, further expanded the scope of research, showing that LSTM-based systems combining handcrafted and embedding features achieved up to 85.4% accuracy.

A major breakthrough came with the advance of transformer-based models. Fine-tuning BERT on the HYPO dataset improved accuracy to 80% (Zhang and Wan, 2021), significantly surpassing earlier methods and confirming the effectiveness of learned contextual representations for hyperbole detection. Further refinements, such as multitask training with literal paraphrases, achieved additional gains (Biddle et al., 2021; Schneidermann et al., 2023).

More recently, research has turned towards large language models (LLMs). While LLMs such as LLaMA, BLOOM, and ChatGPT exhibit strong general-purpose language understanding, studies show that their zero-shot hyperbole detection performance is weak, reflecting an incomplete grasp of this figurative device (Badathala et al., 2023). Even when able to recognise prototypical hyperboles,

Proceedings of the First Workshop on Comparative Performance Evaluation: From Rules to Language Models associated with RANLP 2025, pages 30–38, Varna, Bulgaria, Sep 11, 2025.
https://doi.org/10.26615/978-954-452-102-8-004

LLMs struggle with cases involving overlap with metaphors or context-dependent exaggeration. To address these shortcomings, recent work explores advanced prompting techniques (Zheng et al., 2025; Xu et al., 2024) and hybrid approaches combining LLMs with human expertise, rule-based verification, or emotion-aware modules (Cohen et al., 2025; Qu et al., 2024).

In this paper, we conduct a comprehensive comparison of three distinct approaches for hyperbole detection: (1) a handcrafted rule-based system, (2) fine-tuned transformer models, and (3) prompt-based inference with LLMs in zero-shot and few-shot settings. Our evaluation on benchmark data reveals the strengths and limitations of each paradigm in terms of accuracy, computational efficiency, and generalisability. We show that while fine-tuned transformers achieve the highest performance, LLMs offer competitive results with minimal task adaptation, and rule-based methods remain viable in constrained scenarios.

The contributions of this study are as follows: (i) an empirical analysis and comparative evaluation of hyperbole detection using diverse methodologies ranging from rule-based methods through deep learning models to large language models, (ii) a thorough evaluation of prompt-based LLMs applied to this task, and (iii) insight into the strengths and limitations of each method within the task's challenging landscape.

The rest of the paper is structured as follows. Section 2 overviews related work. Section 3 details the data used in this study. Section 4 presents the experimental setup, outlining the approaches employed, while Section 5 reports the evaluation results. Section 6 offers discussion of the results. Finally, Section 7 summarises the main findings and proposes future research directions.

2 Related Work

While NLP has long studied figurative language phenomena such as metaphor, irony, and sarcasm, hyperbole detection has only recently emerged as a dedicated research topic. It was largely overlooked until Troiano et al. (2018) introduced the HYPO dataset, the first corpus of hyperbolic and literal sentences. Their work framed hyperbole detection as a supervised binary classification problem and demonstrated that handcrafted features grounded in linguistic theory—particularly quantity and quality distinctions—enabled traditional classifiers such as logistic regression to achieve up to 76% F1 score when literal paraphrases were used as negative examples.

Early approaches mainly relied on rule-based methods and lexical heuristics (Burgers et al., 2016). These methods exploited cues such as extreme adjectives, interjections, or polarity intensification (Kunneman et al., 2015) to identify exaggerations. While interpretable, such systems were brittle and lacked scalability to diverse real-world data. The release of HYPO enabled systematic experimentation with machine learning methods, establishing a foundation for subsequent research.

The next wave of studies adopted neural models, motivated by their ability to capture deeper semantic information. Ghosh and Veale (2016) explored early neural network architectures, while Kong et al. (2020) showed that deep learning approaches substantially outperformed feature-based models. Their work introduced HYPO-cn, a Chinese dataset, and demonstrated that an LSTM-based model could achieve 85.4% accuracy by integrating embeddings with handcrafted features.

