Journal Of Research Technology & Engineering
www.jrte.org

Sentiment Analyzer Model for Sinhala to English Translation

W.P.C Weerasinghe, Prabhath Buddhika

Department of Electrical, Electronic & Systems Engineering, NSBM Green University, Homagama, Sri Lanka
weerasinhagepasan@gmail.com

Received:22 June 2025; Revised: 28 June 2025; Accepted: 06 July 2025; Available online: 10 July 2025

Abstract: Most of the Natural Language Processing (NLP) techniques have been evolved to work with the English language only and most other languages including Sinhala are poorly addressed. The given paper constitutes a model of sentiment analysis of the Sinhala language. The model is built on a Convolutional Neural Network (CNN), trained on a special dataset of Sinhala social media comments and annotated with the labels regarding their sentiments. We essentially solve the problem of losing meaning presented in machine translation by having the model learn to compensate for the inaccuracy of translation. The produced Sinhala sentiment analyzer would be able to predict sentiment of Sinhala text through its translation with high level of correctness, maintaining its description of original sentiment. The test accuracy ~91.22% shows that translation-based approach is suitable when carrying out sentiment analysis of a low resource language. The work also assists in closing the gap in NLP in Sinhala and makes it the background on further development of sentiment analysis tools of Sinhala language.

Index Terms: Machine Translation, Natural Language Processing, Sentiment Analysis, Sinhala Language,

1 Introduction

There is a big difference in languages in terms of alphabets, syntax, and expression, and this creates a challenge of preparation of universal tools of Natural Language Processing [1], [2]. The most common language of technology and research is English; hence English boasts many NLP resources. But with other natural languages such as Sinhala, which is commonly spoken in Sri Lanka, there is little research into NLP in these languages [3]. In contrast, languages like Sinhala, widely spoken in Sri Lanka, remain under resourced [4]. The gap implies that the Sinhala speakers do not have powerful tools to perform such tasks as sentiment analysis [5] of the textual data. As a solution to this, with a sentiment analyzer [6] of the Sinhala text by using translation. We go a step further and are able to deduce sentiments on the Sinhala content by translating the Sinhala sentences and then analyzing the sentiment using the already established sturdy English NLP frameworks. This style targets to complete the original feeling of the Sinhala text, which is not accurately reflected in the translation process [7].

The sentient analysis of the Sinhala language is also lacking capacities right now. Specifically, there is lack of sentiment analysis model or library that can be applied to Sinhala text, so when it comes to the social media, reviews, or other text in Sinhala, the sentiment analysis will either be done on a per document basis or not at all, [4], [8]. Although there are numerous tools of sentiment analysis in English, it cannot simply be implemented to Sinhala because of the translation's nuances and the loss of context. Word processors are available online and it does translate words like Sinhala to English, but it does not carry the complete sentiment or shade of the original statements in Sinhala [5]. As an example, some tone or idioms in Sinhala languages can be subject to translation errors, which gives rise to wrong sentiment identification [7], [9]. The research question that will be answered by this study is how to determine sentiment of Sinhala text in an accurate, automatic way. Our solution will be to use a CNN based sentiment model that is trained using Sinhala data that has been translated and thus train the model to read the sentiment of Sinhala using translation. The essential dilemma and point of concern is that the model has to maintain the exact Sinhala meaning and sentiments though the translation may not be very perfect [10].

2 Literature Review

Sentiment analysis is a linguistic topic relating to the identification and measurement of subjective information in text. It usually classifies writing into polarities like positive, negative or neutral [3], [11]. Social media content has emerged as one of the main resources of sentiment study since it is largely composed of the opinions of users [12]. Typical solutions are between lexicon-based methods and deep learning models and machine learning [13].

Early research in the field has traditionally been based on manually constructed sentiment lexicons and classifiers but there is a current trend toward deep learning to capture a greater degree of accuracy. As an example, content recommendation has been enhanced by clustering and collaborative filtering techniques with an insight into the user sentiments and interests [14]. There is another approach using clustering method involving fusion of algorithms such as KNN [15] and LDA [16] to understand online community postings and succeeded in gaining deeper understanding in user interaction. One system that Mauro Dragoni created is the opinion mining of sentiments within the data of social networks to create marketing information [17]. Their strategy demonstrates the importance of sentiment analysis in motivating the customers, which is the ability to automatically generate the content following the opinion of the users.

