Multi-class Clinical Text Annotation and Classification using BERT-based Active Learning

Muhammad Afzal[a], Jamil Hussain[b], Asim Abbas[c], Maqbool Hussain[d]

[a]*Dept. of Software, Sejong University, Seoul 05006, South Korea, mafzal@sejong.ac.kr*
[b]*Dept. of Data Science, Sejong University, Seoul 05006, South Korea, jamil@sejong.ac.kr*
[c]*Dept. of Computer Science, ST. JHON'S University, USA, abbasa@stjohns.edu*
[d]*School of Computing and Engineering, University of Derby, UK, M.Hussain@derby.ac.uk*

Abstract

Data-driven approaches require labeled data for autonomous decision-making, but medical data is recorded in structured and unstructured form mostly with no label specified. Manual labeling is an arduous task, and variations in the notes at different levels, including section, sentence, and terms, add to the challenge. Creating well-structured and accurate data to use as a training dataset for a machine learning (ML) model increases the chance of getting meaningful results from the ML models. This paper proposes a multi-class annotation and classification framework that uses four classes: subjective, objective, assessment, and plan, adopted from a well-known medical protocol called SOAP. The SOAP classes capture a clinical context and help reduce misinterpretations when a concept of a similar type is used in different sections of the same clinical document. To illustrate the effectiveness of the proposed methodology, we conduct a set of experiments on clinical notes acquired from a public dataset (i2b2). We observe that our proposed model has eased the onerous job of annotation and achieved a better performance than competitors. The encouraging results of this research demonstrate the potential of combining active learning, transfer learning, and deep learning for automatic annotation to get accurate text classification.

Keywords:

Text Classification, Text Annotation, Active Learning, Transfer Learning, Deep Learning, BERT, Clinical Text, SOAP

1. Introduction

In today's world, patient data is logged into an electronic health record (EHR) system in both structured and unstructured formats [1]. The unstructured form mainly includes clinical notes, discharge summaries, and diagnostic test reports written in natural language. These reports contain vital information that might help solve clinical questions about patient health conditions, clinical reasoning, and inferencing. However, due to the time limitation, physicians have difficulty examining the unstructured information at the point of care [2].

Traditionally, clinically relevant information from clinical documents is extracted through manual methods with the support of clinical domain experts, which creates hurdles in terms of scalability and costs. At the same time, data availability allows researchers to execute automated algorithms extracting helpful information for efficient disease care [3]. Natural language processing (NLP) shows a significant role in the clinical domain for various applications such as medical concept identification in different clinical documents [4]. Recently, NLP applications are further diversified to use for disease outbreak detection, conversion of free text to structured features for decision support, answering clinical questions, and accessing knowledge embodied in free-text clinical and biomedical resources [5].

The information extraction facilitated with NLP led to automated clinical text classification in clinical predictive analytics that emerges with the huge creation of clinical notes and speedily growing adoption of EHR systems [6]. Two types of techniques: symbolic and statistical machine learning, are commonly used for clinical text classification tasks [7]. Symbolic techniques are used in applications that involve hand-crafted rules by domain experts like logic rules and regular expressions. Although rule-based methods have been shown effective in the clinical domain because of sublanguage properties, it can be laborious to develop a system that requires collaboration between technical NLP experts and clinical domain experts. Moreover, the final applications may have limitations

of portability and generalization beyond the scenario for which it was intended [8].

Machine learning (ML) methods have been proven to be efficient for the tasks of clinical text classification. However, an effective supervised ML model still needs human involvement to annotate a huge set of training data. The efforts by domain experts to unstructured label data is a significant blockade of inefficient data analysis [9]. The annotation problem is of primary focus in the medical domain because of the lack of clinical data available to the public and expert knowledge for accurate annotations. The other popular methods, such as crowdsourcing, are not suitable for creating labeled clinical training data because of the sensitive nature of the domain. Also, the findings of a systematic review[9] show that most datasets used in training ML models for text classification consist of mere hundreds or thousands of records because of annotation blockade.

The manual annotation process issues have been resolved by modern orthogonal approaches such as active learning (AL) and transfer learning (TL) are utilized as machine-assisted pre-annotations [10]. AL provides a subset of high-value training samples by reducing the huge data required for labor-intensive data annotation without losing the quality[11]. The selection of samples is iterative as to start with a high-quality manually annotated subset of samples to automatically generate another subset of annotations, thus increasing the subset to annotated text to use in the subsequent iterations of the process [10]. AL approaches have been applied in a clinical domain to decrease labor-intensive data annotation burden and enhance the model classification performance with a few labeled examples sets [11, 12]. For instance, Li, Muqun, et al. [13] have used AL to reduce annotation requirements in the de-identification workflow by incorporating real clinical trials and i2b2 datasets to show e improved performance of trained models compared to the traditional passive learning framework.

Similarly, Tomanek and Hahn [14] examined the impact of AL in decreasing the time required for data annotation for entities (person, organization, and location) extraction. They noticed that the AL process significantly decreases

up to 33% data annotation time and cost compared to baseline. Chen [15] conducted a simulation experiment to re-annotate a subset of the i2b2/VA 2010 dataset from the concept extraction challenge. Their results showed that the AL-based query strategy reduced the volume of data needed for manual annotation compared to baseline.

AL is used in other domains such as sentiment analysis [15], where the authors proposed a novel Active Deep Network (ADC) to solve the problem of the small dataset in the sentiment classification problem. In another study by Hajmohammaadi et al. [16], they used AL and self-training for cross-lingual sentiment classification and other baseline models to check the effectiveness of their proposed model; their finding shows that AL performance better as compared to baseline models.

In addition to AL, researchers have used TL to learn knowledge from previously learned domains and apply it to newer domains and tasks. Most real-world applications suffer from data deficiency that results in sub-optimal models based on deep learning approaches. TL is touted to address this issue by allowing pre-trained models from domain A to be applied to tasks in another domain B; both A and B are related domains. TL is the dominant approach leveraged by leading language models such as RNNs, LSTMs, and transformer-based language (TBL). These models can be used for any downstream task, language, or domain. The TBL models perform betters on various NLP tasks as compared with other models. In modern NLP techniques, the researcher combines transfer learning methods with large-scale TBL models for achieving better performance. The existing language models based on RNNs, and LSTMs suffer the vanishing gradient problem and cannot handle the longer contextual dependencies. The LSTMs based models such as ELMO (Embeddings from Language Model) or ULMFiT (Universal Language Model FIne-Tuning) are still used for modern NLP tasks. Still, the main limitations of LSTMs based models are challenging to train in a parallel way. The transformer architecture resolves these issues by an attention mechanism, which creates an entire sequence from the whole document and trains the model in a parallel fashion. Various TBL models with slight

differences exist for modern NLP tasks, but the performance of BERT (Bidirectional Encoder Representations from Transformers)-based models is exceptional [17]. According to [18], the SciBERT outperforms the baseline BERT model on biomedical tasks. SCI-BERT is a deep learning-based language model that uses the original BERT model code, trained on scientific articles for the biomedical domain. In recent times, we see a growing amount of biomedical data available in textual form. Substantial advances in the development of pre-training language representation models provide an opportunity for a range of biomedical domain tasks such as pre-trained word embeddings, sentence embeddings, and contextual representations.

