CXR data annotation and classification with pre-trained language models

Nina Zhou[1], Ai Ti Aw[1], Zhuohan Liu[1],
Cher Heng Tan[2], Yonghan Ting[2], Wenxiang Chen[2], and Sim Zheng Ting[3]
[1]Institute for Infocomm Research(I2R), Singapore,
[2]Tan Toch Seng Hospital (TTSH), Singapore, [3]MOH Holdings (MOHH), Singapore
{zhoun,aaiti,Liu_zhuohan}@.i2r.a-star.edu.sg
{cher_heng_tan,Yonghan_Ting,chen_wen_xiang}@.ttsh.com.sg
Jordan.sim@mohh.com.sg

Abstract

Clinical data annotation has been one of the major obstacles for applying machine learning approaches in clinical NLP. Open-source tools such as NegBio and CheXpert are usually designed on data from specific institutions, which limit their applications to other institutions due to the differences in writing style, structure, language use as well as label definition. In this paper, we propose a new weak supervision annotation framework with two improvements compared to existing annotation frameworks: 1) we propose to select representative samples for efficient manual annotation; 2) we propose to auto-annotate the remaining samples, both leveraging on a self-trained sentence encoder. This framework also provides a function for identifying inconsistent annotation errors. The utility of our proposed weak supervision annotation framework is applicable to any given data annotation task, and it provides an efficient form of sample selection and data auto-annotation with better classification results for real applications.

1 Introduction

Previous work (Wang et al., 2019) has showed clinical text classification can significantly benefit from supervised learning approaches when annotated data is available. However, annotation of clinical data is extremely expensive and time consuming since annotation needs human experts' domain knowledge. To circumvent this, some rule-based methods have been developed using expert knowledge, e.g., NegBio (Peng et al., 2018) and CheXpert (Irvin et al., 2019) using rule labeler to automatically detect the presence of observations in radiology reports. McDermott et al. (2020) mentioned that CheXpert is computationally slow, and its output is non-differentiable, so they proposed to train a BERT based classier (CheXpert++) based on the output of CheXpert. Likewise, Smit et al. (2020) proposed to combine automatic labelers with expert knowledge by first fine-tuning BERT classifier on output of CheXpert and then on a small set of expert annotations augmented with automated backtranslation.

While annotation is data and task specific (Irena et al., 2020), it is further complicated by the differences in writing style, structure, language use such as the vocabularies and phrase variability among different institutes and different countries. For example, phrases "airspace changes" and "infective change(s)" are commonly seen in our local data to describe pneumonia, but rarely seen in MIMIC data (Alistair et al., 2019). In addition, label definitions vary among institutes. For example, CheXpert classifies the sentence "suspicious for pneumonia" as "pneumonia uncertainty" based on their rule definition, while our clinicians/radiologists would consider it as an implication of "pneumonia positive". These differences limit the application of open-source tools on different data, and further limit the applications of methods (McDermott et al., 2020; Smit et al., 2021) which heavily rely on open-source tools.

Some active learning (Chen et al., 2015) and interactive learning methods (Wang et al., 2017) have been commonly used for reducing the experts' annotation burden. Chen et al. (2015)

Proceedings of the 29th International Conference on Computational Linguistics, pages 2801–2811
October 12–17, 2022.

proposed uncertainty-based and diversity-based sampling to annotate clinical NERs. Both sampling approaches adopt random sampling and longest sentence sampling, for comparison, to build the initial set for manual annotation. For subsequent annotation and updating, the uncertainty-based sampling relies on model's predictions while diversity-based selects samples based on pair-wise sentence similarity. Pair-wise sentence similarity is calculated based on individual words, syntax or clinical concepts, with the aim to select samples with lower similarity to annotated samples in the initial set. Wang et al. (2017) proposed an interactive learning method, ReQ-ReC, which is very similar to the uncertainty-based sampling. The method leverages on human experts' domain knowledge to build a list of sense-specific contextual words and use them to search for related sentences to form the initial annotation set. For subsequent sampling, it is based on the model's prediction too. The more ambiguous samples will be selected for annotation.

For the above approaches, it is not efficient in practice since they incur many rounds of model retraining (Chen et al., 2015) and multiple cycles of annotation by experts. Time taken for human annotation is normally affected more by the duration of the annotation cycle than by the sample sizes as experts are not readily available, especially in the clinical domain. Pair-wise sentence similarity is limited by using words, syntax or extracted clinical concepts to represent sentence since they cannot capture the semantic meaning of the whole sentence. In our data, it is very common to have sentences with the same clinical concepts annotated differently due to negation or speculation. In addition, distribution and the number of samples selected for initial annotation affects the performance of the model, which will then affect the prediction quality of the remaining samples.

