A Novel Neuro-symbolic Approach to Irony Detection Based on Structural Components of Ironic Statements

Hiroshi Shigenobu[1,*], Michal Ptaszynski[1,*], Shunsuke Dan[1], Fumito Masui[1], Yuzu Uchida[2] and Rafal Rzepka[3]

[1]*Text Information Processing Laboratory, Kitami Institute of Technology, 165 Koen-cho, Kitami, 090-8507, Hokkaido, Japan*
[2]*Faculty of Engineering, Hokkai-Gakuen University, 1-1, Nishi 11-chome, Minami 26-jo, Chuo-ku, Sapporo, Hokkaido 064-0926, Japan*
[3]*Faculty of Information Science and Technology, Hokkaido University, Kita-ku, Kita 14, Nishi 9, 060-0814, Sapporo, Japan*

Abstract

This paper introduces a novel neuro-symbolic method for irony detection that offers both high accuracy and interpretability. Our two-stage approach first uses a Transformer model to translate sentences into symbolic sequences representing their core linguistic components, such as sentiment expressions and irony targets. A machine learning classifier then uses this symbolic representation for the final classification. By explicitly modeling the internal structure of ironic statements, our method outperforms strong end-to-end baselines while providing a transparent, human-readable decision process.

Keywords

Irony Detection, Neuro-symbolic AI, Explainable AI, Natural Language Processing, Computational Linguistics

1. Introduction

Automated content moderation is a critical but challenging task for Natural Language Processing (NLP) systems, largely due to complex linguistic phenomena like irony. Ironic statements often mislead standard models by conveying negative intent through superficially positive language, or vice-versa. For example, in the sentence, "That ragged dress really suits you," the insult is masked by a positive phrase, leading to common misclassification by systems that rely on surface-level sentiment.

While most existing research relies on end-to-end "black box" classifiers that sacrifice interpretability for performance, this study focuses on the internal linguistic structure of ironic statements. We propose a novel, two-stage neuro-symbolic method that first uses a neural model to identify and extract core ironic components—such as the target and clashing sentiment expressions—into a symbolic sequence. An interpretable machine learning model then uses this sequence for the final classification. This approach yields a system that is not only accurate but also provides a transparent, human-readable basis for its decisions.

The remainder of this paper is organized as follows. Section 2 reviews related work in irony detection. Section 3 describes the creation and annotation of our dataset. Section 4 details our proposed neuro-symbolic methodology. Section 5 presents our experiments and results, and Section 6 discusses the findings, limitations, and ethical considerations. Finally, Section 7 concludes the paper and outlines future work.

2. Related Work

Research in automatic irony detection has primarily followed two broad directions. Early studies mainly employed sequence-based neural models, such as recurrent neural networks (RNNs), including LSTM

The 10th Linguistic and Cognitive Approaches to Dialog Agents Workshop, January 27, 2026
*Corresponding author.
†These authors contributed equally.
m3245300092@std.kitami-it.ac.jp (H. Shigenobu); michal@mail.kitami-it.ac.jp (M. Ptaszynski); f-masui@mail.kitami-it.ac.jp (F. Masui); yuzu@hgu.jp (Y. Uchida); rzepka@ist.hokudai.ac.jp (R. Rzepka)

and GRU architectures, which demonstrated that modeling word order and contextual dependencies was effective for irony detection. These approaches established an important foundation for neural irony classification.

More recent work has increasingly adopted end-to-end supervised classification frameworks based on convolutional neural networks and Transformer architectures, in which models are trained to map raw text directly to ironic or non-ironic labels. A representative study by Chia et al. [1] established strong baselines for English by comparing various standard classifiers on Twitter data. Similar neural pipeline-based approaches have also been developed for other languages, including Japanese [2]. While these end-to-end models often achieve high performance, their decision-making processes are difficult to analyze, as predictions are produced without an explicit representation of the internal structure of ironic expressions.

A second line of research focuses on identifying linguistic features that signal irony, with particular emphasis on modeling different forms of incongruity. Some studies detect incongruity by incorporating external knowledge sources, such as Wikipedia, to identify factual contradictions [3]. Others focus on internal semantic conflicts by using lexical or sememe-based resources to capture clashes in word meanings within a sentence [4]. Another feature-based approach links irony to affective properties of language, using features such as hurtfulness as strong indicators of sarcasm [5]. Although these approaches allow researchers to relate predictions to explicit linguistic cues, they often depend on manually designed lexicons or external resources. More recent work has also extended irony detection beyond binary classification to finer-grained tasks, such as distinguishing different types of irony, including sarcasm and satire [6].

