MedJEx: A Medical Jargon Extraction Model with Wiki's Hyperlink Span and Contextualized Masked Language Model Score

Sunjae Kwon[1], **Zonghai Yao**[1], **Harmon S. Jordan**[2],
David A. Levy[3], **Brian Corner**[2], **Hong Yu**[1,3,4,5]
[1]UMass Amherst, [2]Health Research Consultant,
[3]UMass Lowell, [4]UMass Medical School, [5]U.S. Department of Veterans Affairs
sunjaekwon@umass.edu, zonghaiyao@umass.edu, harmon.s.jordan@gmail.com,
david_levy@uml.edu, brian.corner@umassmed.edu, hong_yu@uml.edu

Abstract

This paper proposes a new natural language processing (NLP) application for identifying medical jargon terms potentially difficult for patients to comprehend from electronic health record (EHR) notes. We first present a novel and publicly available dataset with expert-annotated medical jargon terms from 18K+ EHR note sentences ($MedJ$). Then, we introduce a novel medical jargon extraction ($MedJEx$) model which has been shown to outperform existing state-of-the-art NLP models. First, MedJEx improved the overall performance when it was trained on an auxiliary Wikipedia hyperlink span dataset, where hyperlink spans provide additional Wikipedia articles to explain the spans (or terms), and then fine-tuned on the annotated MedJ data. Secondly, we found that a contextualized masked language model score was beneficial for detecting domain-specific unfamiliar jargon terms. Moreover, our results show that training on the auxiliary Wikipedia hyperlink span datasets improved six out of eight biomedical named entity recognition benchmark datasets. MedJEx is publicly available [1].

1 Introduction

Allowing patients to access their electronic health records (EHRs) represents a new and personalized communication channel that has the potential to improve patient involvement in care and assist communication between physicians, patients, and other healthcare providers (Baldry et al., 1986; Schillinger et al., 2009). However, studies showed that patients do not understand medical jargon in their EHR notes (Chen et al., 2018).

To improve patients' EHR note comprehension, it is important to identify medical jargon terms that are difficult for patients to understand. Unlike the traditional concept identification or named entity recognition (NER) tasks, where the tasks mainly center on semantic salient entities, detecting such medical jargon terms takes into consideration the perspective of user comprehension. Traditional NER approaches such as using comprehensive clinical terminological resources (e.g., the Unified Medical Language System (UMLS) (Bodenreider, 2004)) would identify terms such as "water" and "fat", which are not considered difficult for patients to comprehend. Meanwhile, using term frequency (TF) as the proxy for medical jargon term identification will miss outliers such as "shock," which is a term frequently used in the open domain with its common sense: "a sudden upsetting or surprising event or experience." However, EHR notes incorporate its uncommon sense: "a medical condition caused by severe injury, pain, loss of blood, or fear that slows down the flow of blood." (Shock, 2022). Thus, "shock" should be identified as a jargon term from EHR notes since it would be difficult for patients to comprehend, even though its TF is high. In this study, we propose a natural language processing (NLP) system that can identify such outlier jargon from EHR notes through a novel method for homonym resolution.

We first expert-annotated de-identified EHR note sentences for medical jargon terms judged to be difficult to comprehend. This resulted in the Medical Jargon Extraction for Improving EHR Text Comprehension (MedJ) dataset, which comprises 18,178 sentences and 95,393 medical jargon terms. We then present a neural network-based medical jargon extraction (MedJEx) model to identify the jargon terms.

To ameliorate the limited training-size issue, we propose a novel transfer learning-based framework (Tan et al., 2018) utilizing auxiliary Wikipedia (Wiki) hyperlink span datasets (WikiHyperlink), where the span terms link to different Wiki articles (Mihalcea and Csomai, 2007). Although medical jargon extraction and WikiHyperlink recognition

[1] https://github.com/MozziTasteBitter/MedJEx

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11733–11751
December 7-11, 2022 ©2022 Association for Computational Linguistics

seem to be two different applications, they share similarities. The role of hyperlinks is to help a reader to understand an Wiki article. Thus, "difficult to understand" concepts in the Wiki article may be more likely to have hyperlinks. Therefore, we hypothesize that large-scale hyperlink span information from Wiki can be advantageous for our models of medical jargon extraction. Our results show that models trained on WikiHyperlink span datasets indeed substantially improved the performance of MedJEx. Moreover, we also found that such auxiliary learning improved six out of the eight benchmark datasets of biomedical NER tasks.

To detect outlier homonymous terms such as "shock", we deployed an approach inspired by masking probing (Petroni et al., 2019), a method for evaluating linguistic knowledge of large-scale pre-trained language models (PLMs). Meister et al. (2022) suggests PLMs are beneficial for predicting the reading time, with longer reading time indicates difficult for indicating difficulty in understanding. In our work, we propose a contextualized masked language model (MLM) score feature to tackle the homonym challenge. Note that models will recognize the sense of a word or phrase using contextual information. Since PLMs calculate the probability of masked words in consideration of context, we hypothesize that PLMs trained in the open-domain corpus would predict poorly masked medical jargon if senses are distributed differently between the open domain and clinical domain corpora.

We conducted experiments on four state-of-the-art PLMs, namely BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), BioClinicalBERT (Alsentzer et al., 2019b) and BioBERT (Lee et al., 2020). Experimental results show that when both of the methods are combined, the medical jargon extraction performance is improved by 2.44%p in BERT, 2.42%p in RoBERTa, 1.56%p in BioClinicalBERT, and 1.19%p in BioBERT.

Our contributions are as follows:

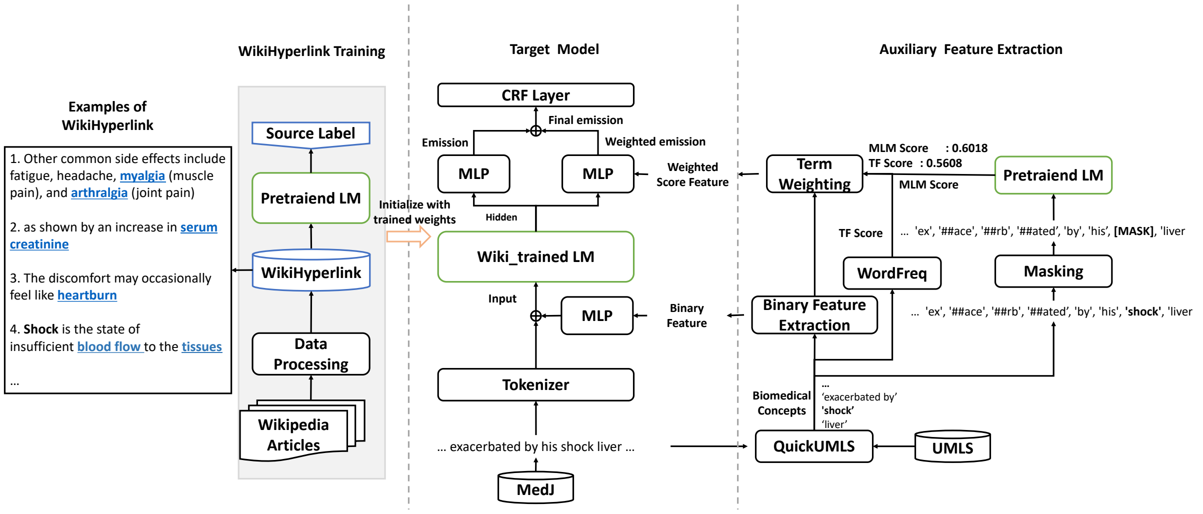
- We **propose a novel NLP task** for identifying medical jargon terms potentially difficult for patients to comprehend from EHR notes.
- We construct **MedJ**, an expert-curated 18K+ sentence dataset for the MedJEx task.
- We introduce **MedJEx**, a medical jargon extraction model. Herein, MedJEx was first trained with the auxiliary WikiHyperlink span dataset before being fine-tuned on the MedJ dataset. It uses MLM score feature for homonym resolution.
- The experimental results show that training on the Wiki's hyperlink span datasets consistently improved the performance of not only MedJ but also six out of eight BioNER benchmarks. In addition, our qualitative analyses show that the MLM score can complement the TF score for detecting the outlier jargon terms.

2 Related Work

In principle, MedJEx is related to text simplification (Kandula et al., 2010). None of the previous work (Abrahamsson et al., 2014; Qenam et al., 2017; Nassar et al., 2019) identified terms that important for comprehension.

