Robustness Analysis Of Lightweight Machine Learning Models For Sentiment Classification Under Noisy Text

Ms. Anjali Jawale[1], Mr. Shreyash Dodekar[2], Mrs. Shital Pashankar[3]
[1,2]*MSc Computer Application, School Of Information Technology, Indira University, Wakad, Pune, India*
[3]*Computer Science, School Of Information Technology, Indira University, Wakad, Pune, India*
doi.org/10.64643/ijirt.208465-459

***Abstract*—Nowadays, sentimental analysis is an important application of natural language processing and widely used to understand whether a review expresses a positive or negative opinion. Many sentiment classification models are trained using clean data. Real-world reviews often contain spelling mistakes, missing or repeated characters, abbreviations and informal language. A model which executes well on clean data may lose accuracy on noisy text, making robustness essential for evaluation. Previous research has mainly focused on the robustness of neural and transformer-based NLP models, while lightweight machine-learning models have received less attention. These traditional models are simple, fast, resource-efficient and easier to interpret compared to complex deep-learning models. Therefore, this study focuses on evaluating the robustness of lightweight machine-learning classifiers when handling noisy text.**

The study assesses Multinomial Naive Bayes, Logistic Regression, Linear SVM using TF-IDF representation for fair comparison. The Amazon Cell-Phone Reviews dataset is used to classify reviews into positive and negative sentiments. To simulate real-world text, five types of noise—deletion, insertion, substitution, swapping and abbreviation are introduced at 10%, 20% and 30% severity levels. The noise is programmatically added while keeping the original sentiment labels unchanged. The models are evaluated on noisy and clean data to measure their sturdiness. Performance is examined using precision, recall, accuracy, performance degradation and F1 score. The study addresses which model remains most robust and which type, level of noise causes greatest performance loss. The study helps to determine the level and type of noise which leads to highest performance loss. This finding helps developers and researchers in choosing lightweight model for sentiment analysis.

***Index Terms*—Machine Learning, Noisy Text, Robustness, Sentiment Analysis, NLP**

I. INTRODUCTION

A. Background

Sentimental analysis is a vital application of natural language processing that targets to identify the opinion stated in a piece of text. It is broadly used for analyzing social media posts, feedback and customer review. Traditional machine learning models like Logistic Regression, Naïve Bayes and Support Vector Machine [1] have been broadly used for text classification. These models are simple to train and can work Efficiently with sparse text representations like TF-IDF. They also use less computer requirements.

However, online text may not be always perfect. Reviews may contain missing characters, spelling mistakes, repeated characters or typing errors. These changes may not remarkably affect a human reader but these changes can affect the features generated from the text. So that, a classifier that performs well on clean reviews may act differently when same reviews with noise.

B. Motivation

Many sentiment-classification studies state clean test data performance. This helps to get useful insights about the predictive Proficiency of a model but it is not able to fully describe how the model will behave in practical situations such as A customer may write “gr8 product” misspell a word, accidentally omit a character or typos. This motivated the present study to assess lightweight sentiment classifiers with noisy conditions. Rather than considering only final accuracy, this study explores how different types and levels of noise affect precision, accuracy, recall and F1-score [4].

C. Research Problem

In this study, main problem is the limited understanding of how lightweight machine learning sentiment classifiers act when review text is influenced by different types of noise. Though Multinomial Naïve Bayes, Logistic Regression and Linear SVM can give good results on clean text but their performance may differ when text contains some mistakes called noise. So, clean data performance alone may not be enough for identifying the most relevant model for real-world noisy data.

D. Research Questions

1. How does textual noise impact the performance of lightweight sentiment classification models, when they are dealing with everyday written language?
2. Which type of textual noise causes the greatest degradation in model performance and how does it differ from other forms of textual noise?
3. Which lightweight machine learning model is most robust, to text and what characteristics make it resilient?
4. How does increasing noise severity from 10% to 20% and 30% affect classification performance and what trends can be observed?
5. Does hyperparameter optimization improve clean-data performance and robustness to text and if so, which parameters are most critical?

