# Automatic Patient Note Assessment without Strong Supervision

**Jianing Zhou[1] , Vyom Thakkar[1] , Rachel Yudkowsky[2], Suma Bhat[1] and William F. Bond[3]**
[1]University of Illinois at Urbana-Champaign
[2]University of Illinois at Chicago
[3]Jump Simulation, OSF Healthcare, University of Illinois College of Medicine at Peoria
[1]{zjn1746,spbhat2,vnt2}@illinois.edu
[2]rachely@uic.edu
[3]william.f.bond@jumpsimulation.org

## Abstract

Training of physicians requires significant practice writing patient notes that document the patient's medical and health information and physician diagnostic reasoning. Assessment and feedback of the patient note requires experienced faculty, consumes significant amounts of time and delays feedback to learners. Grading patient notes is thus a tedious and expensive process for humans that could be improved with the addition of natural language processing. However, the large manual effort required to create labeled datasets increases the challenge, particularly when test cases change. Therefore, traditional supervised NLP methods relying on labelled datasets are impractical in such a low-resource scenario. In our work, we proposed an unsupervised framework as a simple baseline and a weakly supervised method utilizing transfer learning for automatic assessment of patient notes under a low-resource scenario. Experiments on our self-collected datasets show that our weakly-supervised methods could provide reliable assessment for patient notes with accuracy of 0.92.

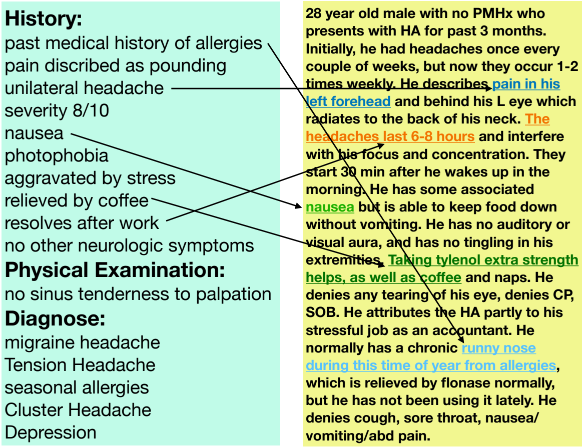
## 1 Introduction

Sponsored by the Federation of State Medical Boards (FSMB) and the National Board of Medical Examiners (NBME), the United States Medical Licensing Examination (USMLE) is a "three-step examination for medical licensure in the U.S. that assesses a physician's ability to apply knowledge, concepts, and principles, and to demonstrate fundamental patient-centered skills, that are important in health and disease and that constitute the basis of safe and effective patient care."[1] Prior to 2020, the USMLE Step 2 exam included a second component, Step 2 Clinical Skills, that used a simulated clinical examination with standardized patients to assess various clinical competencies, including the ability to document relevant patient history and differential diagnoses in a written patient note. After the discontinuation of the USMLE Step 2 Clinical Skills examination, medical schools may have more motivation to include a clinical skills examination that requires patient note writing after observing standardized patients (Tsichlis et al., 2021). Patient notes, as one type of health documents, document clinical findings and reflect examinees' ability to gather information and communicate their findings to patients and colleagues. Therefore, in Step 2 Clinical Skills, examinees' written patient notes were assessed manually by experienced physician raters. More than 30,000 examinees took this examination each year, resulting in more than 330,000 patient notes that were graded by more than 100 raters (Sarker et al., 2019). The case-specific nature of the patient notes and large volume of exams make the human scoring process time-consuming and tedious. Additionally, it is well-documented that human judgement in general is prone to bias and errors (Engelhard Jr et al., 2018). Training of qualified physician graders also requires assessment and feedback from medical experts, costing significant amounts of time. The manual effort required in grading medical examinations makes this a challenging problem to be addressed with the addition of NLP techniques.

NLP has been applied to automatically process health documents, including assessing practical clinical content from patient notes (Latifi et al., 2016; Sarker et al., 2019). Specifically, patient notes after simulated patient encounters are required to contain specific information, which is specified by items in a checklist created through faculty consensus. Figure 1 shows an example of patient note and checklist items. The task of automatic patient note assessment aims to judge if the given checklist items are included in the patient notes by exactly same expressions or synonymous expressions. Equivalents may be true synonyms, acceptable abbreviations, or answer alternatives

[1] https://www.usmle.org/

*Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 116 - 126
December 7, 2022 ©2022 Association for Computational Linguistics



**History:**
past medical history of allergies
pain discribed as pounding
unilateral headache
severity 8/10
nausea
photophobia
aggravated by stress
relieved by coffee
resolves after work
no other neurologic symptoms
**Physical Examination:**
no sinus tenderness to palpation
**Diagnose:**
migraine headache
Tension Headache
seasonal allergies
Cluster Headache
Depression

**28 year old male with no PMHx who presents with HA for past 3 months. Initially, he had headaches once every couple of weeks, but now they occur 1-2 times weekly. He describes pain in his left forehead and behind his L eye which radiates to the back of his neck. The headaches last 6-8 hours and interfere with his focus and concentration. They start 30 min after he wakes up in the morning. He has some associated nausea but is able to keep food down without vomiting. He has no auditory or visual aura, and has no tingling in his extremities. Taking tylenol extra strength helps, as well as coffee and naps. He denies any tearing of his eye, denies CP, SOB. He attributes the HA partly to his stressful job as an accountant. He normally has a chronic runny nose during this time of year from allergies, which is relieved by flonase normally, but he has not been using it lately. He denies cough, sore throat, nausea/vomiting/abd pain.**

