Clinical Phrase Mining with Language Models

Kaushik Mani[1,*], Xiang Yue[1,*], Bernal Jimenez Gutierrez[1], Yungui Huang[2], Simon Lin[2], and Huan Sun[1]

[1]Department of Computer Science and Engineering, The Ohio State University, Columbus, OH, USA
[2]Research Information Solutions and Innovation, Nationwide Children's Hospital, Columbus, OH, USA
{mani.46, yue.149, jimenezgutierrez.1, sun.397}@osu.edu
{Yungui.Huang, Simon.Lin}@nationwidechildrens.org
[*]These two authors contributed equally

***Abstract*—A vast amount of vital clinical data is available within unstructured texts such as discharge summaries and procedure notes in Electronic Medical Records (EMRs). Automatically transforming such unstructured data into structured units is crucial for effective data analysis in the field of clinical informatics. Recognizing phrases that reveal important medical information in a concise and thorough manner is a fundamental step in this process. Existing systems that are built for open-domain texts are designed to detect mostly non-medical phrases, while tools designed specifically for extracting concepts from clinical texts are not scalable to large corpora and often leave out essential context surrounding those detected clinical concepts. We address these issues by proposing a framework, `CliniPhrase`, which adapts domain-specific deep neural network based language models (such as ClinicalBERT) to effectively and efficiently extract high-quality phrases from clinical documents with a limited amount of training data. Experimental results on the two real-world clinical datasets: MIMIC-III and i2b2 show that our method can outperform the current state-of-the-art techniques by up to 18% in terms of F_1 measure while being very efficient (up to 48 times faster).[1]**

***Index Terms*—Clinical Phrase Mining, Clinical Text, Clinical NLP, Language Models, BERT**

I. Introduction

Electronic Medical Records (EMRs) contain a wealth of information available as free text. Most of these texts are detailed reports narrated by physicians and care providers, which contain valuable patient information such as problems, medications, vital signs, etc., and give a longitudinal picture of a patient's health. There has been a lot of work on extracting such valuable information from clinical texts such as clinical information extraction [1], [2], question answering [3] and relation prediction [4]–[6]. When converting unstructured texts to structured knowledge (e.g., detecting medical concepts, entities, relations among entities), it is fundamental and among the very first steps to extract high-quality and meaningful phrases from unstructured texts. Therefore, in this paper, we study the task of *phrase mining on unstructured clinical documents* and propose a new effective and efficient method for this task.

Phrase mining is the process of extracting high-quality phrases from a given text corpus. It could help subsequent downstream tasks in the biomedical domain such as summarizing biomedical documents [7], query searching across multiple clinical documents in an EMR system [8], linking phrases to knowledge bases like UMLS (Unified Medical Language System) [9], [10], and predicting relations between two phrases to discover new biomedical knowledge [11], [12].

To the best of our knowledge, there are very few existing tools that are specifically designed for phrase mining in the clinical domain. Baldwin et. al [13] extract several clinical categories at the phrase level, attempting to provide the necessary context while still keeping the extracted elements concise. They employ a three-stage pipeline which extracts categorized phrases of interest using clinical concepts as anchor points. However, their method highly depends on the existence of two concepts in a sentence to extract a span as well as the coverage of knowledge bases.

The other relevant tools available are used for extracting medical concepts from free-text datasets, such as MetaMap [9], KIP [14], cTAKES [10], QuickUMLS [15], SciSpaCy [16], etc. Clinical-domain methods primarily adopt dictionary matching techniques to map a set of candidate terms extracted from a text corpus to a pre-defined ontology and only keep those terms that can be mapped as good concepts. Many efforts have been made for better pre-processing of raw terms and a more accurate mapping process. However, such direct matching methods suffer from scalability issues when processing large datasets. For example, MetaMap generates hundreds of potential mappings for a candidate term which requires a lot of computation. QuickUMLS tries to solve the efficiency problem of MetaMap and cTAKES by employing an approximate dictionary matching algorithm and is up to 135 times faster than both systems. SciSpaCy uses a NER system based on the chunking model from Lample et. al [17] to extract relevant clinical entities.

However, as we will show in the experiments later, the efficiency of QuickUMLS and SciSpaCy is still far from satisfactory on large datasets. Moreover, all the aforementioned concept-mining tools operate on *concept level* and are unable to operate on *phrase level*, which may provide the necessary contextual information for clinical interpretation. For instance, phrases like *'sensitive to levofloxacin'* and *'thrombosis in the right leg'* contain concepts such as *'levofloxacin'* and *'thrombosis'* respectively, but it is necessary to include the

[1]Our source code, pre-trained models and documentations are available online at: https://github.com/kaushikmani/PhraseMiningLM