This study proposes a methodology for clinical text classification by combining AL and TL learning approaches to minimize human efforts in creating labeled data. The proposed methodology employed a rule-based NLP algorithm based on a lexical approach that automatically annotates the unlabeled input data to create an initial seed dataset. Using the initially labeled dataset, we design an AL approach by training an ensemble learning classifier. The AL output is used to train the proposed text classification model by employing the pre-trained model for feature encoding to classify texts in the biomedical documents into four classes of SOAP (subject, object, assessment, plan) protocol. SOAP is a well-known structure used for patient information organized into four logical compartments.

To demonstrate the usefulness of the proposed methodology, we conduct a set of experiments on clinical notes acquired from a public dataset (i2b2/VA 2010). The findings of the proposed approach indicate a significant reduction in annotation costs by achieving a higher accuracy compared to the existing approaches used for the same task in the past. Furthermore, our approach is unique by applying novel AL methodology enhanced with TL for embeddings to perform text classification tasks using an attention-based deep learning model. This approach is different from traditional NLP approaches in terms of context capturing within SOAP sections. For instance, a medication x may appear in a clinical note in two different forms; x is used currently and x is prescribed for the

future use. Here identifying medication names correctly is not sufficient, but the context is important too. Identifying SOAP sections clearly differentiate between the x medication as; currently in use (subjective) and prescribed for the future (plan).

Our proposed approach provides an end-to-end solution involving clinical text annotation and classification tasks. It exhibits usefulness in different NLP tasks and clinical applications such as question-answering systems, clinical decision support systems, clinical follow-up systems, and health technology assessment processes. The automatic clinical annotations and labeling created with our proposed AL algorithm are helpful for any clinical text classification task that needs labeled data. In summary, the key contributions of this study are as follows.

- Designing algorithms for unstructured clinical text preprocessing and section identification to prepare initial training data with SOAP labels as seed data for the active learning model.
- BERT-based multiclass annotations by developing a robust AL model based on uncertainty-based sampling – least confidence query strategy, which could be reused as an integral part of our proposed framework or as an independent system for annotating unlabled clinical data with SOAP labels.
- BERT-based multiclass classification by developing an attention-based neural network model called Domain Adaptive Semantic-based Attention Network (DASAN), which employ transfer learning (TL) and UMLS-based semantic enrichment (UMLS-SE) to help capture both contextual and semantic information in clinical notes.

2. Materials And Methods

This section describes the proposed framework of SOAP-based data labeling and classification of clinical text. The framework is divided into three steps,

as shown in Figure 1. In the first step, a rule-based algorithm is employed for initial data labeling (seed data annotations). According to the SOAP protocol, the rule-based algorithm includes both syntactic and semantic approaches to annotate different sections in the clinical notes. In the second step, an AL model is designed to create more data with SOAP labels as a training dataset for the classification model. Finally, a pre-trained model is used to create embeddings to enrich the training data for attaining data and gain maximum throughput out of the final deep learning model, which we eventually utilize to classify the unseen clinical notes.

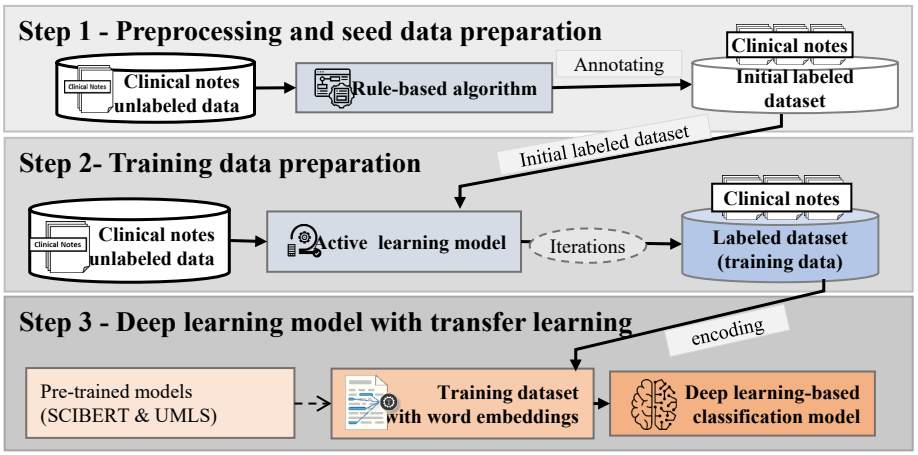

Figure 1: SOAP-based data labeling and classification framework of unstructured clinical notes.

2.1. Seed Data Preparation

We employ the baseline classifier using a rule-based approach to generate the initially labeled dataset, as shown in Figure 2. A clinical note is written by physicians either in semi-structured having sections with headings identified or unstructured having sections with headings unidentified. The rule-based classifier handles both formats. Subseqeuntly data preprocessing, First, it identify the clinical sections boundary, then it checks the clinical notes, identifies headings, and then chooses the appropriate workflow to process the text.

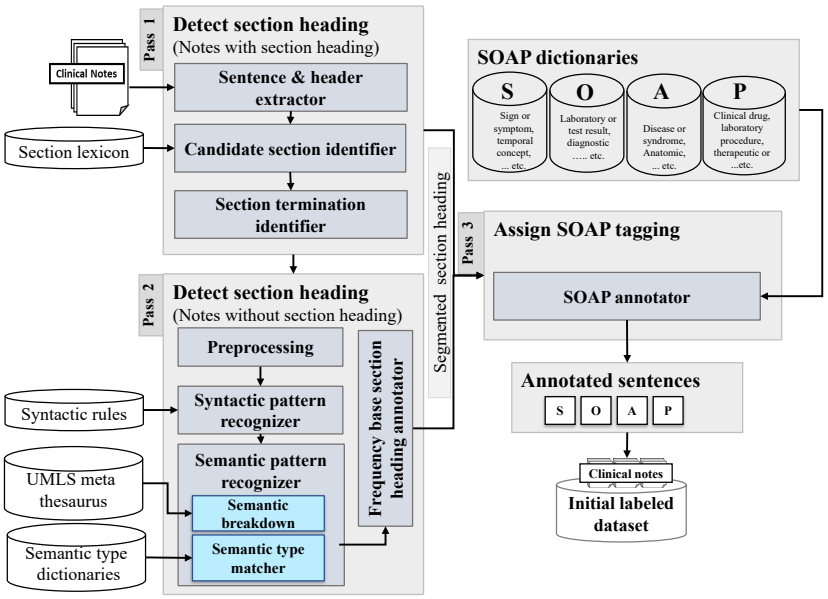

Figure 2: Rule-based classification model for seed data preparation.