E. Research Objectives

1. To implement Logistic Regression, Multinomial Naïve Bayes and Linear SVM [1] for sentiment classification that relies on TF-IDF features.
2. To evaluate these three models on clean Amazon cell-phone reviews.
3. To introduce controlled character-level and abbreviation-based noise contains test data.
4. To evaluate how different noise types and varying severity levels affect classification performance.
5. To compare model robustness using accuracy, recall, precision and F1-score [4].
6. To examine whether hyperparameter optimization changes performance on data or robustness, on noisy data.

II. LITERATURE REVIEW

A. Traditional Machine Learning for Sentiment Classification

Traditional machine learning approaches remain useful for sentiment classification because they can provide competitive results with relatively low computational requirements. A study compared Logistic Regression, Naive Bayes and SVM for sentiment analysis of Azerbaijani tweets [1]. Their research found that these traditional machine-learning algorithms can effectively classify sentiments in tweets without using deep-learning models. This study concentrated on comparing the efficiency of the three classifiers. After comparing Naive bayes and SVM for sentimental analysis, study found that both methods can be used to classify sentiments in reviews [2]. One of the research projects studied the effect of preprocessing techniques on the accuracy of machine-learning algorithms in sentimental analysis [3]. After comparing different machine learning and deep learning models for sentimental analysis using Amazon product review. They found that Logistic Regression and SVM performed well among traditional machine-learning model, while RoBERTa performed among deep-learning models [4].

B. Text Noise and NLP Robustness

Robustness to noisy text has received increasing attention in natural language processing. A study found that how review text and ratings can be used to understand user preferences [5]. The impact of synthetic and naturally occurring noise on neural machine translational systems [6]. One of study investigated whether synthetically generated noise could help NLP systems better handle naturally occurring noise [7]. Adding artificially created noise to clean text can help models handle real-world text variations better after studied how NLP systems perform when dealing with noisy user-generated text like typos, slang and informal language [8]. The study is useful because it shows that controlled noise can be used to test and increase the efficiency of NLP models. A study showed that even small changes can greatly reduce the accuracy of NLP models. The study also found that using a better word-recognition metho can help improve model performance when the text contains such errors [9]. Wild NLP to study how NLP models perform when text contained common errors such as misspelling, keyboard mistakes, missing characters and word changes [10]. A study created a stress- testing approach to find weaknesses in NLP models [11].

C. Adversarial and Character-Level Perturbation

Another line of research investigates robustness through adversarial or controlled text perturbations. A study presented GeepWordBug, a method for testing the weakness of text classification models. It makes small text changes to important words to alter model results. The study showed that even minor text modifications can significantly decrease classification model. The research is relevant to strength testing of sentimental-analysis models against modified text [12]. A method for testing – TextFooler introduced the strength of text classification models. It changes selected words while trying to preserve the original meaning and language rules of the text. The study showed that models such as BERT, CNN and RNN can be vulnerable to small text changes [13]. A study proposed BAE, a BERT – based method for generating challenging. It changes selected words while maintaining meaning and language-related standard as much as possible [14]. A framework for testing and improving the robustness of NLP models [15]. A study proposed CharBERT, a character-aware language model designed to handle character- level information [16]. The impact of intentional and unintentional text perturbations on NLP classification systems was studied [17]. A study determined the robustness of NLP models against OCR-generated text noise. The authors showed that models trained only on clean text can perform poorly on noisy inputs [18].

D. Noise in Classification and Research Gap

Noise has also been studied from a broader classification perspective. Research on noise learning in text classification has highlighted the importance of using controlled and clearly defined noise settings when comparing robustness. A study presented a benchmark for studying noise in text classification. The authors evaluated different noise-learning methods across multiple datasets and noise types. The study found that existing datasets may contain natural label noise, which can affect results. The researchers highlighted the importance of consistent noise definitions and controlled experiments when comparing models [19].

A study reviewed the impact of label noise on classification performance. Their study discussed different types of label noise and methods for making classifiers more tolerant to incorrect or unreliable labels. The research mainly focused on noisy labels rather than corrupted input text and provided a theoretical understanding of how noise can affect classification systems [20].

Research Gap:

This Literature review shows that traditional machine models have been broadly evaluated for sentimental classification, while noisy and adversarial text has been increasingly examined in robustness research. However, these two areas may not always study together. Especially,

Relatively few studies have focused on systematically comparing lightweight classifiers, including Multinomial Naïve Bayes, Logistic Regression and Linear SVM, using a common TF-IDF representation and the same character-level noise conditions. This study addresses this gap by evaluating these models on clean and systematically corrupted Amazon cell-phone reviews at 10%,20% and 30% noise severity.