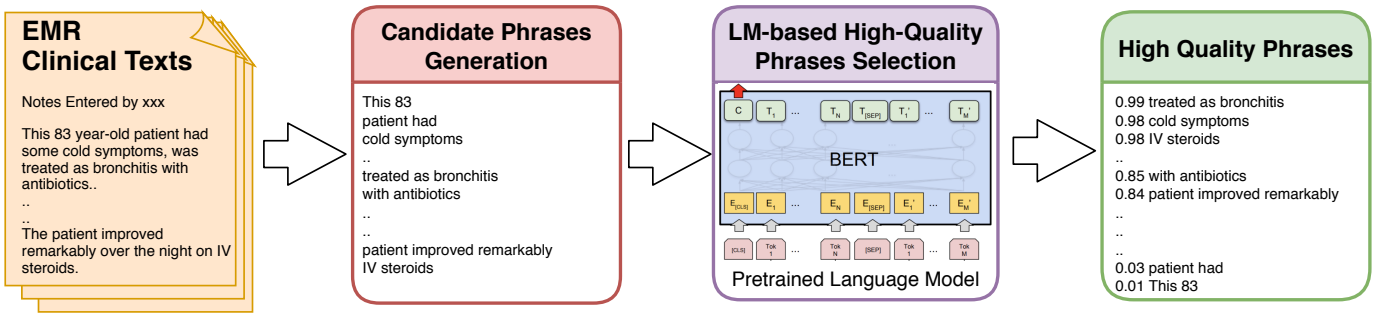

Fig. 1: Overview of our proposed method `CliniPhrase`. Given a set of clinical documents as input, frequent phrases are extracted from the documents as phrase candidates, and features are extracted using a language model for all the phrases to measure the quality of a phrase based on its relevance in the clinical context. These features are then fine-tuned with a classifier, which predicts quality phrases.

contextual information around the concept for the phrases to be helpful to a physician. Besides, many user-generated quality phrases are not available in existing ontologies or knowledge bases, resulting in these methods having a low recall. For example, phrases such as *'verbally responsive'* and *'vitals were stable'* may not be captured by lexical variants of existing concepts in a knowledge base like UMLS [18].

On the other hand, with the availability of large-scale text corpora in the open domain (e.g., Wikipedia articles) as well as the development of statistical machine learning methods, data-driven approaches which employ statistics extracted from a corpus to mine quality phrases, have gained popularity in the phrase mining literature. These methods range from frequency-based methods [19], [20] to more advanced statistical measures derived from a corpus [21], [22]. Most state-of-the-art methods such as TopMine [23], SegPhrase [24], and AutoPhrase [25], are also data-driven methods. TopMine mines significant phrases as part of topical phrase extraction considering frequency and concordance as phrase quality criteria. SegPhrase integrates phrase quality estimation with phrasal segmentation to rectify an initial set of statistical features based on the local context. AutoPhrase further enhances the performance of phrasal segmentation by incorporating the shallow syntactic information in part-of-speech (POS) tags. AutoPhrase does not require any human annotation but instead uses public knowledge bases (e.g., Wikipedia) to train its phrase quality estimator. Open-domain methods, however, do not work well in the clinical domain, because they rely on statistical measures based on open-domain corpora and tend to extract phrases which are not related to the clinical domain (e.g., *'plane rides'* and *'watching television'*) or miss out interesting clinical phrases (e.g., *'walking pneumonia'* and *'malignant melanoma'*) that rarely appear in the open domain.

To overcome the efficiency and low-recall issues of clinical concept extraction tools as well as the aforementioned issues of open-domain phrase mining methods, we propose a new method `CliniPhrase` to effectively and efficiently extract quality phrases from clinical documents. Figure 1 gives an overview of our framework. Given large-scale EMR clinical texts, our framework first extracts candidate phrases based on the assumption that quality phrases should occur with sufficient frequency in a given document collection. We consider two properties: *Downward Closure Property* and *Prefix Property* as constraints for efficient mining of frequent phrases.

Table I: Comparison of existing methods with `CliniPhrase`. *'User-Generated Phrases'* are human written phrases in clinical documents, many of which might not be present in existing ontologies or knowledge bases. *'Clinically-Specific Phrases'* refers to clinical phrases that rarely appear in the open domain.

Methods	Scalability	High recall for User-Generated Phrases	Clinically-Specific Phrases
Medical Concept Mining	✗	✗	✓
Open-Domain Phrase Mining	✓	✓	✗
`CliniPhrase` (Ours)	✓	✓	✓

To further select the most high-quality phrases, in particular, we adapt very recent deep neural network based Language Models (LM), such as BERT [26] and ELMo [27], which are trained on biomedical texts or clinical texts, e.g., BioBERT [28], ClinicalBERT [29], and BioELMo [30] to extract linguistic information and clinical patterns for a candidate phrase, based on which we build a binary classifier to evaluate the quality of the phrase. Experimental results demonstrate that our framework substantially outperforms the state-of-the-art open domain tools as well as those specifically designed for clinical texts. More importantly, our framework is around 48 times more efficient than existing tools in the clinical domain. In addition, we directly apply our model trained on the MIMIC-III [31] dataset to a new corpus (which is unseen in the training phase), i.e., the 2009 i2b2 medication extraction challenge dataset [32] and it still achieves the best performance among all methods, which shows that its greater generalizability can save tremendous labeling effort on the new dataset. We further go on to show the usage of phrases extracted by our model using *query suggestion* as a case study, in which we show similar phrases to a query phrase, and physicians can use those similar phrases together with the query phrase to retrieve more relevant clinical records. To summarize, compared with existing methods (as shown in Table I), our `CliniPhrase` has good scalability, achieves high recall for user-generated phrases, and mostly focuses on clinically-specific phrases.

