**An analysis of the viability of Decision Trees as an Interpretability model for sentiment prediction Neural Networks**

Student: Gerald Freislich  
School of Computer Science  
The University of Adelaide  
a1853059@adelaide.edu.au

Supervisor: Dr Bernard Evans  
School of Computer Science  
The University of Adelaide  
bernard.evans@adelaide.edu.au

**Abstract**

*Within machine learning, there exists an intrinsic obfuscation surrounding how Neural Networks determine their predictions [1]. To provide insight into how a model forms its decisions, Interpretability models have been generated, all suffering from a general lack of coherence and legibility. As Decision Trees are interpretable models, displaying clear decision paths, it was proposed that they could be used as an Interpretability model for Neural Networks. This was investigated with the use of a sentiment-predicting Neural Network, and whilst Decision Trees were determined to not be a viable Interpretability model, a tree-based decision visualisation tool was produced.*

* 1. **Introduction**

Recently, there has been a significant surge in the adoption of machine learning models, specifically, Neural Networks, due to their ability to perform increasingly complex tasks [2]. Problematically, Neural Networks are a black-box machine learning model, and hence, the decisions made by the Network are unclear [3]. Resultantly, significant apprehension exists regarding the applicability of Neural Networks, particularly in high-risk fields such as Medicine or Engineering.

Moreover, recent draft legislation by governing, and highly influential, bodies such as the EU, has emphasised the need for explainable AI, referencing ‘transparency [as] a core social or constitutional power’ [5]. This draft legislation imposes a right for individuals to access an explanation of a decision formed about them, by autonomous systems such as AI [6].

To satisfy this need for transparent machine learning models and explainable predictions, Interpretability models were developed. Interpretability is the ‘extraction of relevant knowledge from a machine-learning model concerning relationships…learned by the model’ [2]. As such, Interpretability models seek to determine the relationships behind the data passed into a model, and the output received, and hence deduce the decisions made by the model. However, as Interpretability is a recent field, its definition differs across papers.

Within this investigation, we seek to demonstrate the applicability of Decision Trees as an Interpretability method for sentiment-predicting Neural Networks. Through this investigation, potential decisions made by the Neural Network to classify sentiment would be illuminated.

* 1. **Motivation**

Decision Trees are a white-box machine learning model, with inherently interpretable characteristics [7]. Firstly, Decision Trees utilise clear decisions and decision paths to produce their conclusions. Moreover, Decision Tree modules have visualisation tools to increase their Interpretability, and hence maximising the use of these tools will increase model Interpretability.

Further, the field of Interpretability is a young field, and hence, there exists significant scope to apply new and old technologies to new problems. Within sentiment prediction, only feature attributive models such as LIME and SHAP have been employed. Hence, by applying Decision Trees to sentiment-predicting Neural Networks the scope for Decision Trees utilisation will increase.

1. **Literature Review**

Within Interpretability for Natural Language Processing (NLP), there are two types of Interpretability models: training and test-based models [8]. Common training-based Interpretability identifies high-influence instances in the training data, which guided the model’s prediction for a given example. Contrastingly, test-based interpretation explains the extent to which features in the test sample contributed to the model’s prediction of the instance. Here we review relevant Interpretability models for NLP’s, involving a discussion of Interpretability models that fall into both training and test-based Interpretability.

**2.1 Local Interpretable Model-agnostic Explanations (LIME)**

Local Interpretable Model Agnostic Explanations aim to evaluate the features influencing a black-box model’s predictions. A screenshot of a graph

Description automatically generated

**Figure 2.1, illustrating partial dependence plots generated with LIME [9].**

As demonstrated in *Figure 2.1*, LIME evaluates these features by influence and thus is limited by its requirement for inference [9]. LIME assesses explanations by minimising its loss function which is a mean squared error multiplied by a proximity measure between a perturbed example of a data entry () and the original data entry (), as displayed in *equations 1.1 and 1.2*.

[1.1]

[1.2]

**Equations 1.1 and 1.2, depicting LIME calculations used to produce the explanations [9].**

In a comparative study by Kulshrestha et al., LIME was tested in a natural language processing healthcare scenario against Logistic Regression, Extreme Gradient Boosted Machines and Convolutional Neural Networks [10]. This foundational study justified the validity of LIME for NLP-based contexts, also, however, highlighting that an increasingly complex black-box model, one with many features, produces proportionally complex LIME influence generations which can be difficult to understand.

