StellEllaStars at MEDIQA-Sum 2023: Exploring Transformer-Based models for Dialogue2Topic Classification

Chi-Yun Chang[1,*], Jiaqi Li[1], Shivangi Kumar[1] and V.G. Vinod Vydiswaran[1,*]

[1]*University of Michigan, Ann Arbor, United States*

Abstract

Topic labeling of clinical notes is an essential task in Clinical Natural Language Processing. In this study, we explore deep neural network-based classification approaches to assign predetermined topic categories to conversation snippets between doctors and patients. Our proposed models[1] include transformer-based models like BERT, traditional machine learning models like Logistic Regression, and deep learning models such as Continuous Bag-of-Words (CBoW). To address the issue of the lack of sufficient data due to a small training dataset, we incorporate oversampling techniques into our methods. Our proposed approaches perform fairly well on the MEDIQA-Sum 2023 shared task on classifying doctor-patient dialogues into topic categories, with our best run which leverages the ClinicalBERT model achieving an accuracy score of 0.765 on the MEDIQA-Sum test set.

Keywords

Natural language processing, Text classification, Deep learning, Clinical notes

1. Introduction

Medical conversations between doctors and patients contain critical clinical information. Developing natural language processing (NLP) approaches that understand these interactions could facilitate numerous applications, such as clinical decision support and patient-centered health information systems. However, accurately identifying the conversation topics is challenging due to the complex and domain-specific nature of medical dialogues. As part of our participation in the Dialogue2Topic Classification subtask (Subtask A) of MEDIQA-Sum 2023 [1], we focused on exploring deep neural network models to identify the most appropriate clinical note section headers for the dialogue snippets, classifying dialogue snippets into one of twenty predefined section labels such as Assessment, Diagnosis, Exam, Medications, and Past Medical History.

Previous studies have explored various machine learning models, especially deep neural-network based models, for similar classification tasks. Li et al. [2] focused on automatically classifying different sections within clinical notes using a supervised Hidden Markov Model (HMM) based on section content and structure. The evaluation on clinical notes from MIMIC-II

[1] The code for the described methods can be found at https://github.com/jiaqili0803/MEDIQA-Sum-2023-StellEllaStars

CLEF 2023 – Conference and Labs of the Evaluation Forum, September 18-21, 2023, Thessaloniki, Greece

*Corresponding author.

stellacy@umich.edu (C. Chang); ellali@umich.edu (J. Li); iamsk@umich.edu (S. Kumar); vgvinodv@umich.edu (V.G. V. Vydiswaran)

CEUR Workshop Proceedings (CEUR-WS.org)

showed promising results outperforming the other methods with 93% accuracy for identifying individual section headers and 70% accuracy for identifying all section headers in a note. Although this work made significant contributions in benchmarking "traditional" machine learning approaches, it highlighted the generalizability concerns of training models on a relatively small dataset derived over ICU notes from the MIMIC-II.

In recent years, the focus of evaluation has been on deep neural network models. Nair et al. [3] identified different sections within clinical notes using transfer learning techniques to leverage pre-trained language models. The proposed methodology involved fine-tuning a pre-trained transformer-based language model, Bidirectional Encoder Representations from Transformers (BERT). Experiments were conducted on two publicly available clinical note datasets, MIMIC-III and i2b2-2010, and compared BERT models against rule-based and "traditional" machine learning models. BERT models achieved the best accuracy of 87% on this task. Similarly, Qing et al. [4] presented a novel neural network-based approach focused on capturing the semantic representations and contextual data included in medical text. Their approach combined a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network. The LSTM network collected long-range dependencies and contextual data, whereas CNN extracted local features from the input medical text. The authors tested their methodology using a medical text dataset with a variety of categories, including disease diagnosis and treatment, and achieved a relative improvement of 12.47% on one of the datasets with CNN over other approaches. Aiming to make use of a larger amount of data, Wang et al. [5] proposed a clinical text classification model that combines weak supervision with deep representation. Their methodology fed weakly labeled data in to a Hierarchical Attention Network (HAN) that captured the hierarchical structure and semantic representations of clinical writing. The evaluation on a large collection of clinical notes from Mayo Clinic showed that the weakly supervised HAN model was able to achieve competitive performance in clinical text categorization.