On the other hand, MedJEx is relevant to BioNER, a task for identifying biomedical named entities such as $disease$, $drug$, and $symptom$ from medical text. There are several benchmark corpora, including i2b2 2010 (Patrick and Li, 2010), ShARe/CLEF 2013 (Zuccon et al., 2013), and MADE (Jagannatha et al., 2019), all of which were developed solely based on clinical importance. In contrast, $MedJ$ is patient-centered, taking into consideration of patients' comprehension. Identifying BioNER from medical documents has been an active area of research. Earlier work such as the MetaMap (Aronson, 2001), used linguistic patterns, either manually constructed or learned semi-automatically, to map free text to external knowledge resources such as UMLS (Lindberg et al., 1993). The benchmark corpora have promoted supervised machine learning approaches including conditional random fields and deep learning approaches (Jagannatha et al., 2019).

Key phrase extraction in the medical domain is another related task. It identifies important phrases or clauses that represent topics (Hulth, 2003). In previous studies, key phrases were extracted using features such as TF, word stickiness, and word centrality (Saputra et al., 2018). Chen and Yu (2017) proposed an unsupervised learning based method to elicit important medical terms from EHR notes using MetaMap (Demner-Fushman et al., 2017) and various weighting features such as TextRank (Mihalcea and Tarau, 2004) and term familiarity score (Zeng-Treitler et al., 2007). In another work, Chen et al. (2017) proposed an adaptive distant su-



1. Other common side effects include fatigue, headache, myalgia (muscle pain), and arthralgia (joint pain)

2. as shown by an increase in serum creatinine

3. The discomfort may occasionally feel like heartburn

4. **Shock** is the state of insufficient blood flow to the tissues

...

Figure 1: This figure demonstrates the overall architecture of MedJEx. There are three components in MedJEx: 1) WikiHyperlink training, 2) auxiliary feature extraction and 3) target model. First, in WikiHyperlink training, we extract hyperlink spans from Wikipedia articles. The examples shows that hyperlink spans (blue colored) represent medical jargons, and ignore easier medical terms such as "fatigue" and "headache". Then, the pretrained language model (LM) is trained with WikiHyperlink. In auxiliary feature extraction, we can see that MLM score of medical jargon "shock" shows relatively high TF and MLM scores, indicating that the MLM score can help detect the medical jargon. Finally, the weight parameters of Wiki-trained LM in the target model are initialized with trained parameters of pretrained LM of WikiHyperlink training. Then, the model is finetuned with MedJ.

pervision based medical term extraction approach that utilizes consumer health vocabulary (Zeng and Tse, 2006) and a heuristic rule to distantly label medical jargon training datasets. A key phrase extraction method using a large-scale pretrained model is being actively studied (Soundarajan et al., 2021).

Unlike the previous BioNER or key phrase identification applications, identifying medical jargon terms is important for patients' comprehension of their EHR notes and represents a novel NLP application. However, not all medical entities are unfamiliar to patients. The brute force approach of capturing every medical entity, the approaches of existing BioNER and key phrase identification applications, may bring about confusion to patients. On the other hand, undetected medical jargon terms will reduce patients' EHR note comprehension. In this paper, we propose MedJEx, a novel application that identifies medical jargon terms important for patients' comprehension. Once jargon terms are identified, interventions such as linking the jargon terms to lay definitions can help improve comprehension.

3 Dataset Construction

This work has two different datasets: 1) *MedJ* for medical jargon extraction and 2) *Wiki's hyperlink span (WikiHyperlink)* dataset for transfer learning.

3.1 MedJ

3.1.1 Data Collection

The source of the dataset is a collection of publicly available deidentified EHR notes from hospitals affiliated with the University of Pittsburg Medical Center. Herein, 18,178 sentences were randomly sampled and domain-experts then annotated the sentences for medical jargon [2].

3.1.2 Data Annotation

Domain-experts read each sentence and identified as medical jargon terms that would be considered difficult to comprehend for anyone no greater than a 7th grade education[3]. Overall, 96,479 medical jargon terms have been annotated by complying with the following annotation guideline.

Annotation Guideline The dataset was annotated for medical jargon by six domain experts from medicine, nursing, biostatistics, biochemistry, and biomedical literature curation [4]. Herein, the anno-

[2] Using these data requires a license agreement.

[3] The rule of thumb is that if a candidate term has a lay definition comprehensible to a 4-7th grader as judged by Flesch-Kincaid Grade Level (Solnyshkina et al., 2017), the candidate term is included as a jargon term.

[4] The annotator agreement scores can be found in Appendix A.1.

tators applied the following rules for identifying what was jargon:

Rule 1. Medical terms that would **not be recognized by about 4 to 7th graders**, or that **have a different meaning in the medical context than in the lay context (homonym)** were labeled. For example:

- accommodate: When the eye changes focus from far to near.
- antagonize: A drug or substance that stops the action or effect of another substance.
- resident: A doctor who has finished medical school and is receiving more training.
- formed: Stool that is solid.

Rule 2. Terms that are not strictly medical, but are **frequently used in medicine**. For example:

- "aberrant", "acute", "ammonia", "tender", "intact", "negative", "evidence"

Rule 3. When jargon words are **commonly used together, or together they mean something distinct or are difficult to quickly understand from the individual parts** were labeled. For example:

- vascular surgery: Medical specialty that performs surgery on blood vessels.
- airway protection: Inserting a tube into the windpipe to keep it wide open and prevent vomit or other material from getting into the lungs.
- posterior capsule: The thin layer of tissue behind the lens of the eye. It can become cloudy and blur vision.
- right heart: The side of the heart that pumps blood from the body into the lungs.
- intracerebral hemorrhage: A stroke.

Rule 4. Terms whose **definitions are widely known** (e.g., by a 3rd grader) do NOT need to be labeled. For example:

- “muscle”, “heart”, “pain”, “rib”, “hospital”

Rule 4.1 When in doubt, label the term. For example:

- “colon”, “immune system”

3.1.3 Data Cleaning

First, we cleaned up overlapped (tumor suppressor *gene*, *gene* deletion) or nested (*vitamin D*, 25-hydroxy *vitamin D*) jargon. We chose the longest jargon terms among nested or overlapped jargon terms. For example, we chose "tumor suppressor gene" as a jargon term, not its nested term "tumor." In all, MedJ contains a total of 95,393 context-dependent jargon terms which we used as the gold standard for training and evaluation of the MedJEx model. The 95,393 jargon terms represent a total of 12,383 unique jargon terms.

3.2 WikiHyperlink

From a Wiki dump data[5], we first cleaned and elicited text by using Wikiextractor (Attardi, 2015). Then, we extracted hyperlink spans with the BeautifulSoup (Richardson, 2007) module. Wiki articles were split into sentences with the Natural Language Toolkit (Bird et al., 2009), then the sentences were split into tokens with the PLM tokenizer. Overall, WikiHyperlink contains more than 114M sentences, 13B words, and 99M hyperlink spans. Finally, the source data consists of the sequence input of the token and hyperlink labels represented in the standard BIOES format (Yang et al., 2018).

4 MedJEx Model

Figure 1 is an overview of MedJEx. First, we trained PLMs with WikiHyperlink (Wiki-trained). Then, the Wiki-trained model was transferred to the target model that we propose by initializing the target model with the weight parameters of the Wiki-trained model. Finally, we fine-tuned the target model with our expert-annotated dataset. Note that, since the pretrain corpora of PLMs used in this work include the Wiki corpus, we noticed that the performance change should derive from the added labels (hyperlink spans). Herein, we extracted UMLS concepts and used them as auxiliary features.

4.1 Wiki’s Hyperlink Span Prediction for Transfer Learning Framework

Although MedJ is a high-quality and a large scale expert-labeled dataset, deep learning models could improve performance with additional data. However, annotation is very expensive. Transfer learning is one of the effective ways to mitigate the

[5] https://dumps.wikimedia.org/enwiki/20211001/

challenge (Ruder, 2019; Mao, 2020). In this paper, we propose to utilize Wiki's articles and hyperlink span as source data. We assumed that hyperlink spans are similar to medical jargon: readers need to read the hyperlinked articles to understand the span concepts (Mihalcea and Csomai, 2007). Indeed, in the example sentence in Figure 1, we can see that some difficult biomedical concepts are hyperlinked, but easier concepts such as "fatigue" and "headache" were not linked. We also expect that this approach can also be generalized for BioNER tasks since hyperlinks are often associated with a biomedical concept.

