Journal of Biomedical Informatics 112 (2020) 103607

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

# Concept placement using BERT trained by transforming and summarizing biomedical ontology structure

Hao Liu *, Yehoshua Perl, James Geller

*Dept of Computer Science, NJIT, Newark, NJ, USA*

ARTICLE INFO

*Keywords:*
Ontology summarization
Machine learning
Ontology placement
Natural language processing
BERT
SNOMED CT

ABSTRACT

The comprehensive modeling and hierarchical positioning of a new concept in an ontology heavily relies on its set of proper subsumption relationships (IS-As) to other concepts. Identifying a concept's IS-A relationships is a laborious task requiring curators to have both domain knowledge and terminology skills. In this work, we propose a method to automatically predict the presence of IS-A relationships between a new concept and pre-existing concepts based on the language representation model BERT. This method converts the neighborhood network of a concept into "sentences" and harnesses BERT's Next Sentence Prediction (NSP) capability of predicting the adjacency of two sentences. To augment our method's performance, we refined the training data by employing an ontology summarization technique. We trained our model with the two largest hierarchies of the SNOMED CT 2017 July release and applied it to predicting the parents of new concepts added in the SNOMED CT 2018 January release. The results showed that our method achieved an average F1 score of 0.88, and the average Recall score improves slightly from 0.94 to 0.96 by using the ontology summarization technique.

## 1. Introduction

The maintenance process of an ontology includes ontology enrichment, namely, the addition of new concepts into the ontology. The enrichment is divided into two parts. The first part is discovery of new concepts to-be-added. The second part is placement of the new concepts into the proper positions in the ontology.

The discovery of new concepts may be initiated by requests of users of the ontology, or by searching literature repositories or knowledge bases. Examples of research regarding the discovery of new concepts have been reported by other authors [1–3]. The curators of the ontology need to assess whether a concept suggested is proper for addition to the ontology, and if so, to which hierarchy of the ontology it should be added. The name of this concept may be changed to follow the naming conventions of the target hierarchy.

The placement of a new concept into a hierarchy of an ontology involves identifying all the proper parents of this concept in order to insert it into the IS-A hierarchy, which is the backbone of the ontology. In SNOMED CT, the target of our investigation, there are 19 hierarchies that are disjoint, thus all of a concept's parent(s) must be in the same hierarchy as the concept. In this paper, we are concentrating on the task of automatically identifying the parents of a new concept within the hierarchy to support the work of SNOMED CT curators.

Finding the right place for a new concept is a fundamental task in the curation of an ontology. The position of a concept, represented by its IS-A relationships to other concepts, determines how accurately it is modeled in terms of granularity. Therefore, considering as many related parent candidate concepts as possible leads to a more comprehensive modeling of this concept. Finding all the parents is a challenging and time-consuming task, because it requires both domain knowledge and ontology skills. Placing concepts is difficult and error prone, because oftentimes parents are missing, wrong, or too general. As a rule, a parent should be the most specific generalization of its child concept. Sometimes the name (the text string) of the parent concept is very different from the name of the new concept, which makes the task even more difficult. As a well-known example, *Legionnaires' disease* (SNOMED CT ID: 035187010) has a parent *Pneumonia due to Gram negative bacteria* (ID: 430395005), but the two concept names have no word in common.

Traditionally, curators rely on classifiers such as Snorocket [4] or HermiT [5] to place concepts into ontologies based on a Description Logic. However, this approach relies on the relationship modeling of the new concept as well as the relationship modeling of existing concepts. Since many concepts in a Description Logic ontology, like SNOMED CT [6], are underspecified in terms of their relationships, the placement by

* Corresponding author.
*E-mail address:* hl395@njit.edu (H. Liu).

https://doi.org/10.1016/j.jbi.2020.103607
Received 18 February 2020; Received in revised form 7 September 2020; Accepted 17 October 2020
Available online 22 October 2020

classifier algorithms may be wrong. In cases where the curator does not manually check the automatic placement by classifiers, concepts may end up in wrong positions in the hierarchy. Hence, a user searching for such a concept, without knowing its name in SNOMED CT, would not find it in its expected location. Thus, a Machine Learning (ML) model that automatically prepares a set of candidate parent concepts for a new concept can assist curators to improve the ease and accuracy of placement of a new concept in the process of ontology curation.

As neural network models have succeeded in computer vision and natural language processing, they also show great promise in addressing ontology related tasks, including insertion of new concepts. Liu *et al.* [7] proposed to use a Convolutional Neural Network (CNN) model [8] to verify an IS-A relationship between a new child concept and an existing parent concept. This model recommends the location of the new concept in the hierarchy. In this approach, concepts are mapped to low-dimensional vectors using a paragraph/sentence embedding model. Zheng *et al.* [9] showed that it is possible to further improve the performance of the CNN model by using summarization of ontologies, based on Abstraction Networks [10–12].

Recently, the research interest in language modeling has shifted from training unsupervised low-dimensional neural embedding models to training general language representation models for their easy reuse in downstream tasks. For instance, the Bidirectional Encoder Representations from Transformers (BERT) [13] model, developed by Google, advanced the state-of-art of many English NLP benchmarks [14–16]. It is easy and efficient to preform Transfer Learning from BERT to the task of interest. Few attempts have tried to employ BERT for ontology related tasks. Liu *et al.* [17] utilized the "next sentence prediction" capability of BERT for IS-A relationship classifications and demonstrated a performance improvement with combining pre-training and fine-tuning of BERT. However, the utilization of the "next sentence prediction" capability of BERT was not optimal.

Hence, we have demonstrated two independent ways to improve on the performance of previous work by Liu *et al.* [7]. The first improvement was achieved by using the BERT model rather than the CNN model and the second by utilizing ontology summarization to provide a more accurate training of a CNN model. In this paper, we consider combining the two improvements by utilizing ontology summarization together with the BERT model and with an improved presentation of the training data to better utilize the "next sentence prediction" capability of BERT. It is a challenge to further improve the performance of the BERT model, which is already high, e.g., it shows a recall of 0.94 [17].

