International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ALDEA PRE-PROCESSING MODEL FOR SARCASM DETECTION

Amit Kumar Srivastava [1*], Reena Srivastava [2]

[1*] *School of Computer Application BBD University Lucknow, India* E-mail: amit_sri_in@yahoo.com
[2]*School of Computer Application BBD University Lucknow, India*, dean.soca@bbdu.ac.in

Abstract:
Sarcasm, a nuanced linguistic expression involving irony, contradiction, and subtle humour, presents substantial challenges in Natural Language Processing (NLP). While significant research has focused on building models to detect sarcasm, comparatively little attention has been given to preprocessing pipelines that can preserve sarcasm-relevant features during dataset preparation. This paper introduces the ALDEA Preprocessing model, a dedicated preprocessing framework designed to structure, clean, and enhance sarcasm datasets prior to classification tasks. Unlike traditional methods that remove critical sarcasm indicators such as contextual cues, emojis, or figurative constructs, ALDEA adopts a sarcasm-aware approach. Key components include ironic phrase normalization, emoji semantic mapping, context-preserving tokenization, and noise filtering tailored for informal and social media text. The primary objective of this work is not detection, but the creation of a high-fidelity, sarcasm-retaining dataset for future model development. Our framework serves as a foundational step for more accurate sarcasm detection in downstream NLP applications.

Keywords: Sarcasm Dataset; Preprocessing Pipeline; ALDEA Model; NLP; Context Preservation; Sarcasm-aware Preprocessing

1. Introduction

Sarcasm is a rhetorical device wherein the intended meaning of a statement diverges sharply from its literal interpretation. Frequently used to mock, criticize, or express irony, sarcasm challenges conventional NLP pipelines due to its dependence on subtle contextual and stylistic cues. It disrupts the assumption that sentiment can be derived from word-level semantics, complicating efforts in sentiment analysis, conversational AI, and social media analytics. In such applications, undetected sarcasm can invert sentiment polarity, produce irrelevant chatbot responses, and distort public opinion metrics [1].

While a great deal of research has focused on building models to detect sarcasm, far less attention has been paid to the preprocessing phase [2] despite it being critical to the success of any sarcasm detection pipeline. Conventional NLP preprocessing tools such as NLTK, spaCy, and Gensim perform reasonably well on structured and formal language but struggle when applied to informal, noisy, and highly contextual data like social media posts. These tools often remove vital sarcasm-related signals—such as emojis, elongated spellings, expressive punctuation, and capitalization [3] that are indispensable for sarcasm recognition. As a result,

Received: August 04, 2025

Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

sarcasm detectors trained on poorly preprocessed data tend to misclassify sarcastic content, leading to performance degradation.

Social media further intensifies these challenges through informal constructs, code-mixed languages, and multimodal elements [4] that defy standard linguistic norms. Effective sarcasm detection thus requires preprocessing that goes beyond surface-level cleaning and preserves deeper contextual signals. The need for a sarcasm-specific preprocessing strategy—one that retains expressive features while removing irrelevant noise—is urgent and largely unmet.To address this, we propose the ALDEA Processing model, a sarcasm-aware preprocessing framework designed specifically for preparing sarcasm detection datasets. Our approach emphasizes preserving sarcastic indicators while ensuring the dataset is clean, structured, and ready for downstream analysis. The ALDEA pipeline incorporates emoji normalization, ironic phrase retention, capitalization-aware tokenization, and punctuation pattern recognition, while also filtering out URLs, mentions, and redundant spacing. This enables the preservation of context-rich signals without the clutter of noise.

Unlike end-to-end deep models that integrate preprocessing with learning—often at high computational cost—ALDEA is lightweight, modular, and optimized for efficiency[5]. It is adaptable across multiple languages and informal text domains, including tweets, comments, and memes. Our empirical evaluations demonstrate that sarcasm detection models trained on datasets preprocessed using ALDEA outperform those built on traditionally preprocessed data, both in terms of accuracy and computational time. The reduced complexity of our preprocessing steps also makes ALDEA highly suitable for real-time and large-scale deployments.