Training We fine-tune a PLM with WikiHyperlinik by following the standard protocol for fine-tuning PLMs for sequence labeling tasks (Devlin et al., 2019). Herein, for a given N number of sentences, a PLM calculated the probability distribution for the C classes of each token in the sentence composed of S tokens. The model was trained to optimize cross-entropy (CE) loss of Eq. 1, where $y_{n,s,c}$ and $\hat{y}_{n,s,c}$ indicate the label and the model's output, respectively, for the s^{th} token's probability of the n^{th} sentence belonging to the class c respectively.

$$\mathcal{L}_{CE} = \frac{1}{NSC} \sum_{n=1}^{N} \sum_{s=1}^{S} \sum_{c=1}^{C} y_{n,s,c} \cdot \log \hat{y}_{n,s,c} \tag{1}$$

4.2 Neural Network-based Medical Jargon Extraction Model

Our jargon prediction model consists of the following two parts: 1) additional learning feature extraction, 2) target model. This section explains the additional feature extraction at first. Then, we describes the structure of the target model.

4.2.1 Auxiliary Feature Extraction

The UMLS concepts incorporate important clinical domain-specific jargon, including disease, surgery, drug, etc (Katona and Farkas, 2014; Chen et al., 2018). Therefore, in this study, we extracted the UMLS concepts from input sentences and then used them as features for medical jargon term extraction. We elicited UMLS concepts from input sentences with QuickUMLS, an unsupervised UMLS concept matching tool (Soldaini and Goharian, 2016). Then, we represented the positions of concepts in binary feature extraction in BIOES binary encoding. The weighting score feature was expressed by multiplying the binary encoding of a concept by the term weighting.

In this study, we employed the widely used TF score and the masked language model (MLM) score as term weighting methods. We normalized MLM scores and TF scores to values between [0, 1] by Min-Max scaling (Al Shalabi and Shaaban, 2006). Details on the expression of additional features are described in Appendix B.

Contextualized MLM Score Frequency score-based methods have been widely used to extract unfamiliar or important terms, since some jargon terms can be rarely observed in the general corpus (Chen et al., 2018). However, term frequency-based approaches do not consider contextual factors, and therefore tend to underestimate the homonym issue. Otherwise, a language model is a probability distribution over a sequence of words. We can calculate the probability of phrases or words for a given context. In particular, in MLMs, it is known that we can understand whether knowledge of a specific concept is included in PLMs by masking part of the sentence (Petroni et al., 2019; Kwon et al., 2019a; Zhong et al., 2021).

We proposed a MLM score that is the negative likelihood of the masked tokens from a text. Eq. 2 is the MLM score of a UMLS medical concept c for a given sentence S. Suppose, T_c is the token length of c and p is the starting position of c. Herein, we mask concept tokens $S_p ... S_{p+T_c-1}$ with a special token "[MASK]" then input the masked will be $\tilde{S}$ to PLMs. As a result, we can get the probability $P(\tilde{S}_i = S_i|\tilde{S})$ of the i^{th} masked token $\tilde{S}_i$ will be S_i from $\tilde{S}$. Then, we calculated the MLM score of c ($MLM(c, S)$) by averaging negative log likelihoods of masked tokens. Overall, when the MLM of the model is low, it means that the masked concept can be predicted easily.

$$MLM(c, S) = -\frac{1}{T_c} \sum_{i=p}^{p+T_c-1} logP(\tilde{S}_i = S_i|\tilde{S}) \tag{2}$$

4.2.2 Target Model

First, an input sentence was split into subword units through a PLM tokenizer. Binary features were input to a multi-layer perceptron (MLP), mapped into the same dimension as the token embedding vector, and then added to the output of the tokenizer. The added input is input to a Wiki-trained LM then we can get hidden. In the Wiki's hyperlink step, the initial parameters of the PLM trained were set to the weight parameters of Wiki-trained LM. Then, the output of the Wiki-trained LM (Hidden) was input to an MLP to create an emission score. Simultaneously, the weighted scores and the Hidden were concatenated and then input to another MLP to create the weighted emission score. We got the final emission score by adding the emission score and the weighted emission score. Then, the final emission was input to the conditional random field (Lafferty et al., 2001) layer. Suppose $\mathbf{P}$ is the final emission, and y is a sequence of output labels. Herein, we calculated the score of the sequence y defined as Eq. 3 with the transition matrix A. Then, we picked the optimal output sequence $\hat{y}$ from all possible sequences of labels $\mathbf{Y}$ by jointly decoding through Viterbi searching (Viterbi, 1967).

$$s(y) = exp\left(\sum_{i=0}^{n} \mathbf{A}_{y_i y_{i+1}} + \sum_{i=0}^{n} \mathbf{P}_{iy_i}\right) \tag{3}$$

$$\hat{y} = \underset{\tilde{y} \in \mathbf{Y}}{\operatorname{argmax}}\ s(\tilde{y}) \tag{4}$$

5 Experiment

5.1 Experimental Set Up

The experiments on the WikiHyperlink span prediction were conducted on the following settings. The Wiki data consists of approximately 26M articles, which correspond to 150M sentences with 99M hyperlinks. We used BERT-cased base (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) base models. We also utilized BioClinicalBERT (Alsentzer et al., 2019b) and BioBERT (Lee et al., 2020) which are the state-of-the-art PLMs pretrained on biomedical text. We mainly set the hyper-parameters of the PLMs by following Gururangan et al. (2020)'s post-training setting. Meanwhile, each input consisted of up to 128 tokens and the learning rate was set to 5e-4. The parameters were updated every 2,048 inputs. We trained PLMs up to 50K update steps, which is slightly less than 1 epoch (about 56K; 7 days). For the remaining hyper-parameters, we used the default setting of the Transformers library (Wolf et al., 2020).

To fine-tune medical jargon extract models, we used the following settings. First, the batch size and maximum epoch were set to 32 and 3, respectively, according to the PLMs' standard training setting. We set the learning rate as 5e-5 for all models. Finally, we randomly split the dataset into a 14,542 training set (80%), a 1,817 validation set (10%), and a 1,819 test set (10%). Hyper-parameters and experimental models were selected with the highest performance in the validation set, and detailed results are described in Appendix C.

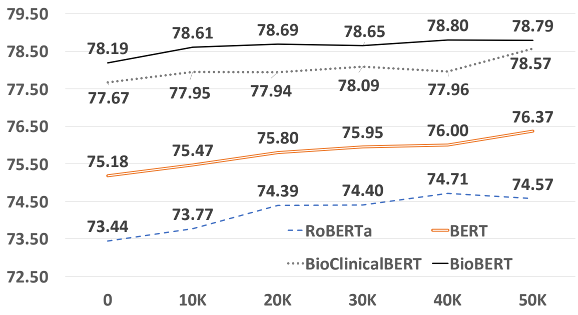


Figure 2: The F1 scores of the vanilla models with the update step of the WikiHyperlink training.

Moreover, a Wordfreq library (Speer et al., 2018) was adopted to calculate TF of the candidate UMLS concepts. We performed Student's t-test (Student, 1908) to assess whether the change in performance between experimental results was statistically significant. Finally, we used the F1 score (Kwon et al., 2019b) to evaluate the performance of the model.

5.2 Experimental Results

The PLMs can be categorized as the following two types: 1) **pretrained** models were initialized with standard pretrained models and 2) **Wiki-trained** models were initialized with the Wiki's hyperlink trained models. **Vanilla** models do not incorporate the UMLS features. The **binary** model has only the binary features. **+TF** and **+MLM** indicate that adding the TF score feature and MLM score feature, respectively. **+TF+MLM** concatenates two features as the weighted input. Finally, The **Ensemble** is a weighted voting of the predictions of four models (Binary, +TF, +MLM, +MLM+TF) designed to reflect various aspects of the features. The algorithm for Ensemble is described in Appendix E.