III. RESEARCH METHODOLOGY

A. Problem Statement

Many sentiment classification models are tested on clean text. Real-world reviews may contain spelling mistakes, extra characters, swapped letter and abbreviations. These changes may affect the features extracted from the text and may reduce model performance.

The study focuses on checking how well lightweight machine learning models handle such noisy text. Logistic Regression, Multinomial Naïve Bayes and Linear SVM [1] are compared using the same TF-IDF features and their performance is measured on both clean and noisy reviews to identify which model remains more reliable when text loses its quality.

B. Scope of the Study

The study highlights binary sentiment classification of English Amazon cell-phone reviews. Three traditional machine learning models were considered:

1) Logistic Regression.
2) Multinomial Naïve Bayes.
3) Linear SVM.

This study uses TF-IDF with unigram - bigram features and evaluates five noise types: insertion, substitution, character deletion, swapping and abbreviation. Each noise type is tested at 10%,20%

and 30% severity. Model performance is compared by using F1 score, precision, recall, and Accuracy.
Hyperparameter tuning is also performed using 5- fold cross-validation to state whether optimized model parameter improve classification performance and robustness.
This study is limited to lightweight machine learning models and does not include transformer-based models or multilingual sentiment classification.

C. Dataset

This study focuses the Amazon Cell-Phone Reviews sentiment dataset, which contained 1,000 labelled reviews

Table 1: Dataset

Property	Description
Total Samples	1,000
Positive Samples	500
Negative Samples	500
Number Of Classes	2
Text Column	Text
Label Column	Label

D. Data Preprocessing

The data was first checked for duplicate reviews. During data cleaning, 10 exact duplicate reviews were found and removed. The final dataset contained 990 unique reviews: 497 negative and 493 positive reviews.

The dataset is divided in ratio of 80:20 as training and testing sets. This produced 792 training and 198 testing reviews.
This study uses TF-IDF to convert review text into numerical features. Unigrams and bigrams were included with maximum of 10000 features. This helped the models to use individual words as well as short word combinations.
The TF-IDF vectorizer was fitted only on the training data and then applied to the test data [4]. After preprocessing, The TF-IDF features were given to the three machine learning models. Then noise introduced into test reviews to examine how trained models performed under different noisy situations.

E. Traditional Machine Learning Models

There are three lightweight machine learning models were selected for this experiment:

a) Logistic Regression: used to classify reviews as positive or negative. it learns the relationship between the TF-IDF features and the sentiment class. It was included as a simple and effective baseline model.
b) Multinomial Naïve Bayes: usually used for text classification because it works well with word based features.
c) Linear SVM : it performs well with high dimensional text data. It separates positive and negative reviews by finding a suitable decision boundary between the two classes.

F. Noise Generation Methodology

Table 2 Textual Noise Types and Perturbation Operations

Noise Type	Description
Character Deletion	Removes selected characters
Character insertion	Inserts additional characters
Character substitution	Replaces characters
Character swapping	Exchanges neighboring characters
Abbreviation	Replaces selected expressions with shorter forms

Table 3: Noise Levels

Severity	Meaning
10%	Low Noise
20%	Moderate Noise
30%	High Noise

G. Hyperparameter Optimization

Hyperparameter Optimization is performed to check whether suitable parameter settings can improve the efficiency of the selected lightweight ML models. However, the main objective of this study is robustness analysis, hyperparameter tuning is treated as a secondary experiment rather than the primary contribution.
The training dataset uses 5-fold cross validation strategy. The test dataset is kept separate throughout the tuning process so that the final evaluation remained unbiased. F1 -score normally used as the major selection criterion for model optimization because it balances precision and recall and is appropriate for binary sentiment classification.

For Logistic Regression, regularization constraint 'C' is enhanced. Then the smoothing parameter 'alpha' is considered for Multinomial Naïve Bayes. For Linear SVM, the regularization constraint C is also optimized. Best values obtained through cross-validation were C=10 fro Logistic Regression, alpha = 0.5 for Multinomial Naïve Bayes and C = 10 for Linear SVM.

