Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Predictive Modeling And Code-Mixing Corpus For Student Learning Behaviors In E-Learning Environments

Bavani Raja Pandian[1], Kamsiah Mohamed [2] & Wan Azlan Wan Hassan @ Wan Harun[3]

[1] Tunku Abdul Rahman University of Management and Technology (TAR UMT), Jalan Genting Kelang, Setapak, 53300 Kuala Lumpur, Malaysia.
[2, 3] Universiti Selangor, Jalan Timur Tambahan, 45600 Batang Berjuntai, Selangor, Malaysia
Email: [1]bavaniraja86@yahoo.com, [2]kamsh@unisel.edu.my, [3] wan.azlan@*unisel.edu.my.*
Orchid Id number: [1] 0009-0000-9870-1988, [2] 0000-0001-5623-2378, [3] 0000-0002-6271-5402
Corresponding Author: Bavani Raja Pandian*

Amid the changing environment of e-learning platforms, the knowledge of the students' activity remains one of the critical challenges for educational organizations. However, the identified literature does not sufficiently capture the social substrates of language or the linguistic features that mediate digital communication and interaction, especially in multilingual settings such as Malaysia which is characterised by English and Malay language code-switching. This linguistic aspect is greatly responsible in properly categorizing the student behavior; the reason why new approaches are called for. In an attempt to fill this gap, this study puts forward a new approach based on code-mixed text data collected from different social media and Learning Management System (LMS) platforms to train the ML models for detecting the student's behavior. In this paper, the authors consider the Support Vector Classifier (SVC), Random Forest (RF), Logistic Regression (LR), and Naïve Bayes (NB) algorithms and part-of-word embedding and Term Frequency-Inverse Document Frequency (TF-IDF). The study is intended to capture linguistic pattern complexity underlying in code-mixed text and then improve accuracy of students learning behavior classification. Moreover, Large Language Model such as Bidirectional Encoder Representations from Transformers (BERT) is also utilized to compare the performance with traditional ML models. The methodology entails data collection from diverse e-learning platforms, preprocessing of code-mixed text data, feature extraction using word embedding techniques, and model training and evaluation. The efficacy of the suggested model is tested through rigorous experiment and validation. The results show that it is capable of precisely classifying student learning behavior. LSTM (TF-IDF) and LSTM (W2V) with augmented datasets appear to be the most effective models in predicting student learning behavior involving code mixing with the accuracy of 100%. These findings underscore the potential of ML models trained on code-mixed text data to provide valuable insights into student behavior in e-learning environments, thereby facilitating personalized and effective educational experiences.

Keywords: code-mixing, machine learning, deep learning, BERT model, prediction, student learning behavior

1) Introduction

E-learning refers to the process of delivering education and helping people learn in unconventional programs and locations. It comprises of moving within online classes, engaging with texts, participation in discussions, seeking assistance and engaging with other students and instructors. Therefore to optimise & use teaching E-Learning strategies effectively and enhance the learning process as well as supporting student achievement in online education, both the educators as well as the institutions need to understand the behaviour of E-Learning. Moreover, the use of code mixing in learning by the students when conversing in their E-Learning platforms makes it difficult to understand their learning process. The intended outcomes of this study are to develop an accurate forecasting model and an English-Malay code-mixing database. The model will be particularly developed for identifying the student actions on the E-Learning platforms with code-switching. Our study aims at enhancing comprehensiveness of the existing data analysis techniques for improving the student learning behaviour prediction models in multi-E-Learning domains using the word embedding and the DL and ML algorithms.