Our work is positioned between neural end-to-end approaches and feature-based methods. Similar to prior neural models, we exploit the representation learning capabilities of neural networks to capture complex patterns in text. At the same time, rather than relying solely on raw text representations, we explicitly model the internal components of ironic statements by translating them into symbolic sequences. Although the use of neural models means that full interpretability cannot be achieved, the proposed neuro-symbolic framework enables the final classification to be grounded in an explicit and human-readable structural representation. This design allows for a clearer analysis of the factors contributing to irony detection compared to conventional end-to-end neural classifiers, without relying on external knowledge bases or predefined sentiment lexicons.

Table 1
Chronological Summary of Related Works in Sarcasm and Irony Processing

Reference	Year	Language	Dataset	Task Type
[2]	2018	Japanese	Twitter	Binary Classification
[1]	2021	English	Twitter	Binary Classification,
[5]	2022	Italian	IronITA (Twitter)	Binary Classification
[4]	2022	Chinese	GuanSarcasm (News)	Binary Classification
[3]	2023	English	SemEval-2018 (Twitter)	Binary Classification
[6]	2025	Chinese	FGVIrony (News)	Fine-grained Classification

3. Dataset

Our neuro-symbolic approach requires a dataset annotated with both sentence-level irony labels and token-level structural component tags. This section details the process of creating this resource, from the initial data collection to the multi-stage annotation process.

3.1. Data Collection

The foundation of our corpus is a Japanese dataset of tweets collected by Uozumi et al. [2] This dataset consists of two parts collected during the same time period, and an overview of the sentence-level re-annotation results is presented in Table 2.

- A set of 2,700 tweets collected by searching for the explicit self-declaratory tag *hiniku* (皮肉, "sarcasm"). This served as the initial set of positive examples.
- A set of 2,700 tweets collected randomly from the same period that did not contain this tag. This subset served as the initial set of negative candidates.

For all experiments, occurrences of the sarcasm tag and its surface variants, including differences in parenthesis width, were removed from the text to ensure that the models did not learn to depend on this superficial feature.

3.2. Data Annotation

The initial keyword-based collection method was inherently noisy, as users do not always apply the "(sarcasm)" tag consistently or accurately. To create a more reliable ground truth, we conducted a two-phase manual annotation process: first at the sentence level to refine the irony labels, and second at the token level to identify the structural components of irony.

3.2.1. Phase 1: Sentence-Level Irony Annotation

To guide our annotation, we first established a working definition of irony based on a review of prior linguistic research:

> *"Irony is a statement in which the speaker uses an evaluative expression that is opposite to their true intention to either affirm or negate a target."*

Based on this definition, we manually re-evaluated all 5,400 tweets. Annotators were asked to classify each tweet into one of five categories, taking into account the amount of contextual information available within the tweet itself. The categories were defined as follows:

- **Ironic**: The tweet contains sufficient context to be unambiguously understood as ironic.
- **Probably Ironic**: The context is limited, but the phrasing strongly suggests an ironic interpretation.
- **Ambiguous**: The tweet could plausibly be interpreted as either ironic or literal.
- **Probably Not Ironic**: The context is limited, but the phrasing suggests a literal interpretation.
- **Not Ironic**: The tweet contains sufficient context to be unambiguously understood as non-ironic.

The results of this re-annotation are summarized in Table 2. A key finding was that the original keyword-based collection method was unreliable. In the dataset collected using the "(sarcasm)" tag, less than half of the tweets were confidently labeled as "Ironic" or "Probably Ironic" with the largest group being "Ambiguous" (842 tweets). Furthermore, a significant portion (25%) was classified as non-ironic. Conversely, the randomly collected dataset contained a small number of tweets (2%) that were identified as ironic. These results highlight the difficulty of irony detection and confirm the need for careful manual annotation rather than relying on noisy self-declaratory tags.

Table 2
Results of Manual Sentence-Level Irony Annotation

Category	Sarcasm Tag Corpus	Random Corpus
Ironic	595	8
Probably Ironic	588	44
Ambiguous	842	243
Probably Not Ironic	249	336
Not Ironic	426	2,069
Total	**2,700**	**2,700**

3.2.2. Phase 2: Irony Component Span Annotation

Our working definition suggests that ironic statements are constructed from core components: a **target** of the irony, the speaker's true **intention**, and an opposing **surface expression**. A preliminary annotation study confirmed these elements and identified three additional pragmatic features that frequently signal ironic intent: modifiers, honorifics, and colloquialisms. Based on this, we established a final tagset of six categories to capture the structural components of irony. The complete tagset is defined in Table 3.

In addition to these core elements, we observed during an initial exploratory analysis of the dataset that several pragmatic expressions repeatedly appeared in sentences judged to be ironic by human annotators. These observations were based on the authors' manual inspection of the data, rather than on prior theoretical claims established in the literature. Specifically, modifiers, honorific expressions, and colloquial or slang expressions were frequently present in ironic tweets and appeared to contribute to the perceived ironic intent.