II. Task Setting

Given a large collection of clinical documents, we aim to extract quality phrases from those documents. The input

documents can be any textual word sequences with arbitrary lengths such as discharge summaries and procedure notes. In general, *a phrase is defined as a sequence of contiguous words appearing in the text, which work together to form a meaningful unit*. In the clinical domain, we additionally require a quality phrase to be *clinically-specific*.

For instance, in the statement *'The patient had no further chest discomfort.'*, *'the patient'*, *'had no'*, *'no further chest discomfort'*, *'chest discomfort'*, etc. are all sequences of words, however *'chest discomfort'* and *'no further chest discomfort'* are the most relevant and useful in the clinical context. We also suggest using the 'longest phrase heuristic' in case of overlapping phrases, since it is better to include as much context as we can. For example, between 'chest discomfort' and 'no further chest discomfort', we suggest using 'no further chest discomfort' as the phrase extracted from this sentence when pursuing downstream tasks.

III. Methods

In this section, we introduce our proposed `CliniPhrase` framework (as shown in Figure 1), which consists of two major components: Candidate Phrase Generation and High-quality Phrase Selection. The former aims to generate frequent word sequences as candidate phrases. We set a minimum frequency and maximum sequence length threshold to filter a number of candidate phrases. Then the High-quality Phrase Selection module is responsible for keeping only high-quality ones using pre-trained neural models. We will now introduce the two modules in detail.

A. Candidate Phrase Generation

Given a document (word sequence) we first aim to extract candidate phrases from the sequence. We claim that any length of contiguous words appearing in the text could be considered as candidate phrases.

To achieve this, we split the sentences in the documents into words and expand contiguous words (i.e., n-gram) in sentences to generate phrase candidates. However, such a generation process is likely to include many low-quality phrases and it is very inefficient if we consider a larger n.

We observe that a quality phrase often occurs with sufficient frequency in a given document collection. Thus any phrase that is not frequently occurring in the entire collection is less likely to be important. So given a corpus, we can accelerate the aforementioned process and improve the generated phrase quality by introducing two constraints: a minimum frequency threshold γ and a maximum phrase length λ. We further introduce the following two properties to help us efficiently mining of frequent phrases.

• **Downward Closure Property:** If a phrase is not frequent, then any of its super-phrases will not be frequent. Therefore, those longer phrases will be filtered and never expanded.

• **Prefix Property:** If a phrase is frequent, any of its prefix units should be frequent too. All frequent phrases are generated by expanding their prefixes.

In addition, phrases that end with stop words are not likely to have a complete semantic meaning. For example, phrases like *'patient is'* and *'heartbeat was'* might occur frequently but are not meaningful phrases. Therefore, we filter these phrases from the list of candidate phrases based on stop words from NLTK [33].

B. High-quality Phrase Selection

Even though we use some constraints in the candidate phrase generation step, many meaningless or irrelevant phrase units are bound to be included. Thus, we leverage pre-trained language models to further filter low-quality phrases and keep high-quality ones. We first introduce the pre-trained models we use.

Pre-trained word representations have become a key component in a number of neural language understanding models. Methods like Word2Vec [34], Glove [35] enabled us to use a vector to properly represent words in a way that captures semantic as well as syntactic relationships.

However, the meaning of a word should be context-dependent, i.e., their embeddings should also take context into account. This necessitated the use of a language model to obtain embeddings for individual words while taking the entire sentence or paragraph into account. Language modeling is a fundamental task in natural language processing which estimates how likely a sequence of words is to appear in a text corpus. Recently, language models have not only proven very effective at automatically capturing a vast range of features from the text but are also able to extract different features from the same word depending on the word's context (i.e., contextualized word representations).

Earlier neural network based language models used recurrent and long short-term memory (LSTM) [36] neural networks. Both recurrent and LSTM networks receive inputs from a sequence one element at a time and produce new hidden states every time which depends on the input and the previous hidden state, making them particularly well suited for sequential data modeling. They encode all previous information up to time t in the hidden state h_t, thereby allowing us to use the hidden states of the language model as features.

More recently, significant improvements have been made in various NLP benchmarks, owing to more complex pre-trained language models that can more effectively capture contextualized information from text such as ELMo [27] and BERT [26]. Embeddings from Language Models (ELMo) [27] use a bidirectional LSTM trained with a language model objective on a large text corpus. ELMo representations are obtained by using a linear combination of the hidden states in a deep language model, which improves the performance of several downstream tasks over just using the top LSTM layer. The main goal of a forward language model is to predict the probability of the next word in a sequence given the previous words in the sequence. Given a historical sequence $w_{1:t} = [w_1, ..., w_t]$ and a fixed vocabulary V, we can get the

distribution over w_{t+1} by applying an affine transformation to the hidden layer of RNN followed by a softmax.

$$P(w_{t+1} = j|w_{1:t}) = \frac{\exp(W_{hy}p^j + b_y)}{\sum_{j' \in V} \exp(W_{hy}p^{j'} + b_y)} \quad (1)$$

In the above equation, W_{hy} is a parameter to be learned, p^j is the embedding at the output layer of LSTM for the token j. The objective of the language model is to minimize the negative log-likelihood loss (NLL) of the training sequence.

$$NLL = -\sum_{t=1}^{T} \log P(w_t|w_{1:t-1}) \quad (2)$$

A backward language model is similar to a forward language model, except it runs over the sequence in reverse, predicting the previous token given the future context.