2) Related work

From [1], it is understood that digital platform and technology is referred to as media that can be used as a replacement for conventional classroom for transmission of teaching and learning materials, as this may be more flexible than traditional media. Based on this, [2] stress the idea of the heightened flexibility and accessibility in learning born of the new digital environment. As such [3] stress that E-Learning is crucial for the proper response to conditions that require distance learning and continuous education disruption using tools like LMS and social media. Combined with E-Learning, [4] discuss how social media support information sharing and text analysis of students' behavior. This is buttressed by [5] who propose that one can learn about communication activities, improvement of engagement, and even sentiment of clients from the social media data. Thus, the analysis of student behavior in E-Learning environments becomes central to designing and implementing useful E-Learning for all the students [6]. However, in the multicultural countries like, Malaysia they use code mixing where English and Malay are mixed, and which represents culture and language integrated [7][8]. This linguistic practice which has been identified in various E-Learning settings to by [9] offered a very useful perspective towards the engagement and performance of students. In relation to this, it has relevance in the larger field of understanding linguistic features in new media communication. To extend such patterns, it is crucial to apply ML models as a key component of educational data mining, which detect and predict outcomes of students' activities according to [10]. The models are counted as logistic regression, decision trees, and random forests that are useful to model data relationships. Moreover, improved approaches like DL models [Li11] and transfer learning models [Li12] provide far superior features for the evaluation of intricate linguistic patterns including code-mixed content. Besides using ML, one needs NLP methods to deal with text data since it is challenging for machines to decipher plain text. Therefore, NLP includes different uses like sentiment analysis and text classification as well [13]. Despite that, DL models and particularly neural networks perform outstandingly at identifying more sophisticated patterns [14], their functioning is computationally intensive. This leads to the [15] use of transformer models which contain self-attention mechanism to address long-range dependencies in a text despite the fact that they are computationally expensive [16]. Last, predictive modeling, to as explained by, is the use of past data to [17] predict events in the future and it can be used in all fields including academic. This approach helps in changing strategies used in educating the students on the basis of students' needs [18]. Therefore, this study proposes the use of DL, ML, and transformer models for an English-Malay code-mixing corpus to assess the ability of the models to forecast student learning patterns.

3) Code Mixing and Predictive Model Development Process

The method used in this study consists of an integrated model to study student learning behaviour in E-Learning environments [19]. Firstly, different social media and LMS data were gathered and raw data in the form of social media interactions was preprocessed by removing unsuitable comments and by diversifying it. In a next step, the models like Random Forest, Linear SVC, Multinomial Naive Bayes Classifier (MultinomialNB), and Logistic Regression (LR) were created and tuned based on distinct parameters as per data characteristics. Moreover, DL models such as CNNs and LSTM networks, were used to capture complex patterns and contexts of the textual data. In addition, transformers models were used to couple contextual relations and semantic data [20]. Figure 1 shows the development process of code mixing and predictive model development which was discussed in as below.