2.1.1. Data Preprocessing

Text preprocessing is essential to eliminate noisy and useless data such as punctuation, stop words, and trim the dimensionality to minimize machine processing time complexity. A few examples of text that need preprocessing steps are described in Table 1. Deleting numerous spaces among words and phrases, particularly those with colons (:) at the end, such as "Chief Complaint :" to "Chief Complaint: ." Splitting the phrases or words end with a colon ":". Removing all colons from the words' ends and changing case to lowercase.

2.1.2. Section Boundary Identification

In the preprocessing, words or sentences that conclude with a colon, were splitted. The proposed technique assigns sentences to the currently active or matched section. The algorithm saved the presently active section index and filtered out phrases with no section at the beginning of a paragraph to determine a section boundary. The algorithm scans the sentences line by line until the next section found. Section index number is updated, and all sentences are assigned

Table 1: Unstructured clinical note with explicit and implicit.

Section Index	Unstructured Clinical Text with Section
0	**Chief complaint :** shortness of breath
1	**History of present illness:** 67 y/o male with worsening shortness of breath. Had abnormal ETT and was referred for cath. Cath revealed severe 3 vessel disease then referred for surgical intervention
2	**Physical Examination:** VS: 65 20 160/100 5'7" 180 # General: WD/WN male in NAD HEENT: EOMI, PERRL, NC/AT Neck: Supple, From, -JVD, -carotid bruits Chest: CTAB -w/r/r Heart : RRR -c/r/m/g Abd: soft, NT/ND +BS Ext: warm, well-perfused - edema, -varicosities Neuro: A&Ox3
3	**Discharge Diagnosis:** Rectal bleeding from inferior mesenteric artery tributaries supplying sigmoid colon.

to the previous section index. For example, in Table I active section is "chief complaint" with index of 0. The proposed algorithm process the document sentence by sentence until the following section "history of present illness" with index 1 is matched. The active section is "history of present illness" now and its index is noted along with text till the next section matched. This process continues until the last section identified, which is located at index 3. Some parts of this work can be referred from our previous work [19] and the section header terminology lexicon (SHTL) is based on the works [20].

2.1.3. Documents with Sections' Heading Identified

Currently, there is no universal standard or format for writing a clinical note worldwide. Usually written in natural language, clinical notes typically use templates to divide their narratives into sections and subsections for readability and shared understanding. Physicians grouped the segments based on frequently used non-standardized terms and labeled them as "section headers." For example, history sections generally contained labels such as "history of present illness," "past medical history," and "physical examination." Sections can be further divided into subsections, such as "cardiovascular exam" within "physical examination" or "substance abuse history" within "social history." To

keep consistency in the labels given to different sections, we use the annotation schema [5] to group the clinical note sections according to the SOAP framework. In the SOAP framework, each section in the clinical notes has its specific meaning and terminology. It groups the background or historical information relevant to understanding the patient's current or future clinical state into subjective, observable, measurable, and quantifiable information into objective, expressions of a diagnosis, impression or differential diagnosis into assessment, and any reporting of planned or implemented treatment actions, education, or follow-up procedures into plans. Using SHTL, we design a rule-based algorithm called sectionTagger to identify section tags within a clinical note.

The sectionTagger procedure identifies a list of sentences within a clinical document to assist section boundary separation as they likely belong to the same section or a subsection using various regular expressions. The sectionTagger algorithm processes the clinical notes sentence-wise to check the possible section headers based on the developed SHTL lexicon. First, sectionTagger tries to locate all explicitly labeled section tags by string pattern matching. Then, it reads the clinical notes document line by line, checks each line exact string similarity against already defined section headers in SHTL up to the next section, and stores the identified section into the dictionary. This process repeats to the end of the clinical document to make a dictionary with assigned SOAP labels for all the identified sections.

2.1.4. Document with Sections' Heading Unidentified

When the algorithm finds a document with no heading, it performs implicit heading matching using syntactic and semantic techniques. First, a clinical note with unidentified sections is preprocessed by applying basic NLP techniques and then applying both syntactic and semantic matching techniques to identify a candidate heading and assign a SOAP label. In the NLP preprocessing, various tasks are performed: tokenization, stop word removal, lemmatization, N-grams (unigram and bigram), and part-of-speech (POS) tagging. For syntactic-based section's heading matching, a linguistic feature using POS tagging is employed,

where the irrelevant features are removed, and the essential features are retained. A regular expression is constructed to retain only meaningful information explicitly like a noun, adjective, and adverb from a list of words. In Eq. 1, "<NN*>" denotes all the nouns phrase, "<JJ*>" represents all the adjectives, and "<RB*> is for the adverbs phrase, and BoWs represents the list of bag of words.

$$BoWs =< NN >< JJ* >< RB* > \tag{1}$$

In addition to POS, the algorithm also identifies word/part of speech pair (word/POS) for each unigram and bigram tokens to remove word sense disambiguation. For example, a word "discharge" POS is (NN) often indicates a clinical finding, where "discharge" (VB) indicates being released from the hospital. Additionally, the verb phrase also plays a vital role in detecting the sentence tense. With the help of sentence tenses, our algorithm classifies the sentence into subjective, assessment, and plan.

Most of the subjective narration focus on patient history, written in the past tense. Assessment section narration is mainly written in the present tense, while plan section narration focuses on future treatment, written in the future tense. Our algorithm encoded every verb phrase in the sentence, present, past, and future, providing meta-information to semantic-based matcher. For example, "she has developed a severe cough" is encoded as past tense, and "she will return if she develops a severe cough" as future tense. For semantic-based section's heading matching, a multi- step process is followed, semantic breakdown, medical concept semantic matching to the dictionary, and concept classification. Semantic breakdown task includes identification of term, concept, semantic type, and entity type [21].The UMLS Metathesaurus is used as a supporting tool to find semantics for accurate identification of SOAP categories. After identifying concepts in the UMLS for each token, semantic types are identified, which are then matched with semantic type in the dictionary created locally. If a semantic

type is matched in the dictionary, the input term is fairly categorized with a suitable SOAP class.