In order to address the long-range dependencies and parallelization issues present in RNN based architectures, the Transformer model [37] uses an attention mechanism [38] which allows the model to focus on relevant information in the input sequence depending on which word is being processed. Transformers use the Scaled Dot-Product Attention [37], which is computed according to the following equation:

$$Attention(Q, K, V) = \text{softmax}(QK^T/\sqrt{d_k})V \quad (3)$$

where Q is the matrix of queries packed together and K and V are the matrices of keys and values packed together, d_k represents the dimensionality of the queries and keys. The Transformer uses the basic encoder-decoder architecture composed of N blocks. The optimization problem is made easier by using layer normalizations and residual connections. Since attention cannot utilize the position of the inputs, the transformer adds explicit position embeddings to the input embeddings. We refer the readers to Vaswani et. al [37] for more details on the Transformer.

BiDirectional Encoder Representations for Transformer (BERT) [26] uses the Transformer architecture to train a language model by taking both the previous and next tokens into account when predicting output. BERT achieves this by using a concept known as Masked Language Modeling. This involves randomly replacing a set percentage of words with a special `[MASK]` token and to require the model to predict the masked token. BERT pre-trained on a large corpus can be fine-tuned by using encoder hidden states as features and feeding it to an arbitrary classifier, which can be used to predict the final output.

We showcase the capability of multiple pre-trained language models in phrase mining, from simple LSTM-based language models such as ELMo to transformer-based language models such as BERT. *We input each candidate phrase to a language model and use the average of the hidden state vector on each word as its **feature***. In particular, we exhibit the advantage of using biomedical versions of pre-trained language models such as BioELMo and BioBERT (pre-trained on PubMed abstracts) as well as ClinicalBERT (pre-trained on the clinical notes of MIMIC-III dataset [31]), which capture domain-specific characteristics of clinical texts. The training and implementation details of different models are given in Section IV-C.

We use the logistic regression based binary classifier for the simple LSTM-based language model, which classifies a phrase as good/bad based on the features extracted. For transformer-based and ELMo-based language models, we fine-tune all pre-trained parameters on the phrase mining task.

IV. Experiments

In this section, we systematically evaluate different language models on mining quality phrases from clinical documents and compare their performance with other methods.

A. Datasets

To evaluate our framework, we use the discharge summaries available in the MIMIC-III dataset [31]. It consists of 59652 discharge summaries comprising of de-identified health data of 40000 critical care patients associated with Beth Israel Deaconess Medical Center. The database is populated with data from several sources including hospital electronic health record databases and archives from critical care information systems. We also consider the generalizability test (i.e., train the model on one dataset and test it on another unseen dataset, see Section V-D) and thus also use the 2009 i2b2 medication extraction challenge [32] dataset. The i2b2 dataset is a collection of fully de-identified discharge summaries from Partners Healthcare, and comprises of clinical notes of a different set of patients.

We rely on human evaluators to label the quality of candidate phrases. More specifically, we randomly sample $2,000$ candidate phrases which are extracted in Section III-A. To make a fair comparison, we ensure that the sampled phrases are extracted by AutoPhrase [25] as well, since it is the state-of-the-art phrase mining tool. These phrases are evaluated by 3 expert reviewers independently. By the rule of majority voting, phrases that receive at least two positive annotations are considered as quality phrases, otherwise they are treated as bad phrases. We use 1500 phrases for training, 200 for validation and 300 phrases for testing. For the i2b2 dataset, we label 200 randomly sampled phrases for generalizability testing purposes. We find that out of these 200 phrases, 64 phrases are completely unseen in the MIMIC-III dataset, further indicating the difference between these two datasets.

B. Compared Methods

We compare our models with different methods from both the open domain and the clinical domain.

• **AutoPhrase** [25] is the state-of-the-art phrase mining tool that combines statistical features such as frequency, inverse document frequency and KL divergence along with POS-based phrasal segmentation to filter phrases. It does not require human annotation and uses public knowledge bases for training its classifier. We use the Medical Subject Headings (MeSH) [39] as a quality knowledge base for AutoPhrase.

• **SegPhrase** [24] is similar to AutoPhrase and requires limited training data. We provide 300 labeled phrases (the number suggested by SegPhrase) to train the model.
• **QuickUMLS** [15] is an unsupervised, approximate dictionary matching algorithm which extracts medical concepts in UMLS metathesaurus from clinical texts. Soldaini et. al [15] show that QuickUMLS achieves a similar performance as MetaMap [9] and cTAKES [10], but is up to 135 times faster.
• **SciSpaCy** [16] is a library containing models for processing biomedical and clinical text. We used the NER model in SciSpaCy which is trained on mention spans from the MedMentions dataset [40].