**2.2 Shapley Additive Explanations**

Stemming from game theory, Shapley values were introduced in 1953 to find likely possible outcomes for a player in an abstract game [11]. In 2017, Shapley values were integrated into a framework for interpreting model predictions, where each feature in the training dataset is assigned an importance value to a particular prediction [12] similar to LIME. The difference between SHAP and LIME is the importance value functions as displayed in equation 2.1:

[2.1]

This value is calculated as the weighted average of all possible differences, where F is the set of all features, and S is the set of all feature subsets. Hence, a perturbation of the input data is required to form the weighted average of all possible differences.

Despite a difference in how the feature importance is calculated, SHAP also provides a framework around the application of its values, a shortcoming of its predecessor LIME. Local Accuracy, Missingness, and Consistency are axioms of the SHAP framework for Interpretability. These axioms give credence to the framework by ensuring that the explanations provided are reliable, consistent, and dynamic. The dynamicity of the explanations stems from missingness, wherein a feature is constrained and tested to see if its absence or perturbation proportionally influences the explanation [12].

Problematically, SHAP does not provide initially comprehensible explanations or truly explanations at all. Instead, both SHAP and LIME evaluate feature importance for a specific input. From such, the observer must infer what decisions the black-box model is making. Yet, SHAP values are useful in corroboration with other visualisation methods. For instance, Hierarchical Explanation via Divisive Generation (HEDGE) can be applied to a Shapley Additive Explanation to provide a tree-shaped explanation for sentiment analysis [13], as displayed beneath.

A diagram of words

Description automatically generated

**Figure 2.3, highlighting the use of HEDGE technology against a SHAP Interpretability model applied to sentiment analysis [13].**

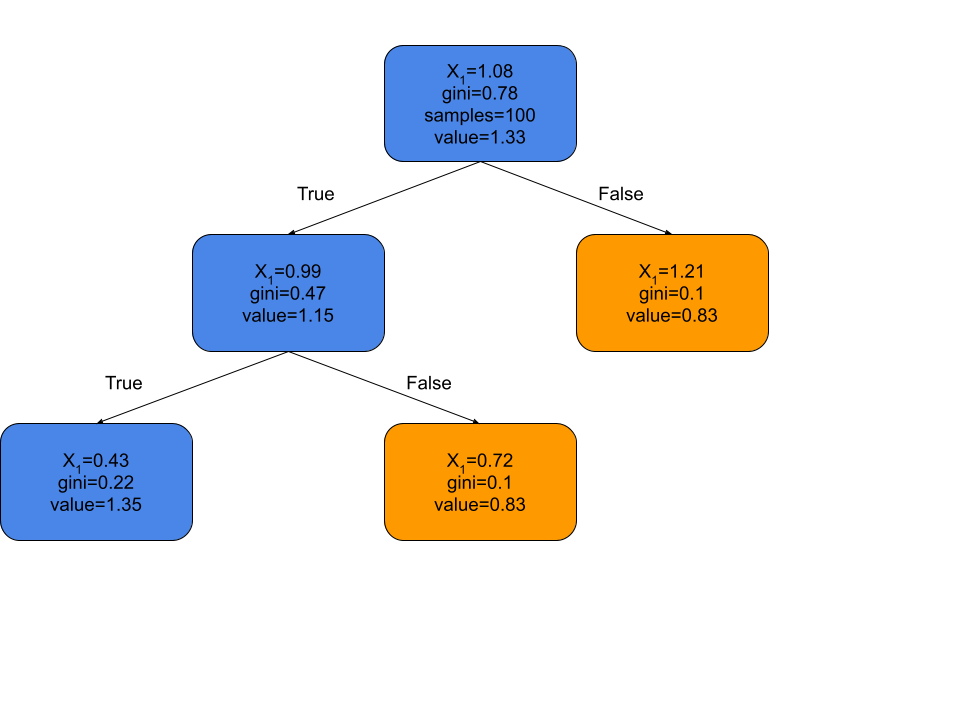
**2.3 Decision Trees**

A map of a person with different colored squares

Description automatically generatedAmong the most comprehensible Interpretability models are decision trees [14]. In early 2023, the viability of decision trees via a surrogate model for Interpretability was framed by Blanco-Justicia A. and Domingo-Ferrer J. [15]. The paper utilised a decision tree trained on input data and the output of a black-box model and highlighted risks associated with large-scale decision tree Interpretability models, namely the extensiveness of their branches if ‘pruning’ is not applied. An example of this complexity is exemplified in *Figure 2.4.* Hence, a Decision Tree model must maintain a fixed size, such that it remains comprehensible to individuals.

