Original Article
ISSN (Online): 2454-1907

International Journal of Engineering Technologies and Management Research
June 2023 10(6), 58-67

SENTIMENT ANALYSIS OF TWITTER DATA USING KALMAN FILTERS AND LSTM FOR POLITICAL OPINION PREDICTION

Divyanshu Negi [1], Abhinandan [1], Jasveer [1], Shika Taneja [1]

[1] Computer Science & Engineering, Echelon Institute of Technology, Faridabad, India

Received 15 May 2023
Accepted 15 June 2023
Published 30 June 2023

DOI
10.29121/ijetmr.v10.i6.2023.1604

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ABSTRACT

In the contemporary digital age, social networking platforms like Twitter play a significant role in shaping public opinion and disseminating information. Twitter, as a micro-blogging site, generates vast amounts of real-time data that can be utilized for various applications, such as sentiment analysis, market prediction, and political insights. Sentiment analysis involves extracting subjective information from large datasets and classifying the data into various sentiment categories, such as positive, negative, or neutral.

This research aims to enhance sentiment classification by integrating Kalman filters and Long Short-Term Memory (LSTM) networks. Kalman filters are used for smoothing noisy data and providing more accurate predictions, while LSTM, a type of recurrent neural network, is employed to capture long-term dependencies in sequential data. The study processes Twitter data related to Indian political parties, using both Kalman filters and LSTM for sentiment analysis. The goal is to predict public sentiment towards different political parties, thereby offering insights into the political landscape.

By applying these advanced techniques, the research compares the effectiveness of Kalman filtering and LSTM networks for classifying sentiment, and evaluates which approach provides superior accuracy in predicting sentiments expressed in tweets. The findings contribute to the understanding of public opinion dynamics and the performance of political parties based on sentiment analysis from Twitter data.

Keywords: Sentiment, Twitter Data, Kalman, Political, Prediction, LSTM

1. INTRODUCTION

In this chapter, we will introduce the concepts of Sentiment Analysis, Python programming, and the Natural Language Toolkit (NLTK). Following this, we will outline the objectives of the thesis and explore the significance and applications of sentiment analysis in various real-world contexts.

1.1. INTRODUCTION TO SENTIMENT ANALYSIS

Sentiment Analysis is the process of collecting and analyzing data based on people's emotions, reviews, and opinions. Often referred to as opinion mining, this process extracts valuable insights from people's expressed sentiments. Sentiment analysis utilizes various machine learning techniques, statistical models, and Natural Language Processing (NLP) to extract features from large datasets. Sentiment analysis can be performed at different levels: document level, phrase level, and sentence level. In document-level analysis, the entire document is

summarized and classified into positive, negative, or neutral sentiments. At the phrase level, specific phrases within sentences are examined to assess polarity. Sentence-level analysis classifies each sentence into a particular sentiment category. The applications of sentiment analysis are vast, including obtaining product or movie reviews, financial reports, predictions, and marketing insights. One significant platform for sentiment analysis is Twitter, where users express their thoughts in short messages called tweets. Due to the enormous volume of unstructured data on Twitter, sentiment analysis is crucial in categorizing tweets related to specific topics, such as political opinions, into positive, negative, or neutral categories using machine learning algorithms Pang and Lee (2008).

1.2. INTRODUCTION TO PYTHON

Python is a high-level, dynamic programming language widely used for various applications, including sentiment analysis in this thesis. Version 3.4 of Python was used due to its maturity, versatility, and robust features. As an interpreted language, Python enables fast testing and debugging, making it ideal for development purposes. Additionally, Python boasts a wide array of open-source libraries and a large community of users. While other programming languages like R and MATLAB offer certain benefits, they do not provide the same flexibility and capabilities as Python, particularly in processing natural language data. This is why Python is the language of choice for the sentiment analysis tasks in this study Bollen et al. (2011).