We then try different pre-trained models to extract features to instantiate our `CliniPhrase` framework.
• **Word2Vec** model [34] is trained using the CBOW method on the MIMIC-III dataset and phrase embeddings are obtained as the average of word embeddings.
• **Sent2Vec** [41] is trained in a similar manner on the MIMIC-III dataset to obtain the embedding of phrases, where the phrase embedding is defined as the average of word embeddings and the word n-gram embeddings in a phrase.
• **ELMo** [27] is a deep contextualized word representation learned from functions of internal states of a deep bidirectional language model which is pre-trained on the 1 Billion Word Benchmark, approximately 800M tokens of news crawl data from WMT 2011.
• **BioELMo** [30] is a biomedical version of embeddings from language model (ELMo), pre-trained on PubMed abstract.
• **BERT** [26] is a base BERT model which learns word representations from a deep bidirectional transformer encoder and is pre-trained on Wikipedia and the book corpus for a very long time.
• **BioBERT** [28] is a pre-trained BERT model that is initialized with weights from BERT trained on the general domain corpora (Wikipedia and book corpus) and then pre-trained on PubMed abstracts and PMC full-text articles.
• **ClinicalBERT** [29] includes multiple BERT models which are: (1) first initialized with weights from either BERT trained on general domain corpora (Wikipedia and book corpus) or BioBERT, and (2) then pre-trained on discharge summaries of MIMIC-III dataset or all clinical notes of MIMIC-III dataset.

Note that SegPhrase, QuickUMLS, and SciSpaCy do not extract all phrases in our testing set and we assume that these tools treat them as bad phrases. As for ClinicalBERT, even if it was pre-trained on MIMIC-III, the whole pre-training process was in an unsupervised manner (i.e., MASK Language Modeling and Next Sentence Prediction, see [29] for more details). Therefore, there are no information leaking issues between pre-training and phrase mining.

C. Implementation details

Our framework is implemented in Pytorch 1.0 [42] and the models are trained with 1 NVIDIA Tesla P100 GPU which is provided by Ohio Supercomputer Center [43]. We set the default value of minimum frequency threshold γ as 10 and maximum phrase length λ as 6, which are the parameters required for frequent phrase mining. We preprocess the dataset by removing the de-identified text patterns in them, and then tokenize the text into sentences and words, and convert the words to lowercase. For transformer and ELMo based language models, we initialize the pre-trained models and fine-tune them with a simple linear classifier according to the phrase mining task and then use the model to filter the phrases. We split the labeled data into training and validation sets with a split ratio of 0.8. The classifier is trained for 100 epochs with an initial learning rate of 0.005. We decrease the learning rate by half if the validation loss does not decrease for more than 10 epochs and use early stopping on validation loss. For the simple LSTM-based language model, we use the PubMed Phrases dataset [44] to pre-train the language model. Then a logistic regression based classifier is trained with similar hyperparameters as mentioned above. We set the threshold score at 0.5 for the classifier to obtain quality phrases. All the experiments are performed on 10 different random states and we show the averaged results to prevent bias towards any particular initialization or train/test set.

D. Evaluation metrics

To evaluate the performance of different methods on the annotated data, we use accuracy, precision, recall and F1 score as our experiment metrics:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Here TP (True Positive) is the number of outcomes where the model correctly predicts the positive class. TN (True Negative) is the number of outcomes where the model correctly predicts the negative class. FP (False Positive) is the number of outcomes where the model incorrectly predicts the positive class. FN (False Negative) is the number of outcomes where the model incorrectly predicts the negative class.

V. Results

A. Overall Performance

The overall results on the MIMIC-III dataset are shown in Table II. We can see that the language models substantially outperform all the compared methods in all evaluation metrics. BERT language models perform better than simple LSTM-based language models and ELMo-based language models. ClinicalBERT initialized with BioBERT weights and pre-trained on discharge summaries of MIMIC III notes and then fine-tuned to the phrase mining task performs the best among all models, however, the performance of other BERT based models are comparable.

Table II: Comparison of different methods on the MIMIC-III dataset

Method Type		Method	Accuracy	Precision	Recall	F1
Baselines	Open Domain	AutoPhrase	62.92	63.06	63.04	62.90
		SegPhrase	61.64	62.10	61.88	61.50
	Clinical Domain	QuickUMLS	55.36	68.77	56.65	47.50
		SciSpaCy	62.08	64.41	62.79	61.17
Ours	Word Embedding	word2vec	57.94	59.24	57.14	54.94
		sent2vec	72.42	72.47	72.32	72.28
	Pre-trained Language Model	Simple LSTM	71.14	71.24	71.00	70.95
		ELMo	73.92	75.42	74.04	73.53
		BioELMo	78.12	79.19	77.61	77.63
		BERT	81.37	79.99	80.00	79.99
		BioBERT	78.60	78.84	78.73	78.59
		ClinicalBERT-(BERT+All MIMIC)	80.00	81.24	80.31	79.89
		ClinicalBERT-(BERT+MIMIC Discharge)	81.00	80.99	81.02	80.99
		ClinicalBERT-(BioBERT+All MIMIC)	81.00	80.98	80.99	80.98
		ClinicalBERT-(BioBERT+MIMIC Discharge)	**82.00**	**82.11**	**82.09**	**81.99**