In other related work, Schloss and Konam [6] proposed medical dialogue modeling to automate the creation of clinical notes by using automatic speech recognition (ASR) to transcribe medical conversations and then implementing deep learning architectures for SOAP (Subjective, Objective, Assessment, Plan) section classification. This study shows the potential for streamlining clinical documentation procedures in healthcare settings and further underlines the significance of context-aware dialogue systems.

Building on these prior work, we investigated numerous machine learning models for Dialogue2Topic classification. Our ensemble of models included ClinicalBERT, a transformer-based model pre-trained on clinical text, traditional machine learning models, and deep learning models such as CNN, LSTM, and Continuous Bag-of-Words. We addressed the inherent class imbalance in the training dataset by oversampling the minority class to improve performance.

2. Methods

2.1. Data preprocessing

As part of the shared task, the MEDIQA-Sum task organizers released two datasets, a training dataset consisting of 1,201 rows and a validation dataset consisting of 100 rows [1]. Each data instance is composed of three columns: ID, section header (the “topic”), and the dialogue snippet.

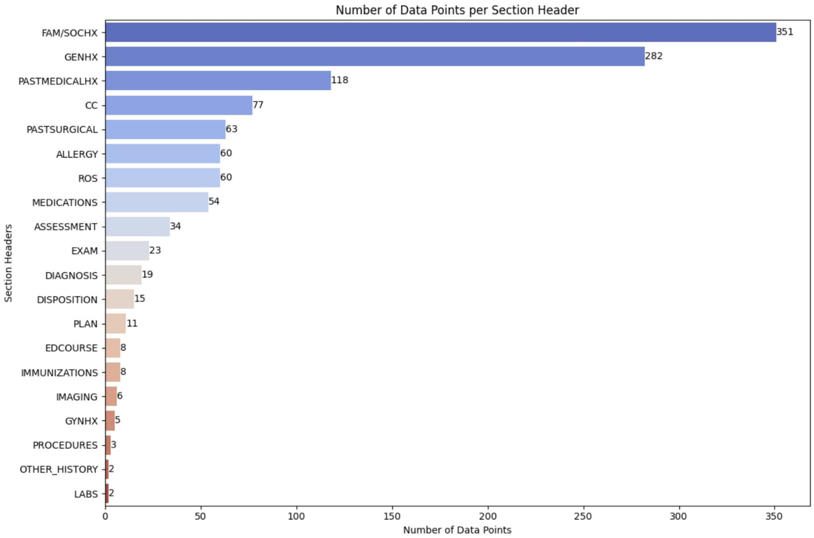

Figure 1: Training Data Distribution Highlighting the Skewed Label Data Distribution

As shown in Figure 1, the training dataset is imbalanced, with FAM/SOCHX and GENHX sections being the most frequent, with more than 250 instances each, while the smallest ten classes has less than 20 instances each. During the test phase, the organizers released a test set consisting of 200 instances with the section header column absent.

We took the following preprocessing steps to clean the dataset for the classification task:

1. **Removing punctuation, digits, and special characters**: We eliminated all non-alphabetic characters. This included punctuation marks such as periods, commas, semi-colons, question marks, and exclamation points, among others as listed next: "@", "#", "$", "%", "^", "&", "*", "(", ")", "-", "+", "=", "[", "]", "{", "}", "|", ":", "’", "”", "<", ">", "/", and "?". Numerical digits ranging from zero to nine were also excluded.
2. **Removing dialogue markers**: Specific dialogue markers such as "Doctor:", "Patient:", and timestamps were removed. This step was taken to ensure the data’s uniformity and to prevent these recurring strings from skewing the analysis.
3. **Removing stop words**: Stop words, which typically include commonly used English words, such as "and", "the", "is", etc., were removed since they do not add semantic value to the text. We utilized the stop words list from the *nltk.corpus* package.
4. **Converting text to lowercase**: All text data was converted to lowercase. This step is essential to guarantee consistency and to prevent different token representations for the same word based on case differences, as "Allergy" and "allergy" would be treated as distinct entities without this conversion.
5. **Tokenizing text**: The text data was split into individual words or "tokens" by identifying word boundaries such as spaces, commas and full stops between adjacent words.
6. **Lemmatization**: Finally, words were reduced to their base or root form (e.g., "cancers" to "cancer") using the WordNetLemmatizer from the *nltk.stem* package.