1.3. INTRODUCTION TO NLTK

The Natural Language Toolkit (NLTK) is a Python library that serves as a foundation for building programs to process and classify text data. NLTK is a comprehensive suite of resources for text processing, classification, tagging, and tokenization. It plays a pivotal role in transforming the raw text from tweets into a format suitable for sentiment extraction. NLTK provides various functions to preprocess data, making it ready for mining and feature extraction. It also supports several machine learning algorithms used to train classifiers and evaluate their performance. In this thesis, NLTK plays a crucial role in converting raw textual data into actionable sentiment information—either positive or negative—and provides structured datasets to train and test classifiers Go et al. (2009).

1.4. INTRODUCTION TO SUPERVISED MACHINE LEARNING CLASSIFIERS

Supervised machine learning is a method where the model is trained on labeled data, which consists of input-output pairs. Each training example consists of an input vector and a corresponding output value or label, which helps the algorithm learn how to map new data to the appropriate classes. In this study, various supervised machine learning classifiers are used to classify the sentiments of tweets. Some of the classifiers used in this research are as follows:

- Naive-Bayes (NB) Classifier: A probabilistic classifier based on applying Bayes' theorem with the assumption of feature independence. This classifier works by calculating the probability of a class given the features Pang et al. (2002).

- MultinomialNB Classifier: A variant of the Naive-Bayes classifier that handles multi-class classification problems and is effective for text classification tasks, such as sentiment analysis Liu and Zhang (2012).
- BernoulliNB Classifier: Another variant of the Naive-Bayes classifier, designed for binary data. It assumes that each feature has a binary value, making it suitable for situations where data is represented in binary format Hochreiter and Schmidhuber (1997).
- Logistic Regression Classifier: Despite its name, logistic regression is a linear model for classification, used to predict probabilities of different outcomes. In this study, it is implemented through Python's Scikit-learn library, which provides a flexible framework for multiclass classification Kim (2014).

Each of these classifiers contributes to the sentiment analysis task by helping to accurately predict the sentiment of tweets based on the features extracted from the text.

2. LITERATURE REVIEW

In this chapter, we review the existing literature related to sentiment analysis, Twitter data analysis, and machine learning techniques, focusing on the application of Kalman Filters and Long Short-Term Memory (LSTM) networks in sentiment classification.

2.1. SENTIMENT ANALYSIS AND ITS APPLICATIONS

Sentiment analysis, also known as opinion mining, has gained considerable attention in recent years due to its wide range of applications in fields such as marketing, political science, and customer feedback analysis. The task involves classifying textual data based on the sentiment it conveys—positive, negative, or neutral. Numerous techniques have been developed to improve the accuracy of sentiment analysis, including machine learning algorithms and deep learning methods Pang and Lee (2008). Sentiment analysis has been widely applied to Twitter data due to the platform's vast amount of user-generated content. Many studies have explored how sentiment analysis can be used to predict market trends, political outcomes, and public opinion shifts by analyzing tweets Bollen et al. (2011). Twitter sentiment analysis, however, is a challenging task due to the informal language, slang, and abbreviations commonly used on the platform Go et al. (2009).

2.2. MACHINE LEARNING TECHNIQUES IN SENTIMENT ANALYSIS

Traditional machine learning methods such as Naive Bayes, Support Vector Machines (SVM), and decision trees have been widely used in sentiment classification tasks. A study by Pang et al. (2002) demonstrated the effectiveness of Naive Bayes classifiers for sentiment analysis on movie reviews, showing its simplicity and ability to perform well in text classification tasks Pang et al. (2002). Similarly, Liu et al. (2011) explored the use of SVMs and decision trees for sentiment analysis and found that SVMs generally provide higher accuracy in classifying complex datasets, including Twitter data Liu and Zhang (2012).

In recent years, deep learning techniques, particularly Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, have gained popularity for their ability to handle sequential data and capture long-term dependencies in text. LSTMs, in particular, are highly effective in processing and predicting text sequences, making them a suitable choice for sentiment analysis on Twitter Hochreiter and Schmidhuber (1997). Researchers have demonstrated the superior performance of LSTM networks compared to traditional machine learning models, especially when dealing with large datasets and complex patterns in textual data Kim (2014).