Based on this combination of established theoretical insights and our own empirical observations, we defined a final tagset consisting of six categories to capture the structural components of irony. The complete tagset is summarized in Table 3. These tags are designed to represent not only sentiment polarity and its target, but also pragmatic cues that were found by the authors to systematically co-occur with ironic usage in the analyzed data.

Table 3
The Six Tags Used for Annotating the Structural Components of Irony

Tag	Definition	Description and Examples
Target	The person, object, or concept being evaluated by the speaker.	Represents the focus of the ironic statement. Can be a noun phrase, proper noun, or pronoun. *Examples: "this game," "the government," "his brilliant idea."*
Positive Exp.	An expression that conveys a positive sentiment or evaluation on the surface.	Typically adjectives, verbs, or phrases with a positive polarity. Includes positive emojis and emoticons. *Examples: "wonderful," "excellent work," "I'm so impressed," (^^) .*
Negative Exp.	An expression that conveys a negative sentiment or evaluation.	Typically adjectives, verbs, or phrases with a negative polarity. Includes insults, negative slang, and negative emojis. *Examples: "awful," "what a mess," "terrible," (T_T).*
Modifier Exp.	An expression that exaggerates, qualifies, or otherwise modifies an evaluative expression, often signaling a non-literal meaning.	Typically adverbs or determiners that intensify or alter the evaluation. *Examples: "really," "absolutely," "so-called," "in a sense."*
Honorific Exp.	The use of polite or honorific language that creates a pragmatic distance or clashes with the context.	Includes formal verb endings and honorific titles that are contextually inappropriate, often signaling insincere politeness. *Examples: -desu, -masu, -sama.*
Colloquial/ Slang	Informal, spoken-language expressions, slang, or symbols that signal a particular tone or stance.	Includes interjections, discourse particles, and internet slang that do not carry inherent sentiment but contribute to the overall tone. *Examples: "wow," "lol," "www," "!"*

For this annotation task, we used the 2,200 tweets classified in Phase 1 as "Ironic," "Probably Ironic," or "Ambiguous," totaling 93,160 characters. To ensure consistency, we created a detailed annotation guideline document[1].

We recruited 20 annotators through the Japanese crowdsourcing platform CrowdWorks. The annotators were provided with the guidelines and trained on the task. The annotation was performed using LightTag, a web-based tool well-suited for team-based token classification. Each tweet was independently annotated by two different annotators. Any disagreements in tag type or span were resolved by a third, senior annotator to produce the final ground truth.

[1]The annotation guideline (in Japanese) is available at: https://t.ly/H54nM

3.3. Final Dataset Statistics

The resulting annotated corpus forms the basis for training and evaluating our neuro-symbolic model. Table 4 presents the overall statistics for each of the six tags across the 2,200 annotated sentences. 'Positive Expression', 'Negative Expression', and 'Colloquial/Slang' are the most frequent tags, suggesting they are common components in Japanese irony on Twitter. In contrast, 'Honorific Expression' is less common, indicating it is a more specialized device. Notably, a 'Target' was identified in only 1,769 of the 2,200 sentences, meaning the target of the irony was implicit in approximately 20% of cases.

Table 4
Overall Statistics for Each Tag in the Annotated Corpus (2,200 sentences)

Tag	Count	Avg./Sentence	Avg. Length	Max Length	Min Length
Target	1,769	0.80	3.97	22	1
Positive Exp.	2,970	1.35	4.42	21	1
Negative Exp.	2,712	1.23	5.32	36	1
Modifier Exp.	1,386	0.63	3.91	18	1
Honorific Exp.	917	0.42	2.67	11	1
Colloquial/Slang	2,935	1.33	2.61	40	1

For our experiments, we divided the 2,200 annotated sentences into a training set of 1,980 sentences and a test set of 220 sentences, maintaining a 90/10 split. The distribution of tags in both the training and test sets is shown in Tables 5 and 6. The distributions are balanced, ensuring that the model is trained and evaluated on a representative sample of the data.

Table 5
Tag Distribution in the Training Set (1,980 sentences)

Tag	Count	Avg./Sentence	Avg. Length	Max Length	Min Length
Target	1,574	0.79	3.85	22	1
Positive Exp.	2,663	1.34	4.32	21	1
Negative Exp.	2,438	1.23	5.09	36	1
Modifier Exp.	1,252	0.63	3.75	18	1
Honorific Exp.	823	0.42	2.75	11	1
Colloquial/Slang	2,647	1.33	2.67	40	1

Table 6
Tag Distribution in the Test Set (220 sentences)

Tag	Count	Avg./Sentence	Avg. Length	Max Length	Min Length
Target	195	0.89	4.10	16	1
Positive Exp.	307	1.40	4.59	16	1
Negative Exp.	274	1.24	5.40	18	1
Modifier Exp.	438	1.99	3.91	14	2
Honorific Exp.	94	0.42	2.60	6	1
Colloquial/Slang	288	1.31	2.49	29	1

4. Methods

To systematically evaluate the benefit of modeling the internal structure of irony, we designed experiments to compare two distinct approaches: a standard end-to-end classification model that serves as our baseline, and our proposed two-stage neuro-symbolic method.

