Ensemble DeBERTa Models on USMLE Patient Notes Automatic Scoring using Note-based and Character-based approaches

Bowen Long[1,*], Fangya Tan[2], Mark Newman[2]

[1]Harrisburg University of Science and Technology

[2]Department of Analytics, Harrisburg University of Science and Technology, Harrisburg, PA 17101, USA

*Correspondence: blong@my.harrisburgu.edu, swjtu.bowenlong@gmail.com

Abstract. To maximize the accuracy and efficiency of the USMLE Step 2 clinical skills examination evaluation process, we proposed an ensemble model that helps automatically score patient notes written by test takers instead of physician raters manually scoring them by appropriate features. This research used DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large as three base models and ensembled them with two different approaches: Note-based and Character-based. We concluded that LSTM Note-based ensemble topped the overall performance with an F1-score of 0.81747 on the validation data, 48% higher than the F1-score of the most effective base model (DeBERTa-v3-large). Furthermore, the performance is robust when breakdown by clinical cases and folds and applied to the testing set (0.88737 accuracy). Finally, the ensemble approach to different base models (BERT-base-uncased and BERT-large-uncased) achieved a 32% F-1 score boost. We demonstrated the ensemble model has excellent potential to improve performance in general Natural Language Understanding tasks.

Keywords: DeBERTa, DeBERTa-v3-large, LSTM, Ensemble, BERT, USMLE.

1. Background

Recording accurate information in clinical notes is vital in the patient's diagnosis plan, acute care, and post-treatment. Consequently, misdiagnosis has led to safety concerns in medical education and training [1,2]. USMLE (United States Medical Licensing Examination) Step 2 clinical skills examination is designed for medical students to learn to record the clinical status of standardized patients, including demographic characters, disease syndrome, condition and diagnosis, medical history, etc. In this exam, test-takers must interact with standardized patients and write patient notes such that physician raters will evaluate and score them by looking for critical clinical concepts or features. The current evaluation process takes up much of the physician's precious time and medical resources [3]. Meanwhile, it's susceptible to the effects of fatigue or human biases, thereby reducing the accuracy of judgment and the fairness of talent selection [4,5]. It is therefore research and practice on how to make patient notes scoring efficient and automated are emerging.

Historically, LSA (Latent Semantic Analysis) has been one of the most popular techniques in automating patient note scoring and obtaining accurate results [6,7]. However, one of the significant limitations of LSA is it's purely statistical that relies on word co-occurrence patterns, so it may fail to capture the whole meaning or the nuances of word usage in different contexts [8], as it's often encountered in clinical interviews, patients use various expressions to describe same disease syndrome or condition. Since transformer architecture was first introduced in a 2017 paper titled "Attention is All You Need" by Vaswani et al., which used a self-attention mechanism to compute contextualized word embeddings. Large Language Model or language model with transformers have been applied in numerous NLU tasks across a wide range of business areas or industries, such as building chatbots [9,10,11], improving product recommendations [12], analyzing financial reports or news articles [13,14] as well as the increasing LLM applications in healthcare [15], which enhanced the efficiency of medical resources allocation and provided appropriate medical services to patients. Its exploration in USMLE patient notes automatic scoring has also achieved significant

results. For instance, Zhou et al. developed a multi-level transfer learning-based model using BERT to identify relevant phrases regarding two specific symptoms, "Headache" and "Abdominal Pain," in patient notes and reported an accuracy of 0.92 [18]. Ganesh's research evaluated a series of mainstream LLMs, including the performance of BERT, RoBERTa, and DeBERTa in generic patient notes, and concluded that DeBERTa achieved the highest F-1 score [19]. DeBERTa is a state-of-the-art language model improved upon previous SOTA PLMs (e.g., BERT, RoBERTa, UniLM) applied to various NLP tasks in medical settings and demonstrated significant potential in automatically analyzing and classifying patient notes [20,21,22]. Lu et al.'s work [23] used DeBERTa as the backbone model. It investigated the impact of preprocessing techniques to score patient notes, which achieved an accuracy score of 0.88658 in the same dataset used in this research.

DeBERTa has iterated three versions. DeBERTa-base is the first iteration of DeBERTa models improved upon BERT and RoBERTa models, which outperformed RoBERTa on most NLU tasks with 80GB training data. DeBERTa-v3 further enhanced the efficiency of DeBERTa using ELECTRA-Style pre-training with Gradient Disentangled Embedding Sharing [39]. Finally, motivated by the high accuracy of DeBERTa models, we decided to use DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large as the base models and further improved their performance with ensemble techniques.

2. Introduction

Ensemble techniques can be particularly useful in analyzing patient notes as the language context is usually complex and diverse [24,25]. Furthermore, combining multiple LLMs makes capturing a broader range of linguistic patterns possible and improves the system's overall performance [26,27]. An ensemble has several ways, including majority voting, weighted voting, bagging, stacking, etc. [28]. This research proposed Note-based and Character-based ensemble methods to stack the output of three base DeBERTa models, including DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large.