To effectively select samples for initial annotation and avoid multiple training cycles and annotation by human experts, in this paper, we propose a new weak supervision annotation framework to overcome the retraining and multiple annotation process. Within the proposed framework, we adopt deep neural networks (DNN) for sample selection and text classification, which can fill the gap of using DNNs in active learning for text classification (Schroder et al., 2020). Our work has the following contributions:

1) We propose a generic weak supervision data annotation framework which relies on sentence embedding for sample selection, error checking and auto-annotation.
2) We propose to select representative samples through sentence clustering to kick start the human annotation process, which is a more efficient approach than random selection and longest sentence selection (Chen et al., 2015).
3) We show that our proposed annotation and training approach achieves better performance and requires fewer number of annotated samples.

2 Related Works

Supervised Sentence Encoder. Earlier sentence encoders are trained in supervised way. InferSent (Conneau et al., 2017) is trained on Stanford natural language inference (SNLI) data with three labels. Universal Sentence Encoder (Cer et al., 2018) augments unsupervised learning on labelled SNLI dataset for improved performance. Reimers et al. (2019) proposed SBERT, built by adding a Siamese network on top of BERT model and then fine tuning on NLI data sets. Their experimental results show that SBERT achieves much better results compared to InferSent and Universal Sentence Encoder on STS tasks (Reimers et al., 2019). For other tasks and domains, retraining SBERT on domain sentence pairs with labels are also preferred.

Unsupervised Sentence Encoder. As to unsupervised approaches, with the advent of pretrained language models (PLMs), Devlin et al. (2018) tried to get sentence embeddings from BERT by either averaging the vectors obtained from the last layer or using [CLS] token. Recently, Wang et al. (2021) proposed the transformer-based sequential denoising auto-Encoder (TSDAE) method by exploiting the encoder-decoder structure of transformer. During training, the encoder converts corrupted sentences into fixed-sized vectors and the decoder reconstructs the original sentences from this fixed-sized vectors. To make reconstruction as good as possible, the sentence embedding from the encoder must well represent the semantic meaning of the sentences. At inference, only the encoder will be used for generating sentence embeddings.

Sentence Textual Similarity (STS) in clinical domain. In clinical domain, there are also related work for evaluating sentence similarity. Mahajan et al. (2020) proposed an iterative intermediate training (IIT) approach for calculating clinical STS by using multi-task learning (MTL). The final system attains promising results for clinical STS tasks by integrating module of Clinical BERT with other language models (BioBERT, MT-DNN, RoBERTa). But the method is not efficient for training sentence encoder as it involves high computation cost for the various pairs of regression tasks due to many possible combinations. Wang et al. (2020b) proposed to take advantage of general domain STS dataset and a small-scale in-domain training data to achieve an impressive result for clinical STS task.

In this paper, we present both supervised and unsupervised sentence encoder training methods in our weak supervised framework for CXR data annotation (See Section 3.2). We evaluate both sentence encoders using a small set of CXR data with pathology labels for a multi-label classification task. We show that our proposed weak supervision framework is effective for semi-supervised data annotation.

3 Methodology

Diseases Sentences	CXR diseases annotation			
	CAT 1	CAT 2	CAT 3	CAT 4
atelectasis is seen at the right lower zone with ***vague air-space changes.***	u	×	×	+
the heart size cannot be accurately assessed in this projection but ***appears to be enlarged. no obvious consolidation is seen.***	-	×	+	×

Table 1. Annotation examples of our CXR data. '×' indicates the pathology is not mentioned

Our annotation task is to label each sentence in the CXR report into four pathologies, mainly pneumonia (CAT1), pneumothorax (CAT2), cardiomegaly (CAT3) and other diseases (CAT4) as illustrated in Table 1. For each pathology, we further label it as being 'positive'(+), 'negative'(-), or 'uncertain'(u). One sentence may have more than one pathology with one pathology as positive, while describing another pathology as negative or uncertain. If there are no pathologies described, we label it as 'no findings'.

3.1 The Proposed Framework

Our proposed framework is depicted in Figure 1. This proposed new weak supervision annotation framework is applicable to any data annotation task. It exploits an efficient sentence encoder to get high quality sentence embeddings. Using these embeddings, 1) we perform unsupervised clustering to obtain the natural grouping of data based on its distribution. 2) We then select representative sentences for human annotation from each sentence cluster using sematic similarity score. 3) Auto error checking is then performed on the human annotations to reduce bias and inconsistences caused by human errors. 4) We then perform automatic annotation on the remaining data in each cluster by measuring their semantic similarity with the annotated samples. Using this approach, we are able to obtain a set of high-quality data for our classification task.

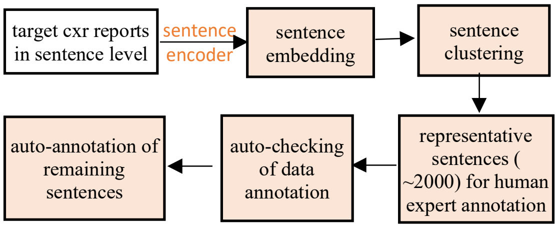
target cxr reports in sentence level
sentence encoder
sentence embedding
sentence clustering
representative sentences (~2000) for human expert annotation
auto-checking of data annotation
auto-annotation of remaining sentences

Figure 1. Proposed weak supervision annotation framework. Sentence Encoder is on our CXR data. Auto-checking checks the inconsistence and bias in the human annotation.