Finally, a sentence is resolved to assign a final SOAP class using a majority vote mechanism. Figure 3 shows an example of semantic-based section annotation. First, the inputted sentence is preprocessed before applying the semantic breakdown using UMLS. Then, after extracting semantic type for each preprocessed token, a dictionary-based annotation process is performed to assign SOAP-based tagging. Finally, the assigned labels are voted to get the final label, "Subjective," in the given an example.

2.2. Automatic Data Annotation using Active Learning

Generally, an active learning approach predicts a label to the most informative unlabeled data and makes it a part of the training dataset [22]. Training a supervised machine learning model with fewer labels is a problem. Involving human experts to create labels for all the data is an expensive task; we need an efficient approach to remove hurdles in data labeling. For this study, we use transfer learning with an AL approach to label the unlabeled data using a small-text framework [23]. In brief, an initial classifier using the seed dataset is trained, which is then applied to predict the next set of unlabeled records. Using a pool-based sampling scenario, we opted for the least confidence query strategy to select a pool of 10 instances per iteration to predict actual labels by the learner. The overall AL model development workflow is depicted in Figure 4, reflecting a step-by-step process of selecting unlabeled instances, predicting labels, and retraining the model. It takes an input of an initially labeled dataset to construct embedding vectors, then selected for model training. We employ a transformer-based model called SciBERT with an uncased model [18].

In AL, we have multiple query strategies for picking an example set for the next iteration. Broadly these strategies are divided into three categories: pool-, stream-, and membership-based selection. We opted for a pool-based sampling query strategy due to its simplistic assumption of a small set of labeled data and a large set of unlabeled data [24]. Under this strategy, we choose the uncertainty

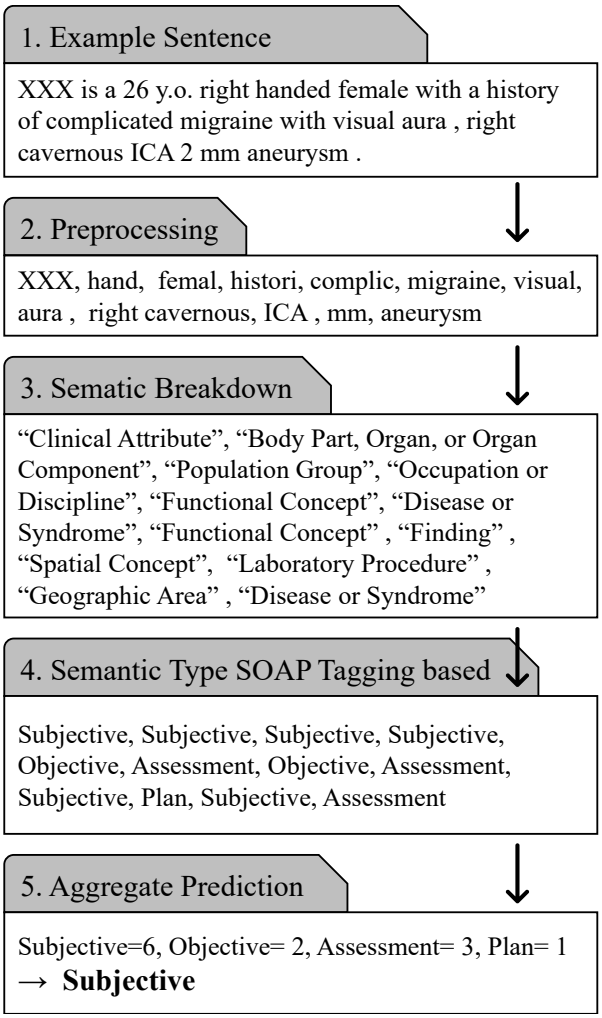

Figure 3: Semantic-based annotation example

sampling (least confidence) strategy, which is the most straightforward and most used query strategy [16].

In this approach, an AL enquires the instances that are least certain to be labeled. The advantage of this model is that it attacks the uncertainty at the beginning, which eventually leads to the correct model. In other words, the probability of correct classification is improved with each iteration. The process of choosing samples for the next iteration stops when we reach the point of

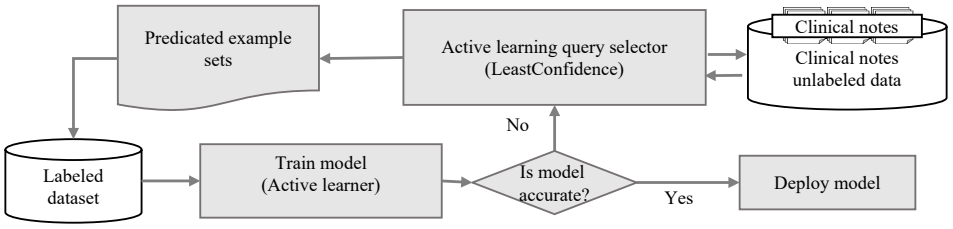

Figure 4: A step-by-step process of automatic text annotation using active learning approach.

convergence when the model stops improving further. The final labeled data is saved in the labeled dataset, which is used in the subsequent steps.

2.3. Classification Model

With AL model we add enough labeled records to the dataset, which is sufficient to use as a training dataset for state-of-the-art deep learning model. Furthermore, we develop an attention-based neural network model named as DASAN. A high-level workflow architecture of the proposed model for classifying clinical notes with SOAP labels is depicted in Figure 5.

The proposed model utilizes TL and UMLS-based semantic enrichment (UMLS-SE) to achieve optimal results. The combination of two networks was intended to help capture both contextual and semantic information in clinical notes. In the model, the weight-tuning operation is activated along with the SOAP-based training dataset to learn specific characteristics of the data. Firstly, the clinical text is normalized using data preprocessing techniques such as removing accented characters, expanding contractions, removing special characters, stemming, and removing stop words. Then, the normalized clinical text is inputted into two proposed networks for predicting the final SOAP label. Both networks combine the concatenation, dropout, and dense layers using the SoftMax activation function. The cross-entropy loss is optimized using Adam and a dropout of 0.3. Finally, an early stopping criterion is applied in model training with the patience of 10, a batch size of 32, and a learning rate of 1e-3.