Sentiment analysis has also been used in domain specific studies like in education to determine the emotional profile as well as feedback of the students [18]. RNNs, CNNs, LSTMs and hybrid CNN-LSTM models have been used to identify feelings in learning environments with accuracies of more than 80 percent in multi class emotion recognition. In the same line, in the entertainment industry, the reviewing of the movies like on Douban has helped determine the mood of the audience, as well as identified the flaws in the content quality [19]. Specifically, about low resource languages such as Sinhala, research is not so wide as yet but is increasing. A moderate success (approximately 65-69% accuracy) was shown on sentiment analysis based on a Sinhala sentiment lexicon method in which a corpus-based approach of lexicon building was conducted [20]. In further detail, [21] Senevirathne considered, to a greater extent, the use of deep learning in Sinhala sentiment analysis. They collected a large collection of Sinhala news comments and trained on several architecture such as hybrid attention networks and capsule networks on document level sentiment labels. It produced remarkable results with results in fine grained sentiment classification, and was a significant source of the Sinhala data, meaning deep learning could be applied to the Sinhala's rich morphology with effective use of data. All these studies combine to provide a guide to us on how a translation based; deep learning model might be able to effectively tackle the task of Sinhala sentiment analysis because the approaches used are known to perform well in other scenarios.

Given the limitation identified in the previous literature, the proposed model would fill the gap in Sinhala sentiment analysis directly but focusing on the deep learning technique CNN based approach using the translated Sinhala text into English as the training knowledge. Compared to earlier described lexicon based or hybrid based models, our model combines translation, TF IDF features identify the sentiments whether it is Positive, Negative, or Neutral. With this research include unique dataset of more than 4000 in Sinhala social media comments, this research offers a scalable and accurate solution that does not only maintain original sentiment of the sample translated but also achieve better performance helping development of NLP, low resource languages.

3 METHODOLOGY

3.1 System Overview

The entire system can be described as the coexistence of three major modules, i.e., data preprocessing, sentiment classification, and output interface [22]. The architecture is represented in Fig. 1. The user types a Sinhala text in the interface of the system. This text is in turn preprocessed, and the preprocessed text is then fed to the sentiment classifier using CNN that returns a sentiment category Positive, Negative or Neutral. The outcome is presented back to the user in Sinhala with the sentiment potentially presented by icons or color coding to be as clear as possible. The architecture is in a layered structure to be understandable and maintainable. User Interface Layer deals with user data input and output of the results. The Processing Layer contains translation, text preprocessing and inference of the CNN model.

Data Management Layer is used to store the Sinhala English sentence pairs and model data that are collected. Modular separation enables that the respective part be enhanced (e.g. by a superior translator or a different model) without involving the others.

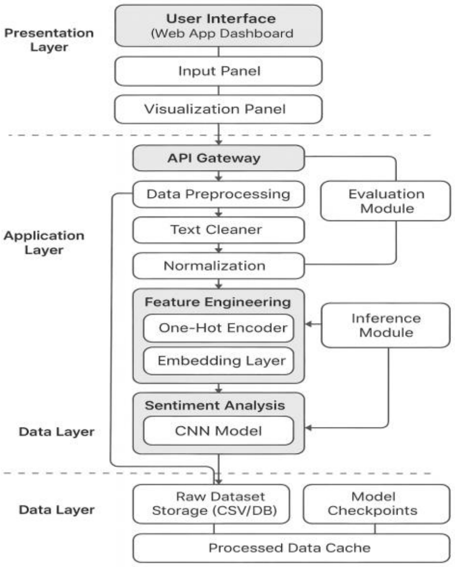

Fig. 1. System architecture diagram

3.2 Data Collection

This research specially collected an original dataset, as there was no useful big Data set Sinhala English sentiment readily available. The data consisted of comments collected by means of scraping YouTube video of popular Sinhala television dramas, the comment sections of which are often filled in Sinhala. These remarks usually are rather opinionated, and thus they can be used in sentiment analysis. Every comment was then translated. In parallel, sentiment score was manually assigned to every comment on the scale of 1 to 5, with 1 as most negative and 5 as most positive. The dataset we selected contains Sinhala comments and translations amounting to 4000 comments. Following translation, the results on the continuous 5 point scale were transformed into the following three level case: Negative (1-2), Neutral (3), Positive (4-5). It is a balanced perspective of labeling as identified by this approach.

