VYTAUTAS MAGNUS UNIVERSITY
FACULTY OF INFORMATICS

Janssens Guillaume Serge C.

INDIVIDUAL ASSIGNEMENT
NEURAL NETWORKS
AUTAUMN 2023

2023

Neural Network-Based Research Paper Classification

Dataset Description

Introduction

- Source and Nature of Dataset: The dataset is sourced from Kaggle, and created by Analytics Vadhya

Basic Statistics

- Data Overview: The dataset is composed of 20972 training and 8989 test research articles.
- Category Distribution: The dataset is built around 6 categories: Computer Science, Physics, Mathematics, Statistics, Quantitative Biology and Quantitative Finance. The testing dataset has the categories of each article attached to them, and their ID, Title and Abstract. The test dataset is composed only of the IDs, Titles, and Abstracts.
- Missing Values Analysis: There are no missing values in the dataset.

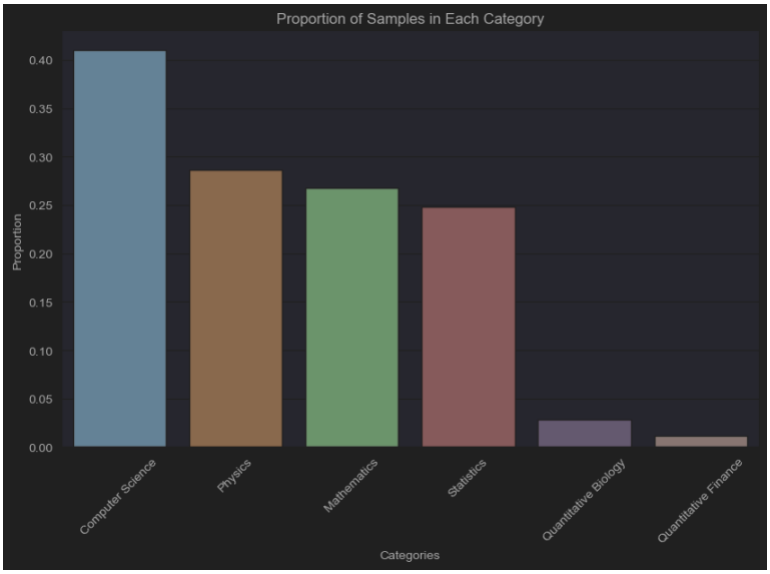

Figure 1 Distribution of the different categories of research articles contained in the dataset.

Goals and Objectives

Purpose

- This model was built and trained to achieve a sorting of research articles only using their title and abstract, which in term could help sorting, and refining research article sourcing.

Specific Goals

The specific objective was to improve classification accuracy and exploring the effectiveness of different neural network architectures.

Data Preparation and Exploratory Data Analysis

Data Cleaning:

In the initial phase of data preparation, our primary focus was on cleaning and structuring the dataset, which consisted of academic paper titles and abstracts. This process involved removing

special characters and HTML tags to ensure the textual data was in a clean, consistent format suitable for analysis. Following the cleaning, I tokenised the text, converting it into a sequence of words, and then applied padding to ensure uniformity in the sequence lengths, which is crucial for neural network processing.

Exploratory Data Analysis:
For the exploratory data analysis, I conducted a thorough examination of the dataset. This included analysing the distribution of papers across various academic categories, such as Computer Science, Physics, Mathematics, and others. I visualised this distribution through bar plots (Figure 1), offering a clear picture of the dataset's composition. Additionally, I examined the length distribution of titles and abstracts using histograms, which provided insights into the textual data's structure. This exploratory phase was vital for understanding the dataset's characteristics and guided our subsequent modelling choices.

Machine Learning Algorithm

Before delving into neural network models, I investigated traditional machine learning algorithms. This is evident from the different approaches taken by my peers on Kaggle, who, like me, applied algorithms such as decision trees and k-NN to classify academic papers into their respective categories. While these methods are foundational in machine learning, they yielded only moderate success with accuracy rates approximately ranging between 60-69%. This achieved a foundational level for our project and demonstrated the possibility for more advanced techniques such as neural networks to improve classification accuracy. The knowledge obtained from this investigation directed our subsequent selection of neural network models, with the aim to exceed the threshold established by these conventional methods.