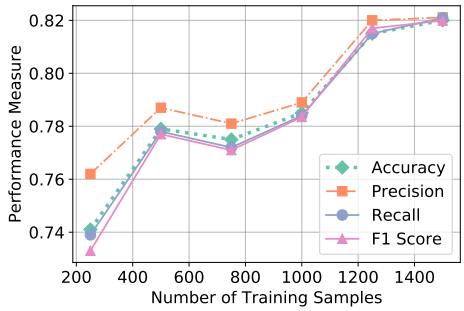

Fig. 2: Impact of training data on performance of ClinicalBERT (BioBERT + MIMIC III Discharge Summaries). The performance improves with increase in amount of training data.

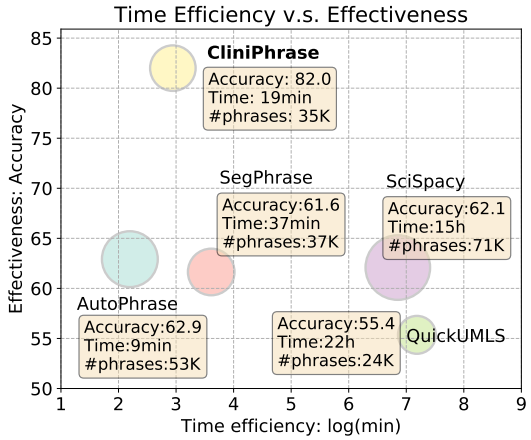

Fig. 3: Efficiency v.s. Effectiveness of different methods. The size of bubbles represents the number of phrases extracted by each method.

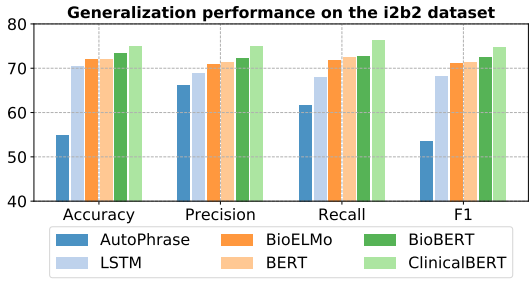

Fig. 4: Generalizability performance of various methods, which are trained on MIMIC-III dataset but evaluated on i2b2 dataset.

B. Effect of Training Set Size

We also study the impact of the size of the training data on our framework. The performance of the ClinicalBERT (BioBERT + MIMIC III Discharge Summaries) on the MIMIC-III testing dataset under different amount of labeled data are shown in Figure 2. From these results, we observe that the performance of the model becomes better as the number of training samples increases, and is already satisfactory when there are around 1200 training samples.

C. Efficiency v.s. Effectiveness

We apply each trained method to discharge summaries in MIMIC-III and compare their efficiency and effectiveness. Fig 3 summarizes the results, where we do not consider the pre-training time for the language models, since they can be pre-trained offline and are ready to be reused on a new dataset.

As shown in Fig 3, our method using the language model (ClinicalBERT) is much faster than methods in the clinical domain. For instance, our method is around 48 times faster than SciSpaCy. AutoPhrase, on the other hand, uses multi-threading techniques and various optimized methods, which makes their framework considerably faster. Besides, the bubble size represents the number of phrases that are extracted from the MIMIC-III documents. Our model can extract roughly the same number of phrases as SegPhrase, larger than QuickUMLS, and less than Scispacy and AutoPhrase. However, when considering the phrase quality (effectiveness), our framework archives much better performance. Thus, our `CliniPhrase` can extract a considerable amount of phrases and achieve very good accuracy while being efficient.

Fig. 5: Phrase clouds: top similar phrases given the query phrase "brainstem stroke", "cortical destruction" and "abnormal bleeding" respectively. The size of each phrase represents how similar it is to the query phrase.

Table III: Given a query phrase, we find the top results of similar phrases extracted by different models.

Query Phrase	Method	Suggested Similar Phrases
"abnormal bleeding"	AutoPhrase	abnormal uterine bleeding, variceal bleeding, bleeding diathesis, abnormal enhancement, abnormal rhythms
	SciSpaCy	bleeding diveriticulus, fulgeration bleeding, gi/rp bleeding, gastrointestal bleeding, dental/gum bleeding
	CliniPhrase	excessive bleeding, unusual bleeding, bleeding disorders, increased bleeding, persistent bleeding

D. Generalizability of the Fine-tuned Language Model

In this section, we conduct an experiment to show the generalizability of different methods to a new dataset without additional human-annotated data. We perform our experiment by using the MIMIC-III dataset to train those methods and then applying them to the 2009 i2b2 medication extraction challenge dataset.

The results of the different models are in Figure 4. We only include AutoPhrase as the baseline in this experiment as it archives the best performance in Table II. We indicate that our method substantially outperforms AutoPhrase and shows better generalizability. Our method can be directly applied to a different clinical corpus without additional training data, thereby indicating the advantage of using language models for phrase mining in the clinical domain.