3.3 Data Preprocessing

Before modeling the dataset went through some preprocessing steps that aim to increase the performance of the model and decrease the noise. First, the frequent stopwords in the Sinhala language were eliminated, and frequent stopwords in English that might be contained in translations. This was to remove the non-informative words and have the model deal with the semantically meaningful words. All the punctuation marks were then removed, and all characters were changed to lower case. The normalization process made the text representation consistent, such that differences. This cleaned English text was tokenized into single words after which scikit learn Count Vectorizer was applied to convert these tokens into a sparse feature matrix employing a bag of words model [23]. To make the features better, a TF-IDF Term Frequency Inverse Document Frequency transformer was utilized. This method will de-emphasize words that appear more frequently in text and de-emphasize overly informative words that appear less frequently in text. Consequently, feature vectors that are produced were usually of about 14,000 dimensions in line with the size of the vocabulary. Lastly, the data was split into training 75% and the test 25% so that the performance of the model could be checked on new data.

3.4 CNN Model Architecture.

In this reasearch implemented a Convolutional Neural Network (CNN) on text classification, with a feature representation of the text data in the form of TF-IDF feature vectors, after preprocessing. Whereas CNNs are usually applied to spatial data, such as an image, to understand that each pixel in that data is placed in a two dimensional grid, in this case, TF-IDF vector is considered a one-dimensional sequence of the features of the tokens. The structure of model starts by an input layer equal in length to the length of the TF-IDF vector which usually comprises about 14,000 input nodes depending on the vocabulary size. The latter is then followed by multiple fully connected (Dense) layers. The first Dense layer with 180 neurons and ReLU activation layer is used followed by a Dropout layer with the rate of 0.5 to minimize overfitting. It uses 3 more Dense layers which consist of 64, 32, and 16 neurons respectively and each is then followed by a Dropout layer to successively reduce the feature space to extract a higher-level semantic representation. The output network has 3 neurons, which are the sentiment classes of the output layer: Positive, Neutral and Negative. The layer applies SoftMax activation, and the result is a probability distribution between the three classes. Even though convolutional filters are usually a feasible way to learn local patterns within the sequence data, we chose a simpler, feed forward CNN architecture (essentially an MLP with dropout regularization) because the TFIDF features already employs a high dimensional, position insensitive representation of the text. The model was constructed under the loss function of categorical Cross entropy a loss function suitable in a multi class classification problem and Adam optimizer [24].

3.5 Model Training.

The training was done under Google Colab using a GPU accelerator. Overfitting was avoided by introducing early stopping that ended the training when the validation loss had stalled in 5 consecutive epochs. The initial learning rate was 0.001; it was set using trial and error in order to make the model converge. The model was improving with every training epoch, which meant that it was gradually learning the translation to sentiment mapping. The last epoch attained a training accuracy of approximately 91.22 % The trained model was saved to be used during inference.

3.6 Deployment

The model is incorporated with a Flask web application. By entering a Sinhala input text on the web interface, preprocesses the input translation, and presents the input translation to the CNN model. The allowed sentiment is once shown to the user. As part of it, we introduced simple logging that will be used to analyze and enhance the system in the future by evaluating the input and output made by the system.

4 Results

4.1 Quantitative Results

On the test set the CNN model reached an overall accuracy of 91.22 % and a precision of 91.55% and recall 91.22 % As neutral sentiment is conceptually harder to classify. Reflecting on the confusion matrix (Table 1.), it is easy to note that the highest percentage of mistakes was associated between the neutral and positive classes some of the slightly positive comments were classified as neutral and some neutral ones as slightly positive. This indicates that in some cases the model can be problematic when it comes to dealing with the passing sentiments near the neutral threshold. Notably, the model achieves much better scores relative to a non-trivial baseline (word count based) classifier that we constructed to compare, achieving about 65–69% percent accuracy. The enhancement underscores the advantage of the CNN and supervised learning solution to the curated data.

Table 1 . Sentiment Classification Metrics

Class	Precision	Recall	F1- score
Negative	0.68	0.75	0.71
Neutral	0.85	0.87	0.86
Positive	0.97	0.95	0.96

The model achieved better performance than our standard of more than 65% performance of the lexicon approaches earlier done in Sinhala. An accuracy of 91.22% implies a strong sentiment prediction when the input is translated.