The corresponding cross-validation F1-scores were 0.8125,0.8349 and 0.8149 respectively. The optimized models were then evaluated under the same clean and noisy test conditions as the baseline models. This evaluation helped determine whether hyperparameter optimization could improve performance on clean data while also enhancing model robustness across different types and levels of textual noise.

Table 3 Hyperparameter Search Space and Optimal Values

Model	Hyperparameter	Best Value	CV F1Score
Logistic Regression	C	10	0.8125
Multinomial Naïve Bayes	alpha	0.5	0.8349
Linear SVM	C	10	0.8149

H. Evaluation Metrics

This study focuses on the performance of all three classifiers using Precision, Accuracy, Recall and F1-score [4]. The metrices were designed for both clean and noisy data.

Accuracy alone does not show how a model behaves with positive and negative sentiment classes that's why use of multiple evaluation measures is important.

TP = True Positive.
TN = True Negative.
FP = False Positive.
FN = False Negative.

1. $Accuracy\ score = \frac{TP+TN}{TP+TN+FP+FN}$
2. $\text{Precision score} = \frac{TP}{FP+TP}$
3. $\text{Recall score} = \frac{TP}{FN+TP}$
4. $\text{F1 score} = 2 * \frac{\text{Precision}+\text{Recall}}{\text{Precision} * \text{Recall}}$

I. Performance Degradation

Along with the standard classification metrics, the effect of noise on model performance is also examined. For each noisy condition, the F1-score obtained on clean data was compared with the corresponding F1-score obtained after noise was introduced. The difference is calculated as:

$F1_{Drop} = F1_{clean} - F1_{Noisy}$

A higher F1-score drop represents the model is more influenced by noise, while smaller drop suggests greater robustness. This measure is use to assess and compare the robustness of Logistic Regression, Multinomial Naïve Bayes and Linear SVM [1] across different noise types and severity levels.

J. Experimental Framework and Procedure

The Experiment Framework is designed to check the performance and robustness of three lightweight classifiers under textual noise. After Preprocessing, the review is presented using TF-IDF and evaluated with Logistic Regression, Multinomial Naïve Bayesand Linear SVM.

Every model is tested on cleaned data and examined under five noise types at 10%, 20% and 30% severity levels. Their noisy data performance is compared with the clean data results to evaluate robustness. Hyperparameter optimization is also performed using five-fold cross validation to check whether model tuning can improve performance and robustness under noisy conditions. The experimental framework is shown in Fig.1.

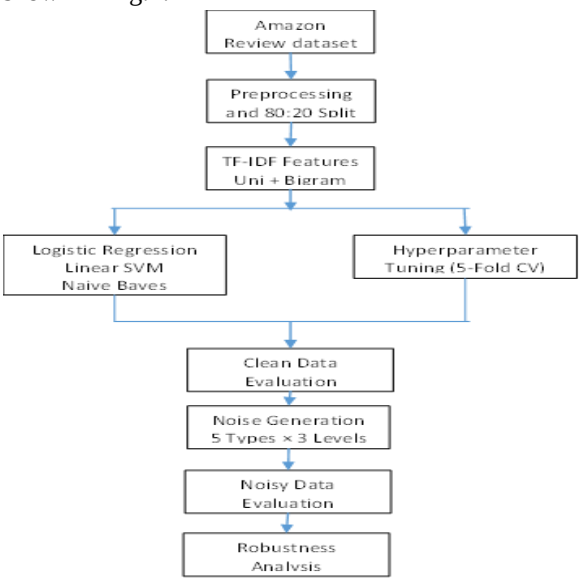

Figure 1 Experimental workflow of the proposed sentiment classification and robustness analysis.

IV. RESULTS AND DISCUSSION

A. Clean-Test Performance of Baseline Model

Table 4 Clean-Data Performance of Baseline Model Before Hyperparameter Tuning

Metric	Logistic Regression	Naive Bayes	Linear SVM
Accuracy	0.8182	0.8030	0.8182
Precision	0.8316	0.7941	0.8182
Recall	0.7980	0.8182	0.8182
F1-Score	0.8144	0.8060	0.8182

From this Result we can identify:
Highest Precision : Logistic Regression
Highest Recall : Multinomial Naïve Bayes
Highest F1 Score: Linear SVM

B. Overall Impact of Textual Noise

This result show that model performance consistently decreases as noise severity increases, with mean F1-score dropping from 0.8129 for clean data to 0.5798 at 30% noise, indicating that higher textual noise drastically decreases sentiment classification performance.