5.2.1 Experimental Results on the Hyperlink Training Step

Figure 2 is the fine-tuning performance of the vanilla models for every 10K update step on the test set. Herein, step 0 indicates the pretrained setting. The results show that the medical jargon term extraction performance tends to be improved as the update step increases. The performance was improved by 1.19%p (p<1e-4) in BERT, 1.13%p (p<1e-3) in RoBERTa, 0.90%p (p=0.11) in BioClinicalBERT and 0.60%p (p=0.07) in BioBERT, although the models were trained with less than 1 epoch of Wiki data. Considering that the pre-

Model	BERT						RoBERTa						BioClinicalBERT						BioBERT					
	Pretrained			Wiki-trained			Pretrained			Wiki-trained			Pretrained			Wiki-trained			Pretrained			Wiki-trained		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Vanilla	75.08	75.27	75.18	75.65	77.11	76.37	72.00	74.94	73.44	73.06	76.14	74.57	77.29	78.06	77.67	77.39	79.78	78.57	77.79	78.59	78.19	78.76	79.40	78.79
Binary	76.44	77.06	76.75	76.31	78.51	77.39	74.00	75.10	74.54	74.63	76.31	75.46	**78.47**	78.85	78.66	**78.80**	79.38	79.09	78.88	78.34	78.61	78.76	79.40	79.08
+TF	75.96	78.04	76.99	75.92	**79.31**	77.58	73.90	76.50	75.18	74.29	76.44	75.35	78.16	79.60	78.88	78.63	79.80	79.21	78.73	79.51	**79.12**	78.69	80.01	79.35
+MLM	76.05	78.09	77.06	75.83	79.16	77.46	72.91	76.00	74.42	74.36	76.45	75.39	78.15	79.65	**78.89**	78.75	79.70	79.22	78.68	79.45	79.06	78.65	80.04	79.34
+TF+MLM	**76.27**	77.39	76.83	**77.67**	77.26	77.26	73.06	76.14	74.57	74.12	76.676	75.37	78.16	79.02	78.59	78.37	79.09	78.73	78.49	78.78	78.64	78.60	79.36	78.98
Ensemble	76.07	**78.62**	**77.33**	76.39	78.90	**77.62**	**74.09**	**76.83**	**75.44**	**74.80**	**76.96**	**75.86**	77.93	**79.88**	**78.89**	78.67	**79.81**	**79.23**	**78.73**	**79.52**	**79.12**	**78.71**	**80.06**	**79.38**

Table 1: The precision (Prec), recall (Rec) and F1 scores of MedJEx models.

training corpora of all models include the English Wikipedia corpus (Liu et al., 2019), we can infer that the improvements are due to the hyperlink span information rather than Wiki's text. Otherwise, the performances of biomedical BERTs are marginally enhanced. We speculate that this is because the models have already been trained on the biomedical literature, so the effect of the task transfer through learning WikiHyperlink span information is relatively small. Nevertheless, these results imply that the task transfer is effective albeit Wiki data are a general corpus. Overall, we can see that the Wiki-training is beneficially transferred to medical jargon extraction models, supporting our assumption.

5.2.2 Impact of the Proposed Methods

Table 1 contains the experimental results for evaluating the impact of the proposed methods. Compared to the vanilla models, the Wiki-trained ensemble models outperformed by 2.44%p in BERT (p<1e-11), 2.42%p (p<1e-9) in RoBERTa, 1.56%p (p<1e-5) in BioClinicalBERT and 1.19%p (p<1e-3) in BioBERT. We can see that the Wiki-trained models improved performance in all 24 cases compared to the pretrained models. This means that the WikiHyperlink span's information is helpfully transferred to training medical jargon. In addition, the binary models demonstrate better performance compared to the Vanilla models. Compared to Binary, TF and MLM features improve performance marginally in BERT, BioClinicalBERT and BioBERT. On the other hand, in the RoBERTa model, while the TF feature improves the performance in the pretrained model, it can be seen that the performance is slightly decreased in other cases. In addition, when both TF and MLM features are included, the performance is marginally changed compared to using each feature. The ensemble models lead to the highest performance in all cases.

	Prec.	Rec.	F1
QuickUMLS	21.69	62.21	32.16
MedCAT	45.89	32.32	37.93

Table 2: The precision (Prec), recall (Rec) and F1 scores on UMLS concept extraction systems.

Type	Datasets	Pretrained	Wiki-trained
Disease	NCBI disease	87.92	**89.21**
	BC5CDR disease	83.93	**84.87**
Drug & Chem.	BC5CDR Chem.	92.07	91.88
	BC4CHEMD	90.06	**90.27**
Gene & Protein	BG2GM	82.30	**83.06**
	JNLPBA	74.95	**77.95**
Species	LINNAEUS	87.59	**89.87**
	S800	74.95	74.85

Table 3: F1 score of BioBERT$_{Vanilla}$ models on Pretrained and Wiki-trained settings in BioNER datasets. Chem. indicates 'chemical.'

5.3 Comparison with UMLS Concept Extractors

To verify that our task is different from existing UMLS concept extraction task. For this, we evaluated the performance of existing UMLS concept extractors, QuickUMLS and MedCAT (Kraljevic et al., 2019), in MedJ's test dataset. The results in Table 2 show that the performance of UMLS extractors was substantially inferior to our models. Even though QuickUMLS extracted all possible UMLS concepts, the recall score was 62.21, indicating that a substantial amount of medical jargon terms (37.79%) in EHR notes is not included in the UMLS concepts. To put it differently, since UMLS concept extractors mainly concentrated on specific types of medical terms, there are some representative jargon types that the UMLS concept extractors frequently fail to predict: 1) abbreviations (e.g., yo: years old, s/p: status post ...), 2) special numerical terms (e.g., 20/40: vision test results, 2-0: a heavy thread used for stitching ...).

5.4 Impact of the Wiki's Hyperlink Span Training on BioNER Datasets

We assessed the generalizability of Wiki training by conducting experiments on eight BioNER benchmarks used in Lee et al. (2020)'s setting [6]. We evaluated the performance of the BioBERT$_{Vanilla}$ model on each data in the pretrained and Wiki-trained settings. Table 3 shows F1 scores on the datasets. We can see that Wiki training positively affected five datasets while it marginally impacted three datasets (S800, BC5CDR Chem. and BC4CHEMD). Especially, in JNLPBA, the performance improveed by 3%p, and in LINNAEUS, the performance improved by 2.29%p. Meanwhile, since BioNER benchmarks are targeted to elicit specific medical concepts, there is some medical jargon that BioNER cannot cover, such as metric units (millliter, mg, ...) and medical techniques (flushing: to use fluid to clean out a catheter) and so on. Detailed experimental settings and results are described in Appendix F.

6 Discussion

6.1 Feature Analysis

MLM Figure 3 represents the histograms of the biomedical concepts on MLM scores. The blue-colored histogram indicates the UMLS biomedical concepts that are not jargon, while the red-colored histogram indicates UMLS biomedical concepts that are jargon. We can notice the heavily tail of the histogram of non-jargon concepts indicating MLM scores are lower. The heavy tail includes concepts that are relatively easy to understand (e.g. shoulder, chest pain, wound management ...). These results show that the MLM score can be an appropriate feature to determine whether a concept will be the medical jargon. Additional analyses for the MLM score are in Appendix G.

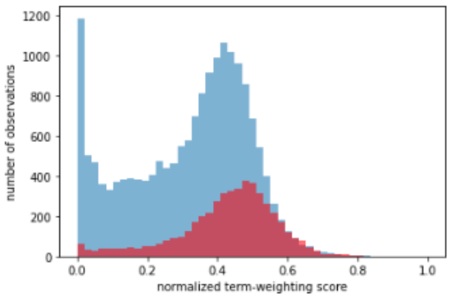


Figure 3: Histograms of the MLM score feature for UMLS biomedical concepts. Red: jargon concepts; blue: non-jargon concepts

TF and MLM We conducted a case study to analyze the impact of the TF and MLM for identifying medical jargon from EHR notes. Specifically, we calculated the TF and MLM scores of candidate UMLS concepts that had been mapped to the medical jargon in our EHR note sentences. Then, we categorized the concepts according to their scores. We used MLM of the pretrained BERT, and the concepts were categorized as a high score (> 0.5; ↑) and a low score (< 0.5; ↓). Note that a high MLM score means that the BERT failed to predict the concept. A high TF score means that the concept was frequently observed in the general corpus. The following is the combination of MLM and TF categories and notable examples.

1. ↑ TF, ↑ MLM: "shock", "drainage", "tissue"
2. ↓ TF, ↑ MLM: "Vancomycin B", "Seroquel", "subdural hematoma"
3. ↑ TF, ↓ MLM: "coma", "gene", "wound"
4. ↓ TF, ↓ MLM: "pneumonia", "membrane", "viral"

The first case is a word frequently observed in a general corpus (↑ TF), but it is a concept that fails to predict in the BERT (↑ MLM). This concept includes rare senses used in medical contexts. For example, "drainage" is used as a synonym for sewer in the general context, while in the medical domain it may mean "extra liquid that is removed from the body." The second case is the most unfamiliar words. The concepts are composed of multiple tokens and medical entities such as disease or drug names. The third case consists of relatively easy-to-understand concepts. The fourth case contains relatively short medical jargon composed of 1 to 2 tokens. We can infer that MLM and TF cannot only be complementary but also can be used together to help solve the challenging homonym issue.