Table 1
Hand-Coded Keywords

Section Header	Keywords
ALLERGY	allergy, allergies, allergic, reaction, hypersensitivity, anaphylaxis
ASSESSMENT	assessment, evaluate, evaluation, examine, analysis, appraisal
CC	chief complaint, complaint, symptom, presenting, concern, Issue
DIAGNOSIS	diagnosis, diagnose, condition, finding, disease, disorder
DISPOSITION	disposition, discharge, admit, transfer, status, outcome
EDCOURSE	ed course, emergency department, treatment, management, care, therapy
EXAM	physical exam, examination, inspection, auscultation, palpation, assessment
FAM/SOCHX	family history, social history, lifestyle, habits, relationships, environment
GENHX	general history, medical history, background, chronic illness, disease
GYNHX	gynecological history, gynecology, reproductive, menstruation, pregnancy, contraception
IMAGING	imaging, x-ray, ultrasound, CT, MRI, radiology, scan, radiograph
IMMUNIZATIONS	Immunizations, vaccinations, vaccine, shot, immunity, inoculation
LABS	labs, laboratory, blood work, test results, analysis, diagnostics
MEDICATIONS	medications, drugs, prescriptions, meds, pharmacotherapy, pharmaceuticals
OTHER HISTORY	other history, additional history, miscellaneous, unrelated, extra, supplementary
PASTMEDICALHX	past medical history, pmh, previous conditions, comorbidities, illnesses, disorders
PASTSURGICAL	past surgical, surgical history, operations, procedures, interventions, surgeries
PLAN	plan, treatment plan, interventions, actions, approach, strategy
PROCEDURES	procedures, interventions, techniques, operations, methods, practices
ROS	review of systems, ros, systematic clinical review, organ systems, body systems

2.2. Machine learning models for text classification

We experimented with several machine learning models frequently employed for classification tasks. Prior to model training, we ran a feature extraction step by utilizing TfidfVectorizer from the Natural Language Toolkit (NLTK) package [7], removing stop words, and generating unigram and bigram features with the maximum document frequency parameter max_df set to 0.9. This resulted in 35,290 features. Next, we ran Grid Search [8] to tune the parameters for the following machine learning models: Support Vector Machines (SVM), Multinomial Naive Bayes, Logistic Regression, Multilayer Perceptron (MLP), Decision Trees, and Random Forest models [9]. We set the Logistic Regression model to run with L2 regularization, with the regularization strength parameter C set to 1.0. We undertook the following measures to further improve our models:

Oversampling : To address the inherent imbalance in our dataset, we oversampled our data using the RandomOverSampler method [10] to generate a more balanced dataset. RandomOverSampler duplicates random instances of the minority class until its count equals that of the majority class.

Creation of Hand-Coded Keyword Sets : We enhanced our features by creating a hand-coded keyword set, shown in Table 1. If a dialogue contained these keywords, we would add a binary feature to the corresponding section header to boost prediction accuracy.

Feature Selection : We also explored mutual information-based feature selection approaches, but did not find it to be highly significant for improving model performance.

2.3. Deep learning models for text classification

In addition to the machine learning models described above, we explored several deep learning models using Keras [11]. We preprocessed the data as follows: we tokenized the text into sequences of indices representing the 20,000 most frequent words, performed padding so that all sequences have the same length, and represented the output classes as a one-hot encoding. The average sequence length was 257, and we set the maximum sequence length of 512.

Continuous Bag-of-Words (CBoW) : A Continuous Bag-of-Words (CBoW) model [12] averages the word embeddings of all words in a sentence to capture the semantics and context of the collection of words to make predictions. We inputted embedded text sequences into a GlobalAveragePooling1D layer, then generated predictions through a Dense layer using softmax as the activation type. We compiled the CBoW model with the loss set to be categorical_crossentropy with adam optimizer, and accuracy metric as the optimization criterion. The model was fitted with a validation split size of 0.1, batch size of 128, and 150 epochs.

Long Short-Term Memory (LSTM) : Next, we initiated a Long Short-Term Memory (LSTM) model [13], which is a Recurrent Neural Network shown to handle sequential data well. LSTM models capture dependencies between words in a sentence, allowing the model to understand the context in which a word appears. These features make it suitable for text classification tasks. Our model consists of an LSTM layer with an output dimension of 128, a dropout rate of 0.2, a recurrent state dropout rate of 0.2, and a Dense layer using softmax as the activation type. We compiled the LSTM model with the loss set to be categorical_crossentropy with adam optimizer, and accuracy metric as the optimization criterion. The model was fitted with a validation split size of 0.1, batch size of 128, and 15 epochs.