4.1. Baseline: End-to-End Irony Classification

Our baseline follows the standard and widely adopted paradigm for text classification. This approach uses a pre-trained Transformer-based language model, such as BERT or RoBERTa, and fine-tunes it for a sentence-level, binary classification task.

In this setup, the model is given the entire raw text of a tweet as input. This text is tokenized and passed through the Transformer's encoder layers to generate a context-aware representation, typically using the embedding of the special '[CLS]' token. A classification head, usually a single linear layer with a softmax or sigmoid activation function, is added on top of the encoder. The entire model is then fine-tuned end-to-end on our labeled dataset to predict a single binary label: 'Ironic' or 'Not Ironic'. This method is powerful because it can learn complex, non-linear relationships directly from the text, but its decision-making process is opaque, functioning as a "black box." A wide range of pre-trained Japanese models were evaluated using this approach to establish a strong performance baseline.

4.2. Proposed Neuro-Symbolic Method

In contrast to the end-to-end baseline, our proposed method is a two-stage neuro-symbolic pipeline designed to be both effective and interpretable. The core idea is to first convert the unstructured text of a sentence into a structured, symbolic representation based on its ironic components, and then perform the final classification based on this representation. The complete workflow of this method is illustrated in Figure 1.

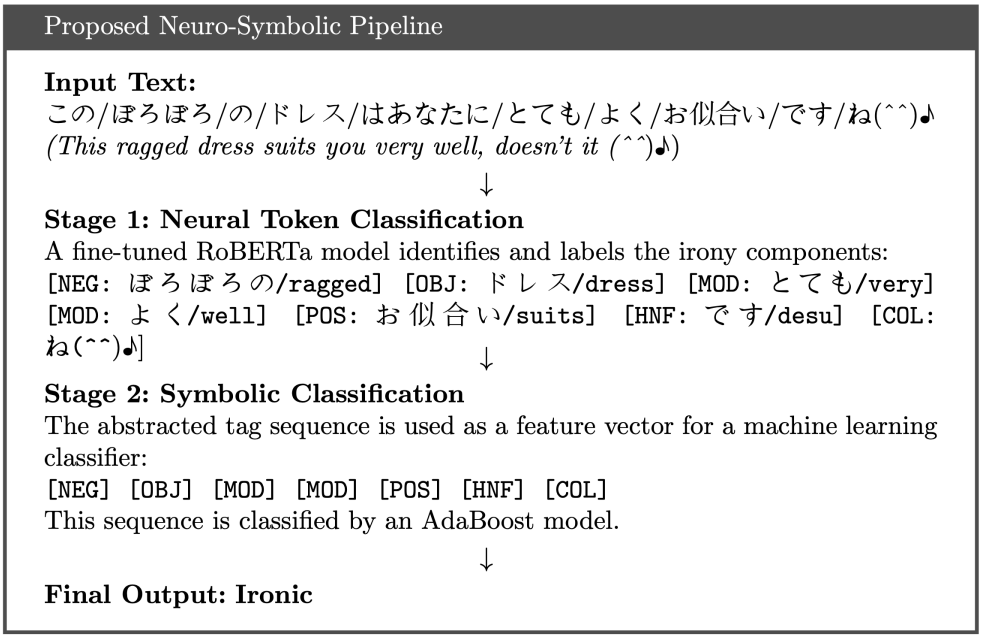

Figure 1: The two-stage workflow of the proposed neuro-symbolic method. The neural model first converts the text into a sequence of symbolic tags, which are then used by a machine learning classifier to make the final, interpretable decision. **Tag abbreviations correspond to those defined in Table 3 (e.g., NEG: Negative Exp., POS: Positive Exp., HNF: Honorific Exp.).**

4.2.1. Stage 1: Automatic Annotation of Irony Components

The first stage of our pipeline is responsible for translating the input text into a symbolic sequence. To accomplish this, we re-purpose a pre-trained Transformer model to perform a token classification task. Specifically, we fine-tune the model on our dataset described in Section 3.2.2, where each token in a sentence is labeled with one of the six irony component tags (`Target`, `Positive Exp.`, `Negative Exp.`, `Modifier Exp.`, `Honorific Exp.`, `Colloquial/Slang`) or a standard 'O' tag for tokens that do not belong to any of these categories.