**Figure 2.4, clarifying the scale to which a complex decision tree Interpretability model can become [15].**

The same paper revealed the usefulness of a Decision Tree for smaller tasks, as depicted in *Figure 2.5* through which the explanations made by a black box model are increasingly comprehensible. Yet, the explanations provided by the Decision Tree in 2.5 is still largely unintelligible as it is composed mostly of equalities.

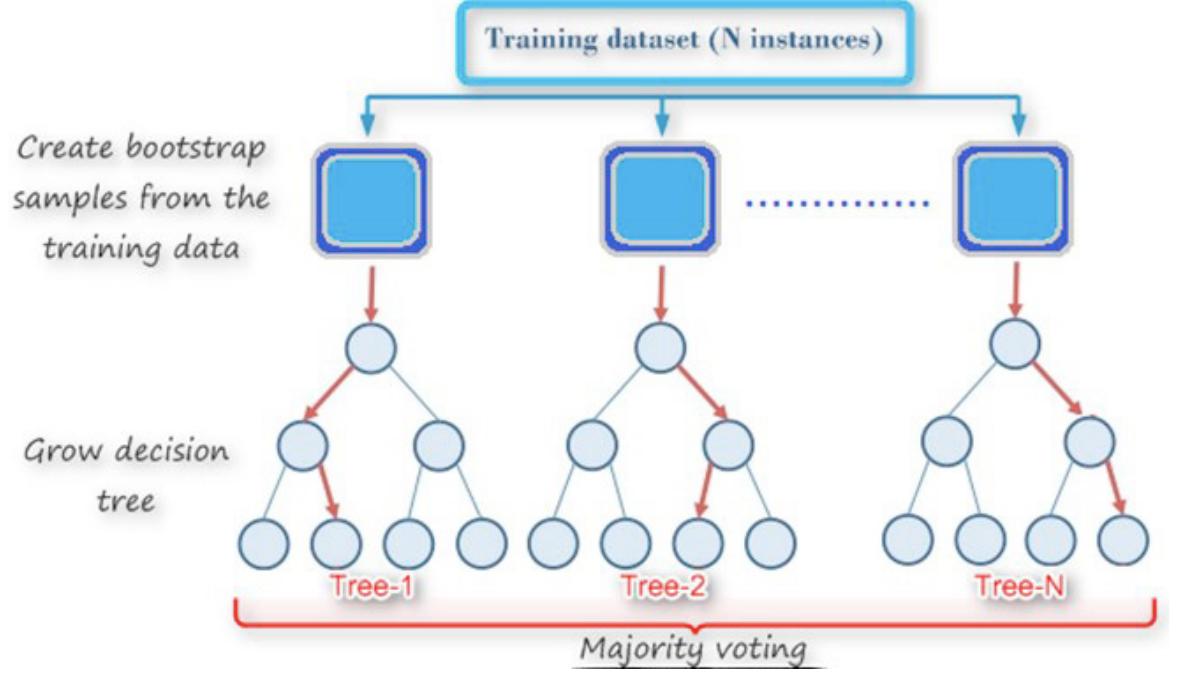


**Figure 2.5, a decision tree explanation formed from a cluster of predictions.**

As introduced in 2022, Mixture of Decision Trees (MoDT), is an approach that minimises the potential sizes of Interpretability trees for generating model explanations [16]. This approach utilises a mixture of Decision Trees trained against inputs, such that a new Decision Tree is generated for each prediction. Each Decision Tree represents a distinct region of the input space and thus contains ‘distinct patterns’ that represent that region. This facilitates a greater accuracy for the model than techniques like Random Forests. Additionally, the approach set a maximum depth to each Decision Tree, which encouraged the forming of more legible DTs.

**2.4 Random Forests**

Random Forests are an ensemble learning method, a collection of classification machine learning models such as Decision Trees [16]. To generate a Random Forest, a Decision Tree is formed from each data instance in the training set, and these trees cast a majority vote to yield the predicted class. This is highlighted in the figure beneath.