Figure 1: An example of patient note (right) and checklist (left). The challenges to NLP include the use of synonyms of checklist items, non-standard abbreviations, different expressions of negation and non-continuous occurrence of checklist items in patient notes.

deemed acceptable. Notes are further complicated by indications of body side (right or left), frequent negations, strings of positive or negative findings, and nonstandard abbreviations used by learners. Learners may use medical terms to describe findings (cholelithiasis) or lay terms (gall stones) and are typically judged the same if correct. Ideally, the NLP model would directly identify the phrases in patient notes correlated with the given checklist items for the most granular grading analysis and feedback to learners in formative settings. Therefore, we study automatic patient note assessment as two tasks: (i) directly judging if the given checklist items are entailed in the patient notes (a natural language inference task), and (ii) identifying the phrases in patient notes correlated with the given checklist items (a named entity recognition task).

Despite its importance, the task of automatic grading of patient notes remains under-explored with only a few works that have studied it (Yim et al., 2019; Sarker et al., 2019). Traditional supervised models have been utilized for this task (Latifi et al., 2016; Yim et al., 2019), but are limited in scope because they rely on large scale annotated datasets. The significant manual effort associated with labeled dataset creation makes these methods difficult and impractical. Besides, the traditional supervised models trained on data with prior clinical cases will be less effective for new clinical cases. Another challenge lies in the inconsistency between the checklist item and the corresponding phrase(s) in the patient note owing to their being non-exact matches occurring as, for instance, synonyms or abbreviations.

To overcome the limitations of previous works and the challenges of traditional supervised models for a low-resource scenario, we propose our method without strong supervision. First we propose a simple baseline unsupervised method with a pipeline framework which could be used in a zero-resource scenario. Then we propose our weakly supervised method utilizing multi-level transfer learning, including data-level and task-level. Data-level transfer learning refers to the ability of transferring knowledge learned from data in one domain to another domain. Task-level transfer learning refers to the ability of transferring knowledge learned from one task to another task. A BERT model (Devlin et al., 2019) pretrained on biomedical texts and a publicly available dataset[2] are used for data-level transfer learning. A key assumption is that judging if the checklist item is entailed in the patient note and identifying the corresponding phrases in the patient note are mutually related and thus we treat the automatic grading as a multi-task learning problem. With experiments on our self-collected datasets, we show that our weakly supervised method achieves a state-of-the-art performance.

Overall, the main contributions are as follows:

- We study an under-explored task of automatic patient note assessment and apply novel NLP methods to solve this task.
- We propose propose a weakly supervised method utilizing multi-level transfer learning at both data- and task-level. Furthermore, a multi-task learning mechanism is proposed for task-level transfer.
- Experimental results on case-specific datasets show that our weakly supervised method achieves SOTA performance. A unique contribution of our work not studied before and critically important for a low-resource scenario is understanding the effect of out-of-domain data. Our analyses show that our method has the ability of data-level transfer learning and task-level transfer learning even using instances that are not case-specific.

[2] The USMLE® Step 2 Clinical Skills Patient Note was made available for research purposes by NBME and can be requested at https://www.nbme.org/services/data-sharing. For more details about the corpus, see (Yaneva et al., 2022).

## 2 Related Work

Being a research task that is currently under-explored, there are very few works studying automatic patient note assessment. The most closely related task that past works focus on is automatic short answer grading (ASAG) for scientific topics (Liu et al., 2016; Hermet et al.; Mitchell et al., 2002; Sukkarieh and Pulman, 2005; Sukkarieh and Bolge, 2010; Dzikovska et al., 2012; D'Mello et al., 2008; Zhu et al., 2022; Haller et al., 2022), which is different from complex domain-specific answer assessment (e.g. medical domain in our work). ASAG aims to grade free text that answers to a prompt categorically or numerically. Produced by ETS, C-rater (Leacock and Chodorow, 2003) is one example system for ASAG focusing on grading school-level examinations based on the presence or absence of required answers. Text goes through a sequence of NLP modules for spelling correction, syntactic analysis, pronoun resolution, morphological analysis and synonym detection. Generated canonical representations are then fed into a maximum entropy model for classification. (Nehm et al., 2012) also focused on a similar task of awarding content points for specific items for college biology essays. Two text analytic platforms are utilized: SPSS Text Analysis 3.0 (SPSSTA) relying on hand-crafted vocabulary and rules and Summarization Integrated Development Environment (SIDE) using a classic bag-of-words representation and support vector machine. With the development of transformers, different transformers and large pre-trained models including BERT and RoBERTa have also been applied (Zhu et al., 2022).