6.2 Error Analysis

We manually examined the outputs. The most common type of false negatives errors was abbreviations, such as "ENT" for "ear, nose, and throat" and "or" for "operating room", and "p.o." for "per os". Another type of error was signs with special meanings such as "q.6 h" for "per every 6 hours", "x2" for "two times", and "3+" for "very strong"

[6] NBCI disease (Dogan and Lu, 2012), BC5CDR disease (Wei et al., 2016), BC5CDR chemical (Wei et al., 2016), BC4CHEMD chemical, (Krallinger et al., 2015), BC2GM (Smith et al., 2008), JNLPBA (Kim et al., 2004), LINNAEUS (Gerner et al., 2010), S800 (Pafilis et al., 2013)

fall into this type. Other notable errors were epoymous person name-based medical concepts (e.g., "Azzopardi effect") and device names (e.g., "BiPap"). MedJEx failed to detect the aforementioned types of medical jargon due to the data sparsity challenge.

6.3 Prospective Downstream Applications

Medical jargon extraction task has divers potential applications. It could be a preprocessing part of BioNLP pipelines and used for downstream medical AI application systems. For example, it could be adapted to medical concept linking systems such as NoteAid (Polepalli Ramesh et al., 2013). In addition, a chatbot-based self-diagnosis system (You and Gui, 2020) could use our approach for the explanation of medical jargons to avoid generating jargons.

6.4 Merits

Prospective downstream applications can promote effective communication between clinicians and their patients by increasing patients' EHR comprehension ability. This, in turn, can help the patients in self-management of their illness (Adams, 2010). Effective communication is also beneficial for preventing physicians' burnout (Aaronson et al., 2019). Thus, we can expect this new task will contribute not only to improve the patients' outcomes.

6.5 Limitations

This task defined medical jargon at a single difficulty-level, disregarding diverse educational levels of users. In particular, setting the difficulty of each medical jargon term will help this task contribute to improving the performance of machines as well as patients, and further educate and support clinicians. Moreover, we did not analyze the jargon types such as acronyms but merely identified the presence of medical jargon, which can limit further analyses.

7 Conclusion

We introduce a novel NLP tasked named MedJEx and present an expert-curated MedJ dataset for the task. We propose two innovative methods: 1) Pretraining Wiki's hyperlink span, and 2) Contextualized MLM score feature for extracting medical jargon from EHR notes. The experimental results show that the Wiki's hyperlink span can be effectively transferred to the medical jargon extraction model, leading to a significant performance improvement. Wiki's hyperlink span training also beneficial in six out of eight BioNER benchmarks. Finally, in a qualitative evaluation, the MLM score feature complements the TF feature to identify common terms (or terms with high TFs) used in the clinical domain (homonyms).

Ethical Consideration

In this study, we legitimately obtained a licensed access to the University of Pittsburgh Medical Center EHR repository, and all EHR notes used were fully de-identified.The experiments described in Appendix A.1 and D were performed in accordance with the recommendations laid out in the World Medical Association Declaration of Helsinki. The study protocol was approved by the institutional review boards of a medical school in the US.

In addition, our model used BERT and its families, so it over-relies on a contextual embedding feature that can cause mis-classification. Specifically, even with the same terminology, the prediction of a model may be different depending on the context.

Acknowledgement

Research reported in this study was in part supported by the National Institutions of Health R01DA045816 and R01MH125027. The content is solely the responsibility of the authors and does not necessarily represent the official views of National Institutes of Health.

References

Emily L Aaronson, Benjamin A White, Lauren Black, David F Brown, Theodore Benzer, Allison Castagna, Ali S Raja, Jonathan Sonis, and Elizabeth Mort. 2019. Training to improve communication quality: an efficient interdisciplinary experience for emergency department clinicians. *American Journal of Medical Quality*, 34(3):260–265.

Emil Abrahamsson, Timothy Forni, Maria Skeppstedt, and Maria Kvist. 2014. Medical text simplification using synonym replacement: Adapting assessment of word difficulty to a compounding language. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 57–65.

Robert John Adams. 2010. Improving health outcomes with better patient understanding and education. *Risk management and healthcare policy*, 3:61.

Herman Aguinis, Isabel Villamor, and Ravi S Ramani. 2021. Mturk research: Review and recommendations. *Journal of Management*, 47(4):823–837.

Luai Al Shalabi and Zyad Shaaban. 2006. Normalization as a preprocessing engine for data mining and the approach of preference matrix. In *2006 International conference on dependability of computer systems*, pages 207–214. IEEE.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019a. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019b. Publicly available clinical bert embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78.

Alan R Aronson. 2001. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association.

Giusepppe Attardi. 2015. Wikiextractor. `https://github.com/attardi/wikiextractor`.

Molly Baldry, Carol Cheal, Brian Fisher, Myra Gillett, and Val Huet. 1986. Giving patients their own records in general practice: experience of patients and staff. *Br Med J (Clin Res Ed)*, 292(6520):596–598.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Jinying Chen, Emily Druhl, Balaji Polepalli Ramesh, Thomas K Houston, Cynthia A Brandt, Donna M Zulman, Varsha G Vimalananda, Samir Malkani, and Hong Yu. 2018. A natural language processing system that links medical terms in electronic health record notes to lay definitions: system development using physician reviews. *Journal of medical Internet research*, 20(1):e26.

Jinying Chen, Abhyuday N Jagannatha, Samah J Fodeh, and Hong Yu. 2017. Ranking medical terms to support expansion of lay language resources for patient comprehension of electronic health record notes: adapted distant supervision approach. *JMIR medical informatics*, 5(4):e8531.

Jinying Chen and Hong Yu. 2017. Unsupervised ensemble ranking of terms in electronic health record notes based on their importance to patients. *Journal of biomedical informatics*, 68:121–131.

Antonio Cuevas, Manuel Febrero, and Ricardo Fraiman. 2004. An anova test for functional data. *Computational statistics & data analysis*, 47(1):111–122.

Dina Demner-Fushman, Willie J Rogers, and Alan R Aronson. 2017. Metamap lite: an evaluation of a new java implementation of metamap. *Journal of the American Medical Informatics Association*, 24(4):841–844.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Rezarta Islamaj Dogan and Zhiyong Lu. 2012. An improved corpus of disease mentions in pubmed citations. In *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, pages 91–99.

Martin Gerner, Goran Nenadic, and Casey M Bergman. 2010. Linnaeus: a species name identification system for biomedical literature. *BMC bioinformatics*, 11(1):1–17.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223.

Abhyuday Jagannatha, Feifan Liu, Weisong Liu, and Hong Yu. 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(1):99–111.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Sasikiran Kandula, Dorothy Curtis, and Qing Zeng-Treitler. 2010. A semantic and syntactic text simplification tool for health content. In *AMIA annual symposium proceedings*, volume 2010, page 366. American Medical Informatics Association.

Melinda Katona and Richárd Farkas. 2014. Szte-nlp: Clinical text analysis with named entity recognition. In *Proceedings of the 8th International Workshop* *on Semantic Evaluation (SemEval 2014)*, pages 615–618.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Citeseer.

Zeljko Kraljevic, Daniel Bean, Aurelie Mascio, Lukasz Roguski, Amos Folarin, Angus Roberts, Rebecca Bendayan, and Richard Dobson. 2019. Medcat–medical concept annotation tool. *arXiv preprint arXiv:1912.10166*.

Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A Folarin, Angus Roberts, et al. 2021. Multi-domain clinical natural language processing with medcat: the medical concept annotation toolkit. *Artificial Intelligence in Medicine*, 117:102083.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(1):1–17.

Sunjae Kwon, Cheongwoong Kang, Jiyeon Han, and Jaesik Choi. 2019a. Why do masked neural language models still need common sense knowledge? *arXiv preprint arXiv:1911.03024*.

Sunjae Kwon, Youngjoong Ko, and Jungyun Seo. 2019b. Effective vector representation for the korean named-entity recognition. *Pattern Recognition Letters*, 117:52–57.

John D Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

John P Lalor, Wen Hu, Matthew Tran, Hao Wu, Kathleen M Mazor, and Hong Yu. 2021. Evaluating the effectiveness of noteaid in a community hospital setting: Randomized trial of electronic health record note comprehension interventions with patients. *Journal of medical Internet research*, 23(5):e26354.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157.

Donald AB Lindberg, Betsy L Humphreys, and Alexa T McCray. 1993. The unified medical language system. *Yearbook of Medical Informatics*, 2(01):41–51.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Huanru Henry Mao. 2020. A survey on self-supervised pre-training for sequential transfer learning in neural networks. *arXiv preprint arXiv:2007.00800*.