Neural Networks for Data Analysis

a. Multilayer Perceptron (MLP)
In our exploration of neural network-based solutions, we first implemented a Multilayer Perceptron (MLP), a foundational architecture in the field of deep learning. The MLP was structured to adeptly handle the textual nuances inherent in academic paper titles and abstracts. At its core, the MLP featured an embedding layer tailored to process the textual data, followed by dense layers with ReLU activation functions to capture non-linear relationships within the data. (Figure 2)

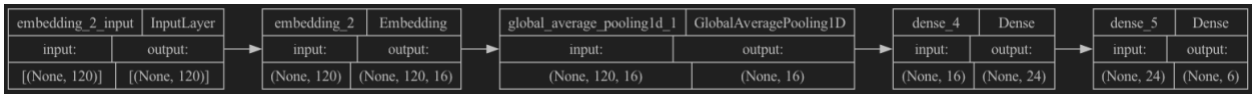

Figure 2 Schematic Plot of the MLP Model for Research Articles Classification

To optimize the learning process and performance, we carefully calibrated the training parameters, setting the batch size to 64 to balance the computational efficiency and model accuracy. The model was trained over 20 epochs, a decision influenced by the need to sufficiently expose the network to the data while avoiding overfitting. Those parameters were found by analysing the plots (Figure 1), which when using a higher batch size, the loss was not dropping enough, and that the number of epochs could be pushed a bit further than 10. But introducing a higher value for epochs was leading to overfitting, which could be seen by a loss curve going up.

A key to the success of this MLP model was the meticulous tuning of its architecture and parameters. This careful configuration was pivotal in enhancing the model's ability to generalize from the training data to unseen data, a crucial aspect of effective machine learning models.

The MLP model's performance was a testament to its robust architecture and fine-tuning, as it achieved a noteworthy validation accuracy of 77.33%. This high level of accuracy not only highlighted the MLP's capabilities in text classification tasks but also set a substantial benchmark for the effectiveness of neural networks in handling complex classification problems, especially in contrast to traditional machine learning methods.

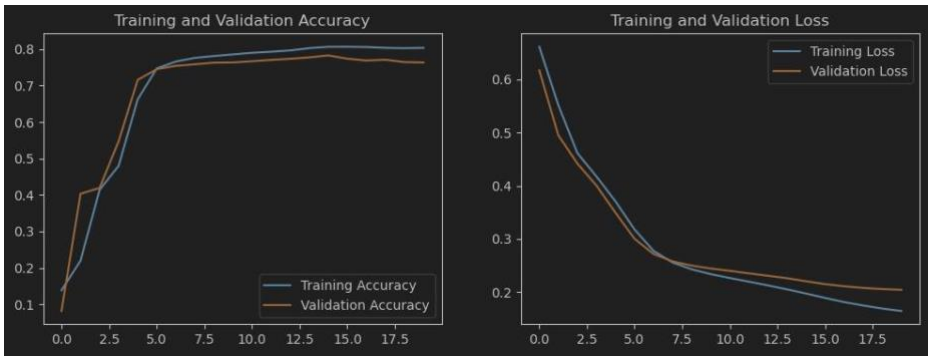

Figure 3 Evolution of training and validation Accuracy and Loss for the MLP Model

b. Convolutional Neural Network (CNN)

In our quest to further refine our text classification capabilities, we advanced our approach by implementing an enhanced Convolutional Neural Network (CNN). This CNN was intricately designed with multiple convolutional layers featuring varying filter sizes to capture diverse textual patterns effectively. Each convolutional layer was followed by batch normalization for stabilization and max pooling for dimensionality reduction, efficiently extracting pivotal features from the academic text.

Recognizing the potential for overfitting in such a complex model, we integrated dropout layers strategically, ensuring a robust and generalizable model. The dropout technique was crucial in preventing the model from learning spurious patterns, thereby enhancing its predictive power on unseen data.