Through a Note-based ensemble, we aimed to synthesize three base models' performances or highlight the best-performing base model in the original note format to identify the spans of characters that capture the features in a patient note. One note-based ensemble method we used is Grid Search; Grid Search is one of the most common methods for hyperparameter tuning. It tries to exhaust the search of a subset of hyper-parameter space and select the value that maximizes model performance [33]. As identified in this research, some base models perform better than others. To give them more contribution when making a prediction, we used grid search to tune the weight of each base model's contribution when predicting the locations of features in a note. As an extension of simply assigning weights to each base model, we trained an LSTM (Long short-term memory) model to stack three base models. LSTM is one of the most powerful algorithms that can predict sequences of variable lengths for both linear and non-linear data. Its application in the medical field is becoming increasingly popular and achieved remarkable success [29,30,31,32]. Xia et al. proposed ensemble algorithms of LSTM to handle the diversity of clinical data and achieved superior performance with the most significant AUCROC value of 0.8451 and demonstrated that LSTM is capable of dynamical prediction in complex clinical situations [29]. Baccouche et al. proposed an LSTM-based ensemble-learning framework to analyze medical features in the EHRs (electronic health records) and showed that the framework could lead to highly accurate models adapted for actual clinical data and diagnosis use [30]. Liu et al. proposed collective learning-based classifier of LSTMs to analyze protected health information (PHI) present in clinical data and obtained the highest micro F1-scores in the CEGS N-GRID NLP challenge [31]. Due to LSTM's adaptability and superior performance in clinical data, we drove in and used it to ensemble the prediction probability vectors of three base models. Like the LSTM ensemble, we trained another new classifier on the base models through Voting Classifier. We implemented three machine

learning algorithms for this method: Decision Tree, KNN and Naïve Bayes. Unlike LSTM, aiming to directly predict the locations of features within a note, in Voting Classifier, we want to further highlight the contribution of the best-performing model by using it as the target in each feature/note pair, thereby, towards different correspondences between note and feature, we want always only to select the best-performing base model to predict feature locations.

Through a character-based ensemble, we aimed to capture each base model's nuances or granular contribution level toward each character. Yoh et al. proposed a character-based model for a language input method that solves the unknown words that word-based models suffered by exploiting character-aligned corpora automatically generated by a monotonic alignment tool and outperformed the word-based baseline model [34]. Mamoru et al. proposed a character-based method for detecting malicious PE files and showed the effectiveness of leveraging specific tokens in the characters to extract features and learn machine learning systems [35]. Ramirez-Orta et al. proposed a novel character-based strategy that splits the input document in character n-grams and combines their corrections into the final output using a voting scheme, which achieved equivalent performance to an ensemble of many sequence models but is much more resource-efficient [36]. Inspired by the proven-tracking records of efficiency and effectiveness of character-based methods, we implemented four machine learning algorithms for each character in the patient note to ensemble the prediction probabilities from three base models to predict whether it captured the feature.

In summary, to optimize the efficiency and accuracy of the scoring evaluation system, we proposed a model that automatically identifies the locations of features within patient notes. We stacked DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large output using Note-based and Character-based ensemble approaches to identify the locations of features within patient notes. We used Grid Search, LSTM, and Voting Classifier in the Note-based approach to building models. In a Character-based ensemble, we made models using Decision Tree, Logistic Regression, Naïve Bayes, and XGBoost for character/feature pairs with and without smote resampling. The proposed model achieved SOTA performance among existing research literature.

3. Data

The dataset we studied is the USMLE Step 2 Clinical Skills Patient Note [16], also used for a Kaggle competition on automated scoring of clinical patient notes [17], which we collected for this research. Table 1 shows the key columns of this dataset. ‘id’ is the unique identifier for each patient note/feature pair. ‘case_num’ indicates which clinical case this pair is for. ‘pn_history’ is the text written by test takers detailing important information related to the patient during his/her assessment of the standardized patient. To score this clinical interview note, the physician is looking for a feature (key concept) relevant to this case: ‘feature_text’ column. ‘location’ is character spans indicating the location of the characters in the note captured by the feature. If the note doesn’t capture the specific feature, it will be an empty list in the table, such as id 30772_310. This research aims to develop an automated system of identifying the locations of relevant features, a.k. ‘location’ column within each patient note, based on the correspondence between the patient note and feature (pn_history/feature_text columns pair).

Table 1: Key columns of USMLE patient note data

id	case_num	pn_history	feature_text	location
30772_310	3	HPI 35 YO M IN OFFICE C/O BURNING EPIGASTRIC ...	Awakens-at-night	[]
81385_807	8	67-year-old female, has come to the physician'...	Hallucinations-after-taking-Ambien	[]
01809_009	0	17 year old male presenting with heart poundin...	heart-pounding-OR-heart-racing	['33 47', '33 38;52 58']
60922_605	6	17 YO MALE C/O CHEST PAIN SINCE YESTERDAY. P...	Exercise-induced-asthma	['535 559']
21372_203	2	Dolores Montgomery, a 44-year-old female, has ...	Sexually-active	['399 414']

In the patient note data, there are 10 clinical cases from standardized patients. Each case has 100 notes collected from test takers with 9 to 18 features that physicians seek to score the notes. Figure

1 shows the word cloud of all patient notes from case-0, Table 2 shows the features the physician raters are looking for to score notes from case-0.