3.2 Sample Selection Strategy

We used HDBScan as our clustering method and divide the clusters into noisy clusters and clean clusters. We define the first cluster as *Noisy Cluster* as HDBScan always put all sentences it cannot group into this cluster. This noisy cluster has many variations on sentence length and keywords (mentions). For example, in Cluster 1 of Table 2, sentence 1 mentions both *cardiomegaly* (enlarged heart) and *pleural effusion* (water in the lung), which are keywords used in CAT3 and CAT4 categories. Sentence 2 is much shorter with only the keyword "*cardiac*" without further information. As the ambiguity is high in this pool, it would be a good source of sentences for human annotation. In our experiment, we selected all samples from this noisy cluster for expert annotation.

We treat all other clusters as clean clusters and further divided them into *Clean-Relevant* and *Clean-Irrelevant*. *Clean-Irrelevant* (see Table 2) *refers* to clusters where most sentences do not mention keywords in any pathologies, and therefore are easily annotated, so we just choose 1 or 2 samples for expert annotation. *Clean-Relevant* clusters normally group sentences with the same pathology, but some sentences may describe more than one or different pathology with similar writing pattern. We selected more samples from them for expert annotation to have a more effective training dataset.

Cluster 1: (*Noisy Custer*) 1. comparing with the previous x-ray dated 0/0/0 findings, cardiac size cannot be completely assessed on the given projection there is increasing homogeneous opacification of the right hemithorax, likely in keeping with underlying pleural effusion. 2. suggest correlation cardiac size cannot be assessed on this suboptimal study 3. nipple markers are noted. … …
Cluster 2: (*Clean-Relevant*) 1. cardiac size is enlarged with perihilar vascular prominence. 2. cardiac silhouette appears enlarged with prominent hilar vessels and upper lobe diversion 3. cardiac size is enlarged with mild perihilar vascular congestion and bilateral perihilar air space shadows. ……
Cluster 3: (*Clean-Irrelevant*) 1. previous image done on 16 July 2016 is reviewed. 2. findings were noted at time of reporting. 3. comparison made with previous study dated 17 Apr. 2013. ……

Table 2. Examples of the *Noisy, Clean-Relevant, Clean-Irrelevant* clusters

Our data selection strategy can be illustrated in *Equation (1)*. Assuming there are N clean clusters and C_i indicates the i-th cluster ($i = 1,2 \ldots N$), n_i indicates the number of samples to be selected from C_i. We first rank sentences based on their length for each cluster. Then we select l_i samples comprising the longest one ($n_{longest}$), the shortest one ($n_{shortest}$) and the one with medium length (n_{medium}). For each sentence in $l_i$samples, we use our trained sentence encoder to compute the cosine similarity between it with the rest of the sentences and select the least similar sentences to obtain n_i samples for human annotation.

$$\begin{aligned} &If\ \ C_i\ \ is\ \ Clean - Irrelvant: \quad n_i = 1\ or\ 2; \\ &If\ \ C_i\ \ is\ Clean - Relevant: \quad n_i = 2 * l_i; \qquad (1) \\ &\quad l_i = n_{longest} + n_{shortest} + n_{medium} \end{aligned}$$

here $n_{longest}$, $n_{shortest}$ and n_{medium} are used to control the length distribution in our training data.

In our experiments, we tried $n_{\{longest,shortest,medium\}} = 1,2,3$, and find out n=2 works best by obtaining enough representative samples selected for efficient expert annotation. There are samples showing that within a cluster, the longer the sentence, the more pathologies it tends to describe. For example, the third sample in cluster 2 describe 'pneumonia' (keyword: '*air space shadows*') and 'other diseases' (keyword: '*vascular congestion'*) besides 'pneumothorax' (keyword: '*cardiac size'* or '*cardiac silhouette*') as the first and second samples do. This is consistent with the assumption observed by Chen et al. (2015) in their clinical NER task too.

3.3 Inconsistent Error Checking and Auto-annotation

We also use the self-trained supervised sentence encoder to assist us in checking the inconsistencies among the human annotations via pairwise sentence comparison among the annotated sentences. If two sentences have high semantic similarity but different labels, we will flag out to the radiologists for label confirmation.

With this high-quality set of human annotated samples, we can auto-annotate the remaining sentences using the same sentence encoder. This is done by assigning the sentence with the same label of the human annotated sentence with highest similarity score. During this automatic labelling process, we set a threshold to reject auto labelling. The rejected sample will be sent for human annotation. A label is assigned only if the cosine similarity value between two sentences is more than 0.9. This threshold has been proved to be efficient in our experiment.

3.4 CXR Sentence Encoder

A good sentence encoder is important for our annotation framework on sample selection, error checking, and auto-annotation. We use supervised SBERT (Reimers et al., 2019) and unsupervised TSDAE (Wang et al., 2021) for sentence encoder training in our experiments (Figure 2).