Transformer-based models soon set the state of the art. Zhang and Wan (2021) reported that fine-tuning BERT on HYPO improved accuracy to 80%, a significant leap over the best traditional baseline of 72%. Biddle et al. (2021), Badathala et al. (2023) and Schneidermann et al. (2023) extended this line of research by using multitask learning and literal paraphrases as privileged information, showing that transformers could exploit more nuanced contextual signals.

More recently, researchers have evaluated LLMs such as LLaMA, BLOOM, and ChatGPT for hyperbole detection. Although these models perform well on a wide range of NLP tasks, their zero-shot performance on hyperbole classification is poor, revealing a limited understanding of exaggeration (Badathala et al., 2023). Even ChatGPT, which can correctly classify prototypical hyperboles, struggles with multi-class cases involving metaphor-hyperbole overlaps. To improve LLM performance, studies have investigated advanced prompting methods, including chain-of-thought reasoning, which helps models articulate reasoning but still fails to capture the emotional and contextual subtleties of hyperbole (Zheng et al., 2025; Xu et al., 2024).

Split	Label	Source	# sentences	Total per label	Total per split
train	0	HYPO L	1979	2917	5834
		HYPO - literal	469		
		HYPO - paraphrase	469		
		HYPO - hyperbole	469		
	1	HYPO L	767	2917	
		HYPO XL	1681		
dev	0	HYPO L	120	360	720
		HYPO - literal	120		
		HYPO - paraphrase	120		
		HYPO - hyperbole	120		
	1	HYPO L	120	360	
		HYPO XL	120		
test	0	HYPO L	120	360	720
		HYPO - literal	120		
		HYPO - paraphrase	120		
		HYPO - hyperbole	120		
	1	HYPO L	120	360	
		HYPO XL	120		
				Total:	**7274**

Table 1: Data splits.

3 Data

In this section, we describe the datasets used in our experiments, along with the procedure for splitting the data into training, development, and test sets.

3.1 Used datasets

We used three existing datasets: HYPO (Troiano et al., 2018), HYPO L, and HYPO XL [1].(Zhang and Wan, 2021)

HYPO contains 2,127 sentences, with 709 examples of hyperbole and 1,418 without. Of the non-hyperbolic sentences, 709 are literal paraphrases of the hyperbolic ones (where hyperbolic words or phrases were replaced with literal equivalents). The remaining 709 non-hyperbolic sentences feature the same phrases in their literal sense.

HYPO L consists of 3,226 sentences: 1,007 with hyperbole and 2,219 without. These sentences were first automatically annotated and then human-verified for accuracy.

HYPO XL is made up of 17,862 automatically annotated sentences, all of which contain hyperbole.

3.2 Data splits

The original datasets exhibit a high degree of class imbalance. To enable robust training and evaluation, we constructed a balanced dataset through a two-stage process.

In the first stage, we merged the HYPO and HYPO L datasets and recast the task as binary classification, assigning a label of 1 to hyperbolic sentences and 0 to non-hyperbolic ones. We then sampled additional hyperbolic instances from HYPO XL to achieve an equal number of examples for each class in the combined dataset.

In the second stage, we partitioned the data into training, development, and test sets. Both the development and test sets are perfectly balanced, containing 720 sentences each—360 hyperbolic and 360 non-hyperbolic—while also maintaining an equal distribution across the original data sources. The training set consists of the remaining 5,834 sentences, evenly split between the two classes. However, in contrast to the development and test sets, the distribution of examples across data sources in the training set is not uniform.

Table 1 summarises the size and composition of each data split.

4 Experimental setup

We frame hyperbole detection as a binary sentence classification task, where each input sentence is labeled as either hyperbolic or non-hyperbolic. In this section, we describe the model architecture,

[1] https://github.com/yunx-z/MOVER

training configuration, and evaluation methodology used in our experiments. The objective is to compare the performance of a rule-based approach with two deep learning models and two large language models for the task of hyberbole detecton.