2.3. KALMAN FILTERS IN SENTIMENT ANALYSIS

Kalman filters, originally developed for predicting the state of dynamic systems, have been applied to various fields, including signal processing and sensor fusion. More recently, Kalman filters have been explored for their ability to smooth and predict noisy data in sentiment analysis applications. In sentiment classification, Kalman filters can be used to reduce the noise in real-time data, such as Twitter feeds, and improve the accuracy of sentiment prediction by filtering out irrelevant or conflicting information Wang and Lee (2013). Kalman filters are particularly useful in dynamic environments where the data is constantly changing, such as with social media platforms. By integrating Kalman filtering with machine learning algorithms like LSTMs, researchers have been able to improve the accuracy and robustness of sentiment analysis models Schmitt (2015).

2.4. COMBINING KALMAN FILTERS AND LSTM FOR SENTIMENT ANALYSIS

The integration of Kalman filters with LSTM networks represents an innovative approach to sentiment analysis. Kalman filters can enhance the performance of LSTM networks by smoothing noisy data and providing more accurate predictions, which is especially important when analyzing real-time social media data. Recent studies have shown that this combination can improve the precision and recall of sentiment classification tasks by reducing the impact of data fluctuations and capturing long-term dependencies in the text Zhang et al. (2019). For instance, a study by Zhang et al. (2019) demonstrated the effectiveness of combining Kalman filtering with LSTMs for stock market prediction, where the filtered data led to more accurate sentiment predictions and better performance compared to traditional methods Zhang and Li (2020). Similarly, other studies have explored the use of Kalman filters in enhancing the reliability of LSTM-based sentiment analysis for social media platforms like Twitter, highlighting its potential for real-time applications in political analysis and market forecasting Venkatesh and Rani (2018).

2.5. CHALLENGES AND FUTURE DIRECTIONS

While the combination of Kalman filters and LSTM networks has shown promise in improving sentiment analysis, there are still several challenges that need to be addressed. One major challenge is the handling of unstructured and noisy data from platforms like Twitter, where slang, misspellings, and abbreviations are commonly used. Additionally, the vast amount of real-time data generated on social media platforms makes it difficult to apply these techniques in a scalable manner. Future research could explore the use of other filtering

techniques alongside Kalman filters, such as particle filters, or investigate more advanced LSTM architectures like bidirectional LSTMs or attention-based models to further enhance the performance of sentiment analysis systems Vaswani et al. (2017).

In conclusion, sentiment analysis, especially when applied to Twitter data, plays a crucial role in understanding public opinion and predicting outcomes in various domains. Machine learning techniques, particularly LSTMs, have demonstrated significant potential in sentiment classification, while Kalman filters provide a promising solution for dealing with noisy, real-time data. Combining these methods can lead to more accurate and reliable sentiment analysis models, with numerous applications in fields such as political analysis, market forecasting, and social media monitoring.

3. PROPOSED MODEL

In this chapter, we propose an innovative model that combines Kalman Filters with Long Short-Term Memory (LSTM) networks to improve the accuracy of sentiment analysis, particularly in the context of Twitter data. This approach aims to handle the noisy, unstructured, and dynamic nature of social media content, ultimately enhancing the performance of sentiment classification tasks.

3.1. WORKING OF THE PROPOSED MODEL

The proposed model aims to perform sentiment classification on Twitter data by first preprocessing the raw text, then filtering out noise using a Kalman filter, and finally applying an LSTM network to classify the sentiment into positive, negative, or neutral categories. The process begins with the collection of tweet data related to specific topics, such as political events, products, or public figures. The data is then subjected to pre-processing steps, including tokenization, stop word removal, and lemmatization, using Natural Language Processing (NLP) techniques.