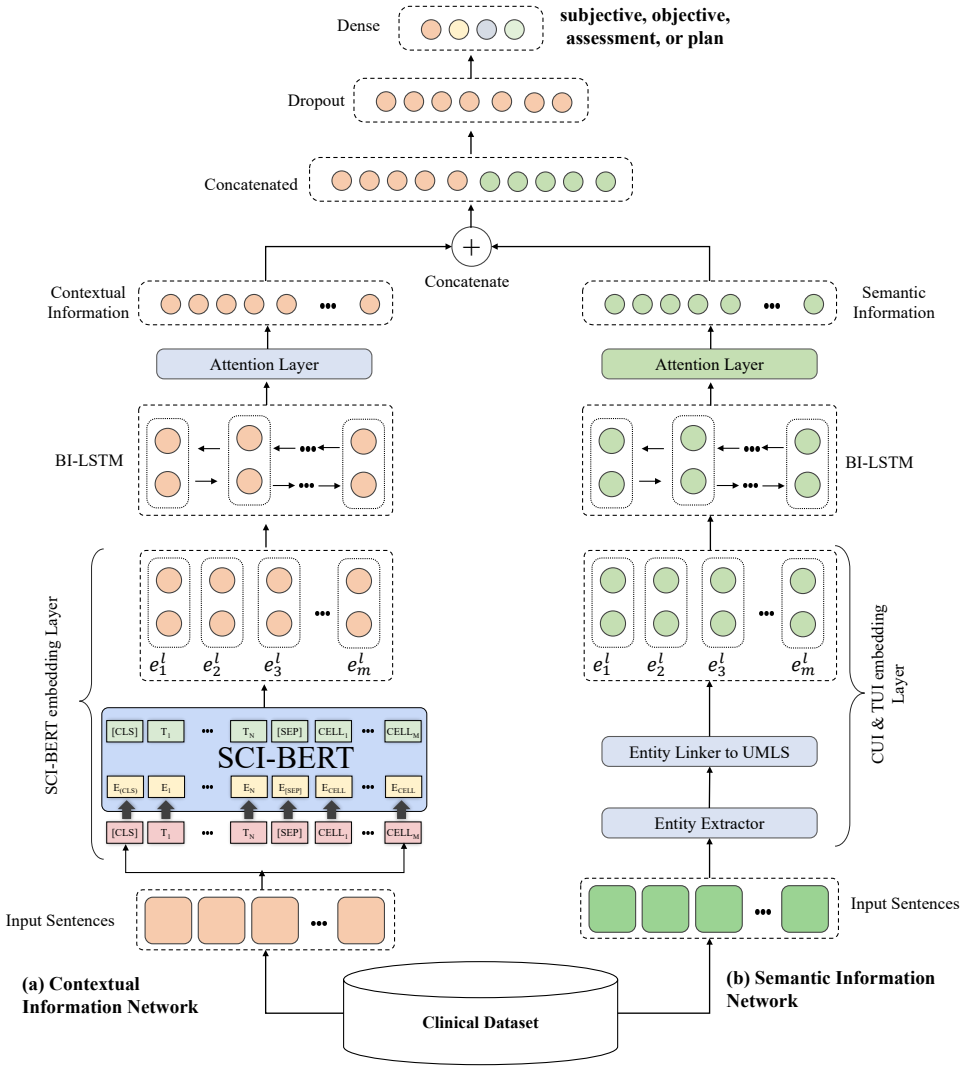

Figure 5: The proposed framework architecture showing two inputs; a) contextual information network and b) semantic information network, concatenated to generate multi-class output: subjective, objective, assessment, and plan.

2.3.1. Contextual Information Network

This network captures contextual information by including three layers: word embedding, encoding, and attention layer. We use a pre-trained SCIBERT-based-uncased model [18] with transformers built on BERT-based architecture

(24 Layers) in the word embedding layer. In BERT-based embedding, initially, words start representing their embedding in the embedding layer. Then, every layer performs a multi-headed attention calculation based on the previous layer for generating the intermediary representation having the same size as shown in Figure 5(a).

For the contextual encoding, a Bi-LSTM layer is incorporated in the model that contains both forward and backward LSTM. A Bi-LSTM is a sequence processing model containing two LSTMs capable of taking the inputs in both directions (forward and backward). Thus, using Bi-LSTMs can effectively improve contextual information by knowing each word immediately next to and preceding a word about the sequence at every step.

Finally, an attention layer is included that uses the output of the contextual encoding layer as an input. This layer assigns a higher weight to the most important words used in the clinical notes. The attention layer solves a problem with Bi-LSTM having a loss of useful encoded information; by keeping that information through an average of the encoded states of the network outputs. All the encoded states of the Bi-LSTM network are equally important, a weighted sum is used of these encoded states to make the final prediction. The attention weights are computed by making a small, fully connected neural network on top of each encoded state. This network has a single-unit final output layer corresponding to the attention weight. Our Attention function involves dense layers back-to-back plus a tanh function from Bahdanau Attention [25].

2.3.2. Semantic Information Network

A semantic information network as shown in Figure 5(b), is used to capture domain-specific semantic information. For extracting the medical entity and their concept from the given text, a component of the scispaCy [26] NER model is utilized, and the UMLS is used as a knowledgebase for entity linker in the scispaCy component. It returns Concept Unique Identifier (CUI), name, definition, Type Unique Identifier (TUI), and aliases. Embeddings are generated from the extracted UMLS semantic information for the inputted sentence, followed by

Bi-LSTM and attention layers as the contextual information network.

3. Results and Evaluations

The results are presented in four parts: (i) Section boundary identification algorithm (ii) Rule-based algorithm for seed data annotations, (iii) AL-based algorithm for enhanced data annotations and (iv) DASAN model for SOAP classification. To measure the merit of algorithms, we use four statistical indicators (recall, precision, F1-score, and accuracy) for the evaluation, and the computing formulas of these metrics are given in Eq. 2.

$$\begin{aligned} Recall &= \frac{TP}{TP+FN} \\ Precision &= \frac{TP}{TP+FP} \\ F1-Score &= \frac{2\,(Rec*Prec)}{Rec*Prec} \\ Accuracy &= \frac{TP+TN}{TP+FP+FN+TN} \end{aligned} \tag{2}$$

Where, TP: True positive, FP: False positive, TN: True negative, and FN: False negative.

We utilized three unstructured clinical discharge summary datasets provided by i2b2 National Center, Partners Healthcare, and Beth Israel Deaconess Medical Center [27] to measure the performance of proposed methods. Partners Healthcare consist of 97 clinical notes, Beth Israel Deaconess Medical Center contain 73 clinical notes, and the test dataset provided by i2b2 National center for system evaluation contain 256 clinical notes. Cumulative we utilized 426 unstructured clinical discharge summaries in the proposed methodology. These clinical notes consist of explicit and implicit defined sections used for section base SOAP annotation.