**Figure 2.6, depicting the training method used to form random forests [18].**

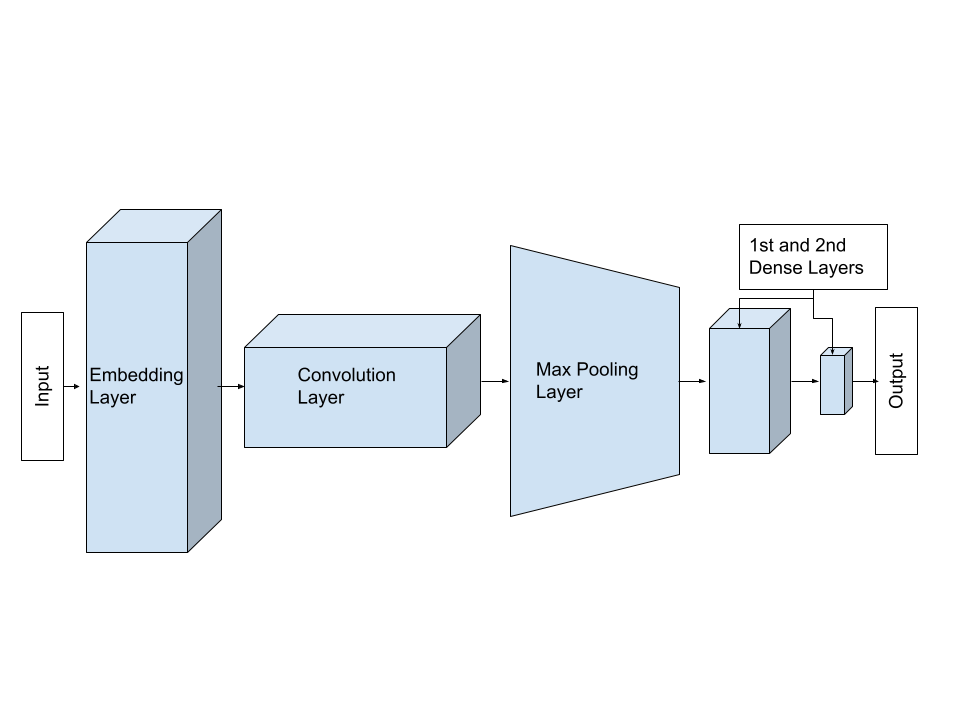
In 2018, Random Forests were used to perform sentiment analysis on Twitter data by Ali Fauzi M. with relative accuracy (82%) with a limited sample size of 386 reviews [19]. Yet, Random Forests have not been used as an Interpretability method for sentiment-predicting neural networks. This is in part due to the increased complexity of Random Forests, as a compromise for increased accuracy when compared to Decision Trees [20]. This increased complexity implies that a fixed Decision Tree cannot explain all decisions made by the model, and thus an increasing amount of inference must be used to make sense of the explanations.

1. **Methodology**

Within this review of the viability of Decision Trees for sentiment-predicting Neural Networks, there were three main stages. First was the analysis of Decision Trees as an Interpretability method for the Neural Network. Next was the analysis of Random Forests as an Interpretability method for Neural Networks. Finally came the production of tree-based visualisation tools for Interpretability. The analysis of Decision Trees and Random Forests was split into two stages, respectively, the first being their use in predicting and explaining the final output of the Neural Network, and the next being their use in predicting and explaining the output of each layer in the Neural Network.

**3.1 Model Used**

A Convolutional Neural Network was used as the Natural Language Processing sentiment classifier, due to historical use in the field. To limit complexity, whilst maintaining accuracy, a 5-layer Convolutional Neural Network was used, as depicted beneath.



**Figure 3.1, displaying the chosen Convolutional Neural Network set up for this investigation.**

This model bore sufficient complexity to accurately predict the sentiment of a given review, whilst not substantially increasing training or testing time.