To obtain our supervised sentence encoder, we utilize sentence transformer[1] for sentence encoder training and testing. We first do a domain adapted fine-tuning on a pretrained language model using our in-house CXR and MIMIC data (Alistair et al., 2019) based on Transformer [2] with default parameters. We use Roberta-large as the pretrained model as it performs significantly better than other pretrained models based on our experiments**.** We further train this fine-tuned model on large Semantic Textual Similarity (STS) tasks using SBERT architecture and apply a pooling operation to our fine-tuned language model to get a fixed size sentence embedding output. Different from Wang *et al* (2020b), we include both general STS and clinical STS in the training and validation. Besides STS tasks, we also study the effect of fine-tuning on Natural Language Inference (NLI) for our data annotation task.

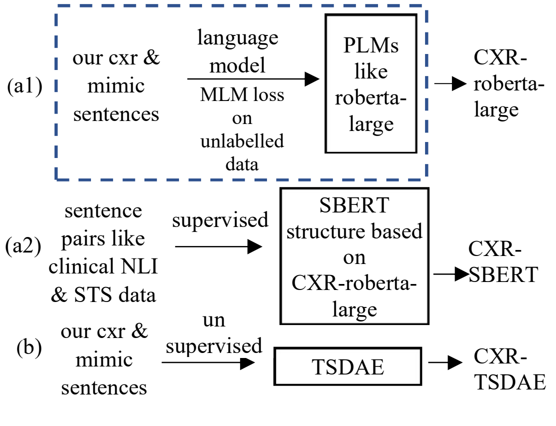
(a1)
our cxr & mimic sentences
language model
MLM loss on unlabelled data
PLMs like roberta-large
CXR-roberta-large
(a2)
sentence pairs like clinical NLI & STS data
supervised
SBERT structure based on CXR-roberta-large
CXR-SBERT
(b)
our cxr & mimic sentences
un supervised
TSDAE
CXR-TSDAE

Figure 2. <u>*Supervised Sentence Encoder Training*</u>: (a1) to get domain adapted model (CXR-Roberta-large) through behavior fine tuning on pretrained model; (a2) further trains the adapted model (CXR-Roberta-large) to obtain a supervised sentence encoder (CXR-SBERT) on labelled clinical NLI or STS data. <u>*Unsupervised Sentence Encoder Training*</u>: (b) train unsupervised sentence encoder on unannotated CXR sentences and Roberta-large.

The unsupervised TSDAE training is shown in Figure 2 (b). The data used for domain adaptation in (a1) is also used to train the TSDAE sentence encoder. We use Roberta-large as the base model and train 10 epochs with the batch size of 16 in our experiment. As to other parameters, we use the default values as Wang et al. (2021) used.

3.5 CXR Report Classification

We frame our CXR disease labelling task as a multi-label classification task. We adopt one-hot encoding label strategy. For each sentence, we use one-hot vector with dimension 13 for the labelling. The first 12 dimensions refer to the labeling of the four diseases, each of which has three dimensions namely ‘1,0,0’ ‘0,1,0’”, ‘0,0,1’ to indicate ‘disease positive’, ‘disease negative’, and ‘disease uncertain’ respectively. The last dimension refers to the labelling of ‘no_findings’, we use ‘1’ to indicate no disease mentioned and ‘0’ for one or more diseases mentioned. For example, if the sentence has pneumonia positive and other disease uncertainty, its one-hot vector label will be [1,0,0,0,0,0,0,0,0,0,0,1,0]. This gives us a maximum of $(3^4 + 1)$ labels, with label values within one disease mutually exclusive to each other and label values among different diseases independent to each other. Let $X = \{X_1, X_2, \dots, X_m\}$ denotes the input space, $\mathrm{Y} = \{Y_1, Y_2, \dots, Y_c\}$ denotes the finite set of labels, in which c is the number of labels, $Y_i = [L_1, L_2 \dots, L_{13}]$ and $L_i \in \{0,1\}$. Our classification task is to build a multi-label classifier H that maps an instance x to its associated set of labels: $H(x) = P(y|x)$, in which $x \in X$, $\mathrm{y} \in Y$.

In this multi-label classification, the prediction may be partially correct, which we do not consider it as a correct prediction. We use precision, recall and F-score to measure the performance for each disease, and use accuracy to measure the performance on sentence or report level.

4 Experiments

4.1 Data Processing

We use 8 years of CXR reports extracted from our local hospitals. All data have been anonymized

[1] https://github.com/UKPLab/sentence-transformers

[2] https://github.com/huggingface/transformers

and annotation is performed by professional radiologists with more than 13 years of experience who are native English speakers.

We segment all reports into sentences by using NLTK sentence tokenization tool (Bird *et al.* 2009). The length of the sentences varies from 1 to 52 words with an average of 10 words per sentence. We filter out all single word sentences as they do not have much context to infer pathology. We also filter out sentences describing other body parts (e.g., Abdomen, Supine, Neck, Back) as they are not related to our pathology. We obtain 12,350 sentences at the end after removing duplicates.