Table 2: Performance Of Different Methods of Identifying Section Headings with Soap Labels

	Subjective			**Objective**			**Assessment**			**Plan**		
Method\Metrics	Rec	Prec	F1	Rec	Prec	F1	Rec	Prec	F1	Rec	Prec	F1
Method-1 (Syntactic)	0.97	0.31	0.46	0.34	0.68	0.45	0.14	0.64	0.22	0.1	0.58	0.17
Method-2 (Semantic)	0.27	0.24	0.25	0.25	0.42	0.31	0.64	0.51	0.56	0.39	0.34	0.36
Method-3 (Hybrid)	0.92	0.93	0.92	1	1	1	1	0.94	0.97	0.87	0.94	0.9

3.1. Section Boundary Identification Algorithm Performance

A sample of 20 clinical notes is chosen from each dataset of i2b2 to measure the section identification algorithm performance. These documents contain explicit sections (79%) and implicit sections (21%). The proposed algorithm and sec tag algorithm are evaluated on these documents. We compare the performance of the proposed algorithm with a "sec_tag" [20] algorithm in terms of precision, recall, and F1-Score, which are standard matrices representing the quality of the information retrieval process. Both algorithms obtain a higher precision on section identification; however, the traditional sec_tag algorithm produced a lower recall of 0.71 than the proposed algorithm with a recall of 0.94. As a result, the proposed algorithm gained about 15% higher F1 score than the competitor.

3.2. Rule-based Algorithm Performance

Table 2 shows the performance of the rule-based algorithm for annotating the initial dataset. A dataset containing 243 SOAP annotated sentences from a set of structured clinical notes with identified sections, which is used as a gold standard for this experiment. Applying three different variations of the experiment, we obtained different results. Method-3 (hybrid – syntactic and semantic combined) consistently performed better than individual syntactic (method-1) and semantic (method-2) for all SOAP classes except the subjective class where method-1 recall score was noted higher than the competitors.

3.3. Active Learning Model Performance

Figure 6 shows the results of the AL model on annotating a dataset containing 243 records obtained as seed data annotated with a baseline rule-based

system. Out of this seed data, 143 are used as the initial training dataset for initializing the AL model, while the remaining records are reserved as unlabeled records. Despite selecting all the instances at once, we opted for ten records per iteration. The reason is to gradually check model performance to reach the point of convergence and avoid burdening human experts to check too many records at once. Accuracy is recorded for each iteration separately, as shown in Figure 6 (a). We evaluated four different query strategies of pool-based sampling, and the least confidence query strategy outperformed the competitors at both the training and testing stages. During training, we observed that model accuracy had reached an optimal level of 94% accuracy at iteration 8. However, the same accuracy is carried over to iterations 9 and 10 without further improvement; therefore, we stopped the AL sample selection process. The same strategy performance is noted higher than competitors at the testing stage, as shown in Figure 6 (b). Finally, we annotated the rest of the clinical notes (457) using the AL model; the final training dataset comprised 700 records.

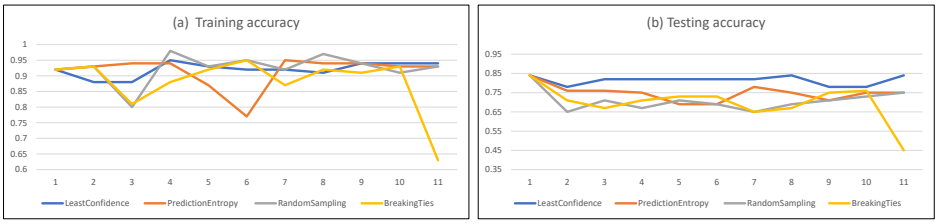

Figure 6: Performance of AL on annotations using different AL query strategies.

3.4. Classification Model Performance

We used the final dataset for training the classification model, with an 8:2 ratio (80% for training and 20% for model evaluation). Table 4 shows the results of the performance of classification models including the proposed model (DASAN) with the following parameter settings as shown in Table 3. After experimenting with different combinations of word embeddings and deep learning models, we found an ideal performer. First, we tested the base model involving

sequence embedding and Convolutional Neural Network (CNN) as a classification model and obtained a lower accuracy of about 25% on the test dataset. Next, we kept the same embedding model but changed the classification models as MLP, Recurrent Neural Network (RNN), and Bidirectional Long-Short Term Memory (BiLSTM) and found the latter model with a higher performance of increasing the base-level accuracy by about 21% with an accuracy of about .73%. Then we changed the embedding layer to a Bert-based embedding called 'Bert-en-cased-L-24-H-1024-A-16' and obtained a bit higher performance (about 7%).

Table 3: Parameter Settings of the Proposed Model

Parameters	**Value**
Max sequence length	10 k
epochs	10
Learning rate	8e-5
Embedding dimensions size	600

Table 4: Performance Of Deep Learning Models with Different Embedding Schemes

	Training				Testing			
Embedding and DL Model	Accuracy	F1	Precision	Recall	Accuracy	F1	Precision	Recall
SE* + CNN	0.595	0.5455	0.5644	0.542	0.3007	0.0604	0.1538	0.0385
SE + RNN	0.9397	0.9226	0.8971	0.9544	0.7359	0.6704	0.6109	0.7426
SE + BiLSTM	0.9463	0.9439	0.9253	0.9664	0.7367	0.6886	0.6625	0.7171
BE* + CNN	0.7049	0.6398	0.7336	0.5699	0.5294	0.0303	0.5	0.0156
BE + RNN	0.9512	0.9163	0.8628	0.9872	0.7849	0.7791	0.7347	0.8417
BE + BiLSTM	0.9663	0.9324	0.8958	0.9808	0.817	0.8137	0.757	0.891
BE + BERT	0.9615	0.9567	0.9386	0.9784	0.8043	0.7476	0.7066	0.7952
XLNet (xlnet-base-cased)	0.9517	0.9337	0.9135	0.958	0.7847	0.7703	0.7252	0.8215
RoBERTa (roberta-base)	0.9149	0.8723	0.8213	0.9412	0.7639	0.7897	0.7742	0.8059
distilbert-base-cased	0.9615	0.9551	0.9301	0.9856	0.7647	0.7834	0.7571	0.8117
ALBERT (albert-base-v1)	0.9507	0.9412	0.9222	0.964	0.7843	0.7594	0.7182	0.8059
Biobert-base-cased-v1	0.9555	0.9529	0.939	0.9694	0.7843	0.7644	0.736	0.7952
Bio-ClinicalBERT	0.9651	0.9643	0.9424	0.9904	0.8235	0.7901	0.7527	0.8322
Clinical-bert-base-128	0.9075	0.8964	0.8584	0.9436	0.8039	0.7836	0.6937	0.9005
PubMedBERT-base	0.9579	0.9498	0.935	0.9676	0.7647	0.7927	0.7802	0.8059
BlueBERT	0.9663	0.9668	0.9502	0.9868	0.8039	0.7842	0.746	0.8273
SBE* + BERT	0.9775	0.9567	0.9386	0.9784	0.8431	0.7827	0.75	0.8215
SBUE* + BiLSTM (Proposed)	0.9809	0.9599	0.94	0.9844	0.8935	0.8025	0.775	0.8322