**3.2 Data Preparation**

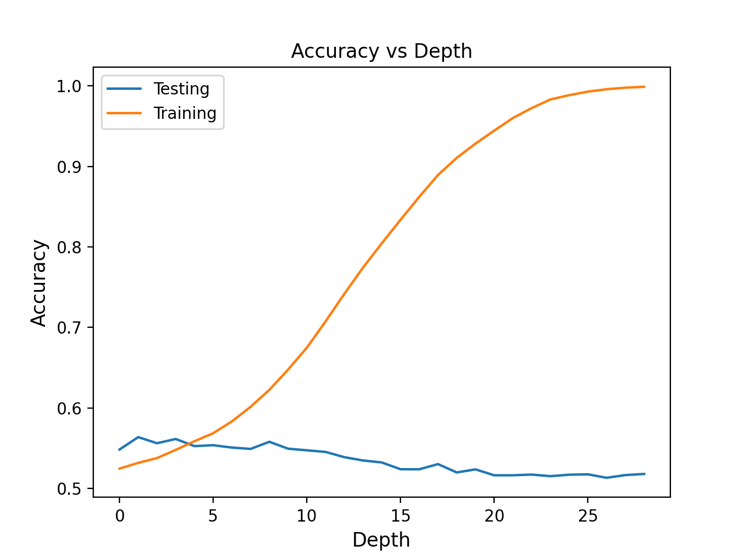
The data used in this investigation was sourced from a Kaggle scrape of 50,000 IMDB reviews and their corresponding sentiments. The sentiments were initially ‘positive’ or ‘negative’, but this was converted to binary classification. Likewise, data was removed randomly to ensure balanced classes in the dataset. When training, testing and validation splits were formed using Scikit-learn’s train\_test\_split stratification around the sentiments occurred to ensure equivalence.

Within each experiment the data was tokenized, using Keras’ Tokenizer class. This provided a sequence of separated numerical values for each review. Then, Keras’ pad\_sequences was used to ensure that equal lengths were given to all sequences, as the CNN requires an equal length of all vectors.

Following the preparation of the data, the input became a vector of numerical values aligned to the initial reviews by index, hence, maintaining the order and complexity of the input.

**3.3.1 Decision Trees and Random Forests for predicting and explaining Neural Network output.**

Decision Trees were produced using Scikit-learn’s Decision Tree classifier with logarithmic entropy as the splitting mechanism. This was chosen due to the reduction in training complexity whilst also maintaining high accuracy in Decision Tree specific tasks. The accuracy of trees vs depth was plotted, as visible in *Figure 3.2*.



**Figure 3.2, depicting the accuracy of Decision Trees against depth**.

This figure highlighted that optimal accuracy in testing occurred around a maximum depth of 5, whilst accuracy tended to overfit with greater depths. Resultantly, a maximum depth of 5 was chosen for the Decision Trees in this investigation.

Comparatively, Random Forests were produced using Scikit-learns RandomForestClassifier with logarithmic entropy as the splitting mechanism. Further, the maximum depth for each Random Forest was left as the maximum number of nodes, enabling each Decision Tree split to optimally represent the input. Finally, the number of estimator trees in the Random Forest was set to 100.

**3.3.2 Decision Trees and Random Forests for predicting and explaining Neural Network layer-wise output.**

To use Decision Trees and Random Forests in predicting the Network’s final output from its layer’s outputs, sub-models were created. These models included trained layers from the CNN. For instance, when using Decision Trees and Random Forests to predict the Network output from the pooling layer’s output a new model was composed of the Embedding, Convolutional and Pooling layers, from the trained CNN. This ensured that all layer-wise outputs were the real outputs from the CNN’s layers.

After these models were made, the data was prepared according to section 3.2, and Keras’ flattening tool was used to reduce the dimensions of the layer’s output. This was required as the layer output was, for most layers, multi-dimensional.

The now flattened layer output was then passed as input into either the Decision Tree or Random Forest and the models were trained on 37500 data instances for each ‘layer’.

**3.4 Deconstructive Neural Network Decision Visualiser**

The original texts were split into halves recurrently and stored in a tree data structure, to be able to effectively visualise the relationship between sections of the input and the CNN’s predictions. These subsections were then fed into the trained Convolutional Neural Network, and predictions were saved correspondingly into the tree data structure.

Tkinter, a Python graphic library, was then used to visualise the data structure, with child nodes being placed on the screen at positions:

Where d is the maximum depth of the subsections, and l is the current layer in the tree-like data structure.

This formula was used, as it enabled a simple generation of position on screen, following the required tree structure. These positions highlighted the relationship between parent subsections and their children’s predictions. To further illustrate these relationships, colour was added, with subsections of a prediction less than 0.33 (negative sentiment) being in red, predictions between 0.34 and 0.66 being in blue, and predictions above 0.66 being in green (positive sentiment).

This deconstructive Neural Network visualisation tool can be used on sentences, phrases or paragraphs, and a paragraph tool and sentence deconstructive visualisation tool were both created.

1. **Experimental Design and Results**

When investigating Decision Trees as an Interpretability model for NLP Neural Networks we used two stages: predicting and explaining the output of a Neural Network with Decision Trees, and then Random Forests.