4.1 Rule-based method

Our rule-based approach to automatic hyperbole detection integrates lexical, syntactic, and semantic cues derived from established linguistic resources and syntactic analyses. The system leverages a combination of handcrafted lexicons, pattern matching, and semantic incongruity detection to identify exaggerated language indicative of hyperbole.

4.2 Data Preprocessing and Linguistic Analysis

For linguistic analysis, input sentences are processed using Stanza POS tagging (Qi et al., 2020), which provides tokenization, part-of-speech tagging, lemmatisation, and dependency parsing. This comprehensive linguistic annotation enables precise syntactic and semantic analysis necessary for detecting subtle forms of exaggeration.

4.2.1 Lexical Resources and Hyperbole Lexicons

We curate several lexicons capturing common hyperbolic expressions across various semantic domains. To construct these lexicons, we manually reviewed the training set exclusively, deliberately excluding the development and test sets to avoid bias.

- **Quantity and Size Adjectives:** Adjectives such as *endless*, *gigantic*, and *limitless* that represent exaggerated quantities or magnitudes.
- **Intense Emotion Verbs and Adjectives:** Verbs and adjectives conveying heightened emotional states (e.g., *die*, *cry*, *terrified*, *ecstatic*), used to detect emotional overstatements.
- **Temporal Exaggerators:** Nouns and adverbs denoting exaggerated durations (e.g., *eternity*, *forever*, *centuries*).
- **Hyperbolic Idiomatic Expressions:** A predefined set of verb-object pairs known to form hyperbolic idiomatic expressions (e.g., *cry me a river*, *break heart*).

4.2.2 Rule-Based Detection of Hyperbolic Patterns

A collection of syntactic and lexical rules is applied to each processed sentence to identify potential hyperbolic cues:

1. **Exaggerated Quantity or Size:** Detection of adjectives from the quantity and size lexicons or large numeric expressions (e.g., *million*, *billion*).
2. **Unrealistic Comparisons:** Identification of comparative constructions typical of hyperbole, such as similes employing patterns like *as . . . as* or *like a*.
3. **Emotional Overstatement:** Recognition of verbs and adjectives associated with intense emotions, with special handling for frequent colloquial hyperbolic phrases (e.g., *so hungry*).
4. **Temporal Exaggeration:** Detection of temporal terms implying extreme duration.
5. **Superlative Forms:** Identification of superlative adjectives (e.g., *biggest*, *most incredible*).

The complete set of rules and the associated lexicons are provided in the Appendix (see Appendix A for the lexicons and Appendix B for the rule set)[2].

4.2.3 Semantic Incongruity Analysis

To complement surface-level rules, we incorporate semantic checks to detect incongruities frequently present in hyperbolic expressions:

- **WordNet Domain Analysis:** Utilizing the WordNet lexical database, semantic domains for verbs and nouns are extracted to assess semantic compatibility. Abstract subjects paired with concrete predicates may signal hyperbole.
- **Verb-Object Selectional Preferences:** By comparing verb domains against expected noun domains, the system flags semantically incongruous verb-object pairs (e.g., *eat horse*).
- **Idiomatic Hyperbole Pairing:** Known idiomatic hyperbolic pairs are matched directly to capture conventionalised exaggerations.

[2] Available at: https://drive.google.com/file/d/1JWRMGPyb7mWrWj0DrEV-JHUjVWge3C_P/view?usp=sharing

4.3 Fine-tuning Transformer Models

For our experiments with transformer-based models, we selected BERT and RoBERTa. For both models, we adopted the standard architecture provided by the Hugging Face Transformers library.

4.3.1 BERT model

Model Architecture We use a standard transformer-based architecture for binary sentence classification, based on the pretrained `bert-base-cased` model from the Hugging Face Transformers library. This version of BERT consists of 12 transformer layers, each with 12 self-attention heads and a hidden size of 768. The model is implemented using the `BertForSequenceClassification` class, which appends a linear classification layer on top of the [CLS] token representation to predict one of two class labels: *hyperbolic* or *non-hyperbolic*. The model is fine-tuned end-to-end on our task-specific data.