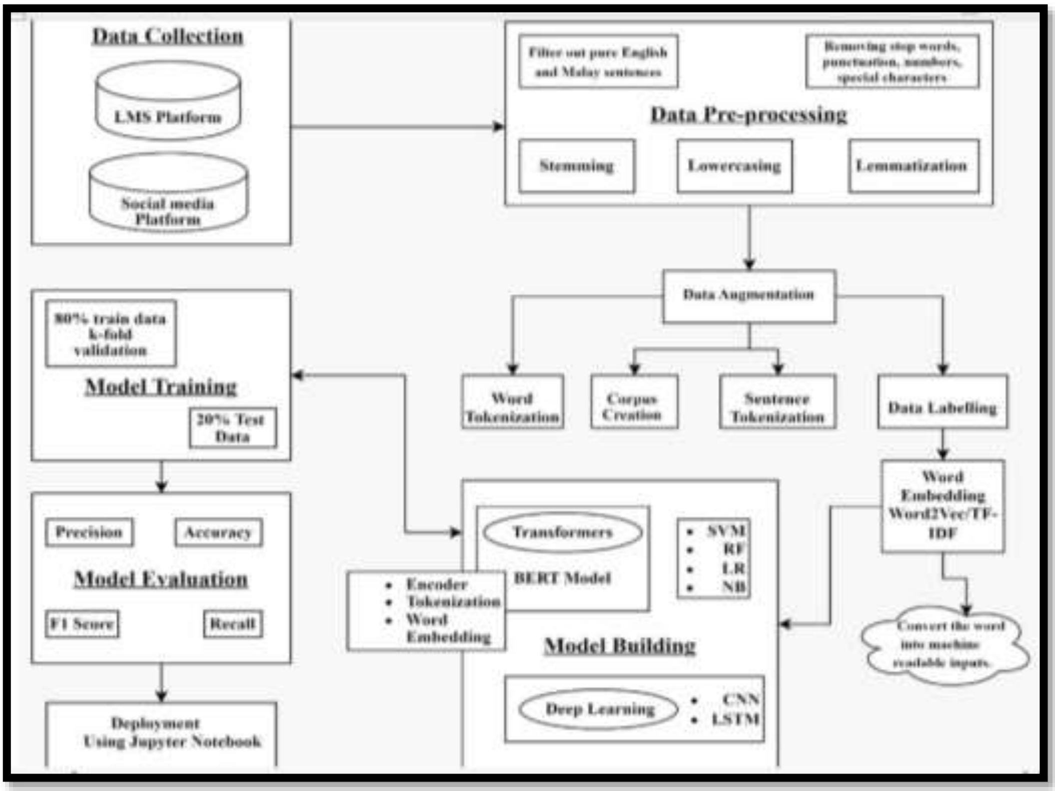


Figure 1 Code Mixing and Predictive Model Development Process

3.1 Data Collection

Facebook, twitter, Reddit, Lowyat, TikTok, KLSE, and Google Classroom are the source of the data sets used in this study. Information generated by interaction on the platform adopted for learning is called LMS data which include the Google classroom. The data set contains posts, and the comments which are gathered from 2017-2023. The overall data collected of datasets were 398539 data from 6 data sources. The dataset was scraped from multiple LMS including social media site scraping through HTML parsing scraping, API scraping, data scraping libraries scraping, third party; Apify. The dataset collected from platforms includes English and Malay code-mixed text data.

In this study, the process dataset will be utilized in two forms. Firstly, the pre-processed dataset used to develop the code-mixing corpus were placing all the words into a text file and reading from it with the Natural Language Toolkit (NLTK) library, which means that all the words from each phrase will be reading from the NLTK library and secondly to establish the labelled dataset for predictive model, It was split into two sections: Therefore for the testing data 20% of the data was employed while 80% was employed in the training data section.

3.2 Data Pre-Processing

The primary stages of data preprocessing will go through several stages as discussed below:

a. **Filtration:** This way of filtering helped achieve the separation of the dataset into two different sets of mainly English only and Malay only sentences. This makes its possible for researchers researching or developing English and Malay languages or language technologies that benefit from monolingual data.

b. **Lowercasing:** All the text is decreased to make certain that words are treated based on their cases of initial letters uniformly lowercasing helps in enhancing consistency in more linguistic analysis. It helps keep first letter case uniform when within the text there are words similar in sense.

c. **Removing punctuation:** Removing all punctuation from the text and leaving only spaces in between words. Many milestones of natural language processing can be reached with little to no consideration for the punctuation of the input and output text. The advantages of pre-processing whereby punctuation is completely excluded are as follows for NLP tasks. Firstly, it helps to exclude 'Non-Essential words and phrases', thus simplifying text and excluding most of the noise it usually contains.

d. **Removing stop words:** Lemmas are terms such as "and," "the," "is" and so on that are not useful in the general meaning of the context. It is also useful to eliminate these low frequency words since they clutter the representation of the text and may not be the most informative or contextually necessary words.

e. **Removing numbers:** In case the symbols are not related to counting, then they can be omitted to get to the textual data. This way, studies may serve to remove a number of features, to simplify analysis, and to minimize noise, that are still relevant to any given text, but are simply irrelevant to a number-based set of analyses.

f. **Removing special characters:** To clean the text, symbol or character that do not contain useful information when analyzing may be eliminated. It can be anything from emoji's to currency symbols or other symbols that are not letters or numbers. Thus, by eliminating them, Studies can clean up language, filter out the noise, or all the irrelevant information actually used in language communication, pursue the kind of language use that is more restricted but then closer to the idealized linguistic assembly line.