While there are some works on ASAG for scientific topics, only three works studied automatic patient note assessment (Latifi et al., 2016; Sarker et al., 2019; Yim et al., 2019). Inspired by the works on ASAG, the first two (Latifi et al., 2016; Yim et al., 2019) studied two systems: a feature based system including an n-gram feature extraction followed by a SVM and a simple BERT based neural network. The third (Sarker et al., 2019) followed previous works on ASAG and leveraged the pipeline framework. Their system employs a sequence of modules including text normalization, lexicon-based matching, fuzzy matching and supervised concept detection all utilizing significant manual annotation and brute force exhaustive searches. Inspired by these works, we also proposed a pipeline model without supervision, which could be used under zero-resource scenario. A key departure from the prior pipeline efforts is our non-reliance on task-specific manual annotation.

| Datasets | Headache | Abdominal Pain |
|---|---|---|
| Total num. of patient notes | 510 | 570 |
| Average Num. of Tokens in patient notes | 132.35 | 97.05 |
| Label Distribution | 258/252 | 337/233 |
| IAA | 0.916 | 0.938 |
| History Checklist | 11 | 8 |
| PEXAM Checklist | 1 | 6 |
| DDX Checklist | 5 | 5 |
| Total | 17 | 19 |

Table 1: Statistics of our datasets. **History**, **PEXAM** and **DDX** represents the number of checklist items on History, Physical Examination and Diagnose. **Total** represents the number of all checklist items. **Label Distribution** is represented as the number of label 1 and the number of label 0. **IAA** refers to inter-annotator agreement evaluated by Cohen's kappa coefficient.

However, the methods proposed in previous works are insufficient for the task of automatic patient note assessment. N-gram features and SVMs are limited for extracting linguistic and semantic features, especially for complex domain-specific text. BERT based model requires a huge amount of annotated data for training, which is usually unavailable for our task, whereas, pipeline models have the obvious problem of error propagation. Therefore, we also proposed an end-to-end model, which utilizes multi-level transfer learning to alleviate the dependence on annotated data.

## 3 Datasets

We used two datasets in our study. The first is a self-collected dataset on two clinical cases— headache and abdominal pain—collectively referred to as **case-specific** datasets. Data from the same clinical case is referred to as in-domain data and data from a different clinical case is referred to as out-of-domain data. Collected data pertains to patient notes written by examinees, where each note covers three sections: (i) *history*, (ii) *physical exam* and (iii) *differential diagnosis*. Patient note in each section should pertain to items from the same domain in the checklist, which includes 17 checklist items for the headache case and 19 for the abdominal pain case. The checklist item may contain fine-grained medical concepts (e.g., 'headache') and general descriptions (e.g., 'pain started two weeks ago'). The medical concepts included in the checklist items for different cases may be similar or vastly different, depending on the clinical condition being portrayed by the patient. As part of the grading process two expert raters, typically physician faculty members, are asked to judge if the checklist items are stated in the patient notes. Interannotator agreements on both cases are reported in Table 1. For both cases, the inter-annotator agreements are above 0.9, which shows the reliability of our constructed dataset. Additionally, for the purpose of our experiments the raters were asked to identify the phrases in the patient notes that correspond to the checklist items when the checklist item was matched. The tokens in the highlighted phrases were labeled following the BIO convention (Ramshaw and Marcus, 1999). Due to the cost of physician faculty rater time, we only collected data from 30 examinees. Of these patient notes from 25 examinees were used for fine-tuning and those from the remaining 5 were set aside for testing.

A second dataset is the USMLE® Step 2 Clinical Skills Patient Note (Yaneva et al., 2022), which contains a total of 43,985 patient note history portions from 10 clinical cases, where 2,840 patient notes (284 notes per case) were annotated with concepts from the exam scoring rubrics. At the time of the writing of this paper the dataset was used for a Kaggle competition on automated scoring of clinical patient notes[3], and only a subset of 100 patient notes from the annotated data were made available to the public (the remaining 184 notes per case were used as a test set for the competition). Therefore, the study presented here has used a subset of 100 annotated patient notes per case, which was not large enough to be directly used for training or fine-tuning the model but was still considered as a diverse but related dataset. This dataset is referred to as **generic** dataset in the rest of the paper.

## 4 Baseline Method

In this section, we introduce our proposed unsupervised method used as a baseline model for written patient note assessment. This approach utilized Amazon Comprehend Medical[4] for the purpose of medical entity extraction. Amazon Comprehend Medical is an API that performs various types of text analysis for the medical domain, and is a service that is provided by Amazon Web Services (AWS). We made use of the medical entity detection feature of this API, that allowed extraction and detection of six different types of medical entities: anatomy, medical conditions, medications, protected health information, test treatment procedures as well as time expressions in the medical context (Bhatia et al., 2019).

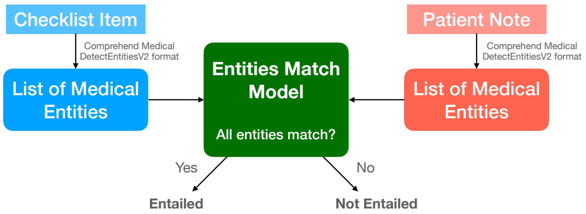


Figure 2: Unsupervised Method Model

### 4.1 Model Architecture

Here we describe the architecture of our unsupervised model presented in Figure 2. Taking the patient note $p = \{w_1^p, ..., w_n^p\}$ and checklist item $c = \{w_1^c, ..., w_m^c\}$ as input, the model aims to predict if the given checklist item is included in the given patient note.

**Medical Entity Extraction.** The first step converts the text of the checklists items and patient notes into the medical entities object format used by Amazon Comprehend Medical. The purpose is to then easily establish matching between medical entities extracted from checklist items and the medical entities extracted from patient notes.