We measured the performance of our proposed method with the two largest hierarchies of the SNOMED CT [18] ontology, the *Clinical Finding* hierarchy and the *Procedure* hierarchy. The SNOMED CT release of July 2017 was used as training data, and the subsequent January 2018 release was used for testing, building on prior art [9,17]. The results of evaluating the placement of 2005 new concepts into the *Clinical Finding* hierarchy and of 911 new concepts into the *Procedure* hierarchy are reported in this paper.

## 2. Background

### 2.1. SNOMED CT

SNOMED CT ® is an internationally leading clinical ontology, managed by SNOMED International. It contains 19 hierarchies covering various subdomains of biomedicine. The largest two hierarchies are the *Clinical Finding* hierarchy and the *Procedure* hierarchy. SNOMED CT is released twice every year on January and July. Each release of SNOMED CT includes three views: "full", "snapshot" and "delta." The "full" view contains all versions of all SNOMED CT components ever released. The "snapshot" view contains the most recent content of all components. In addition, the "delta" view identifies the individual changes of all components between the previous release and the current (snapshot) release. The January 2018 release of SNOMED CT consists of 111,081 active concepts in the *Clinical Finding* hierarchy and 57,806 active concepts in the *Procedure* hierarchy. By comparing it with the previous (July 2017) release using the delta view, we found that 2005 new concepts were added into the *Clinical Finding* hierarchy and 911 new concepts were added into the *Procedure* hierarchy.

### 2.2. BERT

BERT is a general-purpose "language understanding" model trained on a large text corpus (Wikipedia and BookCorpus). Different from traditional *context-free* word embedding models such as *word2vec* [19], *GloVe* [20], or *fastText* [21], BERT generates a representation of each word that is based on the other words in the context. BERT is an *unsupervised*, *deeply bidirectional* system that outperforms previous language processing methods. BERT is based on multi-layer *bidirectional transformer encoders*, which are based on the original implementation proposed by Vaswani *et al.* [22]. BERT can be used for various downstream NLP tasks without heavy task-specific engineering. BERT has advanced the state-of-the-art for several major NLP benchmarks, including named entity recognition on CoNLL-2003 [14], question answering on SQuAD [15], and sentiment analysis on SST-2 [16]. Variants of BERT are also available in the bioinformatics research area, for example, BioBert [23], ClinicalBERT [24] and NCBI BlueBERT [25] are obtained by training the original BERT model with biomedical or clinical research text.

The BERT model is pre-trained with two tasks Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). The training objective of the MLM task is to predict the masked words of a text sequence. The training objective of NSP is to classify whether two sentences are consecutive for any given sentence-pair. The model pre-trained with these two tasks can be easily adapted to other types of NLP tasks.

### 2.3. Areas and Area Taxonomy

Area Taxonomies were introduced by Min *et al.* [10] to achieve summarization of large ontologies. Ontology concepts with exactly the same set of lateral (i.e., non-IS-A) relationships are grouped into an *area*. Areas, considered as nodes, are connected via child-of hierarchical links to form a network, called an Area Taxonomy, since it has only hierarchical relationships. Fig. 1 illustrates the derivation of an Area Taxonomy. Fig. 1(a) shows an excerpt of 14 concepts from SNOMED CT's *Clinical Finding* hierarchy, drawn as labeled ovals. A dashed rectangle contains a set of concepts each of which has exactly the same lateral relationship type(s). The list of relationships for the concepts in each dashed rectangle appears in bold. The arrows denote IS-A links. Lateral relationships are inherited down along the IS-A links between concepts. Fig. 1(b) shows the Area Taxonomy for the excerpt of the subhierarchy in Fig. 1(a). All colored dashed rectangles are represented as "nodes," (shown as colored rectangles) which are connected by hierarchical child-of links (drawn as bold arrows) that are derived from the IS-A relationships in the ontology. The list of the relationship types of an area is used as its name (in bold). For example, because *Bradyarrhythmia* and *Diastolic heart failure* (and two other concepts) in Fig. 1(a) all have the same lateral relationships, *Finding site* and *Has definitional manifestation*, they are grouped together as area node (represented as a green rectangle) in Fig. 1(b).

Areas shown at the same level are displayed in the same color, indicating that all of their concepts have the same number of lateral relationship types. For example, the areas {*Finding site, Occurrence*} and {*Finding site, Has definitional manifestation*} appear in the second level in green. The concept *Heart disease* and its descendants in the grey rectangle in Fig. 1(a) are represented by the area {*Finding site*} in Fig. 1(b). Similarly, the concept *Neonatal bradycardia* and the concept *Fetal bradycardia* are represented by the red area {*Finding site, Has definitional manifestation, Occurrence*} in level 3. Areas inherit relationships along the hierarchical child-of links. For example, the red area inherits its