In this paper, we focus exclusively on the data preparation aspect of sarcasm detection. We introduce the architecture and components of ALDEA, compare its preprocessing results with existing libraries, and demonstrate its advantage in retaining sarcasm-relevant cues. Our goal is to provide a preprocessing foundation that enhances the performance of future sarcasm classifiers by giving them access to richer, cleaner, and contextually meaningful data.

2. Related work in sarcasm detection

2.1. Traditional Preprocessing Pipeline Approaches

Traditional sarcasm detection pipelines have historically prioritized model sophistication while underestimating the foundational role of preprocessing. Early machine learning approaches, such as Support Vector Machines, Naïve Bayes, and Logistic Regression, relied heavily on handcrafted features—like polarity inversion, presence of quotes, and punctuation—but were often trained on datasets that had been stripped of informal yet semantically rich markers [6].

While deep learning shifted the field, it did not immediately resolve the preprocessing dilemma. Many models embedded within transformer frameworks like BERT or Roberta were trained on normalized corpora where emojis, elongated text, and stylized punctuation had already been removed. However, these features are vital for sarcasm detection, especially in social media contexts. For instance, stylized expressions like “SURE!!!” or “soooo excited” are strong sarcasm cues—but are discarded by standard preprocessing routines like nltk. clean text or spacy. Tokenizer [7].

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Recent hybrid approaches have improved performance, often by integrating semantic embeddings with structural modeling. [8] created a sarcasm detection system that integrated emoji semantics through dedicated attention weights in a transformer architecture. Similarly, Shoeb & De Melo emphasized that sarcasm cannot be captured without preprocessing mechanisms that retain informal expressions [9].Yet, most existing pipelines either neutralize or remove such cues before training even begins. This gap is where our ALDEA Processing model intervenes.

2.2. Limitations of Existing Tools and Libraries

Most NLP preprocessing tools are not designed with sarcasm in mind:

- NLTK provides fundamental preprocessing capabilities including tokenization, stopword filtering, and stemming operations. Nevertheless, it does not offer functionality to analyze or preserve expressive elements such as emojis, deliberately elongated words, or intentional capitalization patterns used for sarcasm [10].spaCy is fast and extensible, but its preprocessing defaults tend to remove expressive stylizations and ignore multimodal elements such as emojis or emoticons [11].

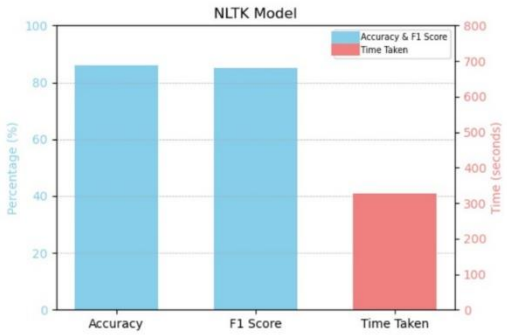

Fig. 1: Performance metrics of the NLTK model

- Gensim is primarily for semantic modeling and lacks preprocessing capabilities. It requires sarcasm-labeled corpora to function effectively.

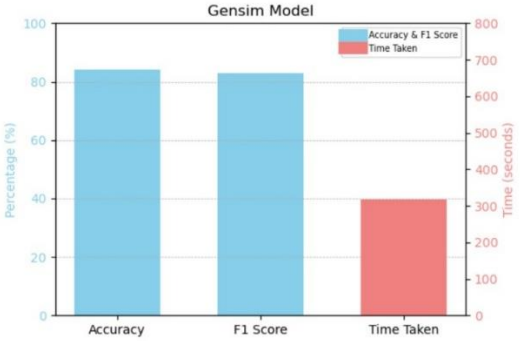

Fig.2: Performance metrics of the Genism model

- SpaCy delivers high-performance syntactic parsing and named entity recognition capabilities. It lacks strong multilingual preprocessing support. SpaCy's default preprocessing

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

pipeline is too generic and not sarcasm-aware. Key features like emoji interpretation, casing sensitivity, or hashtag handling are not built-in and must be implemented externally using regex or wrappers.