**Medical Entity Match.** In the second step, we run a match-detection function based on each medical entity extracted from the checklist item and the medical entities extracted from the patient note. The match-detection function first filters the list of medical entities in the patient note by category. Comprehend Medical has six different entity categories, hence, if we are trying to find a match for a medical condition entity, then only medical condition entities are processed as potential candidate matches. This reduces the search space in the patient note based on medical entity categories. Once we obtain candidate matches by filtering based on these categories, we compare the similarity between the checklist item entity and the candidate entity from the patient note. If there is a surface level similarity (character-by-character equality), then we have found a match. If not, we compute a similarity score between the checklist item and the patient note medical entity using BioWordVec (Zhang et al., 2019). If the similarity score is beyond a certain threshold empirically chosen to be 0.8, only then we characterize the pair of entities

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

[4] https://aws.amazon.com/cn/comprehend/medical/

as a match. A checklist item is considered entailed by the patient note if all of the medical entities in the checklist item have a match in the patient note.

## 5 Weakly Supervised Method

In this section, we provide the details of our proposed weakly supervised method. In our work, the first task of judging checklist items' entailment by the patient note is formulated as a natural language inference task. The second task of identifying phrases that correspond to a checklist item can be treated as labeling the span of corresponding phrases, which is similar to named entity recognition. Therefore, we refer to it as the NER-related task. These two tasks are mutually beneficial in our setting; identification of corresponding phrases directly means the checklist item is entailed by the patient note and the entailment of checklist item indicates that the corresponding phrases are in the patient note. In order to harness this mutual benefit, we propose a multi-task transfer learning setting with a mutual feedback mechanism. Using this method, data from different clinical cases could help the model to learn the basic concepts of our tasks and build appropriate representation for underlying medical concepts. Therefore, we also utilize data from different clinical cases for transferring common medical and task knowledge. Finally, we propose a multi-level transfer learning method including task-level and data-level transfer learning which removes the need for large-scale annotated corpora and is thus weakly supervised.

### 5.1 Model Architecture

Here we describe the architecture of our model, which is related to the task-level transfer learning. Figure 3 shows the architecture of our multi-level transfer learning model. Taking the patient note $p = \{w_1^p, ..., w_n^p\}$ and the checklist item $c = \{w_1^c, ..., w_m^c\}$ as input, the model aims to predict if the given checklist item is entailed by the given patient note and also identifies the span of the expressions corresponding to the given checklist item. BIO labels are used to label the span of the target phrases. In our model, the lower encoder layers are used for extracting the hidden representations of the input text and are shared across all tasks and data while the top task-specific layers with a mutual feedback mechanism are used for different tasks. The mutual feedback mechanism is used for sharing knowledge across different tasks via outputs of different task-specific layers. The architecture details are as follows:

**Encoder Layers.** The encoder layers are used to extract contextual embeddings for input text. We use BERT model as our encoder shared across different tasks. For BERT model, [CLS] is used at the start of the input and [SEP] is used to separate patient note and checklist item. Therefore, the final input to the encoder is $\{[\text{CLS}], w_1^p, ..., w_n^p, [\text{SEP}], w_1^c, ..., w_m^c, [\text{SEP}]\}$. The output contextual embeddings would be $X = \{x_{[\text{CLS}]}, x_1^p, ..., x_n^p, x_{[\text{SEP}]}, x_1^c, ..., x_m^c, x_{[\text{SEP}]}\}$.

**Task-Specific Layers.** For task-specific layers, different layers take different outputs of encoder layers as input. For NLI task, the contextual embedding $x_{[\text{CLS}]}$ is used as input because the whole sequence information are encoded into this embedding (Devlin et al., 2019). For NER task, the contextual embeddings of each token in the patient note $\{x_1^p, ..., x_n^p\}$ are used:

$$[p_s, p_{ns}] = \text{NLI}(x_{[\text{CLS}]})$$
$$[p_B^i, p_I^i, p_O^i] = \text{NER}(x_i^p)$$

where $\text{NLI}(\cdot)$ represents the NLI task layer and $[p_s, p_{ns}]$ is the output distribution with $p_s$ as the probability of checklist item is stated and $p_{ns}$ as the probability of checklist item is not stated. $\text{NER}(\cdot)$ represents the NER task layer and $[p_B^i, p_I^i, p_O^i]$ is the output distribution with $p_B^i$ as the probability of token $i$ is predicted as the beginning of the target phrase, $p_I^i$ as the probability of token $i$ is predicted as inside the target phrase and $p_O^i$ as the probability of token $i$ is predicted as outside the target phrase.

**Mutual Feedback Mechanism** As stated before, the NLI task and NER task can actually benefit each other. Therefore, the output of one task could be used to enhance the input of another task. Then the enhanced inputs would be fed into two new task-specific layers for these two tasks. For the NLI task, the output from the previous NER task layer is used to enhance the input as follows:

$$[p_B^{ave}, p_I^{ave}, p_O^{ave}] = [\frac{1}{n}\sum_{i=1}^{n} p_B^i, \frac{1}{n}\sum_{i=1}^{n} p_I^i, \frac{1}{n}\sum_{i=1}^{n} p_O^i]$$
$$\hat{x}_{[\text{CLS}]} = \text{cat}(x_{[\text{CLS}]}, [p_B^{ave}, p_I^{ave}, p_O^{ave}])$$
$$[\hat{p}_s, \hat{p}_{ns}] = \text{NLI}_{new}(\hat{x}_{[\text{CLS}]})$$

where $[\hat{p}_s, \hat{p}_{ns}]$ is the final output distribution for the whole sequence. The average of the output distribution over all the tokens in the patient note is used to enhance the input. Therefore, if the