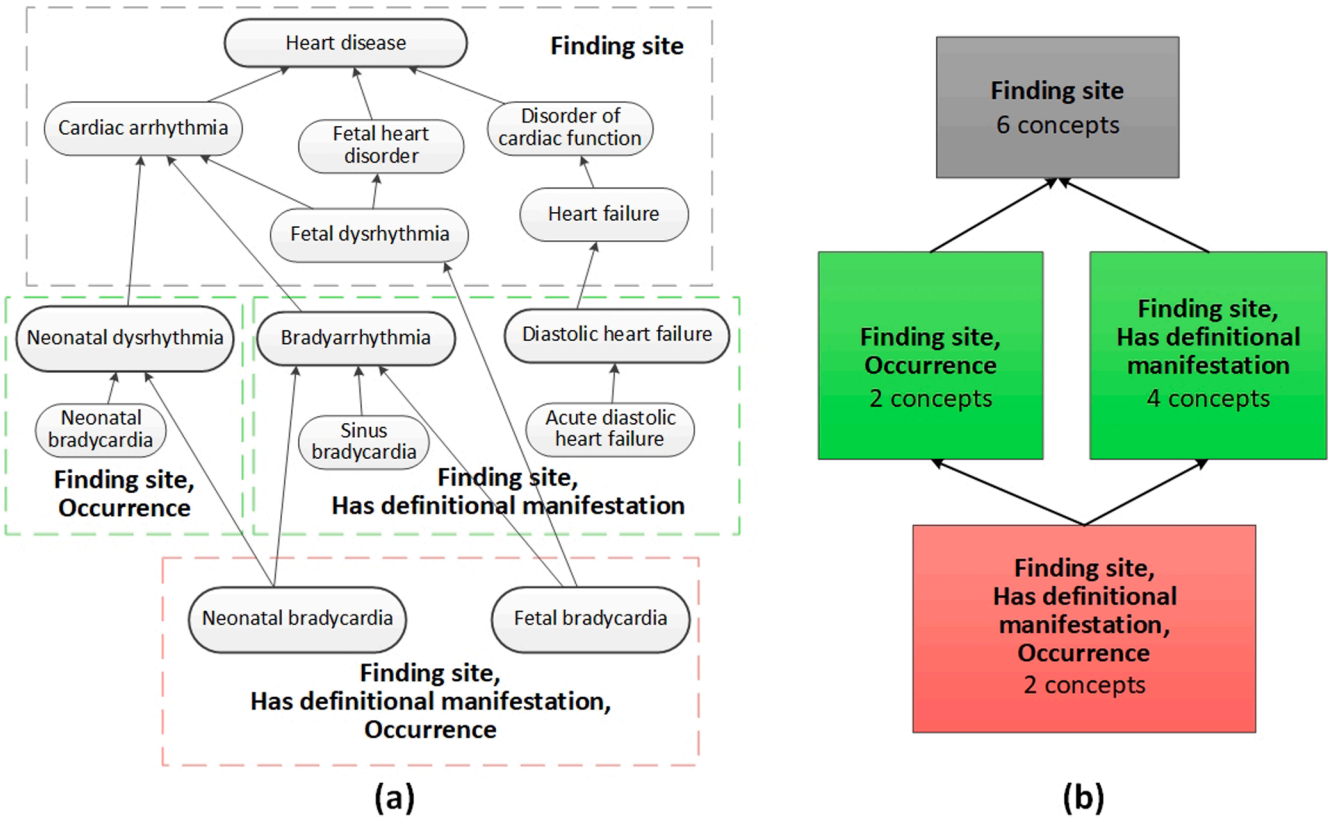



**Fig. 1.** Derivation of Area Taxonomy. (a) Excerpt of a subhierarchy of 14 concepts from SNOMED CT's Clinical Finding hierarchy. (b) Area Taxonomy for the excerpt subhierarchy in (a).

relationships from both green areas.

### 2.4. Quality assurance of IS-A relationship in SNOMED

The IS-A relationship hierarchy is the backbone of the SNOMED ontology. It enables inheritance of lateral relationships from parents to children and to all descendants. In a user study of SNOMED [18,26], users expressed high concerns for errors in IS-A relationship. SNOMED curators use the Snorocket classifier [4] for placement of a new concept into the IS-A hierarchy. The classifier utilizes the lateral relationships to identify the proper position for the new concept. However, many SNOMED concepts are primitive, i.e., they are underspecified, which means that they are missing lateral relationships. As a result, many concepts are misplaced by Snorocket. Hence, quality assurance of IS-A relationship in SNOMED is a high priority.

An example of such placement errors is as follows: the concepts *Cardiovascular operative procedure*, *Implantation to cardiovascular system*, *Procedure on heart*, *Procedure on pericardium removal of device from cardiovascular system* and *Removal of thrombus* are all listed as children of *Procedure on cardiovascular system*. However, *Implantation to cardiovascular system*, *Removal of device from cardiovascular system* and *Removal of thrombus* should be children of *Cardiovascular operative procedure*. *Procedure on pericardium* should be a child of *Procedure of heart*. Hence, out of the above six children of *Procedure on cardiovascular system* only two are correct children while the four other concepts should be grandchildren, connected by IS-A links to these two children, respectively.

Cui et al. [27] presented a hybrid structural-lexical quality assurance method to detect missing IS-A relationships and concepts in ontologies, based on mining non-lattice subgraphs with lexical patterns. The SABOC research group previously developed several techniques to identify concepts with high likelihood of errors. Among the errors found, there were many instances of missing and wrong IS-A relationships, which were reported to SNOMED curators. Examples of categories of concepts with high likelihood of errors are concepts in small partial area [12,28,29], overlapping concepts [30,31], and concepts with multiple parents or many lateral relationships [32].

### 2.5. IS-A prediction in Natural Language Processing

In Natural Language Processing, the IS-A relationship is sometimes referred to as *Hypernymy,* which expresses the subsumption relation between a general concept (hypernym) and its specific concepts (hyponyms), for example, *tiger* IS-A *animal* or *car* IS-A *vehicle*. The subject of hypernymy prediction has been actively studied in the NLP literature, with an evolution from lexical pattern-based methods to embedding-based language representation approaches [33]. Nguyen *et al.* [34] describe HyperVec, a system of hierarchy-oriented embeddings for hypernymy detection, which can distinguish between hypernyms and hyponyms in a hypernymy pair. Carmona *et al.* tested the ability of embeddings to encode hypernymy across different datasets [35]. Wang *et al.* proposed a model to improve hypernymy prediction by coupling an adversarial training algorithm with hierarchical knowledge in Web-scale taxonomies [36]. In addition, various studies have been conducted to extend hypernymy prediction in monolingual and cross-lingual manners [37–39]. The studies mentioned above mainly focus on using distributed word embeddings to represent entities and to train machine learning models to perform pairwise hypernymy predictions. In contrast, our research takes advantage of BERT's Next Sentence Prediction (NSP) capability to identify a concept's proper parent(s) in the hierarchy of an ontology. In addition, our method is specialized to the context of a medical ontology that contain terms, expressions, and semantics that are quite different from a general English corpus.