4.2 Analysis of Errors

We analyzed test cases in which the prediction of the model was different with that of human annotated sentiment. One pattern was that subtle sarcasm or culturally sophisticated statements were misclassified. This highlights a disadvantage: that the act of translation may lose valuable contextual information such as indicators of sarcasm. On another occasion, the mixed sentiment comment both positive and negative words had a confusing effect on the model, which were only able to give one label. Future improvements identified by these cases are to include sentiment scoring that would enable output with mixed sentiment or an ensemble of models with one of them trained on Sinhala text specifically.

4.3 Robustness to Translation Errors

A positive finding is that the model has shown sensible results even though the English translation was not impeccable. To take one example, not all Sinhala slang or informal words are translated correctly by Google nevertheless, this model frequently was able to make the right guess at the sentiment. This implies that even when the general sentiment conveyed words are translated even in rough form then the model can capture the signal. It was suited to learn the types of mistakes that the translator would commit, and that could be found in the training set. With that said, the sentences that are not mistranslated at all which are not numerous in our test sample yield a wrong sentiment output. It might also be possible to make reliability higher by reducing reliance on external translation possibly by training a bilingual word embedding or a translator optimized toward sentiment tasks.

4.4 User Experience

Though this is not the main scope of this paper, we would like to mention that the response time of the application was extremely rapid less than 2 seconds per input end to end taking translation and prediction into consideration. This is far below any reasonable limit of delay that is perceptible to customers so the tool can be used in real time.

4.5 Limitations

The major weakness of the given model is that it still relies on the quality of translation into English. When a critical sentiment signal such as a negation or an idiomatic expression is dropped or misreported by the translation, the model cannot reinstitute it directly, based on the Sinhala original. Also, the model is presently dealing with a sentence on its own, without taking into consideration the context or responses surrounding a comment that would influence sentiment perception. It might be useful to expand the model to take context into account thread level sentiment in the future versions.

5 Conclusion

We have built a sentiment analyzer model, which is efficient because it reads Sinhala language materials through the English translation and a classifier based on CNN. The work is unique in that it addresses a gap that can help in examining sentiments in Sinhala, which is not greatly covered by robust automated approaches. The methodology proves that translation coupled with machine learning may provide a good result in case of low resource languages. Trained on data translated to Sinhala the model learned to deal with certain translation flaws and could still detect sentiment with a test accuracy of 91.22%.

This study is relevant because it reaches into the usability of sentiment analysis to a Sinhala language without the requirement of an extensive native language corpus or a complicated language dependent processing. The model can be directly applicable to practitioners in Sri Lanka because it can be applied in the analysis of social media, customer reviews, or any text based data that is largely written in Sinhala. It also creates opportunities of incorporating Sinhala sentiment analysis in multilingual systems as well.

In regard to future work, The first direction we can consider is enhance the current model, it would be possible to apply Sinhala NLP tools, namely a Sinhala tokenizer and sentiment lexicon, to preprocess the features and enrich them with the help of Sinhala features. Third, mixed or context dependent sentiments: The processed sentiments should be able to improve how they handle mixed sentiment or a context dependent sentiment, maybe producing a sentiment score of a distribution as well as producing one label e.g. to denote that it is not well understood. Lastly, as a second direction, we also hope to study direct Sinhala texts sentiment models such as a multilingual BERT or a sequence to sequence model that translates and classifies sequentially in a single task) to eliminate the need to use third party translation services. Through such study areas we are optimistic that we will further enhance the precision and effectiveness of Sinhala sentiment analysis that will be added into the overall agenda of representative NLP technology across all languages.

REFERENCES

[1] X. T. D. Chau, "Simplifying Sentiment Analysis on Social Media: A Step by Step Approach," Griffith University, Australia, 2022.

[2] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent Trends in Deep Learning Based Natural Language Processing," IEEE Comput. Intell. Mag., vol. 13, no. 3, pp. 55–75, Aug. 2018.

[3] F. Neri, Sentiment Analysis on Social Media, IEEE, Istanbul, 2012.

[4] A. Bansal, A. Joshi, and P. Bhattacharyya, "A Literature Survey on Low-Resource NLP for South Asian Languages," ACM Trans. Asian Low-Resource Language Inf. Process., vol. 18, no. 3, 2020.