Bidirectional Long Short-Term Memory (BiLSTM) : We also investigated the Bidirectional LSTM model [14], which is similar to LSTM, but consists of two layers taking input in both the forward and backward directions. Given its two layers, it is good at capturing both past and future context for each word in a sentence and also long-term dependencies within text. Our BiLSTM model consists of a SpatialDropout1D layer with a dropout rate of 0.2, a Bidirectional layer, a Conv1D layer with a filter size of 64, a GlobalAveragePooling1D layer, a GlobalMaxPooling1D layer, and a Dense layer using sigmoid as the activation type [15]. Similar to the CBoW and LSTM models, the BiLSTM model was compiled with the loss set to be categorical_crossentropy with adam optimizer and accuracy metric as the optimization criterion, and fitted over a validation split size of 0.1, batch size of 128, and 40 epochs.

Convolutional Neural Network (CNN) - Long Short-Term Memory (LSTM) : Next, we combined the Convolutional Neural Network (CNN) and LSTM models, similar to the one explored by Qing et al. [4] for text classification. CNN is set to extract higher levels of word representations, which can then be inputted into LSTM to obtain sentence representations.

We used 1D Convolutional layers to extract local features such as n-grams or character-level features from the input text sequence, and then added pooling layers for dimension reduction. To prevent overfitting, we also added dropout layers for regularization. Our CNN-LSTM models consists of the following layers stacked in this order: a Conv1D layer with a filter size of 64 and a window size of 5, a MaxPooling1D layer with a pool size of 5, a Dropout layer with a dropout rate of 0.2, a Conv1D layer with a filter size of 64 and a window size of 5, a MaxPooling1D layer with a pool size of 5, a Dropout layer with a dropout rate of 0.2, an LSTM layer with an output dimension of 64, and a Dense layer using softmax as the activation type. Similar to the other neural network models above, we compiled the CNN-LSTM model with the loss set to be categorical_crossentropy with adam optimizer and accuracy metric as the optimization criterion, and fitted over the validation split size of 0.1, batch size of 128, and 40 epochs.

While training these deep learning models, we also used some functions in the Callbacks module of Keras to improve overall training performance and efficiency. We used EarlyStopping to drop out from epochs when the validation loss ceases to improve, so as to reduce overfitting. We used ModelCheckpoint to save optimal model weights during the training process.

ClinicalBERT: Last, we instantiated a ClinicalBERT model [16], a variant of the BERT model trained on clinical text, to classify clinical note sections. The ClinicalBERT model's design aims to capture the intricacies of medical language, offering domain-specific knowledge and contextual understanding to enhance performance in healthcare-related natural language processing tasks. Since traditional feature extractors like TfidfVectorizer are not necessary for BERT models, we tokenized the input data and set up attention masks and label encoding. We set the number of epochs at 3, the training and evaluation batch sizes at 8, and the learning rate of 2e-5 during training to dictate the degree of adjustment to the model weights in response to each estimated error update. To prevent overfitting, we implemented a weight decay of 0.01, a regularization technique that adds a small penalty proportional to the L2-norm of the weights to the loss function [17].

2.4. Selecting models for test submission

We chose the three best models based on their performances on the validation data – Logistic Regression, CBoW, and ClinicalBERT models. We chose CBoW over CNN-LSTM models because of higher consistency in results across classes. All models were implemented with the oversampling technique.

3. Results

3.1. Performance on Validation Data

Our baseline model, an untuned Support Vector Classifier (SVC) performed at an accuracy of 0.56 on the validation data. The deployment of the RandomOverSampler method resulted in a considerable enhancement of our overall performance metrics. This was clearly reflected in the ClinicalBERT model, whose accuracy improved from 0.66 to 0.75 as a result of the implementation of RandomOverSampler. More importantly, we observed a substantial increase in the prediction accuracy for the underrepresented classes within our data. For instance, the F1-score for the section header EDCOURSE improved from 0.00 to 0.50. This improvement indicates that our approach provided for a more equitable representation, thereby improving the comprehensiveness and reliability of our model's predictive capabilities.