g. **Lemmatization:** The is another crucial procedure in the text preprocessing step. While stemming just cuts off the suffixes from the word, lemmatization targets to convert the word to its base or the simplest lexical form called lemmas. There are linguistically acceptable forms, grammar, and increased legibility of the text as a result of such work.

h. **Stemming:** It is another text preprocessing method which identifies stems or roots of words by removing their suffixes. In this step we arrive at a form known as the stem.

i. **Data Augmentation:** The next process is data augmentation which can be employed solely for enlarging training dataset. These include replacing certain word with its equivalent word or back translation where the text is translated to another language before being translated back to the translator's desired language.

j. **Tokenization:** The tokenization process that occurred here and there are two types, one is called word tokenization the other is called the sentence tokenization. Word tokenization takes place when preparing a corpus to train the tokenizer for transformers model which is the BERT model but the sentence tokenization takes place in the data collection phase in order to transform the full dataset into single sentence per row to increase easiness in the subsequent annotation process. In other preprocessing steps, it received totality of 78,988 entries.

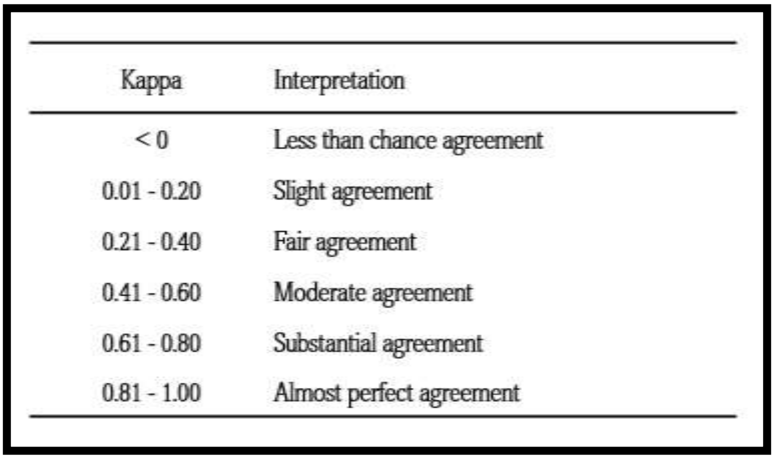
3.3 Data labelling

The annotation in this study was done manually in the sense that the researchers assigned codes and tags to the texts on their own. For the purpose of annotation, the ground truth method was adopted whereby the correct labels for students' learning behaviors were indicated according to the four categories including; activist, pragmatist, reflector and theorist. To achieve significance and reliability for annotation, ten experts were used for the purpose of this research. These experts included psychologists and counsellors who had lots of experience and understanding of students learning behavior. The knowledge enabled them to bring out justified decisions and appropriately use theories and frameworks to categorise the behaviours they noticed. Experience of different coordinators and viewpoints of the experts helped the multicoders to gain the reliability and validity in the process of manual annotation.

3.3.1 Inter-Annotator Agreement

As mentioned by [2], for nominal data, inter-annotator reliability or agreement is exclusively measured by cohen's kappa (κ). It also contributes to making annotations more genuine. Higher inter-annotator reliability as stated by [6] encompasses great quality and enhance the reproducibility of research findings. Higher degree of IA is beneficial in enhancing the risk prediction model, and in clinical handling of patients.