[5] K. Ravi and V. Ravi, "A Survey on Opinion Mining and Sentiment Analysis: Tasks, Approaches and Applications," Knowledge-Based Systems, vol. 89, pp. 14–46, 2016.

[6] Y. Zhang, J. Zheng, and X. Hu, "Cross-lingual Sentiment Analysis via Pretrained Language Models," in Proc. ACL, 2022.

[7] S. K. Behera and K. S. Babu, "Improved Feature Extraction and Classification for Sentiment Analysis," in Proc. Int. Conf. Computational Intelligence and Data Science, New Delhi, India, 2019.

[8] R. Mihindukulasooriya, H. Amarasinghe, and I. Herath, "Attention-Based Deep Learning Model for Sinhala Text Sentiment Detection," in Proc. Int. Conf. Artificial Intelligence and Data Engineering, 2022.

[9] B. Liu, Sentiment Analysis and Opinion Mining, Morgan & Claypool, 2017.

[10] M. Sandaru, D. Seneviratne, and A. Gamage, "Sinhala Sentiment Analysis using Deep Neural Networks," in Proc. Int. Conf. Advancements in Computing, Colombo, 2021.

[11] M. Dragoni, "Computational Advertising in Social Networks: An Opinion Mining-Based Approach," in Proc. 33rd ACM Symp. Applied Computing (SAC), Pau, France, 2018.

[12] M. X. Zhang, J. Ding et al., "Mining User Sentiments from Social Media for E-Commerce Recommendation," in Proc. IEEE Int. Conf. Data Mining Workshops, Singapore, 2018.

[13] S. K. Bharti, S. Varadhaganapathy, R. K. Gupta, P. K. Shukla, M. Bouye, S. K. Hingaa, and A. Mahmoud, "Text-Based Emotion Recognition Using Deep Learning Approach," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 2645381, 2022. [Online]. Available: https://doi.org/10.1155/2022/2645381.

[14] D. Li, S. Wang, L. Zhu et al., "Interest-Based Real-Time Content Recommendation in Online Social Communities," Knowl.-Based Syst., vol. 28, pp. 1–12, 2017.

[15] B. R. K. Nath and R. A. R. Baba, "K-Nearest Neighbor Classification and Regression in Recommender Systems," Foundations and Trends in Machine Learning, vol. 1, no. 3, pp. 233–334, 2008.

[16] S. Ma and M. Ester, "On the Design of LDA Models for Aspect-Based Opinion Mining," in Proc. ACM CIKM, Maui, HI, 2012.

[17] M. Dragoni, "Computational Advertising in Social Networks: An Opinion Mining-Based Approach," in Proc. 33rd ACM Symp. Applied Computing (SAC), Pau, France, 2018.

[18] F. Stasolla *et al.*, “Deep Learning and Reinforcement Learning for Assessing and Enhancing Academic Performance in University Students: A Scoping Review,” *AI*, vol. 6, no. 2, p. 40, Feb. 2025, doi: 10.3390/ai6020040.

[19] Y. Zhang, “Sentiment Analysis Method for Douban Movie Reviews Based on Prompt Learning,” *Applied and Computational Engineering*, vol. 151, pp. 183–191, 2025, doi: 10.54254/2755-2721/2025.23319.

[20] D. Perera, "A Multilingual Approach to Sentiment Analysis Using Cross-Lingual Embeddings," Univ. of Moratuwa, Sri Lanka, 2021.

[21] D. Seneviratne, M. Sandaru, and A. Gamage, "Sinhala Sentiment Analysis using Deep Neural Networks," in *Proc. Int. Conf. Advancements in Computing*, Colombo, 2021.

[22] A. Krouska, M. Virvou, and C. Troussas, “The effect of preprocessing techniques on Twitter sentiment analysis,” Jul. 2016. doi: 10.1109/iisa.2016.7785373.

[23] L. Xuan, Z. Mao, and L. Qing, "Low-rank optimization dictionary training for image classification," in *MATEC Web of Conferences*, vol. 173, 2018. [Online]. Available: https://doi.org/10.1051/matecconf/201817303034

[24] K. Pykes, “Cross-Entropy Loss Function in Machine Learning,” *DataCamp*, tutorial article, Aug. 2023. Available online. (Describes categorical cross-entropy for multi-class settings.