## 3. Methods

We used the SNOMED CT 2017 July release as training set and the

following 2018 January release as testbed. Due to the different sizes and inconsistent modeling schemas across hierarchies, we focused on SNOMED's two largest hierarchies, *Clinical Finding* and *Procedure,* as our data source. The models were implemented with Tensorflow [40]. We trained the models and ran the test cases on a machine with two Nvidia Tesla P100 "Pascal" video cards with 16 GB RAM per GPU and two Intel Xeon E5-2630-v4 CPUs with 2.2 GHz processor speed and 128 GB memory per CPU.

Google released two BERT models: $BERT_{BASE}$ (12 Transformer layers) and $BERT_{LARGE}$ (24 Transformer layers). Both models are trained on their Cloud TPUs (Tensor Processing Units), which have 64 GB of RAM. They are trained on English Wikipedia (2,500 M words of text) and BookCorpus [41] (800 M words of text) with one million update steps. As advised by the BERT creators, we avoided the use of $BERT_{LARGE}$ on GPUs with 16 GB of RAM, because the RAM size limited the number of training instances in each batch to avoid out-of-memory issues. Therefore, we only used $BERT_{BASE}$ in this experiment. The number of parameters for the pre-trained $BERT_{BASE}$ model is 110 M, with the default training settings L = 12, H = 768, A = 12, where L is the number of layers (i.e., Transformer blocks), H is the hidden size, and A is the number of self-attention heads. The feed-forward/filter size is set to 4 times H, i.e., 3072 for H = 768.

Fig. 2 illustrates the process of training and testing an IS-A relationship classifier using taxonomy data from SNOMED CT. In the following sections we described in detail the three major components of the process: (a) pre-train BERT model with concept-level documents (blue arrows), (b) fine-tune BERT model with data derived from Area Taxonomy (black arrows), and (c) test the two trained models with testing data from new release (red arrows).

### 3.1. Pre-train BERT model with concept-level documents

As a general language representation model, BERT was trained with Wikipedia and BookCorpus data, which do not provide a domain specific orientation for our ontology enrichment task. Liu *et al.* [17] demonstrated that pre-training BERT with a task-related corpus can improve the model's classification performance over directly fine-tuning the BERT model. Thus, we use an improved methodology by running additional steps of pre-training the BERT model with *Clinical Finding/ Procedure* hierarchy data, prior to training an IS-A relationship classifier using BERT. For the pre-training setup and process (Fig. 2(a)), we elaborate here on how we extracted data from *Clinical Finding* hierarchy of SNOMED CT (3.1.1 Data Preparation), and converted it into the format that is compatible with BERT (3.1.2 Data Preprocessing) (Fig. 2 (b)).

#### 3.1.1. Data preparation

The original BERT is pre-trained with general English sentences. To pre-train BERT with the knowledge of a hierarchy of a medical ontology is a challenge, because BERT is trained only to handle text, while the hierarchy consists of concepts connected through IS-A relationships. Therefore, we treat each concept's name as a "sentence." The concept's hierarchically closely related concepts are considered as part of its definition in the ontology hierarchy. This is demonstrated in the SNOMED CT browser, where the concept is listed with a larger font in the central blue area, with its parents in the top rectangular frame above and the children in the rectangular frame below (Fig. 3). The synonyms are listed below the concept's name in the central blue area. Thus, for a given concept A, we also consider A's hierarchically closely related concept(s)' names as "sentences."

The challenge is to harness the capability of BERT to model these IS-A relationships between concepts in an ontology. We need to express the hierarchical relationship from a specific concept to a general concept utilizing the features of the BERT model.

BERT was created for tasks such as Next Sentence Prediction (NSP), in which it needs to predict whether one sentence logically (based on human-like intuition, not on formal logic) follows another sentence in a given text. Given two concepts A and B and a relationship A IS-A B, we trained the BERT model to recognize the sentence of A as the next sentence following the sentence of B.

We prepared an ontology-oriented corpus for pre-training BERT with *Clinical Finding/Procedure* IS-A relationships. For each concept, we created a document that consists of the concepts that are hierarchically related to it in a textual form. Though the choices for textual