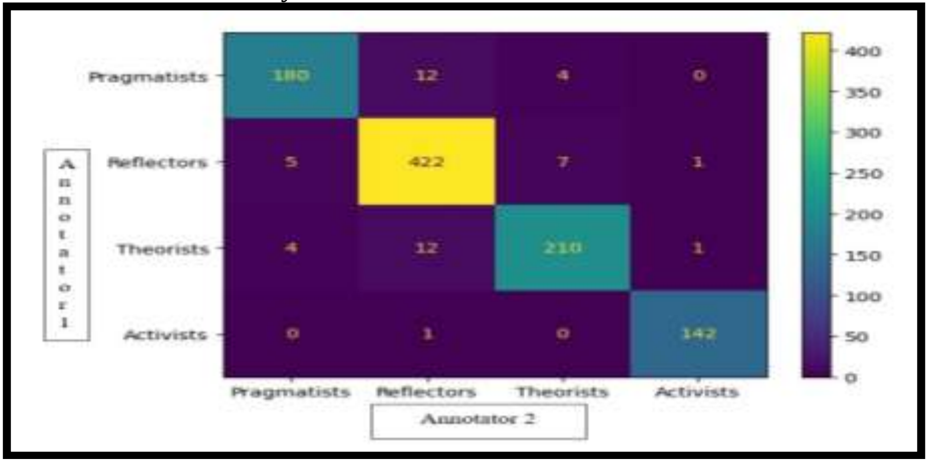
Table 1 Interpretation of kappa



Kappa	Interpretation
< 0	Less than chance agreement
0.01 - 0.20	Slight agreement
0.21 - 0.40	Fair agreement
0.41 - 0.60	Moderate agreement
0.61 - 0.80	Substantial agreement
0.81 - 1.00	Almost perfect agreement

The obtained result is calculated using the formula that is inter-annotator agreement between two annotators. A confusion matrix presented in Figure 3 below was adopted to calculating the level of agreement. This matrix made it possible to diagnose the total number of agreements for each category and to define the kappa's coefficient.

As a result from the above calculation we get a value of kappa equal to 0.928. , thereby proving the high reliability of the annotation as kappa presented in the table 1 is almost perfect. In the current study, both annotators received the same dataset of 1001 instances to label four classes of Student Learning Behaviors (SLB). A high kappa value of 0.928 shows that the opinion of the annotators was highly standardized and they had a good level of agreement in the labeling of dataset. The high degree of intraclass correlation implies that, indeed, the SLB categories were understood and interpreted similarly by the annotators, with minimal opportunities for interannotator variability.



Annotator 1 \ Annotator 2	Pragmatists	Reflectors	Theorists	Activists
Pragmatists	180	12	4	0
Reflectors	5	422	7	1
Theorists	4	12	210	1
Activists	0	1	0	142

Figure 3 Confusion Metrix

3.4 Word Embedding

In the next phase of the embedding technique such as Word2Vec and in all NLP tasks, words are transformed into machine interpretable real values using TF-IDF [3]. These representations become trainable input source to the learning models so that they can be able to reason with and understand textual data. Both Word2Vec and TF-IDF are applicative both in ML and/or DL frameworks, as they shed light into semantic similarity and importance of the word in the given corpus. Nonetheless with the rise of Transformer models such as BERT, a completely different way of doing word embedding is used. In the case of BERT, corpus is used to train the tokenizer that maps the text inputs into machine understandable format for the BERT model. Before these representations are fed into the Transformer model, they undergo three types of embedding's: A Token embedding, positional embedding and segment embedding are three types of embeddings which play significant role in models containing Transformers. The tokenizer is responsible for token embedding and positional embedding represents the position of the token in the given sequence. On the other hand, segment embedding enweights/identifies segments or inputs in the text. After the creation of a corpus and representation of these words, the next step is the actual model building – using these representations on which the NLP model can be built and trained.

4) Result and Discussion

After that, the dataset is split further into the ratio of 80% as the training set and 20% as the testing set of the model. This procedure is termed model training. Our model training using only 5 fold validation shall apply to only ML classifiers Notwithstanding, DL and BERT were trained with epoch training the 7 epoch LSTM and CNN, Fine tuning the Pre train model was 3 epochs, BERT model epoch is 3. After the training of the given model, four matrices, namely precision, accuracy, F1 score, and recall, are used to evaluate the predictive model developed. Identifying the algorithms that produce the highest values of precision, accuracy, F1 and recall turns into an important of the process.