The model architecture consists of the pre-trained Japanese language model followed by a linear layer that outputs a probability distribution over the possible tags for each token. By training on our manually annotated data, the model learns to identify the spans of text that correspond to each

structural component of irony. The output of this stage is the original sentence where key phrases have been annotated with our symbolic tags.

4.2.2. Stage 2: Symbolic Classification of Tag Sequences

The second stage performs the final irony classification using only the sequence of symbolic tags generated by Stage 1, completely disregarding the original words. This abstraction forces the model to base its decision on the detected linguistic structure rather than on specific lexical items.

The process involves two steps:

1. **Feature Extraction:** The sequence of predicted tags (e.g., `[NEG] [OBJ] [POS]`) is first converted into a numerical feature vector. We explore several standard text featurization techniques for this task, including Bag-of-Words (based on tag counts), TF-IDF weighting, and n-grams (e.g., unigrams, bigrams, skip-grams) to capture patterns in how the tags are ordered. For example, a bigram feature would capture the co-occurrence of a `Negative Exp.` followed by a `Positive Exp.`, a common structure in irony.
2. **Classification:** The resulting feature vector is then used as input to a traditional machine learning classifier. To identify the most suitable algorithm for this task, we conduct a comprehensive evaluation of multiple classifiers, including ensemble methods like AdaBoost and Random Forest, Support Vector Machines (SVC), and Naive Bayes models.

By decoupling the process into these two stages, this method allows for a highly interpretable final decision. The success or failure of a classification can be traced back to the specific sequence of structural components identified by the neural model in Stage 1.

5. Experiments

To evaluate our proposed neuro-symbolic method, we conducted a series of experiments designed to answer three key questions: 1. What is the performance of standard, end-to-end Transformer models on this irony detection task? This establishes a strong baseline for comparison. 2. How accurately can a neural model perform Stage 1 of our pipeline, i.e., extracting the structural components of irony? 3. Does our proposed two-stage neuro-symbolic method outperform the end-to-end baseline, and what do its internal components reveal about the structure of irony?

5.1. Experiment 1: Baseline End-to-End Classification

5.1.1. Experimental Setup

To create a clear binary classification task, we constructed a dataset from our sentence-level annotations (Section 3.2.1). Sentences labeled “Ironic” or “Probably Ironic” were combined to form the positive class, while those labeled "Not Ironic" or “Probably Not Ironic” formed the negative class. Sentences labeled “Ambiguous” were excluded from this experiment. To ensure a balanced dataset, we randomly sampled from these categories to create a training set of 2,222 texts (1,111 ironic, 1,111 not ironic) and a test set of 248 texts (124 ironic, 124 not ironic).

We evaluated 22 publicly available, pre-trained Japanese language models. Each model was fine-tuned on our training set for the binary classification task. We tested two different learning rates, 1e-4 and 1e-5, to assess the impact of this hyperparameter on performance. We report accuracy (Acc), precision (Prec), recall (Rec), and the F1-scores for the positive (ironic) and negative (not ironic) classes, as well as the macro F1-score.

5.1.2. Results and Discussion

The results are presented in Table 7 (learning rate 1e-4) and Table 8 (learning rate 1e-5).

With a learning rate of 1e-4, performance varied significantly, and several models failed to converge, resulting in F1-scores of 0. However, when the learning rate was lowered to 1e-5, the results improved across the board. The models that previously failed to learn now achieved respectable scores, and nearly all other models showed an increase in performance. This indicates that a lower learning rate of 1e-5 is more suitable for this task.

Under the optimal learning rate of 1e-5, the top-performing model was *ku-nlp/roberta-base-japanese-char-wwm*, which achieved a macro F1-score of 0.739 and an accuracy of 0.742. This model demonstrated a particularly strong ability to identify ironic statements (F1-positive score of 0.768), making it a robust and challenging baseline for our proposed method. Based on these results, we selected this model for all subsequent comparisons.

Table 7
Performance of Japanese Language Models on End-to-End Irony Classification (Learning Rate = 1e-4)

Model Name	Acc	Prec	Rec	F1-pos	F1-neg	Macro F1
trtd56/autonlp-wrime_joy_only-117396	**0.706**	0.823	0.667	0.736	0.667	**0.702**
KoichiYasuoka/deberta-base-japanese-unidic	0.698	0.742	0.681	0.710	0.684	0.697
colorfulscoop/bert-base-ja	0.694	0.758	0.671	0.712	0.672	0.692
tohoku-nlp/bert-base-japanese-v2	0.685	0.944	0.622	**0.750**	0.576	0.663
ptaszynski/yacis-electra-small-japanese-cyberbullying	0.585	0.726	0.566	0.636	0.516	0.576
tohoku-nlp/bert-base-japanese-char-v2	0.613	**0.968**	0.566	0.714	0.400	0.557
abhishek/autonlp-japanese-sentiment-59363	0.500	0.000	0.000	0.000	0.667	0.333
ku-nlp/roberta-base-japanese-char-wwm	0.500	1.000	0.500	0.667	0.000	0.333