*SE: Sequence Embedding; *BE: BERT-based Embedding; *SBE: SCI BERT-based Embedding; *SBUE: SCI BERT UMLS-based Embedding.

Finally, keeping the same embedding model, we checked for the other three competitors and found BiLSTM once again on the top with an accuracy of 0.93. The BERT model's better performance gave us the confidence to check with other embedding options. Finally, we incorporated the BERT-based embedding layer called 'scibert-basevocab-uncased' together with the UMLS-based embed-

ding layer, which produced the most excellent results of about 0.98 accuracies better than all other configurations and the loss was a minimum of about 1%. The proposed model is tested on multiple points to get the desired number of epochs, and we obtained the optimal results on epochs:10 as shown in Figure 7.

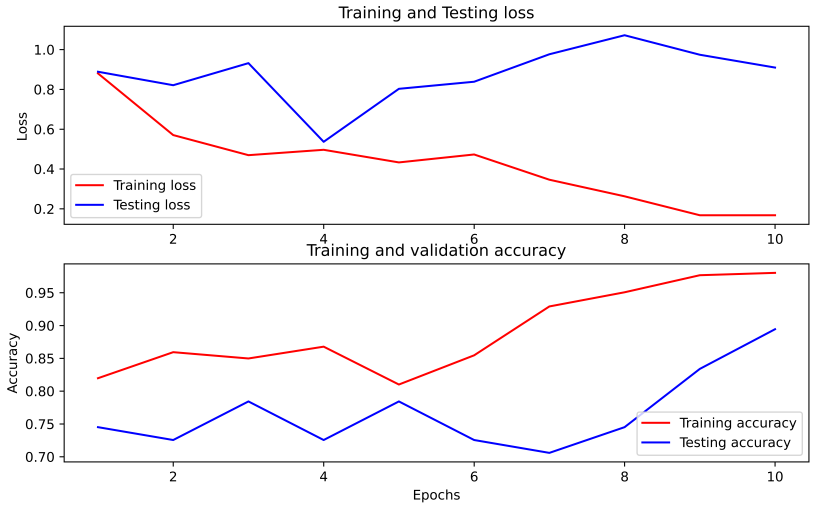

Figure 7: Proposed model accuracy on different Epochs.

4. Discussion

Clinical documents are usually structured with sections' headings identified or sometimes unidentified. The sections' headings are also varied in vocabulary from one setting to another. Our objective was to evaluate the SOAP framework's usefulness for clinical information extraction using state-of-art deep learning models classifying sentences extracted from clinical reports. However, for deep learning models, we were required sufficient annotated data, which was not available. Therefore, we create an initial dataset annotated with a rule-based classifier. We used a rule-based approach for a limited dataset (seed) only and did not carry further with it for two reasons; scaling up the issue and

human expertise required to verify each instance. In addition, developing regular expressions for unseen patterns discovered in new clinical documents is a challenge to cope with top-down rule-based and linguistic approaches. Therefore, we employed the AL model for the rest of the training data annotation. AL is a powerful approach that automates the process of data labeling (annotation) and reduces human involvement compared to the rule-based approach, where human involvement is required for each record verification. We noticed that AL model accuracy increased with each iteration until it reached the eighth iteration. Wang et al. [8] used CNN and simple word embedding for classifying clinical text using weak supervision and deep learning. We evaluated CNN model performance on our data with both sequence embeddings and BERT-based embeddings. However, both models give us lower performance than other models, as mentioned in Table III. Our model, in contrast, uses state-of-the-art embedding techniques that incorporate domain knowledge using UMLS and domain-independent knowledge. As a result, it can be reused for any type of clinical document classification based on SOAP protocol. Moreover, the proposed method minimizes the labor cost of manual data annotation. The proposed AL model is based on the quality of the initial seed (training dataset), which is generated using a rule-based approach. The rule-based approach performs well in the best case (documents with sections' headings) and low performance in the worst case (documents with no sections' headings), which must be addressed to generate a high-quality initial dataset. One exciting feature that can be enhanced in the proposed method is adding an oracle in the loop for validating the predicted instances with low prediction probability scores. Additionally, the proposed model can be deployed in the real-world environment to check the model's effectiveness on the real dataset. Python code of the proposed model is provided on GitHub link https://github.com/BioMeGiX/SOAP_framework.

5. Conclusion

The vast availability of unstructured clinical data offers an opportunity to extract meaningful information for the applications that support the process of clinical decision-making. However, extracting the relevant information from unstructured text into a clinically useful format is a big challenge. Therefore, this work targeted this aspect of information extraction into a well-known protocol (SOAP) used as an information container. The clinical text in the form of SOAP structure enhances information readability, and the individual sentences, i.e., subjective, objective, assessment, and plan, can be used in other add-on applications such as clinical decision support systems. Additionally, it helps the organizations develop multiple individualistic systems such as diagnostic, treatment, and prognostic by utilizing the relevant SOAP section.

References

[1] L. Yao, C. Mao, Y. Luo, Clinical text classification with rule-based features and knowledge-guided convolutional neural networks, BMC medical informatics and decision making 19 (3) (2019) 31–39. doi:10.1186/s12911-019-0781-4.

[2] J. Liang, C.-H. Tsou, A. Poddar, A novel system for extractive clinical note summarization using ehr data, in: Proceedings of the 2nd clinical natural language processing workshop, 2019, pp. 46–54. doi:10.18653/v1/w19-1906.