Among other measures we took to enhance the performances of our models, the creation of hand-coded keyword sets did not result in improvement in model performances, so we discontinued this approach. We conclude that more work is needed to incorporate domain knowledge to improve these hand-coded keywords or to capture them directly in the model.

The best performing model is ClinicalBERT combined with the oversampling technique, which achieves an accuracy of 0.75. The next-best performing model is Logistic Regression, with an accuracy of 0.71. Among the deep learning models, CBoW trained with un-resampled data obtained an accuracy of 0.66, while CBoW trained with resampled data had a slightly improved performance of 0.68. LSTM models and CNN-LSTM models performed slightly better with un-resampled data (0.62 and 0.68, respectively) compared to with resampled data (0.60 and 0.64, respectively); while BiLSTM's performance was better with resampled data (0.64) than with un-resampled data (0.61).

3.2. Qualitative analysis of best performing models

Logistic Regression : This model achieved an accuracy of 0.71 and a weighted average F1-score of 0.66, suggesting balanced performance in precision and recall. High F1-scores, with near perfect precision and recall were observed for FAM/SOCHX and PASTSURGICAL, while perfect scores were observed for DIAGNOSIS, DISPOSITION, and IMMUNIZATIONS. Conversely, precision and recall were zero for ASSESSMENT and EDCOURSE. The class EXAM was over-predicted, thus generating more false positives.

CBoW : The overall prediction accuracy of the model is 0.68, with perfect classification results for IMMUNICATIONS and near-perfect F1-scores for FAM/SOCHX, PASTSURGICAL and ROS. On the other hand, the model showed poor performance with zero precision, recall and F1 scores for ASSESSMENT, EDCOURSE, IMAGING, OTHER_HISTORY, and PROCEDURES.

Clinical BERT : This model achieved the best of the three submitted runs, with an overall accuracy of 0.75 and a weighted average F1-score of 0.734. High F1-scores were achieved for ALLERGY and FAM/SOCHX classes, while F1-scores for ASSESSMENT and DISPOSITION classes were low, and zero for DIAGNOSIS and GYNHX classes. Classes such as EDCOURSE and PLAN demonstrated high precision, but low recall, indicating that while the model does not tend to misclassify instances, it is prone to omitting instances from these classes.

All three models benefited from oversampling techniques and had improved performance on infrequent classes. Even so, the models performed significantly better on larger classes and less so on less frequent ones.

3.3. Performance on Test Data

As summarized in the last column of Table 2, among the three runs we submitted, the ClinicalBERT model performed the best, with an accuracy of 0.765 on the test data. CBoW and Logistic Regression models achieved an accuracy of 0.695 and 0.675, respectively, on the test

Table 2
Best validation performance of all models, and performance of the selected models on test set

Model	Accuracy with unresampled data	Accuracy with resampled data	Best accuracy	Test accuracy
Baseline Model - untuned SVC	0.54	0.56	0.56	
SVC	0.65	0.65	0.65	
Multinomial NB	0.56	0.69	0.69	
Logistic Regression	0.66	0.71	**0.71**	0.675
MLP	0.41	0.65	0.65	
Decision Tree	0.52	0.58	0.58	
Random Forest	0.64	0.67	0.67	
CBoW	0.66	0.68	**0.68**	0.695
LSTM	0.62	0.60	0.62	
BiLSTM	0.61	0.64	0.64	
CNN - LSTM	0.68	0.64	0.68	
Clinical BERT	0.66	0.75	**0.75**	**0.765**

data. We think that the higher performance of the ClinicalBERT model can be attributed to its pre-training on clinical text data, which allows for transfer of domain-specific knowledge and contextual understanding learned from a more extensive corpus of medical language, resulting in enhanced performance on the target task despite limited training data [18]. CBoW models performed similarly over the test set (accuracy of 0.695) as over the validation set (accuracy of 0.68), mirroring the stability of results across classes we noticed on validation data. When compared to the deep learning models, the test performance of the Logistic Regression model was significantly poorer. We think that some of the poor performance can be attributed to potentially missing informative features. While it is known that feature selection plays a critical part in machine learning model training [19], our attempt to optimize our selected feature set for the Logistic Regression model, including the hand-coded features, did not appear to work well.