Training Configuration The model is fine-tuned using the AdamW optimiser with a learning rate of 2×10^{-5} and trained for 3 epochs with a batch size of 16. A linear learning rate scheduler without warm-up steps is used throughout training. Sentences are tokenised using the `BertTokenizer`, with all inputs truncated or padded to a maximum length of 128 tokens.

Evaluation Methodology Model performance is assessed on the test set. We report standard classification metrics, including **accuracy**, **precision**, **recall**, and **F1-score**, computed using the `scikit-learn` library. Evaluation is conducted in batches using PyTorch's `no_grad()` context to disable gradient tracking. Predicted labels are stored alongside the gold labels to support detailed error analysis.

4.3.2 RoBERTa model

Model Architecture We use the multilingual `XLM-RoBERTa base` model for our experiments, treating the task as a binary sentence classification problem. The model follows a standard transformer encoder architecture, consisting of 12 layers, each with 768 hidden units and 12 self-attention heads. On top of the transformer backbone, a classification head is added—a fully connected layer followed by a softmax layer that outputs a probability distribution over two classes: hyperbolic and non-hyperbolic.

Training Configuration The model is fine-tuned using the Hugging Face Transformers library. Input texts are tokenised using the corresponding `AutoTokenizer`, with truncation and padding applied to ensure a maximum sequence length of 128 tokens. Training is performed using the AdamW optimiser with a learning rate of 2e-5, over 3 epochs, and with a batch size of 16. A linear learning rate scheduler without warm-up steps is employed. The model is trained using the cross-entropy loss.

Evaluation Methodology We evaluate model performance on both the development and test sets using standard classification metrics: accuracy, precision, recall, and F1-score. Predictions are obtained by selecting the class with the highest softmax probability.

4.4 LLM-based methods

For the large language model (LLM) experiments, we evaluated two instruction-tuned models: **Gemini** (proprietary, accessed via API) and **LLaMA** (open-weights, accessed via the Hugging Face Transformers library). Both models were tested in *zero-shot* and *few-shot* configurations. The task required the model to predict whether a given sentence contains hyperbole, returning either `"hyperbole"` or `"not hyperbole"`.

4.4.1 Prompting Strategies

In the *zero-shot* setting, each model was given only a natural language instruction along with the input sentence, as shown below:

```
You are a helpful assistant for
    ↪ detecting hyperbole.

Classify the following text into one of
    ↪ two categories: hyperbole or not
    ↪ hyperbole.

Hyperbole is a figure of speech that
    ↪ uses extreme exaggeration to
    ↪ emphasize a point or create a
    ↪ strong impression. It is not
    ↪ meant to be taken literally and
    ↪ is often used for humor or
    ↪ dramatic effect.

Output only the predicted label (either
    ↪ hyperbole or not hyperbole) and
    ↪ nothing else.

Now classify the following text:

Text: {text}
Classification:
```

In the *few-shot* setting, the prompt included the same instruction followed by several example text–label pairs, illustrating both hyperbolic and non-hyperbolic cases. An example prompt is given below:

```
You are a helpful assistant for
    ↪ detecting hyperbole.

Classify the following text into one of
    ↪ two categories: hyperbole or not
    ↪ hyperbole.

Hyperbole is a figure of speech that
    ↪ uses extreme exaggeration to
    ↪ emphasize a point or create a
    ↪ strong impression. It is not
    ↪ meant to be taken literally and
    ↪ is often used for humor or
    ↪ dramatic effect.

Here are some examples:

{examples}

Now classify the following text:

Text: {text}
Classification:
```

The examples were selected to cover a range of syntactic and semantic structures typically associated with hyperbolic and literal expressions.