Figure 4 Schematic Plot of the CNN Model for Research Articles Classification

The model's dense layers were fortified with L2 regularization, further aiding in combating overfitting by penalizing large weights, and a final dropout layer was employed before the output layer. This output layer maintained a sigmoid activation function, aligning with the multi-label nature of our classification task.

The model was compiled with the Adam optimizer (legacy), renowned for its effectiveness in handling sparse gradients and adaptive learning rates. We trained the model for 10 epochs with a batch size of 128, striking a balance between computational efficiency and the ability to generalize. This training strategy was complemented with early stopping and learning rate reduction techniques to prevent overtraining and to fine-tune the learning process.

But before finding the perfect parameters for our model, experiments with a simpler CNN architecture were led and poor results proved that the architecture was not convenient for the task. Thus, a new architecture was built (Figure 4) and trained.

This enhanced CNN architecture proved its efficacy by achieving a significant milestone in prediction accuracy, reaching around 71% on the validation set. This marked improvement from traditional machine learning methods and our initial neural network attempts underscored the potency of well-architected CNNs in handling complex text classification tasks.

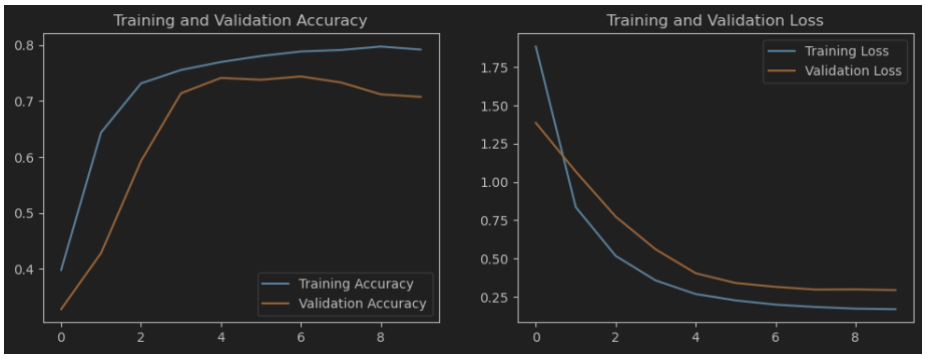

Figure 5 Evolution of training and validation Accuracy and Loss for the CNN Model V2

Accuracy Estimation

In this critical phase of our project, we meticulously evaluated the accuracy of our developed models to ascertain their effectiveness in classifying academic paper titles and abstracts into their respective categories. Accuracy estimation is pivotal in understanding the models' performance and their practical applicability.

Multilayer Perceptron (MLP)
The MLP model underwent rigorous evaluation, and the results were promising. After training the model for 20 epochs with a batch size of 64, the model achieved a remarkable validation accuracy of 77.33%. This high level of accuracy demonstrates the model's capability to understand and categorize complex textual data effectively. The success of the MLP model in this context is indicative of its well-tuned architecture and its suitability for text-based classification tasks.

Convolutional Neural Network (CNN)
Similarly, the enhanced CNN model was evaluated for its performance. The CNN, with its sophisticated architecture comprising multiple convolutional and dropout layers, was trained for 10 epochs, also with a batch size of 64. This model reached a validation accuracy of approximately 71%, a significant improvement over traditional machine learning techniques and indicative of the effectiveness of convolutional layers in capturing essential textual features.

Comparative Insight
These accuracy figures provide a clear insight into the comparative performance of the two models. While both models significantly outperformed traditional machine learning algorithms, the MLP showed a slightly higher accuracy in our specific task. This suggests that while CNNs are exceptionally adept at capturing spatial and local patterns within data, the MLP's architecture might be more suited for this kind of textual classification task. However, the choice between these models can vary based on specific dataset characteristics and requirements.

Overall Evaluation
Overall, the accuracy estimation phase underscored the viability of using advanced neural network models for text classification. The substantial accuracies achieved by both models highlight their potential in automating and enhancing the categorization process in academic and research domains. Furthermore, these results lay a foundation for future explorations into more

complex models and architectures, potentially integrating the strengths of both MLP and CNN models.