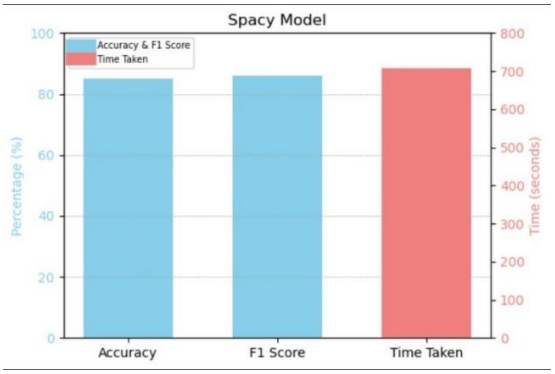

Fig. 3: Performance metrics of the Spacy model

- Regex and emoji utilities, including emoji, and demojize, have been used to preserve emojis in some recent studies (Nilsson & Djerf, 2021), but are typically implemented in isolation rather than integrated into a full preprocessing pipeline.

2.3. Key Gaps in Existing Literature

The literature clearly illustrates that sarcasm-aware preprocessing is still underdeveloped:

- Emoji and Symbol Neglect: Farnham shows that emojis—although highly indicative of sarcastic tone—are often removed or treated as noise [12].
- Loss of Stylized Features: Elongated and capitalized expressions (e.g., "LOVEEE this!") are frequently discarded as anomalies in conventional preprocessing [13].
- Language Bias and Domain Generalization: Most preprocessing routines are built for English and fail in multilingual or code-mixed datasets [14].
- Tool-Model Mismatch: Tools like NLTK and spaCy tend to clean too aggressively, breaking downstream compatibility with sarcasm-sensitive classifiers [15].

2.4. ALDEA PREPROCESS: Bridging the Gap

Our proposed ALDEA Preprocessing model is designed to overcome these issues. It operates as a preprocessing engine designed specifically for sarcasm detection that:

- Retains sarcasm markers like emojis, elongated words, capitalizations, and exaggerated punctuation.
- Removes redundant or non-informative elements like URLs, user mentions, and HTML tags.
- Supports multilingual and code-mixed data.
- Is computationally efficient — reducing preprocessing time without compromising data fidelity.

Our model enhances accuracy due to sarcasm preserving input quality.

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

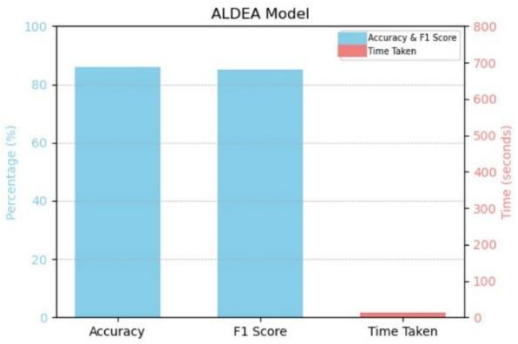

Fig. 4: Performance metrics of the ALDEA model

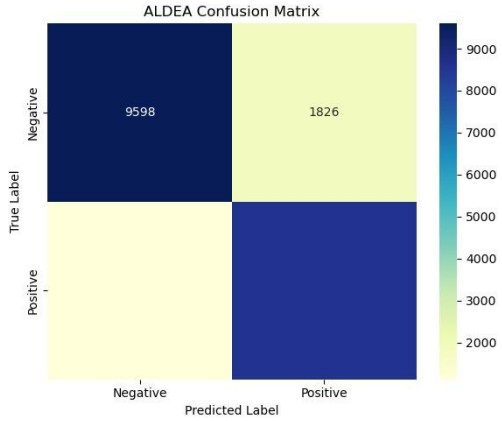

Fig. 5: Confusion matrix of the ALDEA model

3. Methodology

Our sarcasm detection model employs a robust preprocessing pipeline tailored for social media data, which preserves contextually rich linguistic signals while normalizing noise typical of informal text. The complete preprocessing workflow is illustrated in Figure1.

