

2304714_HC_ACCA7026_AI_Assi gnment1

by Hanzhe Cai

Submission date: 19-Apr-2024 12:22AM (UTC+0100)

Submission ID: 229914223

File name: 51943_Hanzhe_Cai_2304714_HC_ACCA7026_AI_Assignment1_571450_1732266747.docx
(173.98K)

Word count: 3395

Character count: 18491

Deep Learning Meta-Ensemble Models for IMDB Sentiment

Analysis

Hanzhe Cai 2304714
2304714@student.uwtsd.ac.uk

Abstract—This study explores the performance of text sentiment classification on the IMDB movie review dataset using CNN, GRU, LSTM, and BERT models. By training and predicting separately on this dataset, not only was the performance of each individual model evaluated, but ensemble learning techniques were further applied to construct an integrated model that employs a stacking approach to integrate the predictions of each base model. The results show that the integrated model achieves a prediction accuracy of 95.10% on the test set of IMDB dataset. This work not only demonstrates the effectiveness of different deep learning models in the task of text sentiment analysis, but also validates the potential of integrated learning approaches in improving prediction accuracy.

Keywords—Ensemble learning, IMDB Dataset, Stacking.

1. Introduction

With the rapid growth of the internet and social media, text data is becoming increasingly important [1]. These text data are rich in sentiment tendencies, which are valuable for understanding public sentiment, guiding product development and improving services. Therefore, text sentiment analysis, especially its application in areas such as film reviews, has received widespread attention in recent years. In this context, deep learning techniques, with their superior ability to handle natural language processing tasks, have become a hot research topic.

Although a single deep learning model has achieved impressive results on many text analysis tasks, differences in the focus of different models have led to their varying effectiveness in processing specific types of data. For example, CNN excel at capturing local correlations [2], [3]. LSTM and GRU excel at dealing with long-term dependencies in textual data [4], [5]. Recently, Bidirectional Encoder Representation Translator (BERT) models have set new standards on several natural language processing tasks by means of pre-training and fine-tuning [6].

In light of this, we ask the question: is it possible to further improve the accuracy of sentiment analysis by integrating these models with complementary strengths? The aim of this study is to explore the performance of different deep learning models on the task of classifying the sentiment of IMDB movie reviews, and to combine the strengths of these models by constructing an ensemble model based on a stacking approach [7], [8].

2. Literature Review

Natural Language Processing (NLP) aims to enable computers to comprehend and process human language [9]. Simultaneously, ensemble learning, which enhances a single model's performance by aggregating predictions from multiple learners, is extensively applied across diverse machine learning tasks [7].

2.1 Natural Language Processing (NLP)

NLP is dedicated to enabling computers to understand and generate human language, and its applications in areas such as language translation, sentiment analysis, and information retrieval are crucial. The technology has evolved from the initial rule-based approaches, to later machine learning techniques, to the current widely adopted deep learning methods such as CNN, RNN, LSTM, Transformer, and pre-trained models such as BERT and GPT, with each step marking a significant enhancement in NLP capabilities [10]. The introduction of deep learning has not only optimised data processing capabilities and model performance, but also achieved technological breakthroughs in several NLP tasks [9].

2.2 Ensemble Learning

In the field of NLP, ensemble learning techniques have been shown to be an effective way to improve model performance and robustness. The core idea of ensemble learning is to obtain more accurate and reliable results than any single model by combining predictions from multiple models. This approach is particularly applicable to NLP, where the diversity and complexity of linguistic data exceeds the processing power of a single model [7].

There are several main types of ensemble learning including bagging, boosting and stacking [11]. Bagging reduces the variance of the model by processing the predictions on different

subsets. The boosting method corrects errors by models correcting each other. The stacking method learns how best to combine the predictions of several different models by training a new model to combine the predictions of these models [7].

In NLP tasks, integrated learning has been widely used in areas such as sentiment analysis, text classification, and machine translation, and has achieved significant performance gains. For example, in sentiment analysis, combining predictions from different types of models provides a fuller understanding of the sentiment tendencies of a text. In addition, integrated learning has shown advantages in dealing with linguistic ambiguity and eliminating data imbalance.