Experiment Summary. We observe that language models substantially outperform all the compared methods by extracting higher-quality phrases and are around 48 times faster than SciSpaCy. We also show that our trained model can be readily applied to a new clinical text dataset, clearly indicating its advantage over other methods.

VI. Case Study

To further demonstrate the usefulness of our method in the real-world scenario, we conduct case studies showing the most similar phrases extracted phrases given a query phrase. In the real case, physicians could use the most similar suggested phrases to find more relevant information in a huge database of clinical records, such as finding patients with similar symptoms, their treatments, etc. We take a phrase as an input and use cosine similarity measure based on ClinicalBERT contextualized embeddings to find similar high-quality phrases that were extracted from the MIMIC III clinical notes.

In Fig 5, we show the top 20 similar phrases in word clouds given the query phrase "brainstem stroke", "cortical destruction" and "abnormal bleeding" respectively. It can be seen that most of the suggested phrases are semantically-correct, logically-coherent, and clinically-specific. More importantly, they are also very relevant to the query phrase, which can be used to help other clinical tasks (e.g., clinical record retrieval).

We also show the top 5 similar phrases extracted by the other two baselines given the query phrase "abnormal bleeding" in Table III. Since AutoPhrase does not produce any embeddings, so we use simple TF-IDF [45] of phrases to calculate the cosine similarity. SciSpaCy uses word embeddings trained from PubMed abstracts [46] to perform NER, so we use the average of word embeddings to calculate their phrase embeddings.

We observe that results obtained from both AutoPhrase and SciSpaCy tend to concentrate on individual words and find similar suggestions to the words rather than the entire phrase. For example, the query *'abnormal bleeding'*, when used with AutoPhrase, tends to produce phrases that are more associated with the word *'abnormal'* such as *abnormal enhancement, abnormal uterine bleeding* and *abnormal rhythms*. SciSpaCy tends to concentrate on the word *'bleeding'*, producing phrases such as *fulgeration bleeding, dental/gum bleeding, etc*. On the other hand, ClinicalBERT produces phrases such as *excessive bleeding, unusual bleeding, increased bleeding*, which are all similar to the complete phrase *'abnormal bleeding'*. This clearly indicates the advantage of using language models and contextualized embedding. Besides, it also shows the quality as well as usage of the phrases extracted by our method.

VII. Conclusion

This paper introduces an effective and efficient method to extract quality phrases from clinical documents. We employ pre-trained deep neural network based language models and fine-tune them to extract quality phrases from the given clinical documents with limited human annotations. We conduct extensive experiments and show that our framework performs substantially better than state-of-the-art techniques including those in the clinical space. Our framework is also scalable to large corpora, and is around 48 times faster than SciSpaCy, a popular tool for extracting clinical entities. We show that our trained model can be readily applied to a new clinical

text dataset, thereby reducing the need for expensive human labeling. We also demonstrate potential usages of the extracted phrases, e.g., using them to suggest similar phrases to a query phrase and hence helping physicians retrieve relevant information from documents. In the future work, we will develop an online tool based on our method to make it more convenient to extract clincial phrases.

REFERENCES

[1] H. Xu, S. P. Stenner, S. Doan, K. B. Johnson, L. R. Waitman, and J. C. Denny, "Application of information technology: Medex: a medication information extraction system for clinical narratives," *JAMIA*, vol. 17 1, pp. 19–24, 2010.

[2] B. De Bruijn, C. Cherry, S. Kiritchenko, J. Martin, and X. Zhu, "Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010," *JAMIA*, vol. 18, no. 5, pp. 557–562, 2011.

[3] X. Yue, B. J. Gutierrez, and H. Sun, "Clinical reading comprehension: A thorough analysis of the emrqa dataset," in *ACL*, 2020.

[4] X. Yue, Z. Wang, J. Huang, S. Parthasarathy, S. Moosavinasab, Y. Huang, S. M. Lin, W. Zhang, P. Zhang, and H. Sun, "Graph embedding on biomedical networks: methods, applications and evaluations," *Bioinformatics*, vol. 36, no. 4, pp. 1241–1251, 2020.

[5] Z. Wang, X. Yue, S. Moosavinasab, Y. Huang, S. Lin, and H. Sun, "Surfcon: Synonym discovery on privacy-aware clinical data," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1578–1586.

[6] Z. Wang, J. Lee, S. Lin, and H. Sun, "Rationalizing medical relation prediction from corpus-level statistics," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[7] R. Mishra, J. Bian, M. Fiszman, C. R. Weir, S. Jonnalagadda, J. Mostafa, and G. Del Fiol, "Text summarization in the biomedical domain: a systematic review of recent research," *Journal of biomedical informatics*, vol. 52, pp. 457–467, 2014.

[8] S. Schulz, P. Daumke, P. Fischer, and M. Müller, "Evaluation of a document search engine in a clinical department system," *AMIA Symposium*, vol. 2008, pp. 647–51, 02 2008.

[9] A. R. Aronson, "Effective mapping of biomedical text to the umls metathesaurus: the metamap program," in *Proceedings of the AMIA Symposium*, 2001, p. 17.

[10] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. K. Schuler, and C. G. Chute, "Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications," *JAMIA*, vol. 17 5, pp. 507–13, 2010.