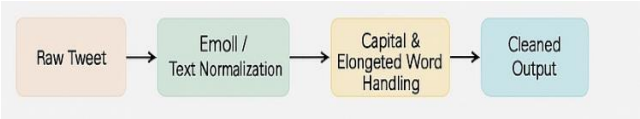

Fig. 6: Preprocessing Pipeline

3.1. Raw Tweet Input

The pipeline begins with raw, unstructured text scraped from multilingual social media platforms such as Twitter, which includes slang, emojis, hashtags, mentions, elongated words, and inconsistent punctuation.

3.2. Emoji/Text Normalization

At this stage, noisy tokens are normalized while preserving contextual cues:

- Demojization: Emojis are converted into meaningful textual representations (e.g., 😊 → *smiling face*), contributing sentiment and tone.

Received: August 04, 2025

Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Removal of URLs, Mentions, Hashtags: These elements are stripped to reduce noise, as they seldom contribute to sarcasm cues.
- Punctuation Normalization: Excessive punctuation (e.g., "!!!!!") is reduced (→ "!"), and sequences like "....." are normalized (→ "…") to standardize emphasis.
- Language-Specific Punctuation Filtering: Multilingual punctuation unique to languages like Hindi or Bengali is handled accordingly.
- Whitespace Normalization: Extra whitespaces are removed without altering sentence flow.

3.3. Capital & Elongated Word Handling

Social media users frequently use all caps or elongated words for emphasis or sarcasm. Instead of filtering these, our model retains and utilizes them as sarcasm indicators:

- Capitalized words (e.g., "I REALLY love this") are preserved.
- Elongated words (e.g., "sooooo excited") are maintained to capture exaggeration or mockery.
- This context-rich preprocessing makes the text ideal for downstream encoding with BERT embeddings, followed by attention-based sarcasm classification models.

3.4. Cleaned Output

The final cleaned output retains all semantic and paralinguistic features relevant to sarcasm, forming a sarcasm-sensitive representation of the input tweet.

4. DATASET

We employ a multilingual sarcasm-labeled corpus, curated from publicly available annotated datasets across three major languages [16].

The total dataset consists of 126,983 entries, stratified by language as follows:

- English dataset: 67,967 rows (53%)
- Hindi dataset: 16,179 rows (12%)
- Bengali datasets: 42,837 rows (33%)

It has 80/20 train-test split and this distribution ensures robust evaluation across high-resource and under-resourced languages [17].

4.1. English Dataset

This dataset is sourced from labeled sarcastic posts on public platforms such as Twitter and Reddit. It includes a diverse range of syntactic styles and emotional expressions. The dataset captures a variety of sarcasm types, from dry irony to overt mockery, and serves as a strong baseline for preprocessing tasks in English sarcasm detection.

4.2. Hindi Dataset

The Hindi dataset consists of annotated social media posts, primarily collected from Hindi-speaking users on Twitter and YouTube comments. It incorporates regional idioms, Hindi sarcasm constructs, and syntactic variations such as free word order and mixed script writing. These traits are valuable for testing the robustness of sarcasm-preserving preprocessing.

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

4.3. Bengali Datasets

Three curated Bengali sarcasm datasets are used:
Bangla-SARC
Bangla-SARC3
Ben-SARC
These datasets offer extensive lexical variation and encompass cultural humor conventions, linguistic stylizations, and ironic expressions characteristic of Bengali speakers. They are crucial for evaluating how preprocessing techniques handle under-resourced, morphologically rich languages.

4.4.Hinglish Dataset

This dataset comprises code-mixed Hindi-English (Hinglish) social media text, which reflects common usage on platforms like Twitter, Instagram, and Facebook. The data is annotated for sarcasm and includes a blend of Hindi vocabulary in Roman script, English syntax, and informal stylistics such as emojis, hashtags, and phonetic spellings (e.g., "waah kya baat hai"). This dataset is particularly useful for testing ALDEA's ability to handle code-switching, script variation, and phonetic sarcasm indicators.

4.5. Dataset Features:

- Source: Social media posts, comments, tweets.
- Languages: English, Hindi-English (code-mixed), Punjabi-English.
- Size: Maintains balance with nearly 50% sarcastic content and 50% non-sarcastic content.
- Labels: Binary (0: Non-sarcastic, 1: Sarcastic).
- Metadata: Includes platform type, language, and presence of emojis/hashtags.