4.4.2 Inference Parameters

To ensure consistent model behaviour across conditions, we fixed the following decoding parameters:

- **Temperature:** 0 (to enforce deterministic output)
- **Max tokens:** 5 (to limit responses to concise labels)

Gemini was accessed via its official API, while LLaMA was executed locally using the Hugging Face Transformers interface with identical prompt structures and generation settings.

The outputs from both models were normalised to binary labels, with "hyperbole" mapped to the positive class (1) and "not hyperbole" to the negative class (0). Any non-standard outputs were either discarded or resolved using simple pattern matching heuristics.

4.4.3 Evaluation Protocol

All LLM outputs were evaluated against the held-out test set used consistently across all models. We computed standard classification metrics, including accuracy, precision, recall, and F1-score. This allowed for direct and fair comparison with both the rule-based baseline and the fine-tuned transformer models.

5 Results

This section presents the evaluation outcomes of the tested approaches on the classification task. We report performance metrics across different experimental setups. The results provide insights into the effectiveness and comparative strengths of each method.

The results, summarised in Table 2, show a clear performance difference across the evaluated methods.

5.1 Rule-Based method

The rule-based method was built using a set of manually designed rules based on common patterns found in hyperbolic expressions—for example, extreme adjectives, intensifiers, or emotional phrases. This system achieved an accuracy of 56%, a precision of 0.55, a recall of 0.60, and an F1-score of 0.58.

These results indicate that the rule-based system is capable of detecting certain prototypical cases of hyperbole, particularly when the language follows well-defined and recognisable patterns. However, its performance declines when faced with more subtle, context-dependent, or creatively expressed instances. This suggests that while rule-based approaches can offer interpretability and precision in constrained settings, they lack the flexibility needed to generalise across the diverse and often ambiguous forms of hyperbolic language found in natural discourse.

While the overall performance is relatively low compared to machine learning models, the rule-based system is still useful. It provides insight into which linguistic features are most important for hyperbole and serves as a transparent and interpretable baseline.

5.2 Fine-Tuned Transformer Models

Both BERT and RoBERTa performed much better than the rule-based system.

- **BERT** achieved an accuracy of 81%, with precision of 0.86 and an F1-score of 0.80. BERT tends to be cautious, favouring precision over recall. This means it is good at avoiding false

Method	Model	Accuracy	Precision	Recall	F1-score
Rule-based	-	0.56	0.55	0.60	0.58
Fine-Tuned Transformer	BERT	0.81	0.86	0.75	0.80
Fine-Tuned Transformer	**RoBERTa**	**0.82**	0.81	0.83	**0.82**
LLM zero-shot	Gemini-2.5-flash-lite	0.71	0.80	0.58	0.67
	Meta-Llama-3-8B-Instruct	0.68	0.80	0.47	0.59
LLM few-shot	Gemini-2.5-flash-lite	**0.78**	0.80	0.78	**0.79**
	Meta-Llama-3-8B-Instruct	0.74	0.68	0.88	0.77

Table 2: Results on the test set.

positives, which is helpful in situations where incorrect detection could be problematic.

- **RoBERTa** performed slightly better than BERT. It achieved the best overall results, with an accuracy of 82%, recall of 0.83, and F1-score of 0.82. RoBERTa was better at finding true cases of hyperbole (higher recall) while still keeping precision high, possibly due to its stronger pre-training.

5.3 Large Language Models (LLMs)

We also tested large instruction-tuned models, Gemini and LLaMA, in zero-shot and few-shot settings. These models were not fine-tuned on our dataset, but we gave them task instructions (and a few examples, in the few-shot setting) at inference time.

- **Gemini Zero-Shot** had moderate performance, with accuracy of 72% and F1-score of 0.67. It was highly precise (0.80) but missed many true cases (recall: 0.58), meaning it was conservative in predicting hyperbole.