Table 5 Overall Model Performance Under Different Noise Severity Levels

Condition	Mean_Accuracy	Mean_Precision	Mean_Recall	Mean F1 score
Clean	0.8131	0.8146	0.8114	0.8129
10% Noise	0.7566	0.7811	0.7145	0.7451
20% Noise	0.6747	0.7106	0.5865	0.6407
30% Noise	0.6350	0.6736	0.5138	0.5798

C. Effect of Different Noise Types

Table 6 Impact of Different Types of Textual Noise on Model Performance

Noise Type	Average F1	Average F1 Drop
Character Deletion	0.6154	0.1974
Character Insertion	0.6217	0.1912
Character Substitution	0.6187	0.1942
Character Swapping	0.6361	0.1768
Abbreviation	0.7842	0.0287

This result shows character deletion caused the greatest performance degradation.

D. Model Robustness to Textual Noise

Before tuning, Multinomial Naive Bayes stood the most robust model according to F1-score, with the smallest F1 degradation (0.1437), while Logistic Regression showed the largest degradation (0.1701).

Table 7 Model Robustness Before Hyperparameter Tuning

Model	Accuracy_change	Precision_change	Recall_change	F1_score
Multinomial Naïve Bayes	0.1226	0.1006	0.1825	0.1437
Logistic_Regression	0.1226	0.0873	0.2135	0.1592
Linearism	0.1279	0.0907	0.2236	0.1701

Table 8 Model Robustness After Hyperparameter Tuning

Model	Accuracy_Drop	Precision_Drop	Recall_ Drop	F1_Drop
Multinomial Naïve Bayes	0.1387	0.1357	0.1677	0.1538
Logistic Regression	0.1182	0.0829	0.2114	0.1566
Linear SVM	0.1461	0.1297	0.2088	0.1743

After tuning, Multinomial Naïve Bayes remained the most robust according to F1-score (0.1538), whereas Linear SVM became the least robust with the highest F1 degradation (0.1743).

Table 9 Best Model at Each Noise Severity Level

Noise Level	Best Model Before Tuning	Best Model After Tuning
10%	Linear SVM (0.7629)	Logistic Regression (0.7581)
20%	Multinomial Naïve Bayes (0.6506)	Multinomial Naïve Bayes (0.6513)
30%	Multinomial Naïve Bayes (0.5934)	Multinomial Naïve Bayes (0.5991)

Linear SVM performed best at 10% noise before tuning, while Multinomial Naïve Bayes consistently performed best at the higher 20% and 30% noise levels both before and after tuning.

E. Effect of Noise Severity on Model Performance

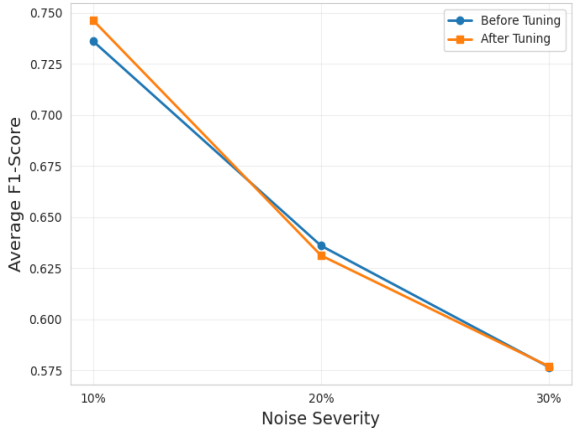

Figure 2 Average F1-Score Across Different Noise Severity Levels

The graph demonstrates a clear decline in F1-score as noise severity increases from 10% to 30%, confirming that increasing textual corruption causes cumulative degradation in sentiment-classification performance.