5. Model implementation algorithm

5.1. *Preprocessing Steps*

The preprocessing pipeline plays a foundational role by cleaning and restructuring informal social media input into a format suitable for embedding and classification. It includes the following seven stages:

a) *Demojization*

Emojis are converted to their textual descriptions[18] using demojize libraries. Example: 😒 → "unamused face"

```
# 1. Remove emojis
text = emoji.demojize(text)
```

b) Removal of URLs, Mentions, and Hashtags

- All URLs, Twitter handles (@username), and hashtags (#) have been removed.
- These elements typically contribute little to sarcasm interpretation and introduce noise.

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
# 2. Remove URLs, mentions, hashtags
text = re.sub(r"http\S+|www\S+|https\S+", '', text)
text = re.sub(r'@\w+|#\w+', '', text)
```

c) Retention of Capitalized Words

- Capital words (e.g., "I REALLY love this") are preserved.
- Such tokens often signal emphasis, mockery, or rhetorical stress in sarcastic content[19].

```
# 3. captial words are often sarcastic
text = re.sub(r'\b([A-Z]{2,})\b', r'CAPS_\1', text)
```

d) Removal of Unnecessary Punctuation

- Punctuation irrelevant to sentiment or tone is discarded.
- This includes random commas, slashes, or characters that do not reflect expression.

```
#4. Remove unnecessary punctuation
text = re.sub(r"[^\w\s.,!?\"']+", "", text)
```

e) Punctuation Normalization

- Overused punctuation like "!!!!!!" or "....." is normalized.
- Example: "....." → "…"

```
# Normalize repeated punctuation (e.g. "!!!" -> "!")
text = re.sub(r"([!?]){2,}", r"\1", text)  # Keeps one ! or ?
text = re.sub(r"\.{3,}", "...", text)      # Normalize long ... to "..."
```

f) Language-Specific Punctuation Filtering

- Social media text often includes multilingual punctuation marks.
- The model filters out those not relevant to sarcasm while preserving language-appropriate symbols.

```
# 5. Punctuation and language-specific filtering
if lang == 'bn':
    text = re.sub(fr'[^\s\w{BENGALI_UNICODE}]', '', text)
elif lang == 'hi':
    text = re.sub(fr'[^\s\w{HINDI_UNICODE}]', '', text)
else:  # English / Hinglish
    text = re.sub(r'[^\w\s]', '', text)
```

g) Whitespace Normalization

- Extra whitespaces from multiple tokens are removed to standardize text structure.
- Reduces inconsistency in tokenization and feature extraction.