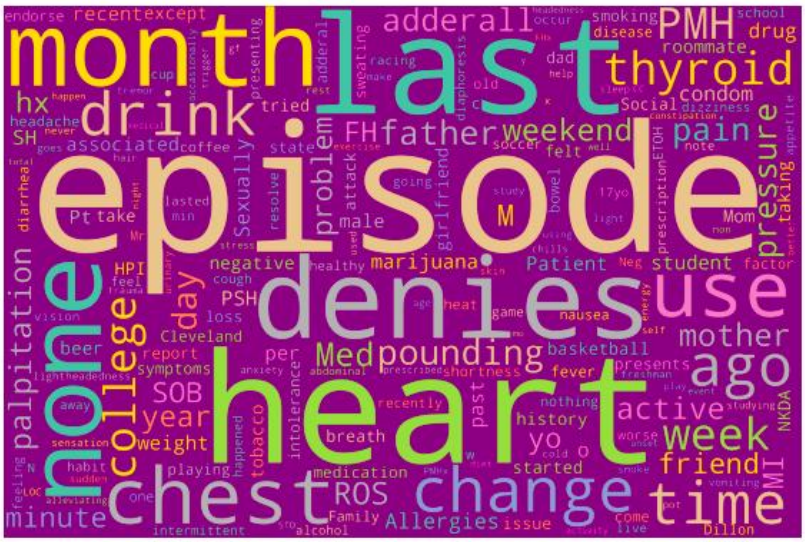

Figure 1: word cloud of patient notes from clinical case-0

Table 2: Features of Clinical case-0

case_num	feature_num	feature_text
0	0	Family-history-of-MI-OR-Family-history-of-myocardial-infarction
0	1	Family-history-of-thyroid-disorder
0	2	Chest-pressure
0	3	Intermittent-symptoms
0	4	Lightheaded
0	5	No-hair-changes-OR-no-nail-changes-OR-no-temperature-intolerance
0	6	Adderall-use
0	7	Shortness-of-breath
0	8	Caffeine-use
0	9	heart-pounding-OR-heart-racing
0	10	Few-months-duration
0	11	17-year
0	12	Male

The note length ranges from 200 to 950 characters. On average, for each note, the number of indexes in the location list that capture features is 8 to 17 by case. This research employed 5-fold cross-validation to split the 14300 instances of the patient note data and develop models for automatic scoring (feature identification). The final model developed will be scored using hidden testing data from Kaggle using micro F1-score, which is the same as the accuracy score in this research. Table 3 shows summary statistics of the patient note data.

Table 3: Summary statistics of Patient Note data

Case	# Features	# Notes	# Instances (Feature/Note Pairs)	Avg Length of Notes	Min Length of Notes	Max Length of Notes	Avg # Indexes in Location List
0	13	100	1300	835	418	950	11
1	13	100	1300	790	359	950	14
2	17	100	1700	841	523	950	13

3	16	100	1600	772	210	950	12
4	10	100	1000	827	469	950	17
5	18	100	1800	839	522	950	15
6	12	100	1200	807	416	950	12
7	9	100	900	837	428	950	15
8	18	100	1800	866	476	950	14
9	17	100	1700	753	200	950	8

After inspecting the training data, we modified the text of feature 2 of case 2 before applying the models. Originally this feature was written as "Last-Pap-smear-I-year-ago". We changed the "I" to '1', so now the feature 2 we want to identify in case 2 become "Last-Pap-smear-1-year-ago".

4. Methods and Models

In this research, we took patient note/feature pairs as input and used DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large as base models to predict feature locations. Then, the three outputs of base models are stacked using Note-based and Character-based ensemble approaches to improve prediction accuracy further. We employed 5-fold cross-validation to develop both base models and stack models. Figure 2 shows the research framework employed in this research. In the Note-based ensemble, we built ensemble models on the prediction probability vectors of three base models using Grid Search, LSTM, and Voting Classifier. In a character-based ensemble, we expanded each note to a sequence of characters and then treated the correspondence between each character and feature as an instance for the ensemble. After this transformation, the data became highly imbalanced, and the percentage of "1" (characters that captured features) is only 1.66%. Therefore, for the character-based ensemble, we further divided it into two categories an ensemble with and without using SMOTE resampling. Within each category, we implemented four machine learning algorithms, Decision Tree, Logistic Regression, Naïve Bayes, and XGBoost.