Analysis of Parameters

The effectiveness of machine learning models, particularly neural networks, is deeply influenced by the choice and tuning of their parameters. In our project, we conducted a thorough analysis of various parameters to optimize our models for the classification task.

Multilayer Perceptron (MLP)

For the MLP, key parameters such as the number of hidden layers, the number of neurons in each layer, and the learning rate were meticulously adjusted. The model's optimal performance at a batch size of 64 over 20 epochs was a result of experimenting with these parameters. The balance achieved in the MLP's architecture highlighted the importance of tuning not only for performance but also for preventing overfitting, ensuring the model's generalizability to new, unseen data.

Convolutional Neural Network (CNN)

In the case of the CNN, parameters like the number and size of filters, kernel size, and the structure of convolutional layers were critical. The interplay between convolutional layers and dropout layers was particularly crucial in achieving an effective model. The chosen configuration, which led to a validation accuracy of about 71%, underscored the significance of parameter tuning in harnessing the power of CNNs for text data.

Impact on Classification Accuracy

The analysis of these parameters provided valuable insights into how different settings affect classification accuracy. It became evident that while there is no one-size-fits-all parameter set, careful tuning tailored to the specific characteristics of the dataset can lead to substantial improvements in model performance.

Comparison of ML Methods

Our project involved a comparative analysis of different machine learning methodologies, spanning from traditional algorithms to advanced neural networks.

Traditional vs. Neural Network Models

The initial benchmarks set by traditional machine learning models, which achieved accuracies in the range of 60-69%, were significantly surpassed by our neural network models. This comparison not only highlighted the advanced capabilities of neural networks in handling complex patterns within text data but also underscored the limitations of traditional algorithms in dealing with high-dimensional and nuanced data.

MLP vs. CNN

Among the neural network architectures, the MLP achieved a higher accuracy (77.33%) compared to the CNN (70.75%). This outcome suggests that for the specific nuances of our textual data, the MLP's structure was more effective, possibly due to its ability to model complex relationships without the spatial constraints inherent in CNNs. However, the CNN's performance indicated its strength in extracting local and positionally invariant features, which could be advantageous in different or more complex textual datasets.

Conclusion from Comparison
The comparison of these methodologies reinforced the notion that the choice of model architecture must be aligned with the specific characteristics and requirements of the dataset. It also opened avenues for exploring hybrid models or ensemble methods that could combine the strengths of different architectures.

Conclusions

In conclusion, our exploration into the realm of machine learning for text classification yielded several key insights and findings.

Model Efficacy
Our project demonstrated the efficacy of neural network models, particularly MLP and CNN, in accurately classifying academic texts. The MLP, with its high accuracy of 77.33%, emerged as a particularly effective model for this task, while the CNN also showed notable strengths in feature extraction.

Parameter Significance
The significant role played by parameter tuning and model architecture in achieving high accuracy was a crucial learning from this project. It emphasized the need for a tailored approach to model building, considering the unique aspects of the dataset at hand.

Potential for Hyperparameter Optimization
An important consideration for future enhancement of our models is the application of systematic hyperparameter optimization, which we would have pursued further given more time. Techniques like Optuna, a hyperparameter optimization framework, offer an efficient and robust method to automatically determine the best hyperparameters. Utilizing Optuna or similar tools would enable a more exhaustive search across the hyperparameter space, potentially uncovering configurations that could further improve model accuracy and efficiency. This approach stands as a promising next step in refining our models, ensuring that every aspect of the neural network architecture is optimized to its fullest potential, and aligning the models even more closely with the intricate patterns and complexities of our textual data.

Future Directions
Looking ahead, the project opens several paths for future exploration. Hybrid models that integrate the strengths of MLPs and CNNs, the application of more advanced techniques like Transformers, or the exploration of ensemble methods present exciting avenues for further research and development in the field of text classification.

Broader Implications
Finally, the success of these models in classifying academic texts holds promising implications for automating and enhancing information retrieval and organization in academic and research settings. It paves the way for more sophisticated, AI-driven tools that can revolutionize how we process and categorize scholarly information.