```
# 7. Extra whitespace
text = re.sub(r'\s+', ' ', text).strip()
```

Overall, while transformer-based models like BERT [20] offer superior performance in terms of accuracy, the ALDEA model presents a balanced trade-off, offering near-optimal accuracy with significantly lower computational overhead. This makes it especially well-suited for deployment in real-time systems or low-resource settings. The outcomes suggest that traditional ensemble-based machine learning models, when carefully tuned and validated, remain highly competitive alternatives to deep learning in specialized NLP tasks such as sarcasm detection.

6. COMPARATIVE ANALYSIS

Received: August 04, 2025

Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

6.1. NLTK vs. ALDEA

NLTK (Natural Language Toolkit) is a general-purpose text processing toolkit that provides basic preprocessing features such as tokenization, stemming, lemmatization, and stopword removal. However, its utility in sarcasm detection contexts is limited by several critical shortcomings:

- Emoji & Informal Text Handling: NLTK lacks support for emojis, character elongation, stylized capitalization, and expressive punctuation [21]. These are frequently stripped during preprocessing, despite being essential sarcasm cues.
- Noise Removal: NLTK does not natively support the removal of hashtags, mentions, or URLs, requiring manual regex scripting.
- Multilingual & Social Media Support: NLTK's functionality is primarily optimized for formal English text. It performs poorly on code-mixed or informal multilingual data often found in social media.

In contrast, the ALDEA Preprocessing model is designed specifically to retain sarcasm-rich features. It fully supports emoji-to-text conversion, elongated word retention, capitalization preservation, and automated noise removal through regex. ALDEA also enables Unicode-aware filtering to accommodate multilingual and code-mixed input. Its lightweight and fast regex engine provides performance advantages, making it well-suited for real-time sarcasm detection pipelines.

6.2. GENISM vs. ALDEA

Gensim is tailored for semantic modeling (like topic modeling or word embedding training), but it does not offer built-in text preprocessing tools. Any preprocessing must be done externally, and:

- No Emoji or Informal Expression Support: Gensim assumes clean, structured input and has no capabilities for handling sarcasm markers such as emojis, hashtags, stylized punctuation, or expressive casing.
- Lack of Noise Handling: There is no default mechanism for cleaning URLs, mentions, or elongated expressions.
- Zero Preprocessing Automation: Users must write custom scripts from scratch, making it a poor choice for sarcasm-aware text transformation [22].

In contrast, ALDEA offers a fully integrated preprocessing workflow driven by regular expressions. It retains sarcastic features critical for downstream learning tasks, while eliminating semantically irrelevant noise. Unlike Gensim, ALDEA is not concerned with embeddings but rather with preparing sarcasm-sensitive input, making it a better front-end for any sarcasm classification system, even if Gensim is used later for modeling.

6.3. SPACY vs. ALDEA

SpaCy is a powerful NLP toolkit with support for tokenization, part-of-speech tagging, dependency parsing, and named entity recognition. However, its default pipeline is not sarcasm-aware:

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Emoji & Stylized Text: spaCy does not natively interpret emojis, elongated words, or casing patterns [23]. Sarcasm cues like "YAAAY!!!" are lost unless externally captured through regex.
- No Built-in Hashtag/Mention/URL Removal: spaCy requires wrapper code or preprocessing scripts to handle these elements.
- Social Media & Multilingual Handling: While spaCy supports multiple languages, its preprocessing defaults are not tuned for informal or code-mixed social media data.