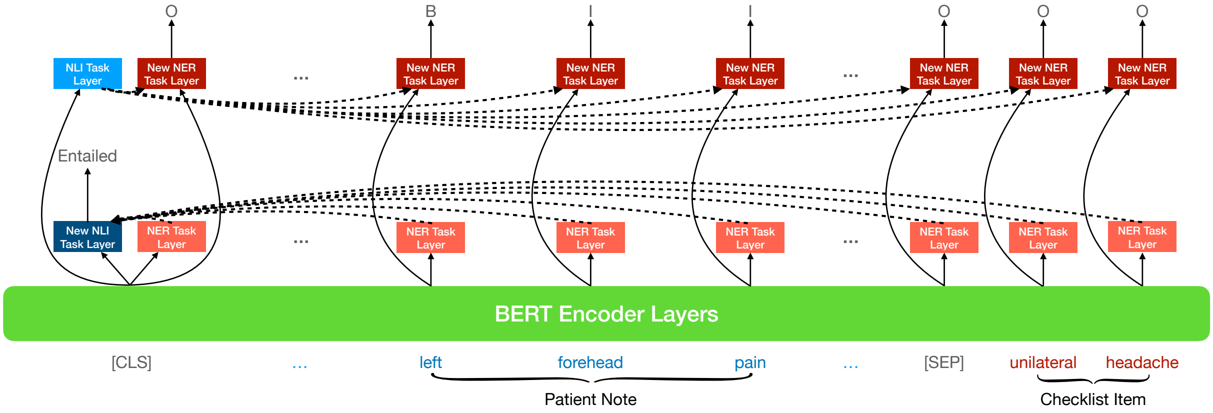


Figure 3: Architecture of our weakly supervised method. Dashed arrows represent outputs from NLI and NER task layers that are used to enhance the input to the new task layers. New NLI and NER task layers take outputs from NLI and NER task layers and outputs from the BERT encoder layers as input and generate the final outputs.

target phrase corresponding to the checklist item is identified, $p_B^{ave}$ and $p_I^{ave}$ would be non-zero, which could be used to guide the new NLI layer. Similarly, for the NER task, the output from previous NLI task layer is used to enhance the input:

$$\hat{x}_i^p = \text{cat}(x_i^p, [p_s, p_{ns}])$$
$$[\hat{p}_B^i, \hat{p}_I^i, \hat{p}_O^i] = \text{NER}_{new}(\hat{x}_i^p)$$

where $[\hat{p}_B^i, \hat{p}_I^i, \hat{p}_O^i]$ is the final output distribution for token $i$. The output distribution from previous NLI layer is used to enhance the input. If the checklist item is found to be stated in the patient note, $p_s$ would be much larger than $p_{ns}$ which could be used as a guidance for new NER layer. Finally, the enhanced input $\hat{x}_i^p$ is fed into the new NER layer.

### 5.2 Training Protocol

We use a simple joint training objective for our model, which is the sum of the sequence classification loss and the token classification loss, each of which is given by the corresponding cross-entropy loss. This training allows the task-level transfer as shown in Section 8.2.

The model is first trained with the generic dataset to learn the basic concept pertaining to the two tasks and the common medical/clinical knowledge. Then for new clinical cases with a few annotated instances, the model is fine-tuned with the case-specific data. Our hypothesis is that with the knowledge of the two tasks and the common medical knowledge learned during training, the model should be able to transfer to new clinical cases without the need for a large scale annotated dataset. In addition, for new clinical cases without any annotated data, our model can still be used because the knowledge of the two tasks and the common medical knowledge learned during training can be transferred to new clinical cases. The ability of data-level transfer is presented in Section 8.1.

## 6 Experiments

### 6.1 Baselines

Due to the fact that related prior works did not release their codes and did not provide enough details for reproduction, we only test one baseline model for comparison with our proposed unsupervised and weakly supervised methods on the NLI-related task. Besides, we also use one baseline model for comparison on the NER-related task.

- **NLI model**: For the NLI-related task of judging if the given checklist item is stated in the patient note, a simple BERT sentence pair classification model is used as the baseline with only the sequence classification loss as the training objective.
- **NER model**: For the NER-related task of identifying corresponding phrases in the patient note given checklist items, a simple BERT token classification model is used as baseline, which generates BIO labels to label the span of the target phrases. For this NER model, only the token classification loss is used as the training objective.

For both the baselines, the experimental settings and the parameters are set to be the same as those in our weakly supervised method. In addition, the baseline models and our weakly supervised model are trained on the same data but with different la-

| Methods | Headache | | | | Abdominal Pain | | | |
|---|---|---|---|---|---|---|---|---|
| | History | PEXAM | DDX | Total | History | PEXAM | DDX | Total |
| **Unsupervised** | 0.72 | 0.30 | 0.87 | 0.63 | 0.68 | 0.58 | 0.93 | 0.73 |
| **NLI baseline** | 0.83 | 0.88 | 0.89 | 0.87 | 0.88 | 0.70 | 0.89 | 0.82 |
| **Weakly Supervised** | 0.91 | 0.94 | 0.94 | 0.93 | 0.91 | 0.90 | 0.93 | 0.91 |