Table 2 summarises an intermodal model comparison with different classifiers and NLP methods. Of all the approaches applied, LSTM(TF-IDF) with an augmented dataset and LSTM(W2V) with an augmented both recorded a 100% accuracy, BERT recorded a 99% whilst CNN(TF-IDF) with an augmented dataset was closely followed by CNN(W2V) with an augmented dataset both at 98%. From these results inference can be drawn that LSTM with TF-IDF as well as LSTM with W2V both with augmented datasets outperformed other models in assessing the given code-mixed sentences by achieving maximum accuracy. Furthermore, augmented data set with BERT & CNN as well as W2V also depicted almost cent percent accuracy. Therefore, LSTM (TF-IDF) and LSTM (W2V) models using augmented datasets are the most promising models for estimating student learning behaviors that include code mixing. Future studies or tests using these models may also shed light on how they can be used to handle code-mixed information. It is then added to Jupyter notebook once it has been built and validated.

Table 2 Inter Model Comparison Table

Classifiers	**Augmented**	**Non-Augmented**	**Precision**	**Recall**	**Accuracy**	**F1**
SVC (TF-IDF)	YES	-	99%	84%	98%	91%
SVC (TF-IDF)	-	YES	83%	49%	93%	62%
SVM (W2V)	YES	-	23%	25%	90%	24%
SVC (W2V)	-	YES	23%	25%	91%	24%
RANDOM FOREST (TF-IDF)	YES	-	23%	25%	91%	24%
RANDOM FOREST (TF-IDF)	-	YES	23%	25%	91%	24%
RANDOM FOREST (W2V)	YES	-	46%	25%	90%	32%
RANDOM FOREST (W2V)	-	YES	23%	25%	90%	24%
LINEAR REGRESSION (TF-IDF)	YES	-	72%	33%	93%	45%
LINEAR REGRESSION (TF-IDF)	-	YES	86%	32%	92%	47%
LOGISTIC REGRESSION (W2V)	YES	-	23%	25%	90%	24%
Logistic Regression (W2V)	-	YES	23%	25%	91%	24%
NAÏVE BAYES (TF-IDF)	YES	-	48%	26%	91%	34%
NAÏVE BAYES (TF-IDF)	-	YES	37%	25%	91%	30%

NAÏVE BAYES (W2V)	YES	-	28%	32%	43%	30%
NAÏVE BAYES (W2V)	-	YES	27%	33%	49%	30%
CNN (TF-IDF)	YES	-	98%	95%	99%	96%
CNN (TF-IDF)	-	YES	76%	52%	93%	62%
CNN (W2V)	YES	-	97%	96%	99%	96%
CNN (W2V)	-	YES	64%	52%	92%	57%
LSTM (TF-IDF)	YES	-	97%	97%	100%	97%
LSTM (TF-IDF)	-	YES	72%	54%	93%	62%
LSTM (W2V)	-	YES	61%	53%	92%	56%
LSTM (W2V)	YES	-	98%	96%	100%	97%
BERT	YES		92%	77%	99%	84%
BERT	-	YES	52%	40%	92%	45%
PRE TRAIN	-	YES	37%	29%	91%	33%
PRE TRAIN	YES	-	54%	48%	95%	51%

5) Conclusion and Future Work

Altogether, this research contributes to the growing body of knowledge on the interaction between CMB and student learning behavior in online learning environments. The broader vision here is to use further the predictive modelling and the analysis of the code-mixing data and the emergent goal is to create a set of e-learning initiatives that may better target the learners' variable linguistic background and preferences in the environment of the digital learning.

As a future work, it would be worthwhile for future work to investigate other communicative channels to which people turn to such as WhatsApp or text messages. Besides, expanding the study beyond datasets with little structure into these platforms might improve the richness and inclusiveness of the research.

6) Acknowledgement: I would like to express my sincere gratitude to Dr Kamsiah and Dr Wan Azlan for their guidance and support throughout this research. Lastly, I extend my appreciation to my colleagues and family for their encouragement and understanding during this process.

7) Funding Statement: There is no fund received for this article

8) Data Availability: Data sharing is not applicable to this article.

9) Conflict of interest: The authors declare that there is **no conflict of interest.**

References

[1] Deliwe, A. P. (2020). The Use of Learner Management System MOODLE in Promoting Teaching and Learning. Universal Journal of Educational Study, **8(12B),** pp. 8383–8392.