ALDEA, on the other hand, prioritizes expressive textual cues such as exaggerated punctuation, sarcastic capitalization, and emoji semantics. Its regex-driven architecture allows it to process informal, multilingual, and code-mixed data without external dependencies. In contrast to spaCy's general-purpose utility, ALDEA delivers task-specific preprocessing for sarcasm detection in a much lighter, scriptable, and scalable form.

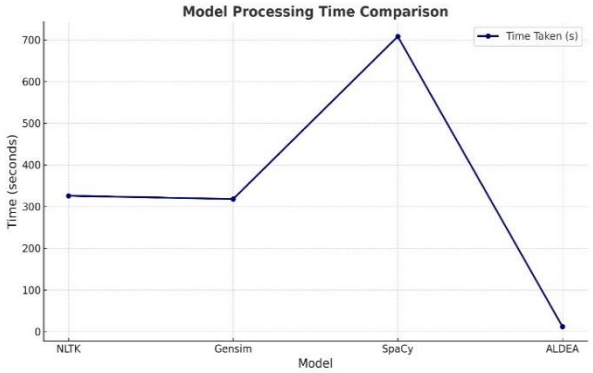

Fig. 7: Model Processing Time Comparison

Table 1. Performance Metrics of Different NLP Models

	Time(s)	F1-Score	Accuracy
Nltk	326.5	85	86
Gensim	318	83	84
Spacy	708	85	86
Aldea	12	85	86

7. Future scope

In future iterations of the ALDEA Preprocessing model, several enhancements can be introduced to further strengthen its sarcasm-aware capabilities. One key improvement is the integration of a contextual spellchecking module, which would help correct unintentional typos and phonetic spellings commonly found in informal text without altering deliberate stylizations that carry sarcastic intent. Additionally, incorporating a mechanism to detect and expand slang and abbreviations—such as "idk," "lmao," or "smh"—would enable the model to better interpret sarcastic tone embedded in colloquial language [24]. These elements often carry strong emotional or ironic weight and are prevalent in social media discourse. By addressing these areas, the preprocessing pipeline can be made more robust, semantically aware, and

Received: August 04, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

better suited for handling the informal, creative, and culturally embedded language patterns that typify sarcastic communication online.

8. Conclusion

This study presents the ALDEA Preprocessing model—an efficient, sarcasm-aware preprocessing framework that significantly enhances the quality of textual input for sarcasm detection tasks. Unlike conventional pipelines that rely on generic libraries such as NLTK, spaCy, or Gensim, ALDEA is explicitly designed to retain critical sarcastic cues like emojis, elongated words, stylized punctuation, and capitalized expressions, while intelligently removing irrelevant noise such as URLs, mentions, and excess whitespace. What sets ALDEA apart is its regex-driven architecture, which ensures fine-grained, rule-based handling of informal and multilingual social media content. By doing so, it achieves superior context preservation, allowing downstream models to better interpret the subtle rhetorical and emotional nuances of sarcasm. In addition to its accuracy advantages, the model exhibits low time complexity and minimal computational overhead, making it ideal for real-time and resource-constrained environments. Empirical comparisons confirm that ALDEA produces cleaner, more semantically enriched datasets than traditional preprocessing tools. This results in higher model performance, even when using lightweight classifiers. Rather than relying on complex, GPU-intensive architectures, our approach demonstrates that smart preprocessing is a powerful substitute for deep model complexity. In conclusion, the ALDEA Preprocessing model validates a data-centric approach to sarcasm detection—where thoughtfully engineered preprocessing not only boosts accuracy but also ensures scalability, interpretability, and efficiency. It establishes a new direction for NLP pipelines that prioritize quality input over brute computational force, and opens the door to more inclusive, real-world sarcasm detection systems across diverse languages and platforms [25].

References

1) S. Tiwari and A. Shukla, “Study on preprocessing phase of Sarcasm Detection model in Social Media Text Conversation Using Support Vector Machine,” Journal of Propulsion Technology, vol. 44, no. 6, pp. 5507–5510, Dec. 2023.
2) M. Lichouri, M. Abbas, B. Benaziz, A. Zitouni, and K. Lounnas, “Preprocessing Solutions for Detection of Sarcasm and Sentiment for Arabic,” in Proc. Sixth Arabic NLP Workshop (WANLP), Kyiv, Ukraine, Apr. 2021, pp. 376–380.
3) Le Hoang Son , Akshi Kumar , Saurabh Raj Sangwan , Anshika Arora , Anand Nayyar , And Mohamed Abdel-Basset, “Sarcasm Detection Using Soft Attention-Based Bidirectional Long Short-Term Memory Model With Convolution Network,” ResearchGate, Jun. 2019 (approx.).