[3] I. Li, M. Yasunaga, M. Y. Nuzumlalı, C. Caraballo, S. Mahajan, H. Krumholz, D. Radev, A neural topic-attention model for medical term abbreviation disambiguation (2019). doi:10.48550/ARXIV.1910.14076. URL https://arxiv.org/abs/1910.14076

[4] S. Sheikhalishahi, R. Miotto, J. T. Dudley, A. Lavelli, F. Rinaldi, V. Osmani, Natural language processing of clinical notes on chronic diseases:

Systematic review, JMIR Med Inform 7 (2) (2019) e12239. doi:10.2196/12239.
URL http://medinform.jmir.org/2019/2/e12239/

[5] D. Mowery, J. Wiebe, S. Visweswaran, H. Harkema, W. W. Chapman, Building an automated soap classifier for emergency department reports, Journal of Biomedical Informatics 45 (1) (2012) 71–81. doi:https://doi.org/10.1016/j.jbi.2011.08.020.
URL https://www.sciencedirect.com/science/article/pii/S153204641100147X

[6] W.-H. Weng, K. B. Wagholikar, A. T. McCray, P. Szolovits, H. C. Chueh, Medical subdomain classification of clinical notes using a machine learning-based natural language processing approach, BMC medical informatics and decision making 17 (1) (2017) 1–13. doi:10.1186/s12911-017-0556-8.

[7] Y. Wang, L. Wang, M. Rastegar-Mojarad, S. Moon, F. Shen, N. Afzal, S. Liu, Y. Zeng, S. Mehrabi, S. Sohn, H. Liu, Clinical information extraction applications: A literature review, Journal of Biomedical Informatics 77 (2018) 34–49. doi:https://doi.org/10.1016/j.jbi.2017.11.011.
URL https://www.sciencedirect.com/science/article/pii/S1532046417302563

[8] Y. Wang, S. Sohn, S. Liu, F. Shen, L. Wang, E. J. Atkinson, S. Amin, H. Liu, A clinical text classification paradigm using weak supervision and deep representation, BMC medical informatics and decision making 19 (1) (2019) 1–13. doi:10.1186/s12911-018-0723-6.

[9] I. Spasic, G. Nenadic, Clinical text data in machine learning: Systematic review, JMIR Med Inform 8 (3) (2020) e17984. doi:10.2196/17984.
URL http://medinform.jmir.org/2020/3/e17984/

[10] M. Kholghi, L. Sitbon, G. Zuccon, A. Nguyen, Active learning reduces annotation time for clinical concept extraction, International Journal of Medical Informatics 106 (2017) 25–31.

doi:https://doi.org/10.1016/j.ijmedinf.2017.08.001.
URL https://www.sciencedirect.com/science/article/pii/S1386505617302009

[11] T. Searle, Z. Kraljevic, R. Bendayan, D. Bean, R. Dobson, Medcattrainer: A biomedical free text annotation interface with active learning and research use case specific customisation (2019). doi:10.48550/ARXIV.1907.07322.
URL https://arxiv.org/abs/1907.07322

[12] J. Khan, Y.-K. Lee, Lessa: A unified framework based on lexicons and semi-supervised learning approaches for textual sentiment classification, Applied Sciences 9 (24). doi:10.3390/app9245562.
URL https://www.mdpi.com/2076-3417/9/24/5562

[13] M. Li, M. Scaiano, K. El Emam, B. A. Malin, Efficient active learning for electronic medical record de-identification, AMIA Summits on Translational Science Proceedings 2019 (2019) 462.

[14] K. Tomanek, U. Hahn, Annotation time stamps — temporal metadata from the linguistic annotation process, in: Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), European Language Resources Association (ELRA), Valletta, Malta, 2010.
URL http://www.lrec-conf.org/proceedings/lrec2010/pdf/652_Paper.pdf

[15] S. Zhou, Q. Chen, X. Wang, Active deep learning method for semi-supervised sentiment classification, Neurocomputing 120 (2013) 536–546, image Feature Detection and Description. doi:https://doi.org/10.1016/j.neucom.2013.04.017.
URL https://www.sciencedirect.com/science/article/pii/S0925231213004888

[16] M. S. Hajmohammadi, R. Ibrahim, A. Selamat, H. Fujita, Combination of active learning and self-training for cross-lingual sentiment classification

with density analysis of unlabelled samples, Information Sciences 317 (2015) 67–77. doi:https://doi.org/10.1016/j.ins.2015.04.003.
URL https://www.sciencedirect.com/science/article/pii/S0020025515002650

[17] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding (2018). doi:10.48550/ARXIV.1810.04805.
URL https://arxiv.org/abs/1810.04805

[18] I. Beltagy, K. Lo, A. Cohan, Scibert: A pretrained language model for scientific textdoi:10.48550/ARXIV.1903.10676.
URL https://arxiv.org/abs/1903.10676

[19] A. Abbas, J. Hussain, M. Afzal, H. S. M. Bilal, S. Lee, S. Jeon, Explicit and implicit section identification from clinical discharge summaries, in: 2022 16th International Conference on Ubiquitous Information Management and Communication (IMCOM), 2022, pp. 1–8. doi:10.1109/IMCOM53663.2022.9721771.

[20] J. C. Denny, R. A. Miller, K. B. Johnson, A. Spickard III, Development and evaluation of a clinical note section header terminology, in: AMIA annual symposium proceedings, Vol. 2008, American Medical Informatics Association, 2008, p. 156.

[21] A. Abbas, M. Afzal, J. Hussain, T. Ali, H. S. M. Bilal, S. Lee, S. Jeon, Clinical concept extraction with lexical semantics to support automatic annotation, International Journal of Environmental Research and Public Health 18 (20). doi:10.3390/ijerph182010564.
URL https://www.mdpi.com/1660-4601/18/20/10564

[22] P. Kumar, A. Gupta, Active learning query strategies for classification, regression, and clustering: a survey, Journal of Computer Science and Technology 35 (4) (2020) 913–945.

[23] C. Schröder, L. Müller, A. Niekler, M. Potthast, Small-text: Active learning for text classification in python (2021). doi:10.48550/ARXIV.2107.10314. URL https://arxiv.org/abs/2107.10314

[24] D. D. Lewis, W. A. Gale, A sequential algorithm for training text classifiers, in: SIGIR'94, Springer, 1994, pp. 3–12.

[25] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate (2014). doi:10.48550/ARXIV.1409.0473. URL https://arxiv.org/abs/1409.0473

[26] M. Neumann, D. King, I. Beltagy, W. Ammar, Scispacy: fast and robust models for biomedical natural language processing, arXiv preprint arXiv:1902.07669.

[27] D. of Biomedical Informatics (DBMI) at Harvard Medical, i2b2: Informatics for integrating biology and the bedside.
URL https://www.i2b2.org/NLP/DataSets/Main.php