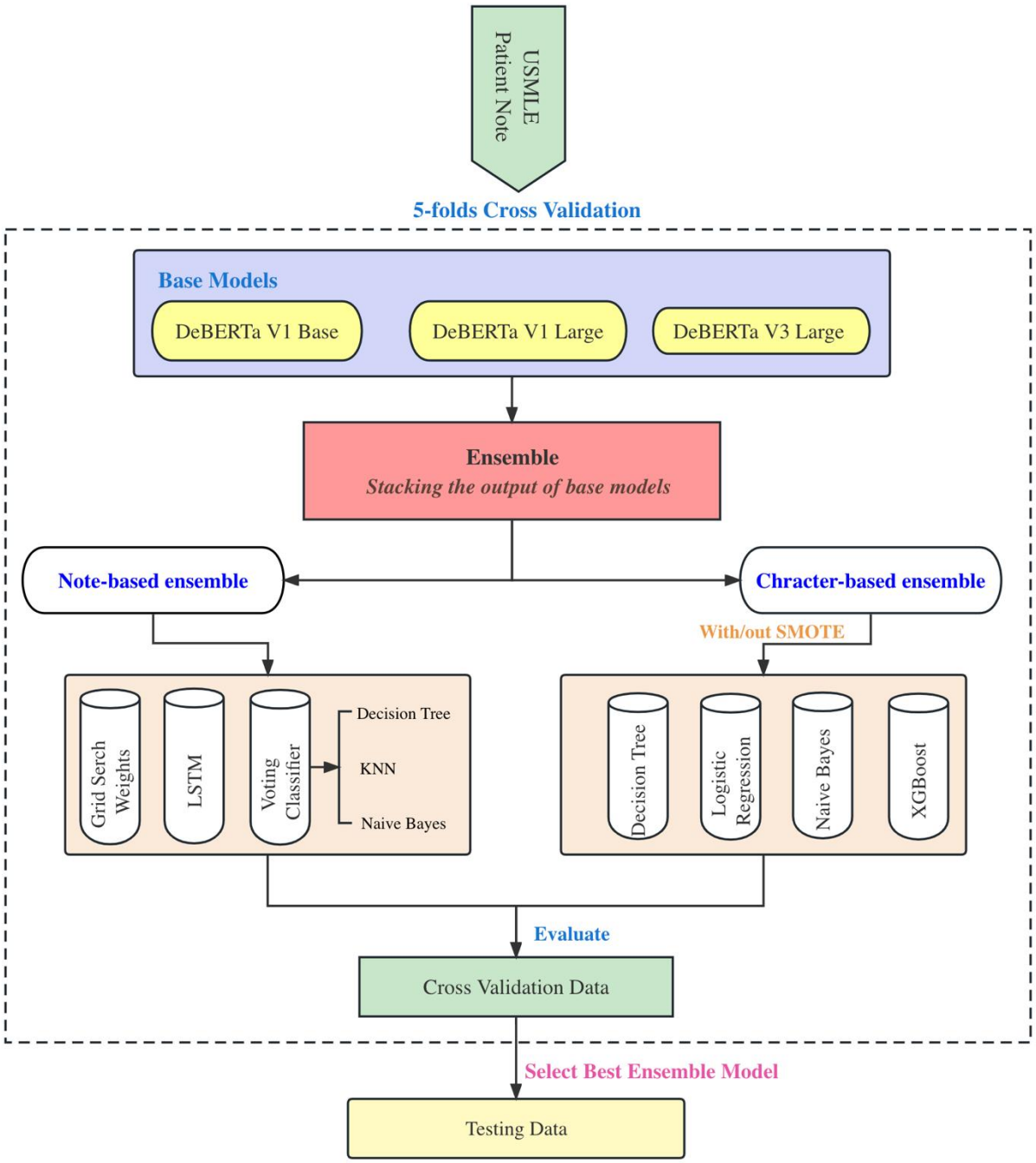

Figure 2. Research framework of USMLE Patient Note Automatic Scoring Modeling

4.1 DeBERTa

DeBERTa is a Transformer-based neural language model opened by Microsoft in 2021. It improved upon previous SOTA PLMs with three novel techniques: a disentangled attention mechanism, an enhanced mask decoder, and a new virtual adversarial training method [37]. Up to now, DeBERTa has iterated three versions. The DeBERTa V3 large model has 24 layers, 304M backbone parameters, and a vocabulary containing 128K tokens. It uses ELECTRA-Style pre-training to improve the efficiency of the model further. This research used DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large as base models and used 5-fold cross-validation to perform extensive hyperparameter tuning including the number of epochs, batch size, dropout rate,

and learning rate. As a result, we obtained batch sizes of 32, 32, and 64, respectively, for DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large, and a dropout rate of 0.2 for the final fully connected layer of all of them. Finally, the results from three base models are used as input to build ensemble models to score patient notes.

4.2 Note-based ensemble

In a Note-based ensemble, we preserved the original format of the data, the correspondence between each note and each feature is an instance, the feature location list is the target, and then we used the vectors of prediction probabilities from the base models to establish predictors and build models. We used three methods for note-based ensemble: Grid Search, LSTM, and Voting Classifier.

4.2.1. Grid Search

First, we assigned weights w1, w2, and w3 to the prediction probabilities of each instance from DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large, and the sum of the weights equals 1. Then we used grid search to divide each weight from 0-1 into 100 equal parts and got 100*100*100 combinations. Lastly, we scored all the combinations on the same validation sets by 5-fold cross-validation used to develop base models and selected the mix with the highest accuracy as the best weights. DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large obtained the best weight combinations of 0, 0.4848, and 0.5152.

4.2.2. LSTM

LSTM (Long Short-Term Memory) is a variant of traditional RNN. Compared with classic RNN, it can effectively capture the semantic association between long sequences and alleviate the phenomenon of gradient vanishing or explosion. In this research, we put three vectors of prediction probabilities from base models as input. We also put the location list of each instance corresponding to the feature as the output vector to model with LSTM. Each output vector is composed of 0 or 1, 1 indicates that the note character corresponding to the index has captured the feature, and 0 does not. Through calculation, it is found that the longest note in the training data has a total of 950 characters. We used 0 padding for the note whose length is less than 950. Therefore, each record of the LSTM model consists of 3*950 input vectors and 1*950 output vectors. Tuning and adjusting included epochs, learning rate, and the number of hidden layers and units using the same 5-fold cross-validation sets developed base models. We selected the optimal LSMT model, which has one hidden layer with 320 output units, 50 epochs, a 0.001 learning rate, and a batch size of 32.