4. Conclusion and Limitations

For the task of Dialogue2Topic classification, our experiments showed the ClinicalBERT model to perform the best among the models tested, with our implementation achieving the best performance of 0.765 on the MEDIQA-Sum 2023 test set. One of the main limitations of our approach was the limited training dataset size available to train traditional machine learning models. While we addressed some of these concerns through oversampling, there were still some classes with only one instance that did not benefit much from the approach. We also encountered the issue of overfitting, which possibly also stems from the limited dataset size. We addressed this problem partially through feature selection and using EarlyStopping [20]. We plan to continue exploring alternate approaches including transfer learning [21], to incorporate domain knowledge into models through better feature and domain representation.

References

[1] W. Yim, A. Ben Abacha, N. Snider, G. Adams, M. Yetisgen, Overview of the mediqa-sum task at imageclef 2023: Summarization and classification of doctor-patient conversations, in: CLEF 2023 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, Thessaloniki, Greece, 2023.

[2] Y. Li, S. Lipsky Gorman, N. Elhadad, Section classification in clinical notes using supervised hidden markov model, in: Proceedings of the 1st ACM International Health Informatics Symposium, 2010, pp. 744–750.

[3] N. Nair, S. Narayanan, P. Achan, K. P. Soman, Clinical note section identification using transfer learning, in: Proceedings of Sixth International Congress on Information and Communication Technology: ICICT 2021, London, Volume 1, Springer Singapore, Singapore, 2021, pp. 533–542.

[4] L. Qing, W. Linhong, D. Xuehai, A novel neural network-based method for medical text classification, Future Internet 11 (2019) 255.

[5] Y. Wang, S. Sohn, S. Liu, F. Shen, L. Wang, E. J. Atkinson, ..., H. Liu, A clinical text classification paradigm using weak supervision and deep representation, BMC Medical Informatics and Decision Making 19 (2019) 1–13.

[6] B. Schloss, S. Konam, Towards an automated SOAP note: Classifying utterances from medical conversations, in: Machine Learning for Healthcare Conference, PMLR, 2020, pp. 610–631.

[7] S. Bird, E. Klein, E. Loper, Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, "O'Reilly Media, Inc.", 2009.

[8] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, Journal of Machine Learning Research 13 (2012) 281–305.

[9] D. Jurafsky, J. H. Martin, Speech and language processing, Prentice Hall series in artificial intelligence, 2. ed., [pearson international edition] ed., Prentice Hall, Pearson Education International, 2009.

[10] H. Han, W. Y. Wang, B. H. Mao, Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning, in: Advances in Intelligent Computing: International Conference on Intelligent Computing, ICIC 2005, volume 1, Springer Berlin Heidelberg, 2005, pp. 878–887.

[11] Chollet, F. and others, Keras, https://keras.io/, 2015.

[12] F. Horn, Context encoders as a simple but powerful extension of word2vec, in: Proceedings of the 2nd Workshop on Representation Learning for NLP, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 10–14. URL: https://aclanthology.org/W17-2602. doi:10.18653/v1/W17-2602.

[13] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, J. Schmidhuber, LSTM: A search space odyssey, IEEE Transactions on Neural Networks and Learning Systems 28 (2016) 2222–2232.

[14] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing 45 (1997) 2673–2681.

[15] Y. Zhang, B. Wallace, A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification, in: Proceedings of the Eighth Inter-

national Joint Conference on Natural Language Processing (Volume 1: Long Papers), Asian Federation of Natural Language Processing, Taipei, Taiwan, 2017, pp. 253–263. URL: https://aclanthology.org/I17-1026.

[16] E. Alsentzer, ClinicalBERT - Bio + Clinical BERT model, https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT, 2023.

[17] H. Wang, C. Qin, Y. Zhang, Y. Fu, Neural pruning via growing regularization, in: 9th International Conference on Learning Representations, ICLR 2021, 2021, pp. 1–16. URL: https://openreview.net/forum?id=o966_Is_nPA.

[18] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, ..., J. Dean, Scalable and accurate deep learning with electronic health records, NPJ Digital Medicine 1 (2018) 18.

[19] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157–1182.

[20] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research 15 (2014) 1929–1958.

[21] K. Weiss, T. M. Khoshgoftaar, D. Wang, A survey of transfer learning, Journal of Big Data 3 (2016) 1–40.