Clara Meister, Tiago Pimentel, Thomas Clark, Ryan Cotterell, and Roger Levy. 2022. Analyzing wrap-up effects through an information-theoretic lens. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 20–28.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. 2021. Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1744–1753.

Rada Mihalcea and Andras Csomai. 2007. Wikify! linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Islam Nassar, Michelle Ananda-Rajah, and Gholamreza Haffari. 2019. Neural versus non-neural text simplification: A case study. In *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association*, pages 172–177.

Evangelos Pafilis, Sune P Frankild, Lucia Fanini, Sarah Faulwetter, Christina Pavloudi, Aikaterini Vasileiadou, Christos Arvanitidis, and Lars Juhl Jensen. 2013. The species and organisms resources for fast and accurate identification of taxonomic names in text. *PloS one*, 8(6):e65390.

Jon Patrick and Min Li. 2010. High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge. *Journal of the American Medical Informatics Association*, 17(5):524–527.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Balaji Polepalli Ramesh, Thomas Houston, Cynthia Brandt, Hua Fang, and Hong Yu. 2013. Improving patients' electronic health record comprehension with noteaid. In *MEDINFO 2013*, pages 714–718. IOS Press.

Basel Qenam, Tae Youn Kim, Mark J Carroll, Michael Hogarth, et al. 2017. Text simplification using consumer health vocabulary to generate patient-centered radiology reporting: translation and evaluation. *Journal of medical Internet research*, 19(12):e8536.

Leonard Richardson. 2007. Beautiful soup documentation. *April*.

Sebastian Ruder. 2019. *Neural transfer learning for natural language processing*. Ph.D. thesis, NUI Galway.

Ilham Fathy Saputra, Rahmad Mahendra, and Alfan Farizki Wicaksono. 2018. Keyphrases extraction from user-generated contents in healthcare domain using long short-term memory networks. In *Proceedings of the BioNLP 2018 workshop*, pages 28–34.

Dean Schillinger, Margaret Handley, Frances Wang, and Hali Hammer. 2009. Effects of self-management support on structure, process, and outcomes among vulnerable patients with diabetes: a three-arm practical clinical trial. *Diabetes care*, 32(4):559–566.

Shock. 2022. in: *Cambridge Dictionary*.

Larry Smith, Lorraine K Tanabe, Cheng-Ju Kuo, I Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M Friedrich, Kuzman Ganchev, Manabu Torii, et al. 2008. Overview of biocreative ii gene mention recognition. *Genome biology*, 9(2):1–19.

Luca Soldaini and Nazli Goharian. 2016. Quickumls: a fast, unsupervised approach for medical concept extraction. In *MedIR workshop@SIGIR*, pages 1–4.

Marina Solnyshkina, Radif Zamaletdinov, Ludmila Gorodetskaya, and Azat Gabitov. 2017. Evaluating text complexity and flesch-kincaid grade level. *Journal of social studies education research*, 8(3):238–248.

Sanjay Soundarajan, Sachira Kuruppu, Ashutosh Singh, Jongchan Kim, and Monalisa Achalla. 2021. Sparclink: an interactive tool to visualize the impact of the sparc program. *bioRxiv*.

Robyn Speer, Joshua Chin, Andrew Lin, Sara Jewett, and Lance Nathan. 2018. Luminosoinsight/wordfreq: v2.2.

Student. 1908. The probable error of a mean. *Biometrika*, pages 1–25.

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer.

Andrew Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the biocreative v chemical-disease relation (cdr) task. *Database*, 2016.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Jie Yang, Shuailong Liang, and Yue Zhang. 2018. Design challenges and misconceptions in neural sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3879–3889.

Yue You and Xinning Gui. 2020. Self-diagnosis through ai-enabled chatbot-based symptom checkers: user experiences and design considerations. In *AMIA Annual Symposium Proceedings*, volume 2020, page 1354. American Medical Informatics Association.

Qing T Zeng and Tony Tse. 2006. Exploring and developing consumer health vocabularies. *Journal of the American Medical Informatics Association*, 13(1):24–29.

Qing Zeng-Treitler, Sergey Goryachev, Hyeoneui Kim, Alla Keselman, and Douglas Rosendale. 2007. Making texts in electronic health records comprehensible to consumers: a prototype translator. In *AMIA Annual Symposium Proceedings*, volume 2007, page 846. American Medical Informatics Association.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [mask]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5017–5033.

Guido Zuccon, Alexander Holloway, Bevan Koopman, and Anthony Nguyen. 2013. Identify disorders in health records using conditional random fields and metamap aehrc at share/clef 2013 ehealth evaluation lab task 1. In *Proceedings of the CLEF 2013 Workshop on Cross-Language Evaluation of Methods, Applications, and Resources for eHealth Document Analysis*, pages 1–8. The CLEF Initiative (Conference and Labs of the Evaluation Forum).

Appendices

A Data Annotation

A.1 Evaluation of the Annotation

To evaluate the annotators' reliability in identifying jargon, an observational study was performed to assess the agreement of the dataset annotators with each other and with laypeople. **Note that, this work is a part of unpublished manuscript.**

A.1.1 Data Collection and Setting

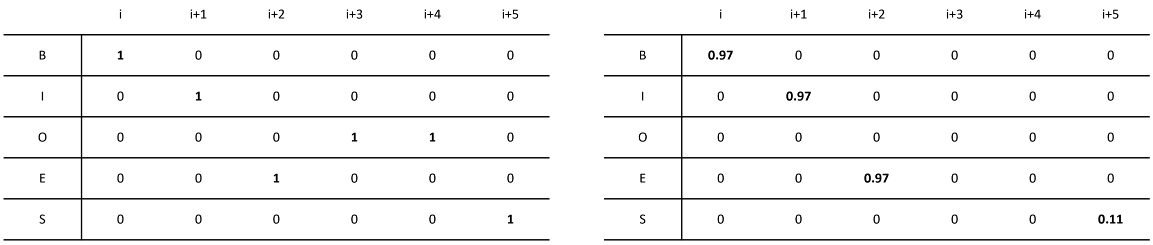
For evaluation, twenty sentences were randomly selected from deidentified inpatient EHR notes from the University of Pittsburgh Medical Center EHR repository. Sentences that consisted only of administrative data, sentences whose length was less than ten words, and sentences substantially indistinguishable from another sentence were filtered out.

Note that, the annotators had never seen the sampled sentences. The twenty sentences were made up of 904 words in total. Common words were discarded so as not to inflate the calculated agreement. These consisted of all pronouns, conjunctions, prepositions, numerals, articles, contractions, months, punctuation, and the most common 25 verbs, nouns, adverbs, and adjectives. Terms occurring more than one time in a sentence were counted only once. Furthermore, to ameliorate double-counting issue, multi-word terms were counted as single terms. Multi-word terms were determined by two members of the research team by consensus. In this work, multi-word terms were defined as adjacent words that represented a distinct medical entity (examples: “PR interval”, “internal capsule”, “acute intermittent porphyria”), were commonly used together (examples: “hemodynamically stable”, “status post”, “past medical history”) and terms that were modified by a minor word (examples: “trace perihepatic fluid”, “mild mitral regurgitation”, “rare positive cells”, “deep pelvis”). After applying these rules, 325 candidate medical jargon terms were utilized. The laypeople consisted of 270 individuals recruited from Amazon Mechanical Turk (MTurk) (Aguinis et al., 2021).

A.1.2 Annotation Reliability

The results showed that there was good agreement among annotators (Fleiss' kappa = 0.781). The annotators had high sensitivity (91.7%) and specificity (88.2%) in identifying jargon terms as determined by the laypeople (the gold standard).

B Details on Feature Representations



	i	i+1	i+2	i+3	i+4	i+5
B	**1**	0	0	0	0	0
I	0	**1**	0	0	0	0
O	0	0	0	**1**	**1**	0
E	0	0	**1**	0	0	0
S	0	0	0	0	0	**1**

(a) Binary feature

	i	i+1	i+2	i+3	i+4	i+5
B	**0.97**	0	0	0	0	0
I	0	**0.97**	0	0	0	0
O	0	0	0	0	0	0
E	0	0	**0.97**	0	0	0
S	0	0	0	0	0	**0.11**

(b) Weighted score feature

Figure 1: Examples of a binary feature and a weighted score feature.