* 1. **Decision Tree Accuracy**

First, a Decision Tree was used to classify raw IMDB reviews on their sentiment, trained on 37500 samples, returning a testing accuracy of 51%. This indicated a poor understanding of raw relationships between text and sentiment, with the Decision Tree virtually using random chance to inform its prediction. After this poor result, the Decision Tree was used to predict the output of the CNN on the same IMDB reviews, producing a minimally increased accuracy of 53%. This alludes to a potential lack of complexity of a Decision Tree to understand the necessary relationships between text and sentiment, whilst also demonstrating an increased understanding of a CNN over the raw text.

To further investigate the Interpretability of a CNN with a Decision Tree, it was determined that the output of each layer would be taken as input, and then a Decision Tree would be used to predict, from the output of each layer, the final output of the CNN. In doing so, it was theorised that the Decision Tree accuracy would substantially increase through each layer, as the input size would decrease, and the correlation between the current layer’s output and the CNN’s final prediction would increase. This experiment produced Table 1.0.

|  |  |  |
| --- | --- | --- |
| CNN Layer Output | Training Accuracy (%) | Testing Accuracy (%) |
| Embedding | 55 | 54 |
| Convolution | 56 | 55 |
| Pooling | 62 | 57 |
| First Dense Layer | 58 | 57 |
| Total | 91 | 53 |

**Table 1.0 displaying the accuracy of a Decision Tree when predicting the output of a CNN from the output of each layer of the CNN.**

As is visible in Table 1.0, the Decision Trees created to predict the output of the CNN from layer outputs produced a marginally better testing accuracy than the Decision Trees used to predict the output of the CNN from the raw IMDB data. Likewise, the highest accuracy being 57%, achieved from predicting the output of the CNN from the outputs of the Pooling and First Dense layer, was a 4% increase, highlighting some potential intrinsic flaws in the use of Decision Trees. This was unexpected as at the level of the first dense layer, it was presumed that the Convolutional Neural Network had effectively formed its prediction.

* 1. **Random Forests Accuracy**

As with Decision Trees, a Random Forest was used to classify raw IMDB reviews based on their sentiment, returning a testing accuracy of 54%, minimally better than the Decision Tree accuracy of 51%. Similarly, Random Forests provided a 57% accuracy when predicting the output of the CNN. This was still low, and the expanded input size was believed to be a potential issue. As such, Random Forests were used to predict the output of the Convolutional Neural Network from the output of each layer in the CNN, yielding the following table. Again, it was proposed that as the output size of the layers decreased the accuracy of the Random Forests would increase.

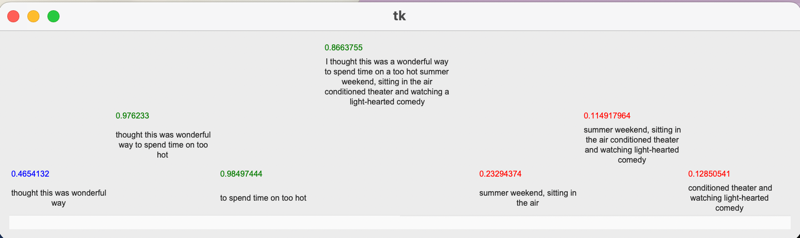
|  |  |  |
| --- | --- | --- |
| CNN Layer Output | Training Accuracy (%) | Testing Accuracy (%) |
| Embedding | 91 | 55 |
| Convolution | 96 | 54 |
| Pooling | 100 | 61 |
| First Dense Layer | 100 | 58 |
| Total | 93 | 57 |

**Table 2.0 displaying the accuracy of Random Forests when predicting the output of a CNN from the output of each layer of the CNN**

Akin to Decision Trees, the accuracy of a Random Forest using the output of the Convolution and Pooling layers to predict the output of the Network exceeded other Random Forests. Yet, Random Forests produced higher test accuracy indicating more overfitting, whereas this was not the case for Decision Trees. However, Random Forests still proved to be flawed through this test, and thus the exchange between Interpretability and complexity led to limited benefits.

* 1. **Deconstructive Neural Network Decision Visualiser**

A significant facet of this investigation was the utilisation of a tree format to highlight decisions made by the Neural Network. This was achieved through an inference-based visualisation tool, wherein the original IMDB reviews were divided into halves and the Neural Network’s predictions for each subset were visualised.



*Figure 5.1 depicting the Decision Visualiser Tool produced, visualising a subsection of a review.*

As illustrated in *Figure 5.1*, the Decision Visualisation tool facilitated a greater understanding of the potential decisions made by the Neural Network. For instance, within the example, the increased influence of the right child node “conditioned theatre and watching light-hearted comedy” when compared to the left child node, in the parent’s evaluation is clear.