Table 2: Performance of different methods on NLI-related task. Accuracy is used for evaluation. **History**, **PEXAM** and **DDX** represents the accuracy averaged on History, Physical Examination and Diagnose checklist items respectively. **Total** represents the accuracy averaged on all checklist items

| Methods | Headache | | | | Abdominal Pain | | | |
|---|---|---|---|---|---|---|---|---|
| | History | PEXAM | DDX | Total | History | PEXAM | DDX | Total |
| **NER baseline** | 0.56 | 0.54 | 0.49 | 0.53 | 0.57 | 0.55 | 0.53 | 0.55 |
| **Weakly Supervised** | 0.60 | 0.59 | 0.63 | 0.61 | 0.61 | 0.62 | 0.63 | 0.62 |

Table 3: Performance of different methods on NER-related task. F1 score is used for evaluation.

bels. That is, for the NLI model, only the sequence-level labels indicating if the given checklist item is entailed or not are used. For the NER model, only the token-level labels indicating if each token belongs to the target phrase are used for training.

### 6.2 Evaluation Metrics

For the different tasks, different evaluation metrics are used. Accuracy defined as $\frac{\text{Num of Correct Predictions}}{\text{Num of All Predictions}}$ is used for NLI-related task, whereas the NER-related task, we use the F1 score, which is widely used for NER.

## 7 Results

The performances of the different models on the NLI task of judging if the checklist item is entailed by the patient note are summarized in Table 2. We find that our proposed unsupervised framework achieves an average accuracy of 0.63 across all the checklist items on the headache dataset and an average accuracy of 0.73 on the abdominal pain dataset. Compared with the unsupervised method, our weakly supervised method achieves a much better performance showing an average accuracy of 0.93 on the headache dataset and 0.91 on the abdominal pain dataset. As shown in Table 2, our proposed weakly supervised method outperforms the baseline NLI model and the unsupervised method across all the sections (checklist items averaged by section—history, physical exam and diagnosis) by a large margin. Looking at the accuracy values averaged across each section, we notice that our proposed weakly supervised method performs consistently well on all the checklist types whereas the baseline NLI model and the unsupervised method only perform well on specific sections.

| Number | Headache Case | | Abdominal Pain | |
|---|---|---|---|---|
| | NLI | NER | NLI | NER |
| **0** | 0.89 | 0.32 | 0.82 | 0.42 |
| **1** | 0.90 | 0.41 | 0.87 | 0.50 |
| **5** | 0.92 | 0.54 | 0.89 | 0.56 |
| **10** | 0.93 | 0.60 | 0.90 | 0.60 |
| **15** | 0.93 | 0.61 | 0.91 | 0.62 |
| **20** | 0.93 | 0.60 | 0.90 | 0.62 |
| **25** | 0.93 | 0.61 | 0.91 | 0.62 |

Table 4: Performance of weakly supervised method. The weakly supervised method is fine-tuned on different number of in-domain data. **NLI** refers to NLI-related task that is evaluated by accuracy. **NER** represents NER-related task that is evaluated by F1 score[5].

For the NER task of identifying the corresponding phrases, our weakly supervised method achieves an F1 score of 0.61 on the headache dataset and 0.62 on abdominal pain dataset, which is better than the performance of the baseline NER model as shown in Table 3. This demonstrates that our proposed weakly supervised method utilizing multi-level transfer learning achieves the SOTA performance in both tasks when compared to all the baselines and our unsupervised method.

## 8 Analysis

In this section, we provide some ablation studies to analyze the contribution of data and the different modules used in our weakly supervised method.

[5] Due to the space limitation, the analysis on the number of in-domain data is provided in the appendix.

| Methods | Headache | | | | Abdominal Pain | | | |
|---|---|---|---|---|---|---|---|---|
| | History | PEXAM | DDX | Total | History | PEXAM | DDX | Total |
| **Unsupervised** | 0.72 | 0.30 | 0.87 | 0.63 | 0.68 | 0.58 | 0.93 | 0.73 |
| **Weakly Supervised wo Fine-tuning** | 0.83 | 0.88 | 0.89 | 0.87 | 0.88 | 0.70 | 0.89 | 0.82 |
| **Weakly Supervised + Out-of-domain Data** | 0.83 | 0.94 | 0.89 | 0.89 | 0.87 | 0.81 | 0.90 | 0.86 |
| **Weakly Supervised + In-domain Data** | 0.91 | 0.94 | 0.94 | 0.93 | 0.91 | 0.90 | 0.93 | 0.91 |

Table 5: Performance of our methods on the NLI-related task. Our weakly supervised method is fine-tuned using different data sizes to show the ability of data-level transfer learning. Accuracy is used as the evaluation metric.

| Tasks | Methods | Headache | | | Abdominal Pain | | |
|---|---|---|---|---|---|---|---|
| | | No Fine-tune | Out-of-domain | In-domain | No Fine-tune | Out-of-domain | In-domain |
| **NLI** | **NLI baseline** | 0.82 | 0.84 | 0.87 | 0.77 | 0.80 | 0.82 |
| | **Ours** | 0.87 | 0.89 | 0.93 | 0.82 | 0.86 | 0.91 |
| **NER** | **NER baseline** | 0.08 | 0.36 | 0.53 | 0.08 | 0.40 | 0.55 |
| | **Ours** | 0.32 | 0.47 | 0.61 | 0.42 | 0.51 | 0.62 |

Table 6: Comparison between baseline models with single-task training and our weakly supervised model with multi-task training. For the NLI task, accuracy is used for evaluation, and for the NER task, F1 score is used.