3. Methodology

This work uses CNN, GRU, LSTM, BERT and meta-models for textual sentiment categorisation performance on the IMDB Movie Review dataset. This section will introduce these methods.

3.1 Convolutional Neural Networks (CNN)

In the field of NLP, the working principle of CNN adapts to the specific nature of textual data to process linguistic information in an efficient and effective way [12], [13].

Suppose there is only one convolutional kernel. For a sentence consisting of n words, $X_{1:n} = X_1 \oplus X_2 \oplus \dots \oplus X_n$, where the word vector $X_n \in \mathbb{R}^k$. Taking a window size of h , the convolved vector $W \in \mathbb{R}^{kh}$, after convolution, the output vector is $c = c_1, c_2, \dots, c_n$, where

$$c_i = f(\mathbf{w}^T \mathbf{x}_{i:i+h-1} + b)$$

where f denotes a nonlinear transformation, for m convolution kernels, each of which can extract a different feature, text convolution is the actual operation of spreading the weights of each convolution kernel into a one-dimensional vector of length $k \times d$, which is then formed into a 2D matrix $W \in (k \cdot d) \times m$, then the convolution of a short sequence corresponding to multiple convolution kernels is:

$$C = \mathbf{x}_{i:i+k-1} \times \mathbf{W} \in \mathbb{R}^m$$

3.2 Gated Recurrent Units (GRU)

GRU is a kind of Recurrent Neural Network [14]. To solve the problem of gradient vanishing in standard RNNs, GRU uses so-called Update Gates and Reset Gates.

We start by computing the update gate $\mathbb{Z}_t$ at time step t using the following equation

$$Z_t = \sigma(\mathbf{W}^{(z)}x_t + \mathbf{U}^{(z)}h_{t-1})$$

When $\mathbb{Z}_t$ is fed into the network cell, it is weighted by its own weights $\mathbb{Z}^{(z)}$ multiplied by it. For $\mathbb{Z}_{t-1}$ is treated in the same way, it keeps the information from the previous $t-1$ cells and is weighted by its own weights $\mathbb{Z}^{(z)}$ multiplied by it. Use the sigmoid function to limit the summed result to 0 to 1.

Essentially, reset doors are used to forget information:

$$r_t = \sigma(\mathbf{W}^{(r)}x_t + \mathbf{U}^{(r)}h_{t-1})$$

First, the storage content of the reset gate is calculated as follows:

$$h'_t = \tanh(\mathbf{W}x_t + r_t \odot \mathbf{U}h_{t-1})$$

As a final step, the network needs to calculate $\mathbb{Z}_t$. To do this the gate needs to be updated. It determines the memory content from the current $\mathbb{Z}'_t$ what is being collected from it, and from the previous steps $\mathbb{Z}_{t-1}$ what to collect from. The operation is as follows:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h'_t$$

3.3 Long Short-Term Memory Networks (LSTMs)

LSTM is also a kind of RNN, capable of learning long-term relevant information and solves the gradient problem of traditional RNNs when dealing with long sequential data through a gating mechanism [15]. The LSTM unit consists of an input gate, a forgetting gate, and an output gate, which control the inflow of new input information, the information that should be forgotten in the previous memory state, and the output value at the current moment,

respectively [16].

3.4 Bi-Directional Encoder Representation Translator (BERT)

BERT is a transformer-based pre-trained language representation model. Its core principle is to pre-train a deep bi-directional representation model using a large amount of textual data, and then use the model to fine-tune subsequent specific tasks [17].

BERT learns deep linguistic representations by pre-training with Masked Language Modelling (MLM) and Next Sentence Prediction (NSP) on a large-scale text corpus [18].

Insights tasks can be used to fine-tune on the pre-trained BERT, and the parameters of the entire model are fine-tuned on new tasks to optimise task-specific performance.

3.5 Stacking

Stacking is the process of obtaining the final result by combining the outputs of the base model as inputs to the metamodel [8].