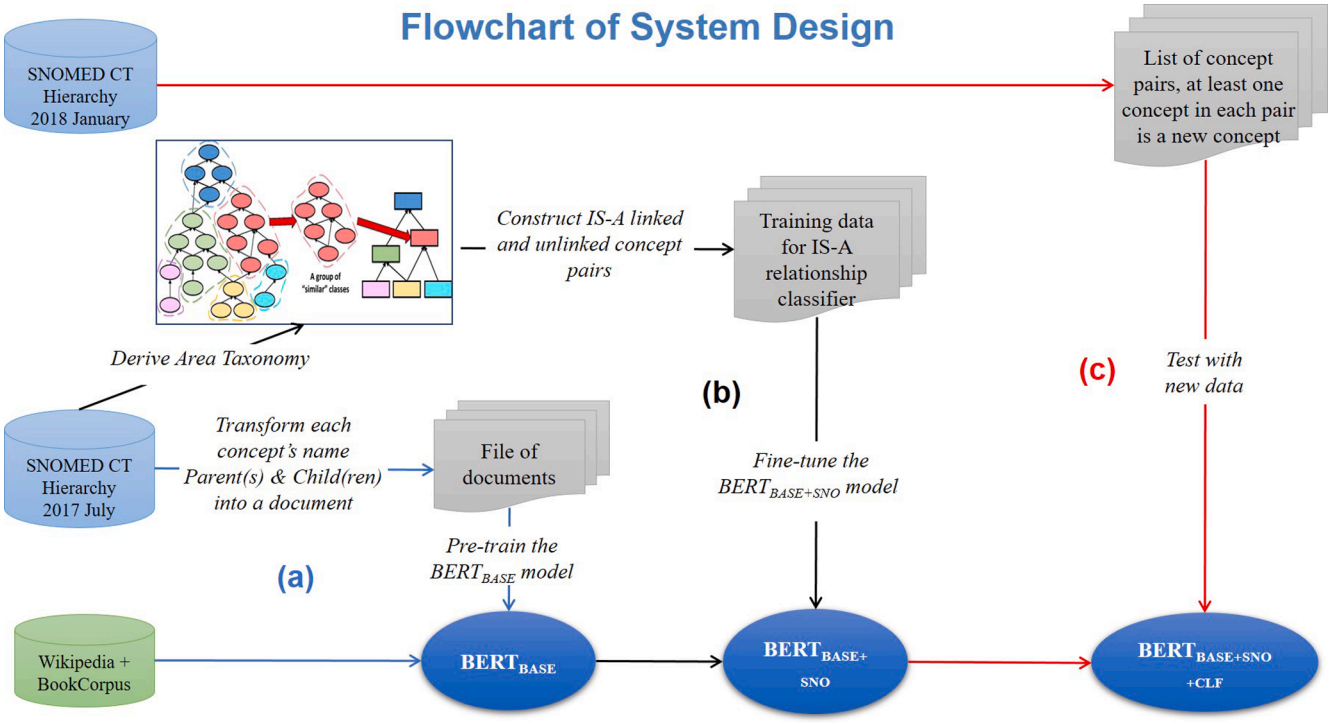



**Fig. 2.** Flowchart of training and testing an IS-A relationship classifier model with summarization data from Area Taxonomy of SNOMED CT (CLF = Classifier).

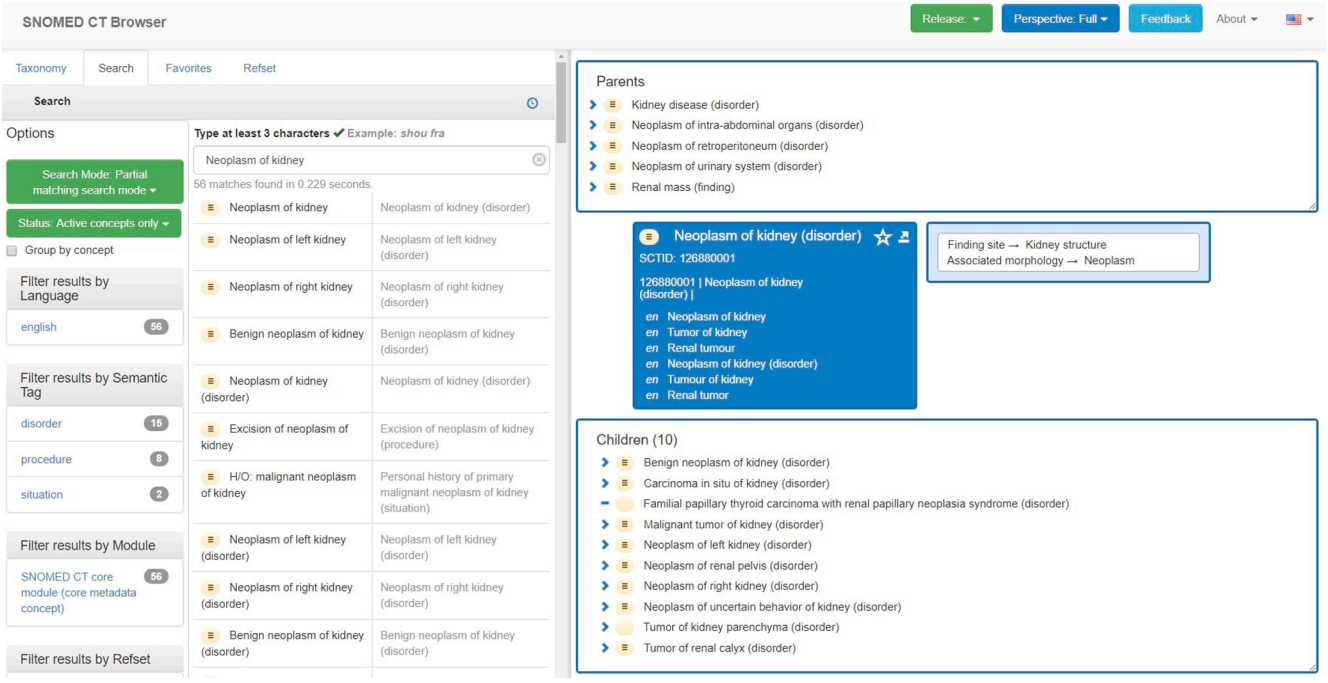
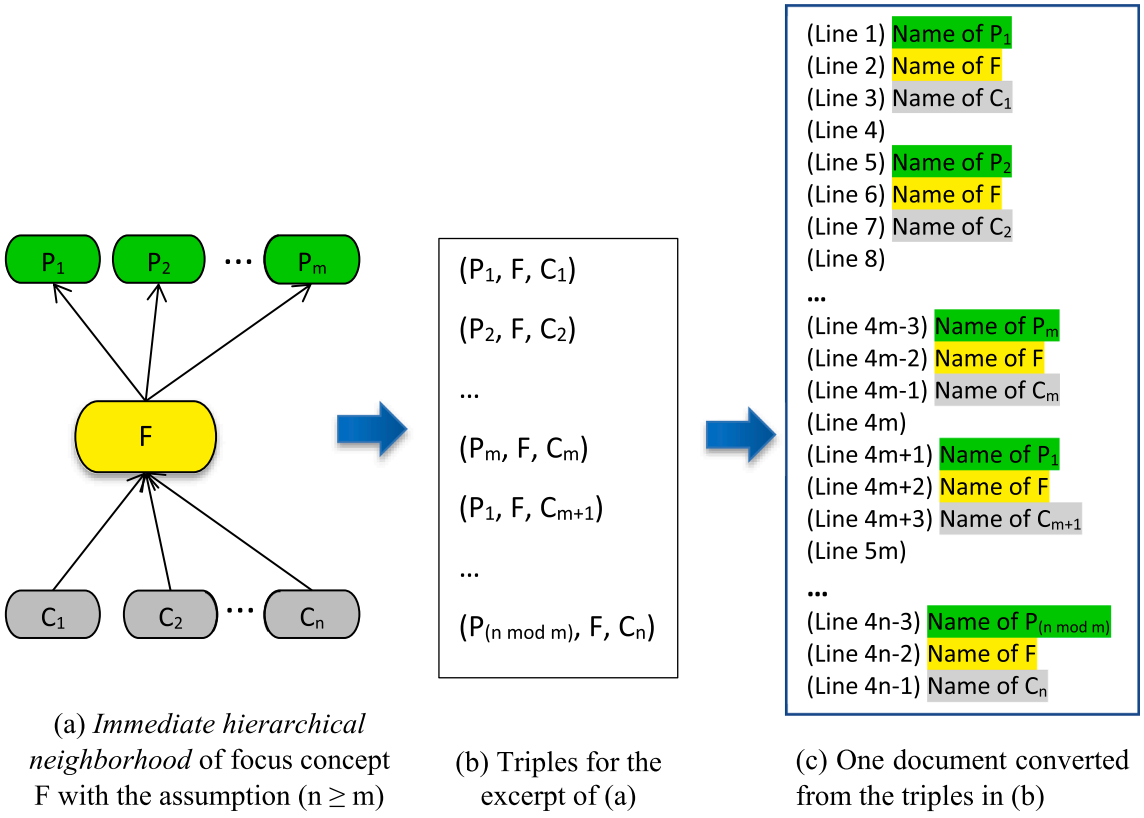