### 8.1 Data-Level Transfer

Here we explore our method's data-level transfer learning ability, which is reflected in the performance of the model with the out-of-domain data. Two settings are used for our experiments. In the first setting we train our weakly supervised model with the generic dataset and then directly test the model on the case-specific dataset with no fine-tuning. In the second setting, we train our weakly supervised model with the generic dataset and then fine-tune it on the case-specific dataset on one of the two clinical cases. After training and fine-tuning, we test our model on the case-specific dataset related to another clinical case (e.g., fine-tuning on abdominal pain and testing on headache).

From the results in Table 5, for the first setting, we see that our model outperforms the other models even when trained on the generic dataset alone. For the second setting, when fine-tuned on the abdominal pain dataset and tested on the headache case-specific dataset, our model's performance improved from 0.87 to 0.89. Similarly, when fine-tuned on the headache dataset and tested on the abdominal pain dataset, our model's performance improved from 0.82 to 0.86. This shows that even the out-of-domain data can aid the performance of our weakly supervised method, suggesting that our method can transfer knowledge learned from out-of-domain data to new cases.

In addition, we also provide an analysis on the influence of training data amount on the performance in the Appendix. It is concluded that our weakly supervised method only requires a small number of in-domain data for fine-tuning to achieve a satisfactory performance for both tasks.

### 8.2 Task-Level Transfer

In this part we show our weakly supervised method's ability of task-level transfer learning via comparison between our model with multi-task training and the baseline models with single-task training. Our model and the baseline models are trained on the same datasets for a fair comparison.

Results are presented in Table 6. When only trained on the generic data and tested on the NLI-related task (corresponding to **No Fine-tune** columns), our model with multi-task training has an averaged accuracy of 0.87 on the headache case and 0.82 on the abdominal pain case whereas the NLI baseline model has an accuracy of only 0.82 on headache case and 0.77 on abdominal pain case. For the NER-related task, when only trained on the data from kaggle dataset, our model has an averaged F1 score of 0.32 on the headache case and 0.42 on the abdominal pain case whereas the NLI baseline model has an averaged F1 score of only 0.08 on headache and 0.08 on abdominal pain.

When in-domain data is used for fine-tuning, our model with multi-task training still outperforms the NLI and NER baseline models on both tasks. When trained using the generic data, fine-tuned on the in-domain data and tested on the NLI-related task (corresponding to **In-domain** columns), our model with multi-task training has an averaged accuracy of 0.93 on the headache case and 0.91 on abdominal pain case whereas the NLI baseline model has an averaged accuracy of only 0.87 on headache case and 0.82 on abdominal pain case. For the NER-related task, our model has an averaged F1 score of 0.61 on headache case and 0.62 on abdominal pain case whereas the NLI baseline model has an averaged F1 score of 0.53 on headache case and 0.55 on abdominal pain case.

Similarly, when out-of-domain data is used for fine-tuning (corresponding to **Out-of-domain** columns), our model with multi-task training also outperforms the NLI and NER baseline models on both tasks. Therefore, as shown by the higher performance compared with the NLI and NER baseline models, our weakly supervised model benefits from the multi-task training and shows a strong ability of task-level transfer learning.

## 9 Practical Impact

Our system is currently undergoing pilot testing by learners and faculty to assess the perceived impact on providing more immediate and automated feedback. The immediacy is important so that the case is fresh, and it will likely impact debriefing of simulation cases, potentially making debriefing more focused on areas of learner struggles identified in the notes. With our system grading all learners, automated feedback could be provided to the users and learners in time, which can be used to help their study of patient notes writing.

## 10 Conclusion and Future Work

In this paper, we study the problem of automatic written medical examination assessment. The complexity and huge manual effort required make data resources for this task very limited. Therefore, traditional NLP systems relying on large annotated corpora are impractical. With these factors in mind, we proposed a weakly supervised method. Our weakly supervised method utilizes multi-level transfer learning including data-level transfer learning and task-level transfer learning to simultaneously judge if the checklist item is stated in the patient note and also identify spans of relevant phrases. Experiments on two self-collected datasets show that our weakly supervised method is able to achieve the SOTA performance on both tasks. Therefore, our weakly supervised method can correctly judge if the checklist item is stated in the given patient note and can also find the relevant phrases most of the time. Our future work involves developing more effective transfer learning mechanisms to improve the performance on identifying the relevant phrases.

## Acknowledgement

This project was funded (in part) by a National Board of Medical Examiners (NBME) Edward J. Stemmler, MD Medical Education Research Fund grant. The project and the views expressed in this publication do not necessarily reflect the position or policy of NBME, and NBME support provides no official endorsement.

The authors wish to acknowledge: Dr. Yoon Soo Park from the University of Illinois College of Medicine at Chicago for contributions to the Stemmler Fund project; Internal Medicine clerkship directors Drs. Saurabh Bansal and Manajyoti Yadav from the University of Illinois College of Medicine at Peoria for grading assistance. We also thank our research staff, including Rebecca Ebert-Allen, Rebecca Ruger, Alyse Dutcher, Ryan Klute for their assistance during the project.

## References

Parminder Bhatia, Busra Celikkaya, Mohammed Khalilia, and Selvan Senthivel. 2019. Comprehend medical: a named entity recognition and relationship extraction web service. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1844–1851. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Myroslava O Dzikovska, Rodney Nielsen, and Chris Brew. 2012. Towards effective tutorial feedback for explanation questions: A dataset and baselines. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 200–210.