- **Gemini Few-Shot** improved significantly. With just a few examples, its accuracy rose to 80%, and its F1-score reached 0.79, showing that few-shot prompting can help LLMs better understand the task.

- **LLaMA Zero-Shot** had weaker performance, with a low recall of 0.46 and F1-score of 0.60, even though precision remained high (0.80). Like Gemini, it was overly cautious.

- **LLaMA Few-Shot** improved the most in recall (0.88), meaning it detected many true hyperboles, but at the cost of lower precision (0.68) and more false positives. This suggests it became overconfident in labelling hyperbole after seeing a few examples.

6 Discussion

Among all models, **RoBERTa** achieved the highest overall performance (F1 in the low 80s), highlighting the effectiveness of fine-tuned transformer models for hyperbole detection. **BERT** and **Gemini Few-Shot** also performed competitively (F1 in the high 70s), showing that both supervised learning and few-shot prompting can yield strong results. Although the gap between fine-tuned transformers and few-shot LLMs is relatively small, it is practically meaningful: supervised transformers consistently generalise better across data splits, while few-shot LLMs offer flexible, annotation-free alternatives that trade a few points of accuracy for drastically lower requirements in labelled data.

While the rule-based method performed less well in aggregate metrics (F1 around the high 50s), it remains valuable in certain settings. One of the main challenges we faced was the difficulty of capturing the full range of hyperbolic expressions through a fixed set of handcrafted rules. Hyperbole often relies on creative, context-dependent language, which makes it hard to exhaustively define through linguistic patterns alone. As a result, the system struggled with generalisation and coverage. Nevertheless, in highly constrained domains where hyperbolic forms are stable and predictable, such systems may perform comparably to neural approaches, especially when transparency and efficiency are prioritised.

Although the zero-shot LLMs (Gemini and LLaMA) were less accurate overall (F1 in the high 50s), they show strong potential in low-resource settings. Their performance improves significantly with just a few examples, making them flexible tools for tasks where annotated data is limited or unavailable. Nevertheless, LLMs are computationally expensive to run and may produce inconsistent outputs depending on prompt design and input formulation.

Taken together, these results highlight a trade-off between performance, data requirements, and computational costs: fine-tuned transformers deliver the strongest accuracy but require labelled data; few-shot LLMs offer near-competitive results with minimal annotation; and rule-based systems, though weakest in absolute performance, provide efficiency and interpretability in specialised contexts.

7 Conclusion and Future work

This paper casts the task of hyperbole detection as a binary classification problem, comparing rule-based methods, fine-tuned transformer models, and large language models (LLMs) in both zero-shot and few-shot configurations. Our findings demonstrate that fine-tuned transformer models—particularly RoBERTa—offer the most robust performance overall, with F1 scores in the low 80s, clearly outperforming both handcrafted rule systems and prompt-based LLMs across standard evaluation metrics.

The relative performance differences are significant: while few-shot LLMs achieved F1 in the high 70s, suggesting they are competitive with fine-tuned transformers in practical terms, their advantage lies in requiring no annotated training data. By contrast, zero-shot LLMs and rule-based methods, both yielding F1 in the high 50s, lag behind in predictive accuracy but retain value in specific conditions—such as absence of labelled data, domain-specific constraints, or the need for interpretability. This performance spectrum indicates that model choice should be guided by resource availability and task requirements rather than accuracy alone.

Future work could explore hybrid approaches that combine the interpretability of rule-based systems with the generalisability of neural models. In addition, improving prompt engineering strategies and model calibration may further enhance the reliability of LLMs in zero-shot settings. Finally, expanding the task to include more nuanced figurative language phenomena, such as irony, may offer a more comprehensive understanding of exaggeration in natural language.

Acknowledgments

This work is supported by GATE project funded by the Horizon 2020 WIDESPREAD-2018-2020 TEAMING Phase 2 programme under grant agreement no. 857155, the programme "Research, Innovation and Digitalization for Smart Transformation" 2021-2027 (PRIDST) under grant agreement no. BG16RFPR002-1.014-0010-C01, and the BROD project, funded by the Digital Europe programme of the European Union under grant agreement no. 101083730.