**Fig. 3.** Concept Neoplasm of kidney shown in SNOMED CT browser.

representation of hierarchically related concepts for a focus concept can vary, we preferred a simple pattern of a triple to form the document with a Parent – Focus concept – Child in each triple. In this way, two IS-A relationships are embedded, one from the focus concept to the parent concept, the other one from the child concept to the focus concept. Since the parents and children are important contextual knowledge elements of a focus concept, an *immediate neighborhood* [42], contains the concept itself plus all concepts connected to it by a single relationship, either




(a) *Immediate hierarchical neighborhood* of focus concept F with the assumption ($n \geq m$)

(b) Triples for the excerpt of (a)

(c) One document converted from the triples in (b)

**Fig. 4.** Generate a document for a focus concept F: (a) the hierarchical structure of a focus concept F with m parent concepts $P_1$ to $P_m$ and n child concepts $C_1$ to $C_n$, assuming $n \geq m$. (b) n (P, F, C) triples obtained from (a). (c) F's document representation by converting each triple in (b) into a three-line paragraph, with one concept's name per line. Each paragraph is separated from another paragraph by an empty line.

hierarchical or lateral. Derived from this definition, the *immediate hierarchical neighborhood* of a concept is defined as follows.

**Definition**. ((*Immediate hierarchical neighborhood*)) A concept's immediate hierarchical neighborhood contains the concept itself plus all concepts connected to it by a hierarchical relationship. That is, the immediate hierarchical neighborhood of a concept contains itself and all concepts at a hierarchical distance of one, i.e., its parents and children.

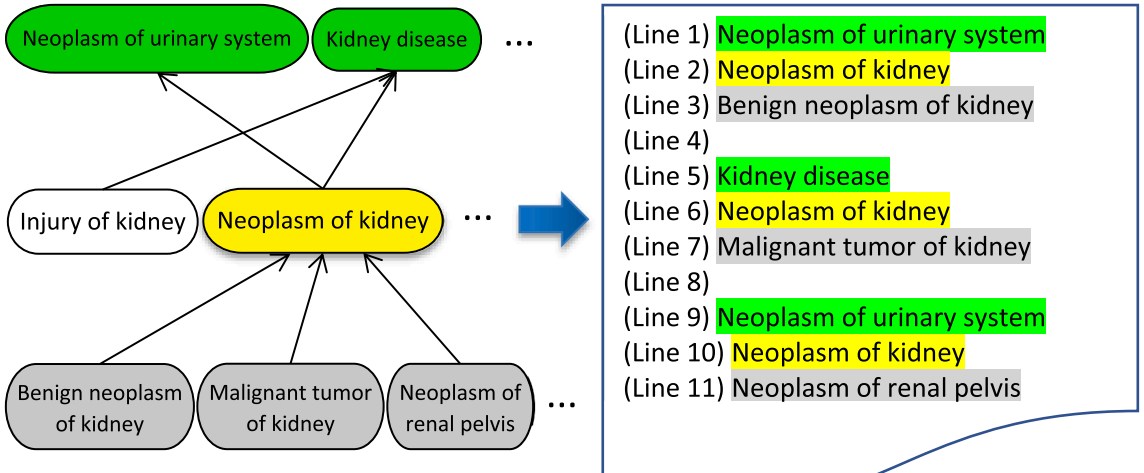
We illustrate a general configuration of a focus concept with its *immediate hierarchical neighborhood* in Fig. 4(a). Consider a focus concept F (in yellow) with its $m$ parents $P_1$ to $P_m$ (in green) and $n$ children $C_1$ to $C_n$ (in grey). We represent this configuration by triples of the form (Parent, Focus concept, Child) to capture all the $(m + n)$ IS-A relationships between the focus concept and its parent(s) and child(ren). We constructed triples by using the focus concept and matching the parents and children by their indexes, e.g. the second parent matching with the second child ($P_2$, F, $C_2$) as shown in Fig. 4(b). Assuming $m$ is less than $n$ (the number of parents is smaller than the number of children), after exhausting all parents, we continue to match remaining children with the parents from the beginning, e.g. ($P_1$, F, $C_{m+1}$), and ending with ($P_{(n \bmod m)}$, F, $C_n$) as shown in Fig. 4(b). In this matching process, for each child with index n $\geq$ m, for cases where n mod m = 0 (e.g. n = 2*m), parent $P_m$ is used instead of $P_{(n \bmod m)}$ to get ($P_m$, F, $C_n$). In the case when $n$ is less than $m$, a similar modification is used to deal with the remaining parents. In Fig. 4 (c) we demonstrate the transition from the ontology dimension, to the textual dimension that is needed for BERT, by converting each triple into a "paragraph." A paragraph consists of three lines to accommodate the three names, the parent concept, the focus concept, and the child concept, one name per line. The collection of $n$ paragraphs for the $n$ triples forms the document for the focus concept.