F. Effect of Hyperparameter Optimization

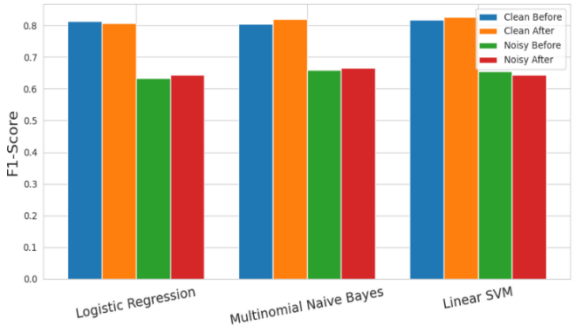

Figure 3 Clean and Noisy F1-Score Before and After Hyperparameter Tuning

Hyperparameter optimization produced model-dependent effects. Multinomial Naïve Bayes and Linear SVM improved their clean-data F1-scores from 0.8060 to 0.8205 and 0.8182 to 0.8265, respectively, whereas Logistic Regression decreased from 0.8144 to 0.8081. For noisy data, Naïve Bayes improved from 0.6623 to 0.6667, Logistic Regression improved from 0.6443 to 0.6515, while Linear SVM decreased from 0.6590 to 0.6523.

G. Overall Comparison of Model Robustness

Multinomial Naïve Bayes shows the lowest F1 degradation in both conditions, confirming it as the most robust model to textual noise formed on F1-score.

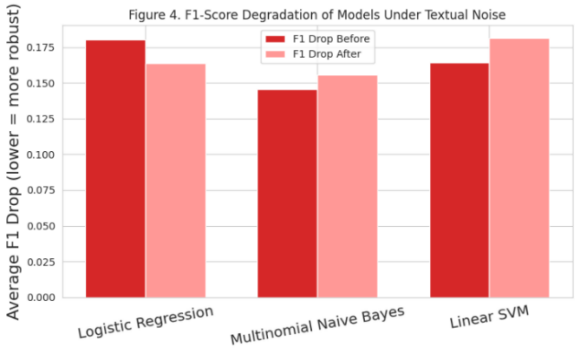

Figure 4 F1 score Degradation of Models Under Textual Noise

H. Discussion

This experiment results explains that textual noise has a considerable effect on lightweight sentiment classification models. Though on clean reviews, three models achieved comparable performance. When noise is introduced, their robustness has been differed. Among the evaluated models, the greatest robustness is achieved by Multinomial Naïve Bayes based on F1-score., also maintain the lowest performance loss before and after hyperparameter tuning. The result also explain that classification performance reduces when noise severity 10% to 30% increasing. Hyperparameter optimization improved robustness for some models but does not provide a universal improvement across all models.

V. CONCLUSION.

This study focuses the impact of textual noise on three lightweight sentiment classification models Logistic Regression, Multinomial Naïve Bayes and Linear

SVM [1]– using different types of noise and severity levels. The results show that noise Considerably reduces sentiment classification performance, with average F1 score decreasing from 0.8129 on clean data to 0.5798 at 30% noise. From the 5 noise types, Character deletion caused the greatest loss, with an average F1-acore drop of 0.1974, while abbreviation noise has the least impact (0.0287). In terms of robustness, Multinomial Naïve Bayes shows the strongest Toughness, achieving the lowest F1-score loss both before (0.1437) and after (0.1538) hyperparameter tuning. When noise severity increasing from 10% t0 20% and 30% consistently reduced model performance, confirming an Accumulative effect of textual corruption. By using hyperparameter tuning, it improved some models under clean and noisy conditions but does not provide consistent improvements among all three classifiers.

Overall, the results show that clean data performance alone is insufficient to determine model robustness testing under different types and levels of textual noise is essential when deploying lightweight sentiment classification models in real world condition.

VI. FUTURE SCOPE

This study currently evaluated smaller dataset it can be extended to larger and more diverse dataset. Additional types of real-world noise may be investigated in future work like punctuation changes, slang, code -mixed language. In current study, noise is mainly used for evaluation. Robustness-aware training can be explored. The study shows that tuning affects different models differently. Future work could examine wider parameter ranges, different techniques of optimization. Future research can further check model robustness using repeated experiments with different data split and random seeds, along with confidence intervals and statistical significance tests. Other possible goal is to develop a hybrid or adaptive approach in which the system first identifies the type or severity of noise and then selects a suitable classifier or preprocessing method.

REFERENCES

[1] S. Hasanli and A. Rustamov, “A comparison of machine learning algorithms for sentiment analysis of Azerbaijani tweets,” in *Proc. IEEE 13th Int. Conf. Application of Information and Communication Technologies (AICT)*, 2019, doi: 10.1109/AICT47866.2019.8981793.

