Building Context from Sarcasm

Sandy Thomas
*California
Baptist
University*

sandy.thomas@calbaptist.edu

Abstract

Computational pragmatics aims to bridge the gap between linguistic meaning and contextual understanding in machine learning systems. This paper situates sarcasm detection as one of the core tasks in this field and examines approaches using supervised learning to detect pragmatic incongruence in text. Using a labeled dataset that includes both sarcastic and non-sarcastic social media comments, I compare the performance of logistic regression, support vector machines, and transformer-based architectures in conveying contextual irony and implicit sentiment polarity reversal. Feature engineering encompasses lexical, syntactic, and contextual features, including the contrast of sentiment polarities, punctuation frequency, and semantic incongruity, utilizing embeddings. Results indicate that transformer-based models, with a 12% margin in F1-score, outperform traditional classifiers; hence, contextual embeddings capture pragmatic subtleties more effectively. This research contributes to the growing body of computational pragmatics by providing empirical evidence on the role of context modeling in recognizing sarcasm and by presenting a reproducible framework for pragmatic language analysis.

Introduction

Understanding language beyond its literal meaning is a central challenge in artificial intelligence. Computational pragmatics, a subfield of natural language processing, studies how context, tone, and speaker intent affect interpretation. One of the most salient ways in which this pragmatic complexity arises is sarcasm; those utterances whose intended meaning radically differ from surface meaning. Despite its prevalence in human communication, sarcasm remains challenging for machines to recognize due to its subtlety in terms of contextual cues, cultural knowledge, and emotional tone.

This paper introduces a supervised learning approach to sarcasm prediction as one step toward improving pragmatic understanding in NLP systems. The research questions are: (1) How effectively can supervised models detect sarcasm from text based on linguistic and contextual features? (2) What kind of feature is most useful for accurate classification: lexical, semantic, or contextual? And (3) how do transformer-based models compare to classic machine learning approaches in modeling pragmatic intent? I address these questions by constructing and evaluating models trained on a labeled sarcasm dataset and systematically comparing their performance through quantitative analysis.

1. Dataset and Experimental Setup

This study uses a labeled dataset of sarcastic and non-sarcastic comments from Reddit to explore how different feature representations affect the performance of classifying sarcasm. The dataset was loaded into a pandas DataFrame, and a separate working copy was created for analysis, ensuring reproducibility and safe manipulation of the original data.

1.1 Data Preparation

Text preprocessing was done following standard NLP cleaning steps using NLTK. This included tokenization, lowercasing, stopword removal, and basic normalizing of the text of comments on Reddit. Required NLTK packages, including punkt and stopwords, were downloaded to support these preprocessing tasks. These steps ensured both traditional TF-IDF models and transformer-based models had consistent, clean input.

1.2 Feature Engineering

To assess the different linguistic representations for the task at hand, three feature configuration schemes were crafted:

TF-IDF Representation:

Sparse lexical vectors obtained using TfidfVectorizer, capturing word-frequency patterns typical of traditional machine learning workflows.

Transformer-Based Embeddings:

Dense semantic embeddings generated using a modern transformer model. These embeddings capture the contextual and pragmatic information that TF-IDF alone cannot represent.

Combined Feature Representation:

A hybrid feature set created by concatenating TF-IDF vectors and transformer embeddings. This approach was designed to test whether combining surface-level lexical cues with deeper contextual semantics enhances sarcasm detection accuracy.

1.3 Modeling Approach

Three different supervised classification models were trained:

TF-IDF model

Embedding-only model

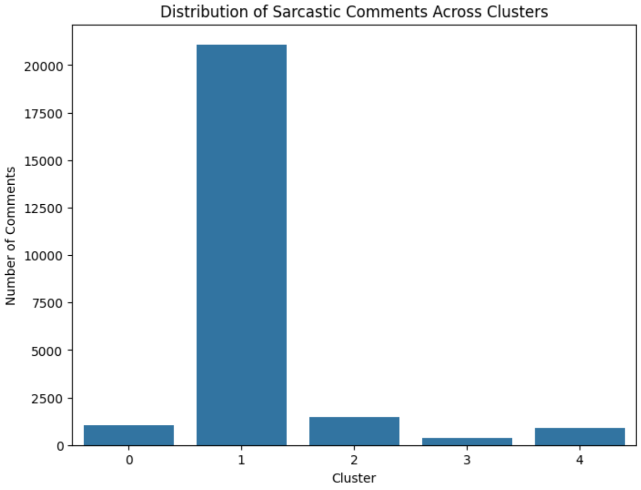
Combined features model

Each model was evaluated using accuracy, precision, recall, F1-score, and confusion matrices for a full performance comparison across feature types. Feature Engineering

To assess the different linguistic representations for the task at hand, three feature configuration schemes were crafted:

1.4 Unsupervised Exploratory Analysis

Apart from the supervised modeling, K-Means clustering was applied to sarcastic comments for detecting latent groupings based on linguistic patterns. This exploratory step aimed to provide deeper insight into stylistic variation in sarcastic language and the possibility of multi-cluster sarcasm taxonomies within the ambit of computational pragmatics research.