Table 8
Performance of Japanese Language Models on End-to-End Irony Classification (Learning Rate = 1e-5)

Model Name	Acc	Prec	Rec	F1-pos	F1-neg	Macro F1
ku-nlp/roberta-base-japanese-char-wwm	**0.742**	0.697	**0.855**	**0.768**	**0.709**	**0.739**
tohoku-nlp/bert-base-japanese-char-v2	0.738	**0.887**	0.683	0.772	0.692	0.732
sonoisa/sentence-bert-base-ja-mean-tokens-v2	0.726	0.790	0.700	0.742	0.707	0.725
tohoku-nlp/bert-base-japanese	0.726	0.798	0.697	0.744	0.704	0.724
vabadeh213/autotrain-iine_classification10-737422470	0.722	0.806	0.690	0.743	0.696	0.720
trtd56/autonlp-wrime_joy_only-117396	0.722	0.823	0.685	0.747	0.691	0.719
hiroshi-matsuda-rit/bert-base-japanese-basic-char-v2	0.710	0.871	0.659	0.750	0.654	0.702
daigo/bert-base-japanese-sentiment	0.702	0.806	0.667	0.730	0.667	0.698
colorfulscoop/bert-base-ja	0.698	0.806	0.662	0.727	0.661	0.694

5.2. Experiment 2: Performance of Irony Component Extraction

5.2.1. Experimental Setup

This experiment evaluates Stage 1 of our pipeline: the automatic annotation of irony components. We fine-tuned the top 6 performing models from Experiment 1 on our token classification dataset (Section 3.3). The task is to predict the correct tag (`Target`, `Positive Exp.`, etc.) for each token in a sentence.

We used the *seqeval* framework for evaluation, which is standard for named entity recognition and other token-level tasks. This metric computes precision, recall, and F1-score based on exact matches of both the tag category and the span of tokens for each annotated entity.

5.2.2. Results and Discussion

The overall token classification performance is shown in Table 9. The best performance was achieved by *ku-nlp/roberta-base-japanese-char-wwm* with a learning rate of 1e-4, reaching a macro F1-score of 0.616. This model is a RoBERTa-based architecture trained on Japanese text using a character-level tokenization scheme combined with whole-word masking, which allows it to robustly handle the lack of explicit word boundaries in Japanese. This result confirms that modern Transformer architectures are capable of learning to identify these abstract, functional components of irony with reasonable accuracy.

Table 9
Performance of Language Models on the Irony Component Token Classification Task

Model Name	Precision	Recall	F1-score
Learning Rate: 1e-4			
ku-nlp/roberta-base-japanese-char-wwm	**0.581**	**0.655**	**0.616**
trtd56/autonlp-wrime_joy_only-117396	0.570	0.636	0.602
tohoku-nlp/bert-base-japanese-whole-word-masking	0.567	0.640	0.601
Learning Rate: 1e-5			
ku-nlp/roberta-base-japanese-char-wwm	0.548	0.649	0.594
vabadeh213/autotrain-iine_classification10-737422470	0.504	0.632	0.561
tohoku-nlp/bert-base-japanese-whole-word-masking	0.507	0.622	0.560

Table 10 provides a breakdown of the F1-scores for each individual tag. The model performed best on 'Honorific Exp.' (0.598) and 'Colloquial/Slang' (0.584). This is likely because these categories often contain specific, predictable lexical items (e.g., formal verb endings for honorifics, "www", which is commonly used in Japanese online communication to express laughter). In contrast, performance was lowest on 'Negative Exp.' (0.306). This category is challenging because negative expressions can range from single words to long, complex phrases, making the exact span difficult to predict. The performance on 'Positive Exp.' (0.464) and 'Target' (0.424) was moderate, reflecting the similar variability of these components.

Table 10
Per-Tag F1-Scores for the Best Token Classification Model

Tag Category	Precision	Recall	F1-score
Honorific Exp.	0.524	0.696	0.598
Colloquial/Slang	0.568	0.600	0.584
Positive Exp.	0.441	0.490	0.464
Modifier Exp.	0.369	0.585	0.452
Target	0.364	0.506	0.424
Negative Exp.	0.251	0.393	0.306

5.3. Experiment 3: Performance of the Neuro-Symbolic Method

This final set of experiments evaluates the full two-stage pipeline and compares it to the baseline.