* 1. **SHAP Visualisation Tool**

A screen shot of a television show

Description automatically generated

**Figure 5.2, demonstrating the hierarchical visualisation tool produced.**

The SHAP visualisation tool highlights words with their accompanying SHAP influence factor value, exemplifying the main words that influenced the model’s prediction for a given IMDB review. It is observable, in this instance, that ‘episode’, ‘mentioned’ and ‘reviewers’ had the greatest influence on the Neural Network’s output.

1. **Discussion**

The results of this investigation into the feasibility of Decision Trees as an Interpretability method for sentiment-predicting Neural Networks demonstrate the limitations of Decision Trees and Random Forests. Despite this, the Deconstructive Neural Network Visualiser clarifies potential decisions made by the model.

* 1. **Decision Trees**

As highlighted within the results, Decision Trees had similar accuracy to random chance when tasked with predicting the true sentiment of a IMDB movie review, and marginally better than random chance when tasked with predicting the CNN’s prediction of an IMDB movie review. This alluded to a slightly better understanding of Neural Networks than of a true sentiment classification task.

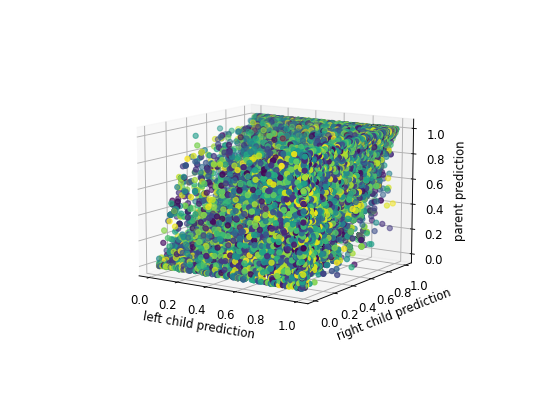


Figure 6.1, depicting the relationship between children and parent subsection predictions of the IMDB reviews.

Furthermore, with sentiment prediction tasks the words tend to become the features that are split by the Decision Tree, and thus the Decision Tree attempts to use individual words to determine the sentiment of the text. Problematically, sentiment is more nuanced than simply the words in a piece of text, being more influenced by the combination of words than the words themselves.

By using Decision Trees to predict the output of a Neural Network, potential decisions made by the Neural Network to reach its output would be visible. However, the lacking accuracy of Decision Trees indicates that only half of these explanations would be utilisable.

* 1. **Random Forests**

Akin to Decision Trees, Random Forests displayed poor accuracy when tasked with both predicting the true sentiment of an IMDB review and the sentiment predicted by a Neural Network. Whilst the accuracy of Random Forests was marginally greater than Decision Trees, the fundamentally low accuracy limits the applicability of Random Forests as an Interpretability method for sentiment-predicting Neural Networks.

Conceptually, Random Forests have greater accuracy than Decision Trees due to their accumulation of Trees to inform their output and consequently have limited Interpretability. Yet, the 5% increase in accuracy between Random Forests and Decision Trees is negated by the increased complexity and hence reduced Interpretability.

* 1. **Deconstructive Neural Network Visualiser**

With the produced Deconstructive Neural Network Visualiser, an inference-based approach is taken wherein the user must apply inference to determine relationships between parents and the children subsections of the text. When expanded to a word level, increasing the depth, potential decisions such as biases against words are clear. However, there is no clear linear relationship between the children subsections and the parent subsections, as displayed beneath.

Figure 6.1 demonstrates a low correspondence between the prediction of children subsections and parent subsections, which highlights the high variance of language. For example, “the sky” which may have a positive sentiment, according to the Neural Network, may be combined with “was black”, which may have a negative sentiment. Yet, instead of producing a neutral sentiment prediction, the overall sentiment of the sentence is negative.

This reaffirms the limitations concerning the use of Decision Trees as an Interpretability method for sentiment-predicting neural networks, as language, and more specifically words, cannot be treated as isolated entities.