4. Experiments

I started this task by individually training and validating four different models, CNN, GRU, LSTM and BERT, using the IMDB dataset. Each model was rigorously evaluated to ensure its performance on unseen data. The validation results were then meticulously collected and used as training data for the metamodels, which were constructed using a stacking approach. This combinatorial strategy aims to exploit the individual strengths of each base model. In this section I will briefly describe the four base model experiments, focusing on the metamodels.



4.1 Data Processing

A dataset of IMDB movie reviews containing 50,000 labelled reviews was used for the experiments. It was downloaded using Torchtext and divided into training, validation and test sets [19]. Text processing consisted of word splitting using spacy, limiting the vocabulary to 10,000 and initialising with GloVe vectors [20]. Text was converted into indexed sequences with overages truncated and underages filled. Sentiment labels were converted to floating point numbers for loss calculation. In the BERT experiments, specific preprocessing was

performed with its Tokenizer, including disambiguation and sequence padding or truncation to 512 tokens, as well as the generation of attention masks.

4.2 Base-Model Experiments

I first implemented a CNN model that starts with an embedding layer that converts word indexes into word vectors. Then, convolutional operations are performed on the embedding layer using three different sizes of convolutional kernels (3, 4, 5) with 100 filters of each size. Each convolutional layer is followed by a maximum pooling layer. The outputs of the convolutional and pooling layers are spliced and passed through a dropout layer, and finally a fully connected layer outputs the sentiment prediction.

Embedding Dimension	Number of Convolution	Dropout Ratio	Optimizer	Epoch	ACC	Recall	F1-Score
50	100	0.5	Adam	5	83.58%	82.79%	82.90%
100	100	0.5	Adam	6	85.77%	85.43%	85.69%
100	100	0.5	Adam	6	87.14%	87.39%	87.18%

Tab. 1. Hyperparameter selection for CNN experiments

The second experiment used the GRU model, which consists of an embedding layer, one GRU layers, and a fully-connected layer. the GRU layers are set up in a bi-directional fashion to capture both forward and backward dependencies.

Embedding Dimension	Hidden Layer Dimension	Number of GRU Layers	Optimizer	Epoch	ACC	Recall	F1-Score
50	128	1	Adam	2	85.67%	85.44%	83.70%
50	256	2	Adam	3	86.33%	83.12%	85.68%
100	256	2	Adam	3	87.00%	84.18%	86.62%

Tab. 2. Hyperparameter selection for GRU experiments

Immediately following this, I tried the LSTM model, similar to the GRU model, the LSTM model consists of an embedding layer, followed by one LSTM layers, and ultimately a fully-connected layer outputting the sentiment prediction. the LSTM layers also use a bi-directional setup.

Embedding Dimension	Hidden Layer Dimension	Number of LSTM Layers	Optimizer	Epoch	ACC	Recall	F1-Score
50	128	1	Adam	5	81.33%	80.94%	81.20%
50	128	1	Adam	5	83.35%	83.14%	82.98%
50	256	2	Adam	7	85.46%	85.23%	84.17%
100	256	2	Adam	7	87.77%	88.32%	87.84%

Tab. 3. Hyperparameter selection for LSTM experiments

Finally, I used a pre-trained BERT model in my experiments and fine-tuned it for sentiment analysis tasks.

After training and validation, I tested their performance on the test set separately:

MODEL	ACC	Recall	F1-Score
CNN	87.14%	87.39%	87.18%
GRU	87.00%	84.18%	86.62%
LSTM	87.77%	88.32%	87.84%
BERT	93.07%	92.78%	93.05%

Tab. 4. Results of base-model experiments(highest value for this indicator in red)

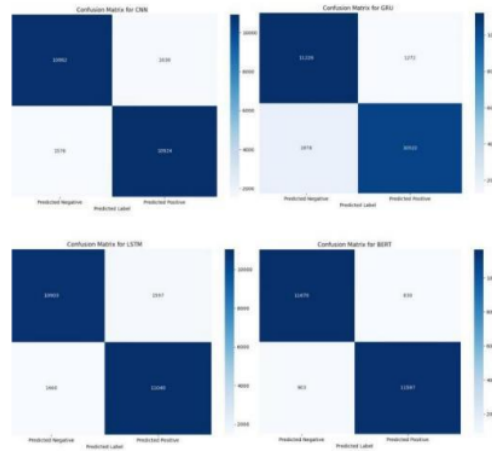


Fig. 1. Confusion Matrix of Base-model experiments

As can be seen in Table 4, BERT scores the highest among the models: 93.07% for ACC,

92.78% for Recall, and 93.05% for F1-Score. The other models - CNN, GRU and LSTM - show similar but lower metrics, with all three performance metrics in the 87% range.