5.3.1. Analysis of Symbolic Features

First, to understand which structural patterns are most indicative of irony, we analyzed the predictive power of different n-gram features extracted from the ground-truth tag sequences. We used a logistic regression model to measure the contribution of each n-gram type. Table 11 shows the results.

The 2-gram analysis reveals that the sequence 'NEG_POS' (a negative expression followed by a positive one) is the single most predictive feature of irony. This provides strong empirical evidence for the classic linguistic theory of irony as a clash of sentiments. Other important bigrams like 'MOD_COL' and 'HNF_COL' show that the interplay between modifiers, politeness, and informal language is also a key structural signal. The 3-gram 'POS_HNF_COL' further reinforces this, indicating that a positive statement made politely but ending with a colloquialism is a powerful ironic pattern.

5.3.2. Selection of the Symbolic Classifier

Next, we evaluated 27 different machine learning classifiers from the Scikit-learn library for Stage 2 of our pipeline. Each classifier was trained on TF-IDF features derived from the tag sequences predicted by our best Stage 1 model. The goal was to find the most effective algorithm for classifying these symbolic representations.

Table 11
Most Predictive N-gram Tag Sequences for Irony Classification

2-grams		3-grams		Skip-grams (w=2)	
Feature	**Coefficient**	**Feature**	**Coefficient**	**Feature**	**Coefficient**
NEG_POS	0.759	POS_HNF_COL	1.617	MOD_COL	0.550
MOD_COL	0.676	POS_OBJ_POS	1.497	NEG_COL	0.468
HNF_COL	0.634	HNF_HNF_NEG	1.408	NEG_HNF	0.458
MOD_HNF	0.592	OBJ_OBJ_COL	1.341	POS_HNF	0.418
POS_HNF	0.543	OBJ_NEG_POS	1.238	MOD_HNF	0.386

As shown in Table 12, the AdaBoostClassifier achieved the highest weighted F1-score (0.77), closely followed by Bernoulli Naive Bayes (0.75). The strong performance of AdaBoost, an ensemble method, suggests it is well-suited to capturing the complex, non-linear interactions between the different irony components. Based on this, we selected AdaBoost as the default classifier for Stage 2.

Table 12
Comparison of Classifiers for Stage 2 (Symbolic Classification)

Model	Weighted F1	Model	Weighted F1
AdaBoostClassifier	**0.77**	SGDClassifier	0.71
BernoulliNB	0.75	RandomForestClassifier	0.68
LinearSVC	0.73	ExtraTreesClassifier	0.65
LogisticRegression	0.73	DecisionTreeClassifier	0.63
GradientBoostingClassifier	0.71	KNeighborsClassifier	0.57

5.3.3. Final Performance Comparison

Finally, we compared the performance of three models on the test set: 1. RoBERTa (Baseline): The end-to-end *ku-nlp/roberta-base-japanese-char-wwm* model from Experiment 1. 2. AdaBoost (Symbolic-Only): The AdaBoost classifier trained on the tag sequences predicted by the Stage 1 RoBERTa model. 3. Proposed Method (Hybrid): A hybrid model that uses the AdaBoost prediction for sentences where at least one irony tag is detected. If the Stage 1 model predicts no tags, it falls back to using the prediction from the end-to-end RoBERTa baseline. This ensures that the model can handle both structurally explicit irony and more subtle, contextual cases.

The results are summarized in Table 13. Our Proposed Hybrid Method achieved the best performance across all metrics, with an accuracy of 0.7863 and a macro F1-score of 0.7829. This represents a substantial improvement of over 4.4 percentage points in both accuracy and F1-score compared to the strong RoBERTa baseline. The symbolic-only AdaBoost model also performed competitively, slightly outperforming the baseline, which demonstrates the power of the structural features alone.

To assess the statistical significance of these improvements, we conducted a McNemar's test. The comparison between the Proposed Method and the RoBERTa baseline yielded a p-value of 0.267. While this does not meet the conventional threshold for statistical significance ($\alpha = 0.05$), likely due to the limited size of our test set (n=248), the magnitude of the performance gain is practically meaningful. The results strongly suggest that explicitly modeling the structural components of irony provides a significant advantage over standard end-to-end approaches.

Table 13
Final Performance Comparison and Statistical Significance Testing (n=248)

Model	Performance Metrics				McNemar's Test vs. Baseline		
	Prec.	**Recall**	**Macro F1**	**Acc.**	χ^2	p	**Sig.**
RoBERTa (Baseline)	0.755	0.742	0.739	0.742	–	–	–
AdaBoost (Symbolic)	0.757	0.750	0.748	0.748	0.012	0.914	ns
Proposed (Hybrid)	**0.805**	**0.786**	**0.783**	**0.786**	1.235	0.267	ns

Note: Bold indicates the best performance. 'ns' indicates the result is not statistically significant ($\alpha = 0.05$).