This section explains details on the binary and weighted score features with an example. Suppose a concept c_1 starts with the i^{th} token and the length is 3, and another concept c_2 starts with the $i + 5^{th}$ token and the length is 1. In this case, binary encoding features can be expressed as a 5-dimensional vector ("B", "I", "O", "E", "S") as shown in Figure 1(a). Since c_1 starts at i and ends at $i + 2$, the i^{th} "B" dimension and $i + 2^{th}$ "E" dimension are set to 1. In addition, because the $i + 1^{th}$ token is the inner of the c_1, the $i + 1^{th}$ 'I' dimension will be 1. Then, c_2 starts and ends at $i + 5$. On the other hand, further assume that the term weighting score of c1 and c2 are 0.97 and 0.11

C Details of the Experimental Setting

In the experiments, we set the models' hyper-parameters with the performance on the validation set via the grid-search. Once, determining hyper-parameters in vanilla models, the same values were used in the other models. We trained WikiHyperlink for 50K update steps. Herein, the number of parameters of all experimental models are about 108M. In all experiments, the random seed was set to 0. All experiments were conducted in the Centos Linux 7 environment using one RTX-8000 GPU, Intel Xeon E5-2620 CPU, and 64GB RAM.

C.1 Hyper-parameter Setting

In the case of MLP, all hidden sizes were set equal to the default hidden size of PLM. Also, the activation function used a hyperbolic tangent function by following Gururangan et al. (2020)'s setting.

Learning Rate	BERT	RoBERTa	BioClinicalBERT	BioBERT
6e-5	67.46	62.80	67.07	70.48
1e-5	71.02	66.91	73.06	73.97
5e-5	**75.92**	**73.64**	**78.53**	**78.51**

Table 1: The F1 scores of the finetuned vanilla models for each learning rate.

In the fine-tuning on the task, we choose the best learning rate of the vanilla models on the validation set among the following set of the candidate learning rates {5e-6, 1e-5, 5e-5}. Overall, the results in Table 1 show that we could achieve the best performances on validation set when the learning rate was set 5e-5.

C.2 Model Selection

BioClinicalBERT	BioBERT	BioMedRoBERTa	BioClinicalRoBERTa
78.53	78.51	73.89	76.27

Table 2: The F1 scores of the fine-tuned pretrained biomedical PLMs on vanilla setting.

In this work, we utilize contextualized PLMs to make jargon prediction models. For this, we use two representative PLMs: BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019). In addition, there are several recent state-of-the-art models pretrained in biomedical domains, recently. To be specific, BioBERT additionally trained BERT with biomedical text corpora (Lee et al., 2020). BioClinicalBERT (Alsentzer et al., 2019b) further trained BioBERT with clinical notes from MIMIC-III (Johnson et al., 2016). In addition, there are some studies, such as BioMedRoBERTa (Gururangan et al., 2020) or BioClinicalRoBERTa (Lewis et al., 2020) that suggested training the RoBERTa model with biomedical text corpora or clinical notes. On the other hand, Michalopoulos et al. (2021) proposed UmlsBERT that integrates UMLS semantic type embedding as an additional input feature during the pretraining step. UmlsBERT is similar to our suggestion in that it uses UMLS concept as an embedding feature. However, our method is slightly different in that it uses span information instead of the UMLS semantic type. Moreover, we show that the performance can be improved using the UMLS features only in fine-tuning.

In this paper, we selected two biomedical PLMs by comparing the performances of state-of-the-art biomedical PLMs in the vanilla models on the validation set. Table 2 presents the experimental comparison among the four representative biomedical PLMs: BioClinicalBERT (Alsentzer et al., 2019a), BioBERT (Lee et al., 2020), BioMedRoBERTa and BioClinicalRoBERTa. The results show that BioBERT and BioClinicalBERT showed no differences ($p > 0.05$) but the other RoBERTa-based models presented inferior scores. Therefore, we choose the BERT-basedbiomedical PLMs for further experiments.

C.3 Experimental Results of UMLS concept extractors

	Prec.	Rec.	F1
QuickUMLS	21.10	**60.74**	31.32
MedCAT	**45.89**	32.32	**37.93**

Table 3: The precision (Prec), recall (Rec) and F1 scores on UMLS concept extraction systems.

Setting	Concept Extractor	Prec.	Rec.	F1
BERT_{Binary}	QuickUMLS	**75.51**	**77.37**	**76.43**
BERT_{Binary}	MedCAT	74.90	77.00	75.93

Table 4: The precision (Prec), recall (Rec) and F1 scores on BERT with a binary setting on the different concept extractors.

Since experimental models rely on a UMLS concept extractor, it is also important to choose appropriate UMLS concept extractors. There are several concept extractors that have been introduced including MetaMap (Demner-Fushman et al., 2017), QuickUMLS (Soldaini and Goharian, 2016), cTAKES (Saputra et al., 2018), and MedCAT (Kraljevic et al., 2021). We compared two extractors, QuickUMLS and MedCAT, which are state-of-the-art concept extractors. Table 3 presents the performance of the concept extractors. Herein, we can see that MedCAT achieved better performance in terms of precision and F1 but QuickUMLS had better recall performance. We preferred higher recall, since a concept extractor was used for candidate concept extraction. Indeed, the performances on the BERT with the binary setting in Table 4 demonstrates that using QuickUMLS led to higher performance than that of using MedCAT.

C.4 Experimental Results on Tagging Schemes

Tagging scheme	F1
BIO	74.25
BIOES	**75.92**

Table 5: Experimental comparison on BIO and BIOES tagging schemes.

Finally, to select a sequence labeling tagging scheme, we compared two representative tagging schemes: Begin, Inside and Outside (BIO) and Begin, Inside, Outside, End, and Singleton (BIOES) (Yang et al., 2018). Table 5 presents the experimental results on the validation set of the BERT's vanilla setting. The results show that the validation performance with the BIO scheme is lower than that of the BIOES scheme.

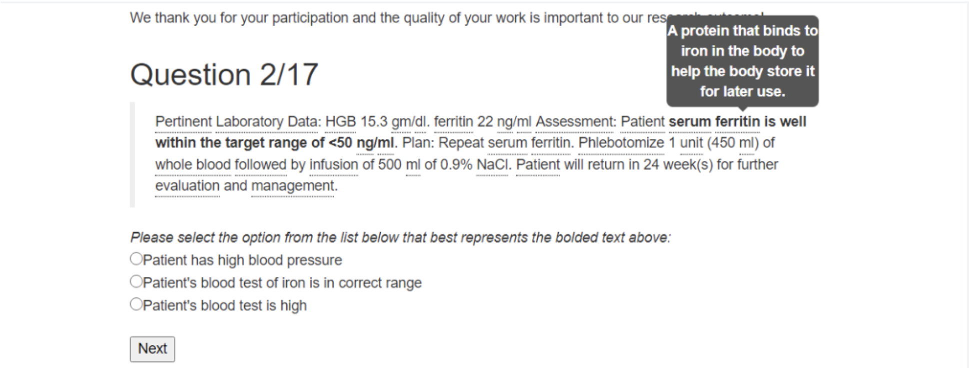
D The Impact on the Understandability to Patients

This section introduces an experiment to verify that providing medical jargon and the corresponding lay definitions can be beneficial to the comprehension of patients. Note that all experimental settings and results are part of Lalor et al. (2021)'s work.

D.1 Experimental Setting

The authors recruited 174 patients from a community hospital in the USA. Herein, the participants took a web-based EHR comprehension test and the participants were randomly assigned to a control (n=85) group or intervention (n=89) group to take the test without or with the support of the medical jargons identification and the corresponding lay definitions, respectively. In addition, 200 participants from MTurk were engaged to take the test (100 participants were assigned to a control group and the other 100 were allocated to an intervention group).

D.1.1 EHR Comprehension Test



We thank you for your participation and the quality of your work is important to our res

A protein that binds to iron in the body to help the body store it for later use.

Question 2/17

Pertinent Laboratory Data: HGB 15.3 gm/dl. ferritin 22 ng/ml Assessment: Patient **serum ferritin is well within the target range of <50 ng/ml**. Plan: Repeat serum ferritin. Phlebotomize 1 unit (450 ml) of whole blood followed by infusion of 500 ml of 0.9% NaCl. Patient will return in 24 week(s) for further evaluation and management.

Please select the option from the list below that best represents the bolded text above:

○Patient has high blood pressure

○Patient's blood test of iron is in correct range

○Patient's blood test is high

Figure 2: An example question of the EHR comprehension test. Herein, you can see that identifying medical jargon "ferritin" and providing its definition can be helpful to understand that the bold text describes a blood iron test.