The data used for unsupervised TSDAE training and supervised SBERT training are summarized in Table 3. MIMIC CXR (Alistair et al., 2019) and in-house CXR are used to fine-tune Roberta-Large and TSDAE on the clinical domain. The data used to fine-tune the SBERT sentence encoders include (i) Combined NLI comprising MEDNLI (Romanov et al., 2018), general domain NLI (StanfordNLI (Bowman et al., 2015) and multi-genre NLI (Nangia *et al*., 2017); (ii) STS-B (STS Benchmark, Cer et al., 2017) consists of a mixture of news, captions, and forums; (iii) SemEval STS 2012-2016 (Agirre et al., 2012-2016) for semantic textual similarity tasks; (iv) Clinical STS 2018 (v) Clinical STS 2019 (Wang et al., 2018 & 2020c) are the only available STS data in clinical domain. We use all the 5 datasets for SBERT training.

data	train/dev/test	average length
data for domain adaptation and TSDAE		
in-house CXR	12350	10
MIMIC CXR	248k	14
data for SBERT sentence encoder		
combined NLI	956k	11.4
STS-B	5749/1500/1379	10.2
STS 2012-2016	23,778	12
clinicalSTS 2018	749/318	25.4
clinicalSTS 2019	1641/412	19.3

Table 3. Data used for sentence encoder training

4.2 Experiments & Results

4.2.1 Sentence Encoder

We studied our proposed framework using three supervised sentence encoders and one unsupervised sentence encoder. All three supervised sentence encoders are based on CXR-SBERT but trained on different datasets. The open-source sentence encoder "sts-robert-large" from sentence-transforms[1] is used as the baseline in our experiment. The settings for the supervised sentence encoder are descried below.

1) CXR-SBERT-nli-stsb: train CXR-SBERT on both NLI and STS-B, followed by continuous training on STS 2012-2016.
2) CXR-SBERT-nli-sts: train CXR-SBERT on both NLI and STS-B, followed by continuous training on all data from STS 2012-2016, clinicalSTS 2018 and Clinical STS2019.
3) CXR_SBERT-sts: train CXR-SBERT on STS-B and STS 2012-2016, followed by continuous training on Clinical STS 2018 and clinical STS 2019.

For training on NLI data set, we use classification objective function with the mean pooling strategy and the mean squared error loss. We set num_epoch as 4 and batch_size as 16. We use Adam optimizer with learning rate 2e−5, and a linear learning rate warm-up over 10% of the training data. For training on STS data, we use regression objective function with mean pooling strategy and cosine similarity loss (Reimbers *et al* 2019). We set num_epoch as 5 and batch_size as 16. Other parameters are the same as Reimers *et al* (2019).

Sentence encoder	Accuracy
sts-roberta-large	93.33
CXR-SBERT-nli-stsb	94.33
CXR-SBERT-nli-sts	**98.33**
CXR-SBERT-sts	**98.33**
CXR-TSDAE	96.67

Table 4. Sentence encoders' comparison on auto-annotation performance on a small CXR data set.

To select a suitable sentence encoder for our task, we use 360 CXR sentences from clean clusters generated by each sentence encoder for experts' annotation. We use 60 as training data and 300 as test data. The 60 sentences are distributed across clusters and are used for auto-annotating the 300 sentences.

The results are shown in Table 4. From the result, we can see that sentence encoder CXR-SBERT-sts and CXR-SBERT-nli-sts produce better results than other encoders, which means the encoders trained on large STS data generate better sentence embeddings on our data. The model CXR-SBERT-nli-stsb has better performance than the out-of-the-box model (sts-roberta-large), which means domain adapted fine tuning is helpful. The unsupervised sentence encoder CXR-TSDAE performs well but is not so good as SBERT encoder CXR-SBERT-sts.

4.2.2 Semi Auto-annotation Strategy

With good quality sentence embeddings obtained, we leverage HDBscan with Umap[3] for clustering. We set parameters n_neighbors as 200 and n_components as 500 (original data dimension is 1024) for Umap. We set HDBscan parameters min_samples as 10, min_cluster_size as 30. Using the above setting, we generate 94 clusters for 12 thousand sentences (See Appendix A.1 for parameters setting).

We follow the selection criterion in Section 3.2 to select different number of samples for annotation (see Table 6). The selected samples are first annotated by two professional radiologists and checked by a third annotator. After human annotation, we conduct label auto-checking through similarity values. Some inconsistent annotations could be found during auto-checking due to human bias or mistake. These inconsistencies are verified by human annotators again and the verified annotated data is added to the reference for automatic annotation of the remaining sentences by measuring the similarity scores of the remaining sentences with these references.

Categories	(+)	(-)	(u)	total
pneumonia	949	489	79	1517
penumothorax	900	294	76	1210
cardiomeglay	1511	520	775	2806
other diseases	3850	974	132	4956
no_finiding	3436			

Table 5. The in-house data statistics, in which "(+) / (-) / (u)" indicates positive/negative/uncertainty of each pathology.