[2] Kumar, V., Pasari, S., Patil, V. P., & Seniaray, S. (2020). Machine Learning based Language Modelling of Code-Switched Data. Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020, Icesc, pp.552–557.

[3] Lin, C. L., Jin, Y. Q., Zhao, Q., Yu, S. W., & Su, Y. S. (2021). Factors Influence Students' Switching Behavior to Online Learning under COVID-19 Pandemic: A Push–Pull– Mooring Model Perspective. Asia-Pacific Education Studies, **30(3),** pp.229–245.

[4] Tounsi, A., & Temimi, M. (2023). A systematic review of natural language processing applications for hydro meteorological hazards assessment. In Natural Hazards (Vol. 116, Issue 3). Springer Netherlands.

[5] Zhou, J., Ran, F., Li, G., Peng, J., Li, K., & Wang, Z. (2022) Classroom Learning Status Assessment Based on Deep Learning. Mathematical Problems in Engineering, 2022.

[6] Mohd Basar, Z., Norhaini Mansor, A., Azhar Jamaludin, K., & Salwana Alias, B. (2021). The Effectiveness and Challenges of Online Learning for Secondary School Students-A Case Study. Asian Journal of University Education, **17(3),** pp.119–129.

[7] Abu Bakar, A. L., Mohd. Esa, S., Ationg, R., & Jawing, E. (2021). the English Language in the Malaysian Education System. International Journal of Education, Psychology and Counseling, **6(43**), pp.122–130.

[8] Rahim, H. A., Hasan, R. A., Hong, A. L., & Joharry, S. A. (2021). The Diachronic Malaysian English Corpus (DMEC): Design, Development and Challenges. GEMA Online Journal of Language Studies, **21(4),** pp.88–109.

[9] Shiri, F. M., Perumal, T., Mustapha, N., & Mohamed, R. (2023). A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU.

[10] Kulkarni, N., Vaidya, R., & Bhate, M. (2021). A comparative study of Word Embedding Techniques to extract features from Text. Turkish Journal of Computer and Mathematics Education, **12(12),** pp.3550–3557.

[11] Nagy, M., & Molontay, R. (2018). Predicting Dropout in Higher Education Based on Secondary School Performance. INES 2018 - IEEE 22nd International Conference on Intelligent Engineering Systems, Proceedings, 000389–000394.

[12] Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2020). Transformer

XL: Attentive language models beyond a fixed-length context. ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, pp.2978–2988.

[13] Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. Multimedia Tools and Applications, **82(3),** pp.3713–3744.

[14] Sghir, N., Adadi, A., & Lahmer, M. (2023). Recent advances in Predictive Learning Analytics: A decade systematic review (2012–2022). In Education and Information Technologies (Vol. 28, Issue 7). Springer US.

[15] Neu, D. A., Lahann, J., & Fettke, P. (2022). A systematic literature review on state-of-the-art deep learning methods for process prediction. Artificial Intelligence Review, **55(2),** pp. 801– 827.

[16] Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The Long-Document Transformer. http://arxiv.org/abs/2004.05150

[17] Fife, D. A., & D'Onofrio, J. (2023). Common, uncommon, and novel applications of random forest in psychological study. Behavior Study Methods, **55(5),** pp**.** 2447–2466.

[18] Behr, A., Giese, M., Teguim K., H. D., & Theune, K. (2020). Early prediction of university dropouts - A random forest approach. Jahrbucher Fur Nationalokonomie Und Statistik, **240(6),** pp.743-789.

[19] Yan, N., & Au, O. T. S. (2019). Online learning behavior analysis based on machine learning. Asian Association of Open Universities Journal, **14(2),** pp.97–106.

[20] Santy, S., Srinivasan, A., & Choudhury, M. (2021). BERTologiCoMix how does code-mixing interact with multilingual BERT? Adapt-NLP 2021 - 2nd Workshop on Domain Adaptation for NLP, Proceedings, pp. 111–121.