[11] B. Rink, S. M. Harabagiu, and K. Roberts, "Automatic extraction of relations between medical concepts in clinical texts," *JAMIA*, vol. 18 5, pp. 594–600, 2011.

[12] X. Lv, Y. Guan, J. Yang, and J. Wu, "Clinical relation extraction with deep learning," *International Journal of Hybrid Information Technology*, vol. 9, no. 7, pp. 237–248, 2016.

[13] T. Baldwin, Y. Guo, V. Mukherjee, and T. Syeda-Mahmood, "Generalized extraction and classification of span-level clinical phrases," *Proceedings of the AMIA Symposium*, vol. 2018, pp. 205–214, 12 2018.

[14] Q. Li and Y.-F. B. Wu, "Identifying important concepts from medical documents," *Journal of biomedical informatics*, vol. 39, no. 6, pp. 668–679, 2006.

[15] L. Soldaini and N. Goharian, "Quickumls : a fast , unsupervised approach for medical concept extraction," 2016.

[16] M. Neumann, D. King, I. Beltagy, and W. Ammar, "Scispacy: Fast and robust models for biomedical natural language processing," 2019.

[17] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in *Proceedings of NAACL'16*, Jun. 2016, pp. 260–270.

[18] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D267–D270, 2004.

[19] H. Ahonen, "Knowledge discovery in documents by extracting frequent word sequences," *Library Trends*, vol. 48, 1999.

[20] A. Simitsis, A. Baid, Y. Sismanis, and B. Reinwald, "Multidimensional content exploration," *PVLDB*, vol. 1, pp. 660–671, 2008.

[21] C. Ramisch, A. Villavicencio, and C. Boitet, "Multiword expressions in the wild?: the mwetoolkit comes in handy," in *Proceedings of COLING*, 2010, pp. 57–60.

[22] M. Danilevsky, C. Wang, N. Desai, X. Ren, J. Guo, and J. Han, "Automatic construction and ranking of topical keyphrases on collections of short documents," in *Proceedings of the 2014 SIAM International Conference on Data Mining*. SIAM, 2014, pp. 398–406.

[23] A. El-Kishky, Y. Song, C. Wang, C. R. Voss, and J. Han, "Scalable topical phrase mining from text corpora," *Proceedings of the VLDB Endowment*, vol. 8, no. 3, pp. 305–316, 2014.

[24] J. Liu, J. Shang, C. Wang, X. Ren, and J. Han, "Mining quality phrases from massive text corpora," in *Proceedings of the SIGMOD*. ACM, 2015, pp. 1729–1744.

[25] J. Shang, J. Liu, M. Jiang, X. D. Ren, C. R. Voss, and J. Han, "Automated phrase mining from massive text corpora," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, pp. 1825–1837, 2017.

[26] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *Proceedings of NAACL'19*, 2019.

[27] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proc. of NAACL*, 2018.

[28] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 09 2019.

[29] E. Alsentzer, J. R. Murphy, W. Boag, W. Weng, D. Jin, T. Naumann, and M. B. A. McDermott, "Publicly available clinical BERT embeddings," *CoRR*, vol. abs/1904.03323, 2019.

[30] Q. Jin, B. Dhingra, W. W. Cohen, and X. Lu, "Probing biomedical embeddings from language models," *CoRR*, vol. abs/1904.02181, 2019.

[31] A. E. Johnson, T. J. Pollard, L. Shen, H. L. Li-wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific data*, vol. 3, p. 160035, 2016.

[32] Ö. Uzuner, I. Solti, and E. Cadag, "Extracting medication information from clinical text," *JAMIA*, vol. 17 5, pp. 514–8, 2010.

[33] E. Loper and S. Bird, "Nltk: the natural language toolkit," *Proceedings of the ACL-02 Workshop on Effective tools and methodlogies for teaching natural language processing and computational linguistics*, 2002.

[34] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[35] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *EMNLP*, Oct. 2014.

[36] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[38] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014.

[39] C. E. Lipscomb, "Medical subject headings (mesh)," *Bulletin of the Medical Library Association*, vol. 88, no. 3, p. 265, 2000.

[40] S. Mohan and D. Li, "Medmentions: A large biomedical corpus annotated with UMLS concepts," *CoRR*, vol. abs/1902.09476, 2019.

[41] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised learning of sentence embeddings using compositional n-gram features," *arXiv preprint arXiv:1703.02507*, 2017.

[42] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[43] O. S. Center, "Ohio supercomputer center," 1987. [Online]. Available: http://osc.edu/ark:/19495/f5s1ph73

[44] S. Kim, L. Yeganova, D. C. Comeau, W. J. Wilbur, and Z. Lu, "Pubmed phrases, an open set of coherent phrases for searching biomedical literature," *Scientific data*, vol. 5, p. 180104, 2018.

[45] G. Salton and M. J. McGill, "Introduction to modern information retrieval," 1986.

[46] S. Pyysalo, F. Ginter, H. Moen, T. Salakoski, and S. Ananiadou, "Distributional semantics resources for biomedical text processing," in *Proceedings of LBM 2013*, 2013, pp. 39–44.