A total of 11,114 sentence samples are annotated using our proposed framework, during which 150 confusing samples were sent for further verification by our annotators. The final data statistics is shown in Table 5. Most of the data (88%) include only one pathology, while 12% of data include multiple pathologies.

We perform analysis on the size of human annotation samples with respect to the accuracy of auto-annotation of the remaining data. From the results shown in Table 6, we can see the annotation accuracy increased from 69.98% to 90.05% by annotating about 19.8% of data, from a test set of 1,236 reports.

#sents selected (% data)	Auto annotation Accuracy
876 (7.8%)	69.98%
1236 (11.12%)	81.07%
1656 (14.9%)	85.60%
2200 (19.80%)	90.05%

Table 6. Performance of auto-annotation performance on different number of samples selected from further annotation. '%data' indicates the percentage of data selected from 11,114 sentence samples.

4.2.3 Pathology classification

In the experiment, we used SimpleTransformer[4] library for the multi-label classifier training and testing. We used train and evaluation batch of 8, epoch of 3, learning rate of 4e-5 and threshold of 0.5. For other parameters, we use the default values. We split the annotated sentence samples into train data 9879, dev (1235) and test data (1236). The experiment results are shown in Table 7. We can see that most of the categories have f-score at around or more than 95% except two uncertainty cases (pneumonia and others). This model also has been recently tested on 988 reports from local hospital with an accuracy of 98.1% on report level. This successful application of this framework on CXR data boosts our confidence on its application on other annotation tasks (See Appendix A.2).

5 Discussion

5.1 The Robustness of the classifier

We observe that although the SBERT sentence encoder was trained on out-domain labelled STS data, it performs better than the unsupervised TSDAE sentence encoder. The result is consistent with the observation in Schick et al. (2021), who mentioned and demonstrated supervised sentence encoders perform better than unsupervised

[3] https://umap-learn.readthedocs.io/en/latest/clustering.html#umap-enhanced-clustering

[4] https://github.com/ThilinaRajapakse/simpletransformers

sentence encoders. Our auto-annotation experiments demonstrate very good performance on our CXR data. One possible reason can be due to the characteristics of our data. Our data is in sentence level and some sentences are very similar (not much variation), so clustering can generate some very clean clusters which make the annotation inside those clusters very efficient; Besides, the average length of our data is about 10 words, which possibly captures a good embedding for the sentence representation.

Category	precision	recall	f-score
pneumonia (+)	93.30	98	95.59
pneumonia (-)	94.70	94.74	94.74
pneumonia (u)	85.71	75.00	78.00
pneumothorax(+)	97.92	96.91	97.41
pneumothorax(-)	86.96	95.24	94.05
pneumothorax(u)	100	100	100
cardiomegaly (+)	96.27	99.36	97.79
cardiomegaly(-)	100	98.15	99.07
cardiomegaly (u)	98.55	94.44	96.45
others (+)	90.41	94.83	92.57
others (-)	90.70	92.86	91.77
others(u)	77.78	50	60.87
no_finding	97.80	93.93	95.36

Table 7. The classification comparison on each pathology, in which "(+) / (-) / (u)" indicates positive/negative/uncertainty of each pathology.

Though sentence encoder could potentially be used as an annotation tool, it has limitations. It cannot differentiate multi-labels because it treats multi-label as one power set of labels. A classifier trained on annotated data is much more robust than a sentence encoder for label assignments because during classifier training, the weights of the classifier in multiple layers are iteratively updated to represent a sentence in a more accurate way. On the other hand, a sentence encoder converts each sentence to a vector in a fixed way and treats each dimension of sentence embeddings equally (Reimers et al., 2019). Therefore, we fine-tuned cxr-roberta-large on the annotated data to obtain a more robust classifier (see Section 4.2.3).

5.2. The Effectiveness of our method

To demonstrate the effectiveness of our proposed annotation method, we compare our method with the uncertainty-based sampling method with the initial set selected randomly and based on sentence length (Chen et al., 2015). The reasons of using the two baselines of random sampling and sentence length-based sampling are because 1) although random sampling is simple and straightforward, it performs competitive to most sophisticated strategies (Schroder et al., 2020); 2)The length strategy is a data driven strategy, which is simple and has been tested to be effective for medical data (Chen et al., 2015), and it is slightly better than random sampling for medical data.

We use the total number of samples selected for manual annotation as an evaluation measure for the annotation efforts among different methods. Note that other methods will incur additional time for updating model and waiting time of more cycles of experts' annotation. In this experiment, we used the same library and setting as Section 4.2.3. For data setup, we have 10k+ data for annotation experiments and 2000 data used for model evaluation. For the batch sizes used in active learning, we use batch size of N= [100, 300, 600, 1000, 1500, 2200, 2800, 3600, 4500, 5500, 6700, 10350] for 12 iterations in our experiment, for the comparison among different annotation methods (see Figure 3).