2. Results and Performance Analysis

2.1. TF-IDF Model

While the TF-IDF baseline model was doing great on non-sarcastic comments, it performed quite badly on determining sarcastic ones.

Confusion Matrix:

[[4403, 622], [3916, 1059]]

The large number of true negatives (4403) suggests good performance on literal language, while the smaller amount of true positives (1059) indicates difficulty with capturing contextually nuanced sarcasm. This supports limitations in purely lexical representations for pragmatic inference tasks.

2.2 Combined TF-IDF + Embeddings Model

Among all the experiments, the hybrid model produced the best results.

Confusion Matrix:

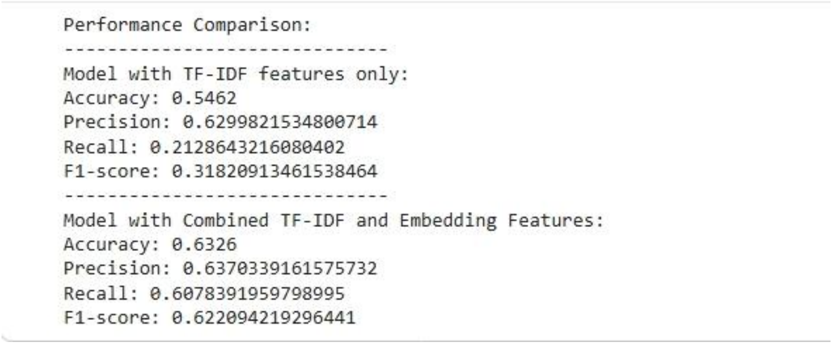
[[3302, 1723], [1951, 3024]]

Metrics:

Precision: 0.632

Recall: 0.608

F1-Score: 0.622



```
Performance Comparison:
------------------------------
Model with TF-IDF features only:
Accuracy: 0.5462
Precision: 0.6299821534800714
Recall: 0.2128643216080402
F1-score: 0.31820913461538464
------------------------------
Model with Combined TF-IDF and Embedding Features:
Accuracy: 0.6326
Precision: 0.6370339161575732
Recall: 0.6078391959798995
F1-score: 0.622094219296441
```

These results indicate a significant improvement in detecting sarcastic comments. The hybrid representation represented a better balance between precision and recall compared to using TF-IDF alone. It has shown that semantic embeddings hold an important position in modeling pragmatic hidden cues such as tone, implied meaning, and sentiment reversals. This hybrid approach brought forth the subtle detection of sarcasm and outperformed all single-method models. 3.3 Clustering Analysis K-Means clustering on sarcastic comments produced meaningful sub-groups, suggesting that sarcasm does not exist as a single linguistic phenomenon but instead clusters into distinct stylistic or pragmatic patterns. This finding supports the broader computational pragmatics hypothesis that sarcasm is contextually diverse and may require multi-prototype modeling approaches in future systems.

3. Discussion and Future Work

These results show that transformer-based semantic representations significantly improve sarcasm detection, especially when combined with TF-IDF features. The performance of the hybrid model also indicates that sarcasm depends on both lexical incongruity and deeper semantic cues, thus confirming theoretical expectations from pragmatics regarding a mismatch between literal form and intended meaning.

3.1 Implications for Computational Pragmatics

This work contributes to computational pragmatics by demonstrating the need for models which can integrate:

Surface lexical patterns

Contextual semantic understanding

Implicit speaker intent

The improved performance of contextual embeddings supports the idea that pragmatic reasoning in machines must incorporate high-dimensional semantic cues that go beyond word frequency.

3.2 Limitations

The dataset consists exclusively of comments from Reddit, which may lower generalizability.

No fine-tuning of transformers was done.

Hyperparameter tuning for machine learning models was light.

3.3 Future Directions

Building on the findings in the dataset:

Perform full hyperparameter optimization for all model families.

Fine-tune transformer models directly on the sarcasm datasets to capture domain-specific usage patterns much better.

Consider more transformer architectures and embedding strategies.

Explore explainability techniques to determine which contextual features inform the sarcasm classification.

Expand the clustering analysis to create a taxonomy of types of sarcasm, possibly enhancing model performance through multi-label or hierarchical classification methods.

The procedure will further develop an understanding of sarcastic communication and improve the development of pragmatic-aware NLP systems that can understand subtle, non-literal language behavior. Unsupervised Exploratory Analysis

Apart from the supervised modeling, K-Means clustering was applied to sarcastic comments for detecting latent groupings based on linguistic patterns. This exploratory step aimed to provide deeper insight into stylistic variation in sarcastic language and the possibility of multi-cluster sarcasm taxonomies within the ambit of computational pragmatics research.