4.3 Meta-Model Experiments

 constructed a logistic regression meta-model to improve the performance of sentiment analysis by integrating the prediction results of four base models (CNN, GRU, LSTM and BERT) on the IMDB movie review dataset. These predictions were stacked to form a new feature set for training the meta-model. Logistic regression was chosen as the meta-model due to its simplicity and adaptability to binary classification tasks.

MODEL	ACC	Recall	F1-Score
CNN	87.14%	87.39%	87.18%
GRU	87.00%	84.18%	86.62%
LSTM	87.77%	88.32%	87.84%
BERT	93.07%	92.78%	93.05%
Meta-Model	95.10%	94.83%	95.09%

Tab. 5. Comparison of results from all experiments (highest value for this indicator in red)

We can see the performance of the individual base models compared to the meta-model in Table 5. The meta-model has an accuracy of 95.10%, a recall of 94.83%, and an F1 score of 95.09%, which is significantly better than the individual models.

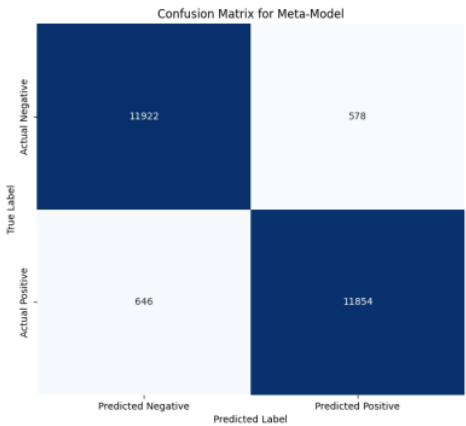


Fig. 2. Confusion Matrix of Meta-model experiments

In addition to the main experiment, I found that the meta-model dynamically adjusts the

importance of the individual base model predictions during the learning process, so I experimented with manually assigning weights to the individual base models to see the impact:

Weight Allocation [CNN:GRU:LSTM:BERT]	ACC	Recall	F1-Score
[0.25 : 0.25 : 0.25 : 0.25]	92.98%	87.14%	92.54%
[0.25 : 0.25 : 0.2 : 0.3]	94.80%	94.10%	94.77%
[0.3 : 0.3 : 0.3 : 0.1]	91.46%	90.54%	91.38%

Tab. 6. Metamodel results with different base model weights (highest value for this indicator in red)

In the table, equal weights of 0.25 for each model resulted in 92.98% accuracy, 87.14% recall, and a 92.54% F1-score. Adjusting weights to favor BERT (0.3) over LSTM (0.2), while keeping CNN and GRU at 0.25, improved recall to the highest at 94.10%, with accuracy at 94.80% and the best F1-score of 94.77%. Increasing weights for CNN and GRU to 0.3 and reducing BERT's to 0.1 decreased accuracy to 91.46% and F1-score to 91.38%, but recall was still high at 90.54%.

On top of that, I tried different models in the quest for better results:

MODEL	ACC	Recall	F1-Score
Logistic Regression	95.10%	94.83%	95.09%
SVM	94.78%	94.36%	94.10%
FNN	94.25%	93.83%	93.17%

Tab. 7. Results of meta-modelling using different models

Table 7 shows a comparison of the three models: logistic regression, SVM, and FNN. logistic regression tops the list with the highest accuracy (95.10%) and F1 score (95.09%), and the recall is 94.83%. SVM comes in a close second, and perhaps I set up the FNN to be too simple, and so it is a little bit behind in all the metrics [21].