Sidney D'Mello, Tanner Jackson, Scotty Craig, Brent Morgan, P Chipman, Holly White, Natalie Person, Barry Kort, R El Kaliouby, Rosalind Picard, et al. 2008. Autotutor detects and responds to learners affective and cognitive states. In *Workshop on emotional and cognitive issues at the international conference on intelligent tutoring systems*, pages 306–308.

George Engelhard Jr, Jue Wang, and Stefanie A Wind. 2018. A tale of two models: Psychometric and cognitive perspectives on rater-mediated assessments using accuracy ratings. *Psychological Test and Assessment Modeling*, 60(1):33–52.

Stefan Haller, Adina Aldea, Christin Seifert, and Nicola Strisciuglio. 2022. Survey on automated short answer grading with deep learning: from word embeddings to transformers. *arXiv preprint arXiv:2204.03503*.

Matthieu Hermet, Stan Szpakowicz, and Lise Duquette. Automated analysis of students' free-text answers for computer-assisted assessment1.

Syed Latifi, Mark J Gierl, André-Philippe Boulais, and André F De Champlain. 2016. Using automated scoring to evaluate written responses in english and french on a high-stakes clinical competency examination. *Evaluation & the health professions*, 39(1):100–113.

Claudia Leacock and Martin Chodorow. 2003. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):389–405.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas. Association for Computational Linguistics.

Tom Mitchell, Terry Russell, Peter Broomhead, and Nicola Aldridge. 2002. Towards robust computerised marking of free-text responses. In *Proceedings of the 6th International Computer Assisted Assessment (CAA) Conference*.

Ross H Nehm, Minsu Ha, and Elijah Mayfield. 2012. Transforming biology assessment with machine learning: automated scoring of written evolutionary explanations. *Journal of Science Education and Technology*, 21(1):183–196.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Abeed Sarker, Ari Z Klein, Janet Mee, Polina Harik, and Graciela Gonzalez-Hernandez. 2019. An interpretable natural language processing system for written medical examination assessment. *Journal of biomedical informatics*, 98:103268.

Jana Sukkarieh and Eleanor Bolge. 2010. Building a textual entailment suite for the evaluation of automatic content scoring technologies. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

Jana Z Sukkarieh and Stephen G Pulman. 2005. Information extraction and machine learning: Auto-marking short free text responses to science questions. In *Proceedings of the 2005 conference on artificial intelligence in education: Supporting learning through intelligent and socially informed technology*, pages 629–637.

Jason T Tsichlis, Andrew M Del Re, and J Bryan Carmody. 2021. The past, present, and future of the united states medical licensing examination step 2 clinical skills examination. *Cureus*, 13(8).

Victoria Yaneva, Janet Mee, Le Ha, Polina Harik, Michael Jodoin, and Alex Mechaber. 2022. The usmle® step 2 clinical skills patient note corpus. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2880–2886.

Wen-wai Yim, Ashley Mills, Harold Chun, Teresa Hashiguchi, Justin Yew, and Bryan Lu. 2019. Automatic rubric-based content grading for clinical notes. In *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, pages 126–135.

Yijia Zhang, Qingyu Chen, Zhihao Yang, Hongfei Lin, and Zhiyong Lu. 2019. Biowordvec, improving biomedical word embeddings with subword information and mesh. *Scientific data*, 6(1):1–9.

Xin Hua Zhu, Han Wu, and Lanfang Zhang. 2022. Automatic short answer grading via bert-based deep neural networks. *IEEE Transactions on Learning Technologies*, pages 1–1.

# A Appendix

## A.1 Training Data Amount

First, we analyze the influence of training data amount on the performance. We use different amounts of in-domain data to fine-tune our weakly supervised method. As shown in Table 4, we notice that when the data used for fine-tuning is less than 10 patient notes the amount of training data has a big influence on the performance, with the performance improving with the increase of training data. When the training data is increased from 0 to 10 patient notes, the averaged accuracy increased from 0.89 to 0.93 and the F1 score increased from 0.32 to 0.6 on headache case. case of 0 training instances corresponds to the out-of-the-box performance of the weakly supervised model. On the abdominal pain case, the averaged accuracy increased from 0.85 to 0.9 and the F1 score increased from 0.42 to 0.6. However, when the data used for fine-tuning is more than 10, the performance did not change significantly with the increase of training data. Based on this, we note that our weakly supervised method only requires a small number of in-domain data for fine-tuning to achieve a satisfactory performance for both tasks.

## A.2 Limitations

Although our weakly supervised model shows a satisfactory performance on NLI-related task after fine-tuning on in-domain data, the performance on NER-related task is still limited. Therefore, our weakly supervised model is limited on relating phrases in the patient notes with the given checklist item. Besides, without fine-tuning on in-domain data, the performance on NLI-related task is not good enough, which means that our weakly supervised model still relies on annotated in-domain data. In addition, it is obvious that our unsupervised model has a much worse performance compared with our weakly supervised model. Therefore, one important limitation lies on the relying of in-domain data given that unsupervised model's performance is unsatisfactory and weakly supervised model need in-domain data for fine-tuning.

Another important limitation lies in the data used for testing. In our experiments, we only use patient notes from 5 examinees for testing, which is not a large test set. Therefore, future studies should consider validating these results with larger samples and wider variety of cases.