4.2.3. Voting Classifier

A Voting Classifier is a machine learning estimator that trains various base models or estimators and predicts based on aggregating the findings of each base estimator. Compared with a standard Voting Classifier, which typically using the exact target of based models to train an ensemble model, we used the best-performing base estimator out of DeBERTa-base, DeBERTa-large, DeBERTa-v3-large as the target based on their corresponding accuracy for each instance. Besides, instead of using the original prediction probabilities from the base models as input, we engineered nine variables based on the characteristics of the base model results, including the average probability, the number of probabilities predicting “1”(probability of the index >=0.5) of each base estimator, and the overlap percentage between each base estimator’s vector of prediction probabilities. Then, we implemented three machine learning algorithms, including Decision Tree, KNN and Nave Bayes. These algorithms used the nine engineered variables as the predictors, a base estimator with the highest accuracy of each instance as the target for modeling to select the best estimator based on the characteristics of base estimator results. Tuning the hyper-parameters of each algorithm, we set the decision tree fully grown, the KNN with neighbors 5, and the Naïve Bayes with a Gaussian distribution.

4.3 Character-based ensemble

In a character-based ensemble, we expanded each note to a sequence of characters and then treated the correspondence between each character and feature as an instance for the ensemble. So for a note with a length of 950, to determine whether it captures a specific feature, there will be 950 records in a character-based ensemble model. Each record has three individual prediction probabilities from DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large the ground truth of each record will be whether this character in the note captured the targeted feature. We implemented four machine-learning algorithms for the character-based ensemble. They are Decision Tree, Logistic Regression, Naïve Bayes, and XGBoost. In addition, we calculated that when we transformed the original data into this character and feature correspondence format, the data became highly imbalanced, and the percentage of 1 is only 1.66%. To tackle the imbalance issues, we also implemented SMOTE to resample the data after transformation and compared it with model performance without resampling. After hyper-parameter tuning, we set the number of estimators for XGBoost to 150, while other models perform best with default settings.

5. Results and Discussion

In our experiments, we used 5-fold cross-validation and developed our base and ensemble models using 14300 note/feature pairs. Further, we expanded each note to a series of characters when developing character-based ensemble models. There are 817 characters per note on average, so the cross-validation data became 11680163 character/feature pairs. After extensive model tuning and adjustment, we selected the model with the best performance under each methodology. The key parameters of each model have been discussed in Section 4. Table 4 shows the performance of base models. Table 5 shows the performance of each ensemble model. The results reported are based on the aggregated performance of validation sets by 5-fold cross-validation.

Table 4. Performance of base models on validation sets by 5-fold cross-validation

Base Models	Accuracy	Precision	Recall	F1-Score
DeBERTa-base	0.86380	0.56552	0.51408	0.53857
DeBERTa-large	0.87703	0.57418	0.52195	0.54682
DeBERTa-v3-large	0.88639	0.58031	0.52752	0.55266

Table 5. Performance of ensemble models on validation sets by 5-fold cross-validation

Ensemble	Models	Accuracy	Precision	Recall	F1-Score
Note-based	Grid Search	0.89579	0.58646	0.53312	0.55852
	LSTM	0.89625	0.81307	0.82192	0.81747
	Voting Classifier-Decision Tree	0.89550	0.52599	0.58154	0.55237
	Voting Classifier-KNN	0.89555	0.52847	0.58100	0.55349
	Voting Classifier-Naïve Bayes	0.89527	0.52620	0.57862	0.55117
Character-based	Decision Tree	0.86401	0.48407	0.50769	0.49560
	Logistic Regression	0.86533	0.51871	0.55678	0.53707
	Naïve Bayes	0.86424	0.49283	0.57868	0.53231
	XGBoost	0.86537	0.51511	0.56609	0.53940
	Decision Tree-smote	0.85278	0.47778	0.50109	0.48916
	Logistic Regression-smote	0.85408	0.51196	0.54954	0.53009
	Naïve Bayes-smote	0.85301	0.48642	0.57116	0.52539
	XGBoost-smote	0.85412	0.50842	0.55873	0.53239

The results demonstrated that the best base model is DeBERTa-v3-large with an accuracy of 0.88639 and an F1-score of 0.55266. Using the note-based ensemble approach, all five ensemble models performed better than the base models. Using the character-based ensemble approach, on average, models without smote resampling achieved better performance than those with smote resampling; however, all eight character-based ensemble models got lower accuracy than the best base model (DeBERTa-v3-large). Figure 3 compares the accuracy and F1-score of the best model under each methodology. Based on the results, LSTM note-based ensemble achieved the best result of all methods with an accuracy of 0.89625 and an F1-score of 0.81747, 48% higher than the best base model (DeBERTa-v3-large with an F1 score of 0.55266). Between two ensemble approaches, from an accuracy perspective, note-based ensemble > character-based ensemble without resampling > character-based ensemble with resampling. Therefore, the ensemble closer to the original vector format of prediction probabilities of base models, the better. This pattern also existed when we broke down the performance of 5 note-based ensemble models: LSTM and Grid Search ensembled directly on the original prediction vectors achieved higher accuracy and F1 score than three voting classifier models that are ensembled on the engineered characteristics of the original vectors.