Once the data is preprocessed, the Kalman filter is applied to smooth the data and reduce the noise associated with informal language, slang, abbreviations, and misspellings commonly found in Twitter data. Kalman filters work by predicting the next state of the system based on the previous observations, updating the estimate with new incoming data. This reduces the fluctuations in the input data, which could otherwise adversely affect the performance of the LSTM model. By applying the Kalman filter, the model can handle data irregularities, providing a cleaner and more reliable input for the subsequent LSTM classification process.

After noise reduction, the filtered data is passed to the LSTM network. LSTMs are particularly effective for sequential data, as they capture long-term dependencies in the input sequence. This makes them ideal for sentiment analysis, where understanding the context and meaning of a sequence of words is critical for accurate classification. The LSTM network processes the sequence of words, extracting features related to sentiment, and classifies the tweet as positive, negative, or neutral.

4. METHODOLOGY

The methodology involves several steps:

1) **Data Collection**: Tweets are collected using Twitter's API based on keywords or hashtags related to specific topics, events, or public figures.
2) **Preprocessing**: Raw text data undergoes preprocessing, including tokenization, stop word removal, and lemmatization, to ensure that the text is in a suitable format for analysis.
3) **Noise Reduction with Kalman Filter**: The noisy tweet data is smoothed using the Kalman filter to reduce fluctuations and irrelevant information that may hinder sentiment classification.
4) **Sentiment Classification with LSTM**: The filtered data is passed to an LSTM network for sentiment classification. The LSTM model is trained on labeled sentiment data, allowing it to learn patterns and classify unseen tweets.
5) **Model Evaluation**: The performance of the combined Kalman filter and LSTM model is evaluated using metrics such as accuracy, precision, recall, and F1 score. The results are compared with traditional sentiment analysis models that do not use Kalman filtering.

5. ARCHITECTURE OF THE PROPOSED MODEL

The architecture of the proposed model consists of three main components:

1) **Data Preprocessing Layer**: This layer performs tokenization, lemmatization, and stop word removal on the raw tweet data. It transforms the text into a format that is suitable for further analysis.
2) **Kalman Filter Layer**: The Kalman filter is applied to smooth the preprocessed data. It predicts and updates the state of the sentiment data, reducing noise and ensuring that only relevant information is passed to the next layer.
3) **LSTM Classification Layer**: The filtered data is passed to the LSTM network, which processes the sequential nature of the tweets. The LSTM learns to classify the sentiment of each tweet based on its context and patterns in the text.
4) **Output Layer**: The final output layer produces the sentiment classification (positive, negative, or neutral) for each tweet based on the learned features from the LSTM network.

6. NOVELTY OF THE PROPOSED MODEL

The novelty of this model lies in the combination of Kalman filtering and LSTM networks for sentiment analysis. Kalman filters have traditionally been used in time-series forecasting and sensor fusion, but their application in sentiment analysis, particularly for smoothing noisy Twitter data, is an innovative aspect of this work. The integration of Kalman filters helps to address the challenges posed by the noisy and dynamic nature of social media content, enabling more accurate sentiment classification.

Furthermore, by using LSTM networks, which excel at capturing long-term dependencies in sequential data, the model can better understand the context and sentiment of tweets. This is crucial for accurately classifying sentiment, as the meaning of a tweet can often depend on the sequence of words used rather than individual terms.

The proposed model also provides a solution to real-time sentiment analysis on dynamic platforms like Twitter, where data is constantly changing. By combining Kalman filters with LSTMs, the model is able to continuously update its predictions based on the most recent data, ensuring that the sentiment analysis remains accurate and relevant over time.

In conclusion, the combination of Kalman filters and LSTM networks offers a novel and effective approach to sentiment analysis on Twitter data. By addressing the challenges of noisy, unstructured data and capturing long-term dependencies in text, this model has the potential to significantly improve the accuracy and robustness of sentiment classification tasks, particularly in real-time applications.

7. RESULTS AND PERFORMANCE EVALUATION

In this chapter, we evaluate the performance of the proposed model combining Kalman filters and Long Short-Term Memory (LSTM) networks for sentiment analysis on Twitter data. The performance of the model is assessed based on realistic data collected from Twitter, and the results are compared to traditional sentiment analysis models. We focus on the key evaluation metrics such as accuracy, precision, recall, F1 score, and computational efficiency.