Finally, I compare my meta-model with currently popular NLP models:

MODEL	ACC
OCaTS	93.06%
Roberta-Ba  5	94.74%
DV-ngrams-cosine + NB-weighted BON	93.68%
Space-XLNet	94.88%
GPT-4	96.40%
Meta-Model of this work	95.10%

Tab. 8. Results of different popular NLP models

Table 8 is full of work that has performed well on the IMDB dataset over the last two years, with the meta-model achieving an accuracy of 95.10%, a performance that is slightly inferior to GPT-4's 96.40% but higher than the other models and significantly reduces training costs[22]-[24]. This suggests that the ensemble model, whilst not outperforming the best single model currently available, still performs well and significantly reduces training costs, reflecting the potential of combining multiple techniques and approaches.

5. Discussion

Firstly, looking at the performance of the base models, all of them except BERT have relatively average performance. I chose the pre-trained BERT model to train on the IMDB dataset, because BERT itself is already very powerful and performs best.

The results of the meta-modelling show that the integrated approach effectively leverages the strengths of the individual models and achieves better overall predictive accuracy in sentiment analysis tasks.

At the very beginning when the gap between the LSTM model performance and the BERT model performance was too large, the meta-model was even less effective than the base model, so when the performance of the different base models varies greatly, the meta-models may not be able to take full advantage of their respective strengths. Through the experiment of adjusting the weights, I found that the performance of the meta-model suffers when the poorer-performing model has higher weights, and even if the model chooses the optimal

weights through learning, its performance is still only slightly higher than that of the optimal BERT model by two percentage points. Therefore, I believe that when the performance of each basic model is close, the stacking method may be the best way to improve the performance.

6. Conclusion

This work provides a comprehensive analysis of deep learning models on the IMDB dataset. Our results highlight the effectiveness of CNN, GRU, LSTM and BERT in capturing sentiment information from textual data. In addition, this work integrates the superiority of the methods and achieves an accuracy of 95.10% on the test set. Our findings help to advance the state-of-the-art in sentiment analysis and provide insights on the benefits of integrating learning techniques combined with diverse deep learning architectures. Future research could explore additional ensemble methods and investigate the application of these models to other sentiment analysis tasks and datasets.

REFERENCES

- [1] Z. Khan and T. Vorley, "Big data text analytics: an enabler of knowledge management," *Journal of Knowledge Management*, vol. 21, no. 1, pp. 18-34, 2017.
- [2] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of CNN and RNN for natural language processing," *arXiv preprint arXiv:1702.01923*, 2017.
- [3] S. Liao, J. Wang, R. Yu, K. Sato, and Z. Cheng, "CNN for situations understanding based on sentiment analysis of twitter data," *Procedia computer science*, vol. 111, pp. 376-381, 2017.
- [4] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 606-615.
- [5] S. Yang, X. Yu, and Y. Zhou, "Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example," in *2020 International workshop on electronic communication and artificial intelligence (IWECAI)*, 2020: IEEE, pp. 98-101.
- [6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [7] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Frontiers of Computer Science*, vol. 14, pp. 241-258, 2020.
- [8] Y. Wang, D. Wang, N. Geng, Y. Wang, Y. Yin, and Y. Jin, "Stacking-based ensemble learning of decision trees for interpretable prostate cancer detection," *Applied Soft Computing*, vol. 77, pp. 188-204, 2019.
- [9] Y. Kang, Z. Cai, C.-W. Tan, Q. Huang, and H. Liu, "Natural language processing (NLP) in management research: A literature review," *Journal of Management Analytics*, vol. 7, no. 2, pp. 139-172, 2020.
- [10] X. Liu *et al.*, "GPT understands, too," *AI Open*, 2023.
- [11] Z.-H. Zhou and Z.-H. Zhou, *Ensemble learning*. Springer, 2021.
- [12] N. I. Widiastuti, "Convolution neural network for text mining and natural language processing," in *IOP Conference Series: Materials Science and Engineering*, 2019, vol. 662, no. 5: IOP Publishing, p. 052010.
- [13] A. K. Sharma, S. Chaurasia, and D. K. Srivastava, "Sentimental short sentences classification by using CNN deep learning model with fine tuned Word2Vec," *Procedia Computer Science*, vol. 167, pp. 1139-1147, 2020.
- [14] M. Xia, H. Shao, X. Ma, and C. W. de Silva, "A stacked GRU-RNN-based approach for predicting renewable energy and electricity load for smart grid operation," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 7050-7059, 2021.
- [15] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020.
- [16] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to monitor machine health with convolutional bi-directional LSTM networks," *Sensors*, vol. 17, no. 2, p. 273, 2017.
- [17] Y. Hao, L. Dong, F. Wei, and K. Xu, "Visualizing and understanding the effectiveness of BERT," *arXiv preprint arXiv:1908.05620*, 2019.
- [18] G. Jawahar, B. Sagot, and D. Seddah, "What does BERT learn about the structure of language?," in *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*, 2019.