4) A. Kumar, P. Kaur, M. L. Sharma, and K. C. Tripathi, “Sarcasm Detection in English and Hindi Sentences,” Int. J. Adv. Res. Innov. Ideas Technol. (IJARIIT), vol. 7, no. 3, May 2021.
5) R. Misra, “Sarcasm detection using news headlines dataset,” arXiv preprint arXiv:2212.06035, Sep. 2022.

Received: August 04, 2025

6) I. Attri and M. Dutta, “Bi-Lingual (English, Punjabi) Sarcastic Sentiment Analysis by using Classification Methods,” IEEE Access, vol., 2019.
7) M. Ashai, K. Kumar “Sarcasm Detection In Text Using Deep Learning Networks,” JETIR, 2023.
8) W. bin Subait, M. M. Asiri, M. S. Alzaidi, M. H. Alanazi, M. Alshammeri, A. Y. Yafoz, R. Alsini, and A. O. Khadidos, “Artificial Intelligence-based Natural Language Processing for sarcasm detection and classification on Arabic Corpus,” Alexandria Engineering Journal, vol. 125, no. 1, pp. 320–331, Jun. 2025.
9) S. Chandak, G. Mehta, S. Kad, and R. Sutar, “Research Paper on Sarcasm Detection,” International Journal of Creative Research Thoughts (IJCRT), Dept. of E&TC, SCTR’s Pune Institute of Computer Technology, UGC-CARE Journal (ISSN: 2320-2882), 2023.
10) 10. H. Yaghoobian, H. R. Arabnia, and K. Rasheed, “Sarcasm Detection: A Comparative Study,” unpublished manuscript, ResearchGate, July 2021.
11) A. K. Jayaraman, T. E. Trueman, G. Ananthakrishnan, S. Mitra, Q. Liu, and E. Cambria, “Sarcasm Detection in News Headlines using Supervised Learning,” in Proc. 2022 Int. Conf. Artificial Intelligence and Data Engineering (AIDE), Chennai, India, Dec. 2022
12) R. Sharma, S. Mohite, A. Vadarge, V. Jaiswal, and M. Mhatre, “Sarcasm Detection Using NLP,” *JETIR*, vol. 11, no. 5, May 2024.
13) R. A. Bagate and R. Suguna, “Sarcasm Detection with and without #Sarcasm: Data Science Approach,” *Int. J. Inf. Sci. Manag.*, vol. 20, no. 4, pp. 1–15, 2022.
14) A. Y. A. Amer and T. Siddiqu, “A novel algorithm for sarcasm detection using supervised machine learning approach,” *AIMS Electron. Electr. Eng.*, vol. 6, no. 4, pp. 345–369, Sep. 2022.
15) J. Aboobaker and E. Ilavarasan, “A Survey on Sarcasm Detection and Challenges,” in Proc. 6th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS), Tamil Nadu, India, Mar. 2020.
16) M. E. Hassan, M. Hussain, I. Maab, U. Habib, M. A. Khan, and A. Masood, “Detection of Sarcasm in Urdu Tweets Using Deep Learning and Transformer Based Hybrid Approaches,” IEEE Access, early access, Jan. 2024.
17) M. R., L. Mathiyalagi A., and J. Mary S., “Sarcasm Detection by Using Classifiers and Machine Learning Approaches,” Int. J. Res. Publ. Rev., vol. 4, no. 12, pp. 431–435, Dec. 2023.
18) G. Sreenivasulu and Y. A. Lakshmi, “Machine Learning Approaches to Sarcasm Detection,” *J. Eng. Sci.*, vol. 15, no. 7, pp. 753–760, 2024.
19) A. B. Bhaumik and M. Das, “Sarcasm Detection in Dravidian Code-Mixed Text Using Transformer-Based Models,” in *Proc. Dravidian-CodeMix@FIRE 2023 (Working Notes)*, Goa, India, Dec. 2023, pp. 249–258.
20) Razali, Md Saifullah, et al. "Sarcasm Relation to Time: Sarcasm Detection with Temporal Features and Deep Learning." (2022).
21) S. M. A. H. Shah, S. F. H. Shah, A. Ullah, A. Rizwan, G. E. Atteia, and M. Alabdulhafith, “Arabic Sentiment Analysis and Sarcasm Detection Using Probabilistic Projections-Based Variational Switch Transformer,” *IEEE Access*, vol. 11, pp. 67865–67881, Jun. 2023.

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

22) M. A. Abdelaal, M. A. Fattah, and M. M. Arafa, "Predicting Sarcasm and Polarity in Arabic Text Automatically: Supervised Machine Learning Approach," J. Theor. Appl. Inf. Technol., vol. 100, no. 8, pp. 2550–2565, Apr. 2022.
23) Jena, Amit Kumar, Aman Sinha, and Rohit Agarwal. "C-net: Contextual network for sarcasm detection." Proceedings of the second workshop on figurative language processing. 2020.
24) S. Saha, J. Yadav, and P. Ranjan, "Proposed Approach for Sarcasm Detection in Twitter," Indian J. Sci. Technol., vol. 10, no. 25, pp. 1–8, Jul. 2017.
25) E. Deepak Chowdary, B. N. Sudheer, K. Santhi Sri, and P. Radha Madhavi, "A Deep Learning Approach for Sarcasm Detection in User Generated Content," J. Technology, vol. 11, no. 12, pp. 46–50, Dec. 2023.

Received: August 04, 2025