[2] T. Rahat, M. Kahir, and M. Masum, “Comparison of Naive Bayes and SVM for sentiment analysis,” in *Proc. IEEE Smart Technologies for Smart Nation (SmartNations)*, 2019, doi: 10.1109/SMART46866.2019.9117512.

[3] M. Alam and J. Yao, “The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis,” *Computational Intelligence and Neuroscience*, 2018, doi: 10.1007/s10588-018-9266-8.

[4] M. Daraghmi and M. Zyadeh, “Sentiment classification of Amazon product reviews based on machine and deep learning techniques: A comparative study,” *Future Internet*, vol. 18, no. 3, p. 138, 2026, doi: 10.3390/fi18030138.

[5] J. McAuley and J. Leskovec, “Hidden factors and hidden topics: Understanding rating dimensions with review text,” in *Proc. 7th ACM Conf. Recommender Systems (RecSys)*, 2013, pp. 165–172, doi: 10.1145/2507157.2507163.

[6] Y. Belinkov and Y. Bisk, “Synthetic and natural noise both break neural machine translation,” *arXiv preprint arXiv:1711.02173*, 2018.

[7] V. Karpukhin, O. Levy, J. Eisenstein, and M. Ghazvininejad, “Training on synthetic noise improves robustness to natural noise in machine translation,” in *Proc. 5th Workshop on Noisy User-generated Text (W-NUT)*, 2019, doi: 10.18653/v1/D19-5506.

[8] V. Vaibhav, A. K. Singh, et al., “Understanding and improving robustness of neural machine translation to noisy inputs,” in *Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019, doi: 10.18653/v1/N19-1190.

[9] D. Pruthi, B. Dhingra, and R. J. Lipton, “Combating adversarial misspellings with robust word recognition,” in *Proc. 57th Annu. Meeting Association for Computational Linguistics (ACL)*, 2019, pp. 5582–5591, doi: 10.18653/v1/P19-1561.

[10] P. Rychalska, M. Wróblewska, et al., “Models in the wild: On corruption robustness of neural NLP systems,” in *Proc. 2019 Int. Conf. Artificial Intelligence and Soft Computing*, 2019, doi: 10.1007/978-3-030-36718-3_20.

[11] A. Naik, R. Ravichander, N. Rose, and A. H. H. Chang, "Stress test evaluation for natural language inference," in *Proc. 27th Int. Conf. Computational Linguistics (COLING)*, 2018.

[12] J. Gao, J. L., et al., "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *Proc. IEEE Security and Privacy Workshops*, 2018, doi: 10.1109/SPW.2018.00016.

[13] D. Jin, Z. Jin, J. Zhou, and P. Szolovits, "Is BERT really robust? A strong baseline for natural language attack on text classification and entailment," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 34, no. 5, 2020, pp. 8018–8025, doi: 10.1609/aaai.v34i05.6311.

[14] S. Garg and G. Ramakrishnan, "BAE: BERT-based adversarial examples for text classification," in *Proc. 2020 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2020, doi: 10.18653/v1/2020.emnlp-main.498.

[15] J. Morris, E. Lifland, J. Yoo, J. Grigsby, D. Jin, and Y. Qi, "TextAttack: A framework for adversarial attacks, data augmentation and adversarial training in NLP," in *Proc. 2020 Conf. Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 119–126, doi: 10.18653/v1/2020.emnlp-demos.16.

[16] R. Ma, X. Wang, Y. Zhang, et al., "CharBERT: Character-aware pre-trained language model," in *Proc. 28th Int. Conf. Computational Linguistics (COLING)*, 2020, doi: 10.18653/v1/2020.coling-main.4.

[17] S. Bhalerao, et al., "Adversarial text perturbations for robust NLP classification," *arXiv preprint arXiv:2202.09483*, 2022.

[18] Y. Xu, X. et al., "Robust NLP models to OCR-generated text noise," *arXiv preprint arXiv:2107.07113*, 2021.

[19] X. Liu, X. et al., "Noise learning for text classification," in *Proc. 29th Int. Conf. Computational Linguistics (COLING)*, 2022, pp. 4557–4567.

[20] S. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014, doi: 10.1109/TNNLS.2013.2292894.