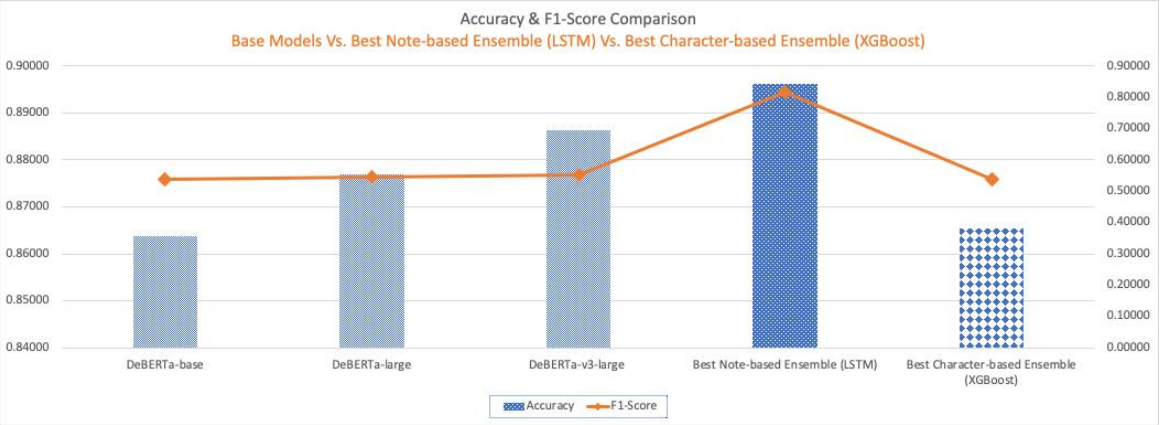

Figure 3: Accuracy & F1-score comparison on validation sets

Figure 4 shows the accuracy pattern of our LSTM experiments. The accuracy increased with the increase of the number of epochs, batch size, learning rate, and complexity of hidden layer structure until reaching 50 epochs, batch size of 32, learning rate of 0.0001, and one hidden layer with 320 units. Therefore, we selected this set of hyperparameters as our best LSTM model for the highest accuracy. As the accuracy of LSTM and grid search are relatively close, to choose the most accurate and robust model to test and score on the hidden testing set provided by Kaggle, we broke down the model performance of LSTM and Gird Search by ten clinical cases and five folds on the validation sets by 5-fold cross-validation. Figure 5 compares the two models' accuracy by clinical cases, and Figure 6 compares the accuracy by folds. From the results, LSTM constantly achieved higher accuracy across all five folds and all 10 cases other than case-2. Therefore we determined LSTM as our final model for testing.

Figure 4: Accuracy pattern of LSTM on validation sets

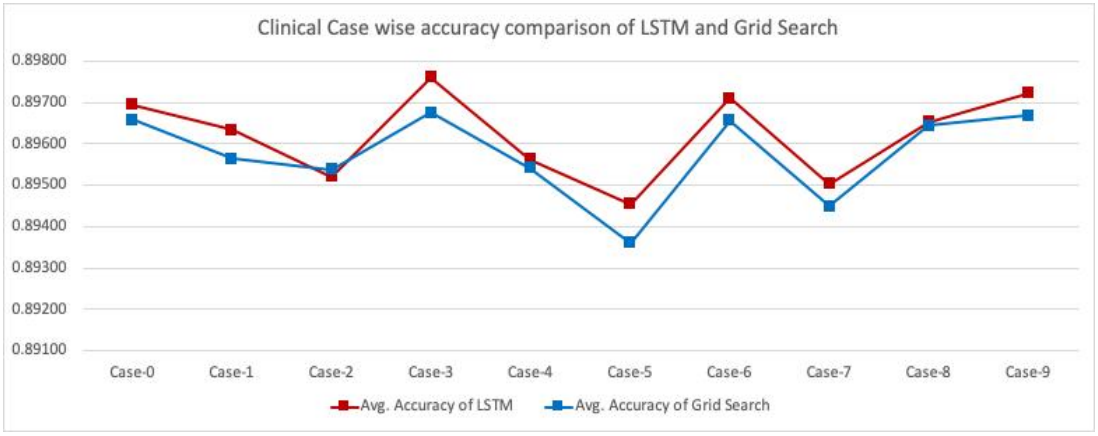

Figure 5: LSTM & Grid Search accuracy comparison by clinical cases

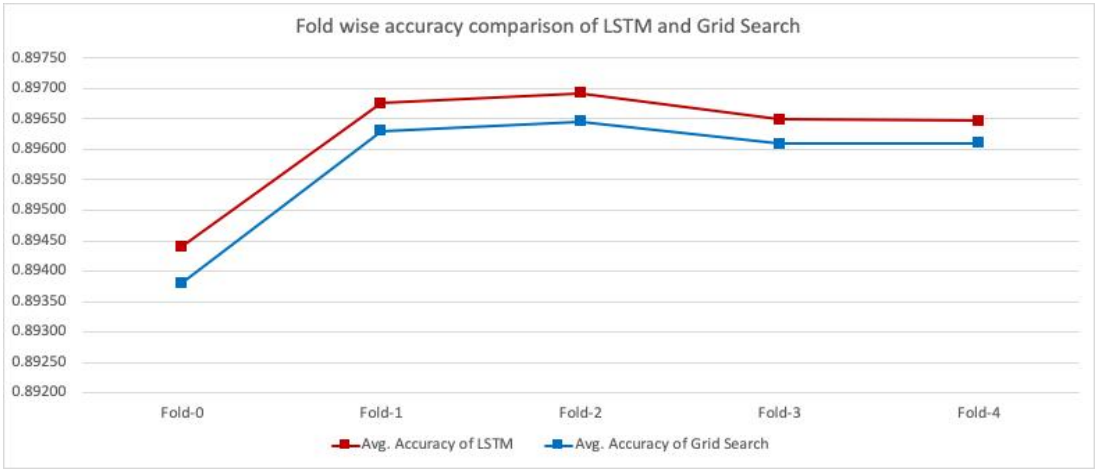

Figure 6: LSTM & Grid Search accuracy comparison by 5 folds

Based on the hidden testing test sponsored by NBME, our final model used DeBERTa-base, DeBERTa-large, and DeBERTa-v3-large as three base models and ensembled with LSTM note-based approach achieved an accuracy score of 0.88737, which is better than all three base

models as observed on the cross-validation data. To further demonstrate the consistency and practicality of this ensemble approach, we used BERT-base-uncased and BERT-large-uncased as two base models and ensembled with LSTM. Table 6 shows the performance. Compared to BERT-base-uncased and BERT-large-uncased, the LSTM ensemble achieved a 32% higher F1 score.