From Figure 3, we can see that with more data for training, the performances of all models increase. The performance increases faster when the number of samples is less than 2000. After 2000 samples, the increasing trend slows down. Our proposed method demonstrates better and faster performance improvement than other two methods as our method selects more representative sentences for human annotation and model training. The auto-checking further assists us to check for errors and control our data quality. The sampling strategy based on longest sentences are better than the random selection, but when more and more samples are selected for training, the gap becomes smaller. Our experiment shows our proposed approach requires only 2200 manually annotated samples to perform auto-annotation of the remaining samples to reach the best performance where other methods need to manually annotate almost all the sentences to reach around 95.06% accuracy. This can contribute to our auto-annotation strategy with auto-checking process which reduces human annotation bias and errors and has a positive impact on the quality of the annotation data.

6 Conclusion

We propose a semi-supervised annotation scheme which avoids multiple model re-training and expert annotation which is applicable to CXR text data and other domain data annotation (Appendix A.2). Within the framework, we investigate a gap mentioned by Schroder et al. (2020) by using fine-tuning-based models in active learning for text classification. We utilize a self-trained sentence encoder for effective sample selection through clustering, error auto-detection and sample auto-annotation. Based on the annotated data, we further fine-tune a pre-trained language model to obtain a robust classifier which demonstrates high performance on CXR data disease detection. This method greatly improves data annotation efficiency and relieves human annotation burden.

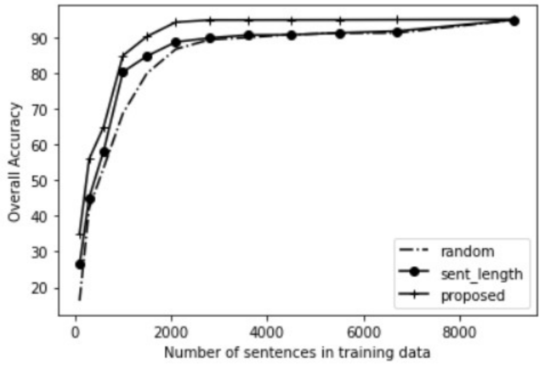

Figure 3. Comparison of different annotation methods

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, Matthew McDermott, *Publicly Available Clinical BERT Embeddings,* Proceedings of the 2nd Clinical Natural Language Processing Workshop, pp.72-78, 2019

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. *Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation*. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pages 497–511.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 2015. *Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability*. In Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), pages 252–263.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. *Semeval-2014 task 10: Multilingual semantic textual similarity*. In Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014), pages 81–91.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor GonzalezAgirre, and Weiwei Guo. 2013. *Semeval-2013 shared task: Semantic textual similarity*. In Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, pages 32–43.

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. *Semeval-2012 task 6: A pilot on semantic textual similarity*. In Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pages 385–393. Association for Computational Linguistics.

Steven Bird, Edward Loper and Ewan Klein 2009, *Natural Language Processing with Python*. O'Reilly Media Inc.

Zalan Bodo, Zsolt Minier, Lehel Csató, Active Learning with Clustering, JMLR: Workshop and Conference Proceedings, 2011.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. *A large annotated corpus for learning natural language inference*. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Daniel Cer, Mona Diab, Eneko Agirre, Iigo LopezGazpio, and Lucia Specia. 2017. *SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation*. SemEval-2017, pages 1–14, Vancouver, Canada

Daniel Cer, et al., 2018, *Universal Sentence Encoder*, https://arxiv.org/pdf/1803.11175.pdf

Yukun Chen,Thomas A. Lasko, Qiaozhu Mei, Joshua C. Denny, and Hua Xu, *A Study of Active Learning Methods for Named Entity Recognition in Clinical Text*, J Biomed Inform. 2015 Dec; 58: 11–18.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Lo¨ıcBarrault, and Antoine Bordes. 2017. *Supervised Learning of Universal Sentence Representations from Natural Language Inference Data*. In Proceedings of the 2017 Conference on

Empirical Methods in Natural Language Processing, pages 670–680,

Alexis Conneau and Douwe Kiela. 2018. *SentEval: An Evaluation Toolkit for Universal Sentence Representation*s. arXiv preprint arXiv:1803.05449.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv preprint arXiv:1810.04805.

Suchin Gururangan et al., 2020. *Don't stop pretraining: Adapt language models to domains and tasks*. ACL 2020, pages 8342–8360.

Jeremy Irvin et al., *CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison*. AAAI 2019

Alistair E. W. Johnson et al., 2019. *MIMIC-CXR-JPG, A large publicly available database of labeled chest radiographs*, https://arxiv.org

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. *On the sentence embeddings from pre-trained language models*. EMNLP 2020, pages 9119–9130.

Diwakar Mahajan, et al., *Identification of Semantically Similar Sentences in Clinical Notes: Iterative Intermediate Training Using Multi-Task Learning,* JMIR Med Inform. 2020 Nov; 8(11): e22508

L. McInnes, J. Healy, S. Astels, *HDBSCAN: Hierarchical density based clustering* In: Journal of Open Source Software, The Open Journal, volume 2, number 11. 2017

Matthew B. A., McDermott, et al., *Chexpert++: Approximating the chexpert labeler for speed, differentiability, and probabilistic output*. MLHC 2020.