We demonstrate the above transformation with a concrete example with the focus concept *Neoplasm of kidney* in Fig. 5. Fig. 5(a) shows the neighborhood network of *Neoplasm of kidney* in yellow with its two parents (*Neoplasm of urinary system*, *Kidney disease* in green), and three children (*Benign neoplasm of kidney*, *Malignant tumor of kidney*, *Neoplasm of renal pelvis* in grey). These concepts are used to construct (Parent, Focus concept, Child) triples in which the focus concept is fixed and the parent concept and child concept are matched by their indexes, e.g. (*Kidney disease*, *Neoplasm of kidney*, *Malignant tumor of kidney*) is a triple matching the second parent with the second child. We stop generating triples when every concept is used in at least one triple. Each triple is converted to a three-line paragraph with one concept's name per line (Fig. 5(b)). Two paragraphs are separated by an empty line. For example, starting in Line 5 there are "(Line 5: Parent) *Kidney disease* – (Line 6: Focus concept) *Neoplasm of kidney* – (Line 7: Child) *Malignant tumor of kidney* – (Line 8) EMPTY LINE …". Thus, we generated a list of ontology-oriented documents by creating one "document" per concept. For simplicity, all the generated documents are concatenated in one text file, separated by two empty lines, as the input to pre-train the $\text{BERT}_{\text{BASE}}$ model.

### 3.1.2. Data preprocessing

After we obtained the list of ontology-oriented documents, we needed to perform preprocessing of the data prior to training BERT. The samples were preprocessed in three steps: 1) Text normalization (E.g., Fournier's gangrene, → fournier's gangrene), 2) Punctuation splitting (E.g., fournier's gangrene, → fournier ' s gangrene), and 3) WordPiece tokenization (fournier ' s gangrene, → four ##nier ' s gang ##ren ##e). BERT employs the WordPiece tokenizer [43] to segment a word into subword-level tokens, when necessary. Specifically, the Out-Of-Vocabulary (OOV) words are split as the combination of existing tokens in the vocabulary, e.g., "gangrene" is split into three tokens "gang," "##ren," and "##e." $\text{BERT}_{\text{BASE}}$ then converted the preprocessed samples into input embeddings, which are the sum of the token embeddings, the segmentation embeddings and the position embeddings [13]. We refer the reader to the BERT paper [13] for input embedding details, as we can only present the operations that are essential for pre-training.

In Fig. 6, we demonstrate an example of how a training instance is formed from the concept-level document. For example, *Formestane allergy* is the child of *Estrogen antagonist* in the SNOMED CT *Clinical Finding* hierarchy. The input sequence will be "1 *Estrogen antagonist (\t) Formestane allergy (\t)*". The first token of the sequence is the classification embedding ([CLS]), representing a classification label. It is "1" for positive instances and "0" for negative instances in the training data. A special token ([SEP]) is used to separate sentences. Out-of-vocabulary words are split into word pieces and denoted with ##. For example, "estrogen" is denoted as two items "est" and "##rogen." This input will be converted into one training instance as "[CLS] est ##rogen antagonist all ##ergy [SEP] form ##est ##ane all ##ergy [SEP]" as shown in Fig. 6(a). Similarly, *Main spoken language Turkmen* is not a child of *Born in Scotland*. The input sequence "0 *Born in Scotland (\t) Main spoken language Turkmen (\t)*" will be converted to "[CLS] born in scotland




(a) *Immediate hierarchical neighborhood* of *Neoplasm of kidney*

(b) One document for the excerpt of (a)

**Fig. 5.** Pre-training data: Serializing (a) the hierarchical structure of Neoplasm of kidney into (b) one document.

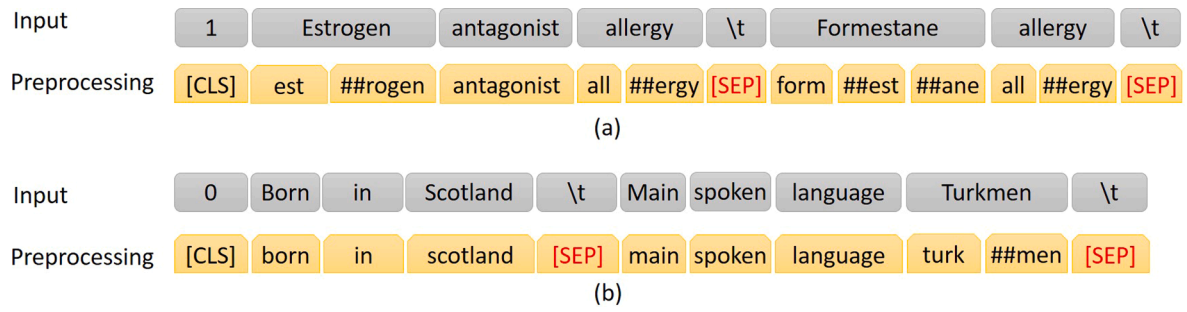



**Fig. 6.** Preprocessing (a) IS-A and (b) non-IS-A concept pairs.

[SEP] main spoken language turk ##men [SEP]" in Fig. 6(b).

#### 3.1.3. Pre-train BERT model

The goal of pre-training is to embed ontology knowledge into BERT's language model. Therefore, we advanced to training $BERT_{BASE}$ with concept-based documents (prepared above) from SNOMED CT. To ensure the obtained model is compatible with the original BERT model, we adopted the same two training tasks, Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) that BERT was originally trained for, since BERT is intended to process text. The training objective of the MLM task is to predict only the masked words. The training objective of NSP is to learn relationships between sentences (concepts, in our case) for any given sentence-pair. For the MLM task, 15% of the words are randomly masked out among all the concept-based documents, and for each document, an upper bound for the number of masks is set. Then the $BERT_{BASE}$ model is trained to output the masked words rather than other possible words. For the NSP task, the training objective is to learn the IS-A relationships between concepts: Given two concepts A and B, is B a child of A, or not. In Fig. 7, we extracted two "sentences" *Skin finding* and *Centrifugal rash* from the document for the focus concept *Centrifugal rash*. After preprocessing these two concepts (treated as two "sentences") as shown in the middle level, we masked out two token – "##ri" and "rash." The $BERT_{BASE}$ model was trained to raise the probabilities of two correct tokens "##ri" and "rash" over other tokens in the vocabulary. In addition, as *Centrifugal rash* IS-A *Skin finding*, the $BERT_{BASE}$ model was also trained to raise the probability for the correct classification label "IsNext." The obtained model is denoted as $BERT_{BASE+SNO}$ (SNO = SNOMED CT).