Table 6: Performance of LSTM note-based ensemble when use BERT models as base

Models	Accuracy	Precision	Recall	F1-Score
BERT-base-uncased	0.83380	0.54588	0.49622	0.51987
BERT-large-uncased	0.84657	0.55424	0.50382	0.52783
LSTM note-based Ensemble	0.85598	0.72051	0.65497	0.68618

6. Conclusion and Future Work

This is one of the few published studies using machine learning approaches to automate the scoring (feature identification) of patient notes. The predicted results positively impact improving the efficiency and accuracy of manual scoring that trained physicians conduct currently.

Of the two ensemble approaches employed, the note-based ensemble achieved better performance than character-based ensembles. LSTM topped the overall performance with an accuracy score of 0.89625 and an F1-score of 0.81747 on the validation data, 48% higher than the F1-score of the best base model (DeBERTa-v3-large). Furthermore, the performance is robust when breakdown by clinical cases and folds and applied on a testing test. Based on all the ensemble methodologies employed and their accuracy, we concluded that the ensembles with less feature engineering on the original prediction vectors of base models perform best. Finally, to demonstrate the consistency and practicality of the proposed ensemble approach using LSTM, we applied it to different base models (BERT-base-uncased and BERT-large-uncased). We achieved an equivalent performance boost compared with using DeBERTa models as a base. Therefore, our proposed ensemble approach has excellent potential to improve the performance of general NLU tasks when the ensemble is considered. On a high level, we summarized our findings into 3 bullet points:

- The Note-based ensemble is better than the Character-based ensemble
- Ensembles with less feature engineering on the original output of base models performs best
- LSTM Note-based ensemble can potentially improve general NLU task performance

One limitation of the research is that the performance of the final model is evaluated on a hidden dataset only using an accuracy score (micro F1-score). However, considering the exceptional F1-score and recall rate of the LSTM ensemble and utility when considering patient notes scoring using these metrics [40], the actual usefulness of the proposed approach needs to be more valued. Also, it is worthwhile to explore the impact of data preprocessing, such as stopwords removal, stemming, and lemmatization [23], and the effect of feature reduction techniques [38] on model accuracy.

Funding: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: https://www.kaggle.com/c/nbme-score-clinical-patient-notes/data

Acknowledgments: We thank Pavel Motloch for his helpful resources of techniques. His blog can be found here:

https://www.motloch.net/blog_entries/nbme_ours.html

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Singh, H., & Graber, M. L. (2015). Improving diagnosis in health care--the next imperative for patient safety. The New England journal of medicine, 373(26), 2493-2495.

[2] Newman-Toker, D. E., Peterson, S. M., Badihian, S., Hassoon, A., Nassery, N., Parizadeh, D., ... & Robinson, K. A. (2022). Diagnostic Errors in the Emergency Department: A Systematic Review.

[3] Salt, J., Harik, P., & Barone, M. A. (2019). Leveraging natural language processing: Toward computer-assisted scoring of patient notes in the USMLE Step 2 Clinical Skills exam. Academic Medicine, 94(3), 314-316.

[4] Kahol, K., Leyba, M. J., Deka, M., Deka, V., Mayes, S., Smith, M., ... & Panchanathan, S. (2008). Effect of fatigue on psychomotor and cognitive skills. The American Journal of Surgery, 195(2), 195-204.

[5] Croskerry, P. (2003). The importance of cognitive errors in diagnosis and strategies to minimize them. Academic medicine, 78(8), 775-780.

[6] Spickard III, A., Ridinger, H., Wrenn, J., O'brien, N., Shpigel, A., Wolf, M., ... & Denny, J. (2014). Automatic scoring of medical students' clinical notes to monitor learning in the workplace. Medical Teacher, 36(1), 68-72.

[7] Gefen, D., Miller, J., Armstrong, J. K., Cornelius, F. H., Robertson, N., Smith-McLallen, A., & Taylor, J. A. (2018). Identifying patterns in medical records through latent semantic analysis. Communications of the ACM, 61(6), 72-77.

[8] Hu, X., Cai, Z., Wiemer-Hastings, P., Graesser, A. C., & McNamara, D. S. (2007). Strengths, limitations, and extensions of LSA. In Handbook of latent semantic analysis (pp. 413-438). Psychology Press.

[9] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.

[10] Wang, C., Dai, S., Wang, Y., Yang, F., Qiu, M., Chen, K., ... & Huang, J. (2022). Arobert: An asr robust pre-trained language model for spoken language understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 30, 1207-1218.

[11] Chen, H., Liu, X., Yin, D., & Tang, J. (2017). A survey on dialogue systems: Recent advances and new frontiers. Acm Sigkdd Explorations Newsletter, 19(2), 25-35.

[12] Ray, B., Garain, A., & Sarkar, R. (2021). An ensemble-based hotel recommender system using sentiment analysis and aspect categorization of hotel reviews. Applied Soft Computing, 98, 106935.