- [19] R. Puri and B. Catanzaro, "Zero-shot text classification with generative language models," *arXiv preprint arXiv:1912.10165*, 2019.
- [20] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532-1543.
- [21] A. T. Imam, A. Alhroob, and W. Alzyadat, "SVM machine learning classifier to automate the extraction of SRS elements," *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2021.
- [22] B. Csanády, L. Muzsai, P. Vedres, Z. Nádasdy, and A. Lukács, "LlamBERT: Large-scale low-cost data annotation in NLP," *arXiv preprint arXiv:2403.15938*, 2024.
- [23] S. Tytarenko and M. R. Amin, "Breaking Free Transformer Models: Task-specific Context Attribution Promises Improved Generalizability Without Fine-tuning Pre-trained LLMs," *arXiv preprint arXiv:2401.16638*, 2024.
- [24] I. M. Stogiannidis, P. Malakasiotis, and S. Vassos, "OCaTS: an Online Cost-aware Teacher-Student Framework to Reduce the Calls to Large Language Models," 2023.

APPENDIX

VIDEO LINK: <https://youtu.be/OjLYtiYZIZA?si=n9gZao1ttqIPOhtL>

2304714_HC_ACCA7026_AI_Assignment1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

80/100

Demonstrated the ability to build and evaluate AI models. Results are well presented. However writing skill and the presentation of the review of literature and methodology need to be improved

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5



Comment 1

Methodology section should clearly demonstrate the methods and materials used to accomplish the study.



Comment 2

Not suited for academic writing

PAGE 6

PAGE 7

PAGE 8



Comment 3

?

PAGE 9



Comment 4

Further, this study applied different models to get better results.

PAGE 10



Comment 5

Provide the in-text citation of all these compared models

PAGE 11

PAGE 12

PAGE 13

PAGE 14

GRADING FORM: ACCA 7026 GRADING

HANZHE CAI

80

ENGAGEMENT WITH LIT. SKILLS

(including current research, referencing, academic conventions and academic honesty)



Engagement with a wider range of literature.
However it lacks, analysis and synthesis of the
reviewed literature.

11

KNOWLEDGE&UNDERSTANDING SKILLS

(A systematic, conceptual understanding of knowledge, and a critical awareness of current problems and/or new insights, much of which is at, or informed by, the forefront of the discipline; a comprehensive understanding of techniques applicable to their own research)



Knowledge has a well defined focus and concepts are well explained.

7.5

COGNITIVE&INTELLECTUAL SKILLS

(Critical evaluation of current research and methodologies and develop critiques of them and, where appropriate, to propose new hypotheses; make sound judgements in the absence of complete data.)



All models are explained well in methodology. However, it lacks the details of the methods and materials.

17

PRACTICAL APPLICATION SKILLS

(Originality / creativity in the application of knowledge, tools and techniques and in tackling and solving problems in complex and unpredictable professional situations; practical use of established techniques of research and enquiry to create and interpret knowledge in the discipline.)



Base models and ensemble based on stacking is implemented and results are presented very well. Results are compared against the existing literature.

37

TRANSF. SKILL & PROF.PRACTICE

(Exercise of self-direction, autonomy and personal responsibility; plan and implement tasks at a professional level; independent learning; use appropriate media to communicate effectively and professionally to a variety of audiences; fluency of expression; systematic approach; clarity and effectiveness in presentation and organisation.)



Can communicate professionally in an appropriate way. The results are well discussed by comparing and contrasting with the literature. However, the presentation of the methodology section need to be improved.

7.5