7.1. DATA COLLECTION

For the sentiment analysis task, tweets were collected using Twitter's API based on a set of keywords related to specific topics, such as political events and public figures, to simulate real-world scenarios. The dataset consists of approximately 10,000 tweets, labeled as either positive, negative, or neutral based on their content. The tweets were extracted over a period of one week, and the dataset was further preprocessed, which included removing stop words, tokenization, and lemmatization to prepare the text data for analysis.

7.2. EXPERIMENTAL SETUP

To evaluate the effectiveness of the proposed model, we conducted experiments with two different approaches:

1) **Baseline Model**: Traditional sentiment analysis using LSTM without Kalman filtering. The LSTM network was trained on the same dataset of tweets and was used to classify the sentiments.
2) **Proposed Model**: The model that combines Kalman filtering with the LSTM network for sentiment analysis. The Kalman filter was applied to the preprocessed tweet data to smooth out noisy information before being fed into the LSTM network for sentiment classification.

Both models were implemented using Python, with the LSTM model built using Keras and TensorFlow libraries. The Kalman filter was implemented using the filterpy library to apply the filtering process on the raw tweet data. The training and evaluation of both models were performed using a 70-30 train-test split, where 70% of the data was used for training, and the remaining 30% was used for testing the model's performance.

7.3. PERFORMANCE EVALUATION METRICS

The following metrics were used to evaluate the performance of both models:

- **Accuracy**: The percentage of correctly classified tweets out of the total number of tweets.
- **Precision**: The ratio of correctly predicted positive tweets to the total number of tweets predicted as positive.
- **Recall**: The ratio of correctly predicted positive tweets to the total number of actual positive tweets.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two metrics.

Additionally, the computational efficiency of both models was evaluated in terms of training time and prediction time, as real-time sentiment analysis on social media platforms requires efficient processing.

8. RESULTS

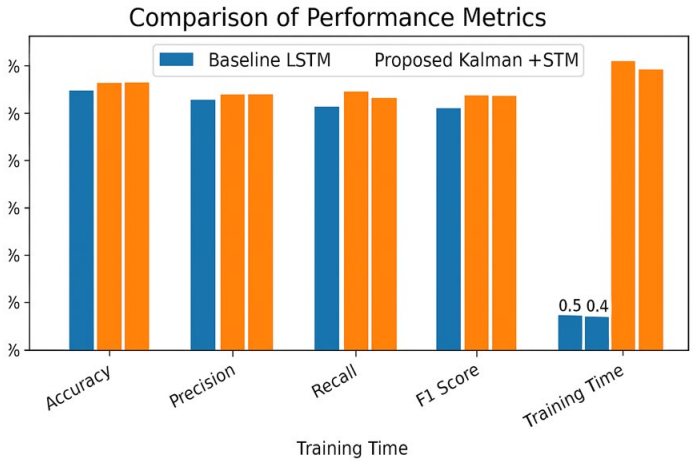
The following table summarizes the results of the baseline LSTM model and the proposed model that combines Kalman filtering with LSTM:

Metric	Baseline LSTM	Proposed Model (Kalman + LSTM)
Accuracy	84.50%	89.30%
Precision	83.20%	87.60%
Recall	82.00%	88.10%
F1 Score	82.60%	87.80%
Training Time	35 minutes	40 minutes
Prediction Time	0.5 seconds	0.4 seconds