To assess a user's comprehension of EHR notes, we conducted the EHR comprehension test. Table 2 is an example of the EHR comprehension test. This test consists of 14 paragraphs extracted from de-identified EHR notes and relevant multiple-choice questions curated by physicians. In previous work, it has been verified that the EHR comprehension test reflects the participant's education level and understandability of the medical literature. In this experiment, we provided definitions for medical jargon only to the intervention group.

D.2 Experimental Results

Source	Condition control	Intervention
MTurk	0.756±0.246	0.830±0.201
Local hospital	0.646±0.179	0.727±0.191

Table 6: Experimental evaluation on patients' understandability. Herein, the values in the table indicate the average score and standard deviation of each group on the EHR comprehension test.

Table 6 presents the experimental results of the evaluation of the customers' understandability. The results of ANOVA (Cuevas et al., 2004) show that providing medical jargon and the corresponding lay definitions significantly enhances the patients' comprehension of the EHR notes in both groups ($p < 0.01$).

E Algorithm for the Ensemble Model

Algorithm 1 Pseudo code for the weighted voting ensemble prediction

```
Require: Validation Set (V), Test Set (T), Trained Models (M)
 1: $S \leftarrow \varnothing$
 2: for $M_i$ in $M$ do
 3:     $L \leftarrow \varnothing$
 4:     for $V_j$ in $V$ do
 5:         $\mathbf{X}_{V_j}, \mathbf{y}_{V_j} \leftarrow V_j$
 6:         $L \leftarrow L || M_i(\mathbf{X}_{V_j})$
 7:     end for
 8:     $S \leftarrow S || F1(L, V)$
 9: end for
10: $O \leftarrow \varnothing$
11: for $T_j$ in $T$ do
12:     $L \leftarrow \varnothing$
13:     for $M_i$ in $M$ do
14:         $\mathbf{X}_{T_j}, \mathbf{y}_{T_j} \leftarrow T_j$
15:         $L \leftarrow L + S_i \times M_i(\mathbf{X}_{T_j})$
16:     end for
17:     for $L_k$ in $L$ do
18:         $O \leftarrow O || \operatorname{argmax} L_k$
19:     end for
20: end for
21: Return $O$
```

We first evaluated the performance on the validation set of each model and then set this as the weight of each model (line 2 to 11). In the line 1, we set of models' F1 scores S as $\varnothing$. Herein, in line 7, we got the j^{th} sequence of optimal predicted labels of a model ($M_i(\mathbf{X}_{V_j})$) and appended it to the list of predictions L. Then, we calculated $F1$ score of each model (M_i) for the given validation set V in line 8.

In the test set, weighted voting was performed based on the scores of the models (line 11 to 20). In line 15, we got the result of multiplying the model's score by the predicted sequence of labels ($S_i \times M_i(\mathbf{X}_{T_j})$) for a test input ($\mathbf{X}_{T_j}$). After that, the label with the highest weighted score was selected from each token in line 18. Finally, the selected labels O are returned in line 21.

F Impact of the Proposed Methods on BioNER Benchmarks

Setting	Entity Type	Datasets	Vanilla	Binary	TF	MLM	TF+MLM
Pretrained	Disease	NCBI disease	87.92	87.62	88.31	88.24	87.53
		BC5CDR disease	83.93	84.10	84.62	84.74	83.96
	Drug/chem	BC5CDR chemical	92.07	91.58	92.08	92.09	91.64
		BC4CHEMD	90.06	90.30	90.04	89.97	90.14
	Gene/protein	BG2GM	82.30	82.87	82.83	82.81	82.67
		JNLPBA	74.95	78.04	77.51	77.41	77.90
	Species	LINNAEUS	87.59	87.86	85.69	85.81	85.92
		Species-800	74.95	75.11	76.88	77.12	76.67
Wiki-trained	Disease	NCBI disease	**89.21**	85.86	87.90	87.85	**88.24**
		BC5CDR disease	**84.87**	**85.47**	**85.23**	**85.35**	**85.62**
	Drug/chem	BC5CDR chemical	91.88	**92.17**	**92.09**	**92.13**	**91.84**
		BC4CHEMD	**90.27**	90.27	**90.21**	**90.13**	**90.50**
	Gene/protein	BG2GM	**83.06**	**83.23**	**83.07**	**83.26**	**83.17**
		JNLPBA	**77.95**	**78.34**	**77.69**	**78.33**	**77.70**
	Species	LINNAEUS	**89.87**	**88.67**	**89.10**	**88.83**	85.85
		Species-800	74.85	**75.76**	75.97	76.12	75.10

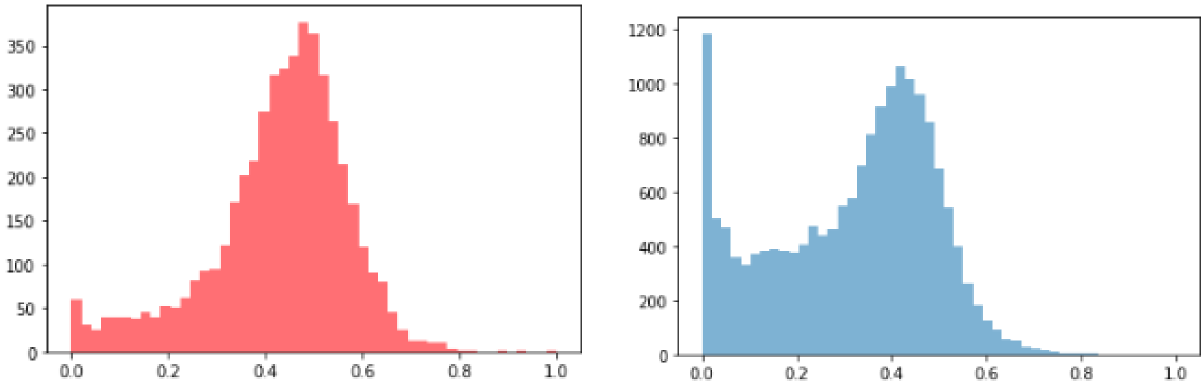
Table 7: Experimental results on proposed methods on BioNER benchmarks. Herein, the values are presented in **bold** if performance is improved in Wiki_setting.

We examined the impact of our suggestions on BioNER benchmarks. Herein, we mainly compare the impact of WikiHyperlink span training method. Furthermore, we kept the hyper-parameters of other experiments and we used the F1 score as the main evaluation criterion.

Table 7 presents the experimental results. We can see that the performances were enhanced in almost all cases. Among 40 experimental settings performances were improved on 30 settings. In addition, when we conducted a t-test on the performance of the experimental settings, we found that the mean and standard deviation of the settings were significantly different ($p < 0.005$). However, in Species-800 and NCBI disease, overall performance marginally decreased after Wiki_training application, and it can be found that the performance marginally changed in the BC4CHEMED data.

On the other hand, additional features did not the affect performance improvement. This is due to the structure of our model. Note that the candidate medical terms are extracted by QuickUMLS and encoded as a binary form (see Appendix B). This can be advantageous in the MedJ task that extracts comprehensive medical terms. However, the BioNER task aims to extract entities of specific semantic types. Therefore, our approach can confuse the NER models. For instance, "Von Willebrand's factor deficiency" is a syndrome name and part of it, "Von Willebrand's factor", is a protein name. In our setting, "Von Willebrand's factor deficiency" is input as a medical jargon. However, if a task is gene/protein name extraction, the input signal can mislead the model. To ameliorate this issue, we can utilize semantic type information from biomedical concept extractors. Specifically, we can use semantic type embedding (Michalopoulos et al., 2021) or a semantic-type span feature (Kwon et al., 2019b) as an additional input.

G Additional Analysis of the MLM Score Feature



(a) MLM scores of UMLS concepts which are jargon

(b) MLM scores of UMLS concepts which are not jargon

Figure 3: Histograms of the MLM score feature. The x-axis is the normalized MLM score, and the y-axis is the number of observations.

In this section, we show that MLM features can be valid for extracted UMLS biomedical concepts. Figure 3 is a histogram of biomedical concepts for MLM scores. In this case, Figure 3(a) is a histogram for medical jargons, and Figure 3(b) is a histogram for non-jargons. As a result of the experiment, we confirmed that, in the case of non-medical jargon, the histogram showed a heavily tailed distribution in the section with the low MLM score. On the other hand, medical jargon was observed relatively infrequently at low MLM scores.

The mean and standard deviation of the jargon's MLM score were 0.43 ± 0.14, and the mean and variance of non-jargon concepts were 0.32 ± 0.17. As a result of performing the statistical test, we can see that the two distributions were significantly different ($p < 0.01$).