Nikita Nangia, Adina Williams, Angeliki Lazaridou, Samuel R. Bowman, *The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations.* Proceedings of the 2nd Workshop on Evaluating Vector-Space Representations for NLP, pages 1–10. 2017

Hieu T. Nguyen and Arnold Smeulders, Active Learning Using Pre-clustering, ICML 2004.

Peng Y, Wang X, Lu L, Bagheri M, Summers R, Lu Z. *NegBio: a high-performance tool for negation and uncertainty detection in radiology reports*. AMIA Summits on Translational Science Proceedings. 2018;2017:188.

Pedregosa *et al.*, *Scikit-learn: Machine Learning in Python*, JMLR 12, pp. 2825-2830, 2011.

Jason Phang, Thibault Fevry, Samuel R. Bowman, *Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-data Tasks*, https://arxiv.org/pdf/1811.01088.pdf

Nils Reimers and Iryna Gurevych, *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*, EMNLP 2019

Alexey Romanov and Chaitanya Shivade. Lessons from natural language inference in the clinical domain. arXiv preprint arXiv:1808.06752, 2018.

Timo Schick and Hinrich Schütze, Generating Datasets with Pretrained Language Models, https://arxiv.org/pdf/2104.07540.pdf, EMNLP 2021

Christopher Schröder, Andreas Niekler, A Survey of Active Learning for Text Classification using Deep Neural Networks, 2020. https://arxiv.org/pdf/2008.07267.pdf

Akshay Smit et al., *CheXbert: Combining Automatic Labelers and Expert Annotations for Accurate Radiology Report Labeling Using BERT*, EMNLP2020, pp 1500-1519

Irena Spasic and Goran Nenadic, *Clinical text data in machine learning: systematic review*, JMIR Med Inform. 2020 Mar; 8(3): e17984

C. Shivade, *MedNLI - A Natural Language Inference Dataset For The Clinical Domain* (version 1.0.0). PhysioNet. https://doi.org/10.13026/C2RS98.2019

Bin Wang and C-C Jay Kuo, (2020a). *A Sentence Embedding Method By Dissecting BERT-based Word Models*, IEEE/ACM Transactions on Audio, Speech, and Language Processing,vol 28, 2146-2157.

Kexin Wang, Nils Reimers, Iryna Gurevych, TSDAE: Using Transformer-based Sequential Denoising Auto-Encoder for Unsupervised Sentence Embedding Learning, EMNLP2021.

Yuxia Wang, Karin Verspoor, Timothy Baldwin, (2020b) *Learning from Unlabelled Data for Clinical Semantic Textual Similarity*, Proceedings of the 3rd Clinical Natural Language Processing Workshop ACL 2020, pages 227–233.

Yue Wang, Kai Zheng, Hua Xu, and Qiaozhu Mei, *Clinical Word Sense Disambiguation with Interactive Search and Classification*, AMIA Annu Symp Proc. 2017; 2017: 2062–2071.

Wang Y, et al., *A clinical text classification paradigm using weak supervision and deep representation*, BMC Medical Informatics and Decision Making 2019.

Y. Wang, Afzal N, Liu S, Rastegar-Mojarad M, Wang L, Shen F, Fu S, Liu H. *Overview of the BioCreative/OHNLP Challenge 2018 Task 2: Clinical Semantic Textual Similarity*. Proceedings of the BioCreative/OHNLP Challenge. 2018.

Y. Wang et al., (2020c) *The 2019 n2c2/OHNLP Track on Clinical Semantic Textual Similarity: Overview*. JMIR Medical Informatics 2020, 8(11), p.e23375.

Yan Zhang et al., *An Unsupervised Sentence Embedding Method by Mutual Information Maximization*, https://arxiv.org/pdf/2009.12061.pdf, MNLP2020

A Appendix

A.1 HDBSCAN & UMAP parameters setting

We had performed extensive clustering experiments on this CXR data. The UMAP is for dimension reduction and n_neighbors and n_components are two of the most important parameters. The bigger n_neighbors, it will look at more global manifold structure. If there are no ground truth labels, it is hard to know which values are best for those two parameters. And those values may change for different data.

HDBSCAN has two important parameters min_cluster_size and min_samples, which can cause quite different clustering if we change them in a wide range. The large min_cluster_size, more data points will be rejected. Our strategy is to obtain the initial results using the default parameters and then adjust the values within a range to get good clustering. This clustering has been tried on another set of finance data and compared with K-means which needs to have the number of clusters specified first. The clusters generated by HDBSCAN with UMAP is much more sensible and preferred by clients.

A.2 Application of the proposed method on anther data

We have used the method on another financial in-house data set for topic classification. The number of sentences used for training is 5044, and the average length of sentences is around 11, with maximum length 48 and minimum length 1. The number of clusters generated is 24. Through the auto error checking and auto-annotation, we achieved satisfactory classification result.