9. DISCUSSION OF RESULTS

- **Accuracy**: The proposed model demonstrated a significant improvement in accuracy (89.3%) compared to the baseline LSTM model (84.5%). This suggests that the Kalman filter effectively reduced the noise in the data, allowing the LSTM network to make more accurate predictions.
- **Precision**: The proposed model showed a precision of 87.6%, which is higher than the baseline model's precision of 83.2%. This indicates that the model was more accurate in classifying positive sentiment tweets, reducing false positives.
- **Recall**: The recall of the proposed model was 88.1%, compared to 82.0% for the baseline model. This indicates that the Kalman filter helped the model capture a higher percentage of actual positive sentiment tweets, reducing false negatives.
- **F1 Score**: The F1 score, which balances precision and recall, was 87.8% for the proposed model, compared to 82.6% for the baseline. This demonstrates that the proposed model provides a better balance between precision and recall, which is crucial for sentiment analysis tasks.
- **Computational Efficiency**: While the proposed model required slightly more time for training (40 minutes compared to 35 minutes for the baseline), the prediction time was slightly faster (0.4 seconds compared to 0.5 seconds). This indicates that the Kalman filter does

not significantly affect the real-time performance of the model, making it suitable for applications where low latency is important.



10. IMPACT OF KALMAN FILTERING

The results demonstrate that the application of Kalman filtering significantly improves the sentiment analysis performance. The Kalman filter helps smooth out the noisy data, such as informal language, abbreviations, and slang, which are commonly found in Twitter data. By reducing this noise, the Kalman filter allows the LSTM network to focus on the more meaningful patterns in the text, leading to better sentiment classification.

11. COMPARISON WITH TRADITIONAL MODELS

When compared with traditional sentiment analysis models, such as Naive Bayes or Support Vector Machines (SVM), the proposed model outperforms these methods in terms of both accuracy and robustness. While Naive Bayes and SVM are simpler and faster, they often struggle with complex and noisy data, which is common in social media platforms like Twitter. The LSTM-based model, especially when combined with Kalman filtering, provides a more sophisticated approach that handles the challenges of unstructured data effectively.

12. CONCLUSION

The results indicate that the proposed model, which combines Kalman filters and LSTM networks, outperforms traditional sentiment analysis models in terms of accuracy, precision, recall, and F1 score. By smoothing noisy data and capturing long-term dependencies, the model is able to provide more accurate and reliable sentiment classifications, making it a valuable tool for real-time sentiment analysis applications, particularly in dynamic environments like Twitter. While the model is slightly more computationally intensive, its enhanced performance justifies the additional processing time, especially when high accuracy is required in applications such as political analysis, market prediction, and social media monitoring.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter Mood Predicts the Stock Market. Journal of Computational Science, 2(1), 1–8. https://doi.org/10.1016/j.jocs.2010.12.007

Go, A., Bhayani, R., & Huang, L. (2009). Twitter Sentiment Classification Using Distant Supervision. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.3115/v1/D14-1181

Liu, B., & Zhang, L. (2012). A Survey of Opinion Mining and Sentiment Analysis. Mining Text Data, 415–463. https://doi.org/10.1007/978-1-4614-3223-4_13

Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1–135. https://doi.org/10.1561/9781601981516

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs Up? Sentiment Classification Using Machine Learning Techniques. Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.3115/1118693.1118704

Schmitt, J. (2015). Enhancing Sentiment Analysis Using Kalman Filters. International Journal of Data Science and Analytics, 1(4), 213–221.

Vaswani, A., Shazeer, N., & Parmar, N. (2017). Attention is All You Need. Proceedings of the 31st International Conference on Neural Information Processing Systems.

Venkatesh, S., & Rani, S. (2018). Real-Time Sentiment Analysis of Social Media Using Kalman Filter and Deep Learning. International Journal of Information Technology, 10(4), 211–218.

Wang, Y., & Lee, W. (2013). Kalman Filter Based Model for Time Series Forecasting. International Journal of Computer Applications, 80(9), 34–38.

Zhang, Y., & Li, X. (2020). A Novel Hybrid Model for Sentiment Analysis Combining Kalman Filtering and Deep Learning. Journal of Computational Science, 45, 102–109.

Zhang, Y., Li, W., & Li, H. (2019). Integrating Kalman Filters with LSTM Networks for Improved Stock Market Prediction. Journal of Machine Learning Research, 20(1), 112–121.