This work has also been partially supported by the CIDEXG/2023/12 project, funded by the Generalitat Valenciana.

References

Naveen Badathala, Abisek Rajakumar Kalarani, Tejpalsingh Siledar, and P. Bhattacharyya. 2023. A match made in heaven: A multi-task framework for hyperbole and metaphor detection. *Annual Meeting of the Association for Computational Linguistics*.

Rhys Biddle, Maciek Rybinski, Qian Li, Cecile Paris, and Guandong Xu. 2021. Harnessing privileged information for hyperbole detection. In *Proceedings of the 19th Annual Workshop of the Australasian Language Technology Association*, pages 58–67, Online. Australasian Language Technology Association.

Christian Burgers, Britta C. Brugman, Kiki Y. Renardel de Lavalette, and Gerard J. Steen. 2016. Hip: A method for linguistic hyperbole identification in discourse. *Metaphor and Symbol*, 31(3):163–178.

WangQun Chen, Fuqiang Lin, Xuan Zhang, Guowei Li, and Bo Liu. 2022. Jointly learning sentimental clues and context incongruity for sarcasm detection. *IEEE Access*.

Kevin Cohen, Laura Manrique-G'omez, and Rub'en Manrique. 2025. Historical ink: Exploring large language models for irony detection in 19th-century spanish. *Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities*.

C. Eke, A. Norman, and Liyana Shuib. 2021. Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and bert model. *IEEE Access*.

Aniruddha Ghosh and T. Veale. 2016. Fracking sarcasm using neural network. *WASSA@NAACL-HLT*.

Li Kong, Chuanyi Li, Jidong Ge, B. Luo, and Vincent Ng. 2020. An empirical study of hyperbole. *Conference on Empirical Methods in Natural Language Processing*.

Florian Kunneman, Christine Liebrecht, Margot van Mulken, and Antal van den Bosch. 2015. Signaling sarcasm: From hyperbole to hashtag. *Information Processing & Management*, 51(4):500–509.

Silviu Vlad Oprea and Walid Magdy. 2019. isarcasm: A dataset of intended sarcasm. *Annual Meeting of the Association for Computational Linguistics*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Xingwei Qu, Ge Zhang, Siwei Wu, Yizhi Li, and Chenghua Lin. 2024. Overview of the nlpcc 2024 shared task on chinese metaphor generation. *Natural Language Processing and Chinese Computing*.

Nina Schneidermann, Daniel Hershcovich, and Bolette S. Pedersen. 2023. Probing for hyperbole in pre-trained language models. *Annual Meeting of the Association for Computational Linguistics*.

Enrica Troiano, C. Strapparava, Gözde Özbal, and Serra Sinem Tekiroğlu. 2018. A computational exploration of exaggeration. *Conference on Empirical Methods in Natural Language Processing*.

Yanzhi Xu, Yueying Hua, Shichen Li, and Zhongqing Wang. 2024. Exploring chain-of-thought for multimodal metaphor detection. *Annual Meeting of the Association for Computational Linguistics*.

Yazhou Zhang, Chunwang Zou, Zheng Lian, Prayag Tiwari, and Jing Qin. 2024. Sarcasmbench: Towards evaluating large language models on sarcasm understanding. *arXiv.org*.

Yunxiang Zhang and Xiaojun Wan. 2021. Mover: Mask, over-generate and rank for hyperbole generation. *North American Chapter of the Association for Computational Linguistics*.

Limin Zheng, Sihang Wang, Hao Fei, Zuquan Peng, Fei Li, Jianming Fu, Chong Teng, and Donghong Ji. 2025. Enhancing hyperbole and metaphor detection with their bidirectional dynamic interaction and emotion knowledge. *Annual Meeting of the Association for Computational Linguistics*.