1. **Conclusion**

Throughout this investigation, Decision Trees were determined to be an infeasible Interpretability model for sentiment-predicting Neural Networks. With an accuracy of 52%, any potential decisions highlighted by the Decision Tree would be marginally better than random chance, and hence cannot be trusted. Moreover, the 57% accuracy of Random Forests exemplified the limitations of Decision Tree-like structures. Yet, the Deconstructive Neural Network Visualiser, while relying on inference by the observer, highlights influential sections of the input, alluding to potential decisions made by the Network. Hence, Decision Trees lack the required complexity to be an Interpretability model for sentiment-predicting Neural Networks, however, Tree-like visualisation tools may be useful in finding influential sections of the input that inform the Neural Network’s predictions.

**References:**

### [1] Carter, A., Imtiaz, S., & Naterer, G. (2023). Review of interpretable machine learning for process industries. *Process Safety And Environmental Protection*, *170*, 647-659. doi: 10.1016/j.psep.2022.12.018

# [2] Murdoch J. W. Singh C., Kumbier K., Abbasi-Asl R., Yu B., (2019) Definitions, methods, and applications in interpretable machine learning <https://doi.org/10.1073/pnas.1900654116>

[3] Ribeiro M., Singh S., Guestrin C. (2016) Model Agnostic Interpretability of Machine Learning arXiv:1606.05386

### [4] Bohr, A., & Memarzadeh, K. (2020). The rise of artificial intelligence in healthcare applications. *Artificial Intelligence In Healthcare*, 25-60. doi: 10.1016/b978-0-12-818438-7.00002-2

[5] European Union, 2016. General Data Protection Regulation. Regulation (EU) 2016/679. https://gdpr-info.eu 7.

[6] The Royal Society (2019), Explainable AI: the basics Policy Briefing, ISBN: 978-1-78252-433-5 <https://ec.europa.eu/futurium/en/system/files/ged/ai-and-interpretability-policy-briefing_creative_commons.pdf>

#### [7] *Blanco-Justicia A., &* Domingo-Ferrer J., 2023, Machine Learning Explainability Through Comprehensible Decision Trees, https://inria.hal.science/hal-02520062/file/485369\_1\_En\_2\_Chapter.pdf.

### [8] Sun, X., Yang, D., Li, X., Zhang, T., Meng, Y., Qiu, H., Wang, G., Hovy, E. and Li, J. (2021) *Interpreting Deep Learning Models in Natural Language Processing: A Review*, *arXiv.org*. Available at: https://arxiv.org/abs/2110.10470

[9] Ribeiro M.T., Singh S., Guestrin C. Anchors: High-precision model-agnostic explanations; Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence; New Orleans, LA, USA. 2–7 February 2018.

### [10] Kulshrestha, S., Dligach, D., Joyce, C., Gonzalez, R., O’Rourke, A. P., Glazer, J. M., Stey, A., Kruser, J. M., Churpek, M. M. and Afshar, M. (2021) "Comparison and interpretability of machine learning models to predict severity of chest injury", *JAMIA Open*, 4(1). doi: 10.1093/jamiaopen/ooab015.

### [11] Ribeiro, M. T., Singh, S. and Guestrin, C. (2016) *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*, *arXiv.org*. Available at: https://arxiv.org/abs/1602.04938

#### [12] Lloyd S Shapley. 1953. A value for n-person games. Contributions to the Theory of Games 2.28, page 307–317.

#### [13] Lundberg, S. and Lee, S. (2017) *A Unified Approach to Interpreting Model Predictions*, *arXiv.org*. Available at: https://arxiv.org/abs/1705.07874

#### [14] Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. 2020. Generating hierarchical explanations on text classification via feature interaction detection. arXiv preprint arXiv:2004.02015.

#### [15] Chen, H., Zhang, H., Boning, D., and Hsieh, C.-J. 2019. Robust decision trees against adversarial examples. arXiv preprint arXiv:1902.10660.

#### [16] Brüggenjürgen, S. et al. (2022) *Mixture of Decision Trees for Interpretable Machine Learning*, *arXiv.org*. Available at: https://arxiv.org/abs/2211.14617

#### [17] What is Random Forest? IBM (2018) Available at: https://www.ibm.com/topics/random-forest

#### [18] Al Amrani, Yassine, Mohamed Lazaar, and Kamal Eddine El Kadiri. 2018. *Procedia Computer Science* 127: 511–20. doi:10.1016/j.procs.2018.01.150.

#### [19] Fauzi, M. A. (2018) *Random Forest Approach for Sentiment Analysis in Indonesian Language*, *Indonesian Journal of Electrical Engineering and Computer Science*.

#### [20] Grigg, T. (2019) *Interpretability and Random Forests*, *Medium*. Towards Data Science.