6. Discussion

This study introduced and evaluated a neuro-symbolic method for irony detection, comparing it against standard end-to-end models. This section interprets the experimental findings, acknowledges the limitations of the work, and discusses the ethical implications of this research.

6.1. Interpretation of Results

Our experiments showed that explicitly modeling the internal structure of irony leads to a more accurate and interpretable system. The proposed hybrid neuro-symbolic method significantly outperformed a strong RoBERTa baseline, and the analysis of its symbolic features validated core linguistic theories. For instance, the high predictive power of the NEG_POS bigram confirmed that a clash of opposing sentiments is a key structural marker of irony. The success of the final hybrid model highlights the complementary strengths of its components: the symbolic classifier excels at identifying irony with clear structural cues, while the end-to-end neural model, used as a fallback, captures more subtle cases that depend on deeper contextual understanding. This synergy creates a more robust and comprehensive detection system.

6.2. Study Limitations

Our findings are promising, but the study has limitations. The primary challenge is the difficulty of the token-level component extraction task. Performance was constrained by the variable length of expressions like Negative Expression and a strict, exact-match evaluation metric that may underestimate the model's practical ability. This difficulty was compounded by the nature of our dataset, which consists of short, context-dependent texts from Japanese Twitter that are often inherently ambiguous. Furthermore, our annotation schema has some boundary issues, such as the occasional overlap between the Colloquial/Slang tag and sentiment-bearing expressions. These factors mean the generalizability of our current model to other domains or languages requires further investigation.

6.3. Ethical Considerations

The application of this technology, particularly in content moderation, requires careful ethical consideration. Models trained on public social media data risk developing biases that unfairly penalize the linguistic norms of specific communities. Since irony is often used for humor and social bonding, misclassifications could lead to unwarranted censorship. Therefore, any deployment of this technology should incorporate human oversight and provide clear channels for users to appeal automated decisions. The interpretability of our method is intended to assist, not replace, human judgment, as over-reliance on its structural rules could create new, rigid biases.

7. Conclusion

In this paper we introduced a novel, two-stage neuro-symbolic approach for irony detection that improves both performance and interpretability over standard end-to-end models. By first translating sentences into a symbolic representation of their core linguistic components, our method empirically validated that irony is constructed from predictable structural patterns, such as the clash of opposing sentiments. Our hybrid model, which combines this structural analysis with a powerful neural baseline, achieved a significant performance gain, confirming that explicitly modeling linguistic structure is a highly effective strategy for this task. This research provides a foundation for developing more transparent and reliable NLP systems for nuanced language understanding.

Future work will focus on three key areas. First, we will enhance the accuracy of the initial component extraction stage by refining our annotation schema and expanding the training dataset. Second, we will test the generalizability of our method on different domains and adapt it for other languages. Finally,

we plan to explore more sophisticated strategies for integrating the symbolic and neural components of our hybrid model to further improve its performance.

Declaration on Generative AI

During the preparation of this work, the authors used Gemini 2.5 Pro in order to correct grammar and spelling.

References

[1] Z. L. Chia, M. Ptaszynski, F. Masui, G. Leliwa, M. Wroczynski, Machine learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection, volume 58, Elsevier, 2021, p. 102600. doi:10.1016/j.ipm.2021.102600.

[2] Y. Uozumi, Y. Uchida, *Hiniku kenshutsu ni okeru kanjō seiki yōin no yūkōsei* [effectiveness of emotional factors in sarcasm detection], The 17th Forum on Information Science and Technology, Volume 2 (2018) 163.

[3] Y. Ren, Z. Wang, Q. Peng, D. Ji, A knowledge-augmented neural network model for sarcasm detection, Information Processing & Management 60 (2023) 103521. doi:10.1016/j.ipm.2023.103521.

[4] Z. Wen, L. Gui, Q. Wang, M. Guo, X. Yu, J. Du, R. Xu, Sememe knowledge and auxiliary information enhanced approach for sarcasm detection, Information Processing & Management 59 (2022) 102883. doi:10.1016/j.ipm.2022.102883.

[5] S. Frenda, A. T. Cignarella, V. Basile, C. Bosco, V. Patti, P. Rosso, The unbearable hurtfulness of sarcasm, Expert Systems with Applications 193 (2022) 116398. doi:10.1016/j.eswa.2021.116398.

[6] Z. Wen, R. Wang, Q. Wang, L. Gui, Y. Long, S. Chen, B. Liang, M. Yang, R. Xu, FGVIrony: A Chinese dataset of fine-grained verbal irony, Information Processing & Management 62 (2025) 104169. doi:10.1016/j.ipm.2025.104169.