[13] Araci, D. (2019). Finbert: Financial sentiment analysis with pre-trained language models. arXiv preprint arXiv:1908.10063.

[14] Liu, H. (2018). Leveraging financial news for stock trend prediction with attention-based recurrent neural network. arXiv preprint arXiv:1811.06173.

[15] Arora, A., & Arora, A. (2023). The promise of large language models in health care. The Lancet, 401(10377), 641.

[16] Yaneva, V., Mee, J., Ha, L. A., Harik, P., Jodoin, M., & Mechaber, A. (2022, July). The USMLE® step 2 clinical skills patient note corpus. Association for Computational Linguistics.

[17] NBME – Score Clinical Patient Notes, Identify Key Phrases in Patient Notes from Medical Licensing Exam. Retrieved Mar 7.2023 from

https://www.kaggle.com/c/nbme-score-clinical-patient-notes/data

[18] Zhou, J., Thakkar, V. N., Yudkowsky, R., Bhat, S., & Bond, W. F. (2022, December). Automatic Patient Note Assessment without Strong Supervision. In Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI) (pp. 116-126).

[19] Ganesh, J. (2022). Transformer-based Automatic Mapping of Clinical Notes to Specific Clinical Concepts. Arizona State University.

[20] Cao, L., Gu, D., Ni, Y., & Xie, G. (2019). Automatic ICD code assignment based on ICD's hierarchy structure for Chinese electronic medical records. AMIA Summits on Translational Science Proceedings, 2019, 417.

[21] McMaster, C., Chan, J., Liew, D. F., Su, E., Frauman, A. G., Chapman, W. W., & Pires, D. E. (2023). Developing a deep learning natural language processing algorithm for automated reporting of adverse drug reactions. Journal of Biomedical Informatics, 137, 104265.

[22] Kalyan, K. S., Rajasekharan, A., & Sangeetha, S. (2021). AMMU: a survey of transformer-based biomedical pretrained language models. Journal of biomedical informatics, 103982.

[23] Lu, S. Y. F., Balaji, S., Shenoy, N., Bakhtawar, M., Chan, J. H., & Thanapattheerakul, T. The Impact of Preprocessing on the Automated Scoring of the USMLE Step 2 Clinical Skills Exam.

[24] López-Úbeda, P., Martín-Noguerol, T., Juluru, K., & Luna, A. (2022). Natural Language Processing in Radiology: Update on Clinical Applications. Journal of the American College of Radiology.

[25] Jagannatha, A., Liu, F., Liu, W., & Yu, H. (2019). Overview of the first natural language processing challenge for extracting medication, indication, and adverse drug events from electronic health record notes (MADE 1.0). Drug safety, 42, 99-111.

[26] Huang, K., Altosaar, J., & Ranganath, R. (2019). Clinicalbert: Modeling clinical notes and predicting hospital readmission. arXiv preprint arXiv:1904.05342.

[27] Jagannatha, A., Liu, F., Liu, W., & Yu, H. (2019). Overview of the first natural language processing challenge for extracting medication, indication, and adverse drug events from electronic health record notes (MADE 1.0). Drug safety, 42, 99-111.

[28] Polikar, R. (2006). Ensemble based systems in decision making. IEEE Circuits and systems magazine, 6(3), 21-45.

[29] Xia, J., Pan, S., Zhu, M., Cai, G., Yan, M., Su, Q., ... & Ning, G. (2019). A long short-term memory ensemble approach for improving the outcome prediction in intensive care unit. Computational and mathematical methods in medicine, 2019.

[30] Baccouche, A., Garcia-Zapirain, B., Castillo Olea, C., & Elmaghraby, A. (2020). Ensemble deep learning models for heart disease classification: A case study from Mexico. Information, 11(4), 207.

[31] Liu, Z., Tang, B., Wang, X., & Chen, Q. (2017). De-identification of clinical notes via recurrent neural network and conditional random field. Journal of biomedical informatics, 75, S34-S42.

[32] Long, B., Tan, F., & Newman, M. (2023). Forecasting the Monkeypox Outbreak Using ARIMA, Prophet, NeuralProphet, and LSTM Models in the United States. Forecasting, 5(1), 127-137.

[33] Géron, A. (2022). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. " O'Reilly Media, Inc.".

[34] Okuno, Y., & Mori, S. (2012, December). An ensemble model of word-based and character-based models for Japanese and Chinese input method. In Proceedings of the Second Workshop on Advances in Text Input Methods (pp. 15-28).

[35] Mimura, M. (2022). Evaluation of printable character-based malicious PE file-detection method. Internet of Things, 19, 100521.

[36] Ramirez-Orta, J. A., Xamena, E., Maguitman, A., Milios, E., & Soto, A. J. (2022, June). Post-ocr document correction with large ensembles of character sequence-to-sequence models. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 36, No. 10, pp. 11192-11199).

[37] He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. arXiv preprint arXiv:2006.03654.

[38] Umer, M., Imtiaz, Z., Ullah, S., Mehmood, A., Choi, G. S., & On, B. W. (2020). Fake news stance detection using deep learning architecture (CNN-LSTM). IEEE Access, 8, 156695-156706.

[39] He, P., Gao, J., & Chen, W. (2021). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. arXiv preprint arXiv:2111.09543.

[40] Hicks, S. A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M. A., Halvorsen, P., & Parasa, S. (2022). On evaluation metrics for medical applications of artificial intelligence. Scientific Reports, 12(1), 5979.