The training parameters used for Pre-training are as follows: batch size = 64, sequence length = 128, training steps = 200,000, learning rate = $2e^{-5}$, dropout rate = 0.1, and activation function = gelu (Gaussian error linear unit).

### 3.2. Fine-tuning BERT with data prepared using Area Taxonomy

#### 3.2.1. Data preparation

To fine-tune a BERT model into an IS-A relationship classifier, we needed to train the model with both IS-A connected concept pairs (positive instances) and concept pairs with no IS-A connections (non-IS-A concept pairs, in short, i.e., negative instances). The IS-A connected concept pairs are explicitly defined in the ontology's hierarchy. As discussed in Section 2.4, there may be errors in the IS-A hierarchy of SNOMED, and we have studied the identification of such errors extensively in the past. Nevertheless, almost all SNOMED IS-A relationships can be assumed to be correct. Thus, we are using existing IS-A relationships to prepare positive training instances. More precisely, the positive training data consists of all IS-A concept pairs in the *Clinical Finding and Procedure* hierarchies, respectively. However, the selection of negative training data (non-IS-A concept pairs) is critical for the accuracy of the model. To compare the performance of models trained with and without the Area Taxonomy-based summarization technique, we prepared two sets of negative training data using the following two methods.

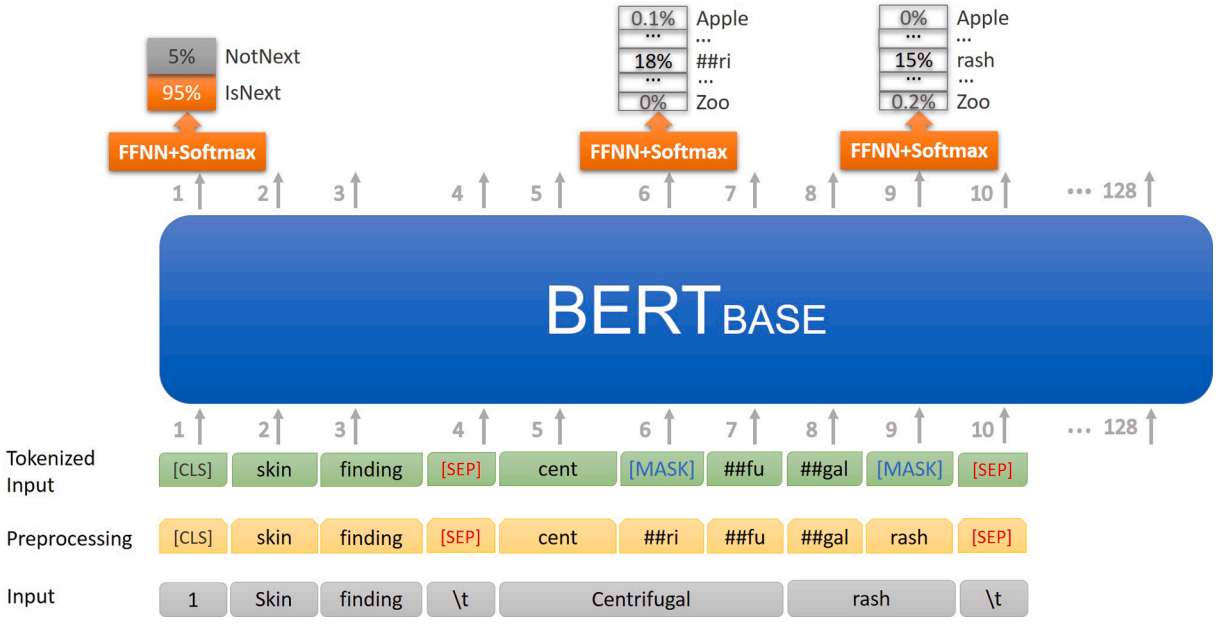



**Fig. 7.** Pre-training the $BERT_{BASE}$ model with concept-based documents to obtain $BERT_{BASE+SNO}$ model. FFNN is short for Feedforward neural network.

*3.2.1.1. Negative training data from hierarchy.* In the previous study [7], for a CNN model, we limited the non-IS-A pairs only to nephew-uncle pairs in the same hierarchy. The rationale was that a non-IS-A pair formed by two randomly sampled concepts that are likely completely unrelated concepts, will result in negative examples with a large semantic distance, that do not contribute to learning the "border surface" between positive and negative instances. Thus, for a given IS-A pair A *IS-A* B, it is more useful to learn differences between IS-A and non-IS-A pairs from a non-IS-A pair (A, C), where C is a "near miss," close to the border surface. We chose to use an uncle concept, i.e., a sibling of B, rather than an arbitrary concept D, which is likely not relevant to concept A, but still semantically close (see Fig. 8). The training data including all IS-A pairs and nephew-uncle pairs from the same hieachy will be referred to as **Hierachy data**.

*3.2.1.2. Negative training data from Area Taxonomy.* Utilizing the Area Taxonomy, the nephew-uncle pairs can be further divided into two types: uncle and nephew concepts are from the same area or uncle and nephew concepts are from different areas. A classification model can benefit more from training with nephew-uncle non-IS-A pairs from the same area, because in a pair from the same area the two concepts are more closely related than in a pair from different areas. Thus, the classification model can learn the features representing the subtle differences between IS-A pairs and nearby non-IS-A pairs. As a result, the extracted features will enable the classification model to better verify whether a concept pair should be connected by an IS-A link or not, achieving better testing performance.

We demonstrate the above observation with a concrete example (Fig. 9) from the *Clinical Finding* hierarchy. Let the nephew concept be *Hearing difficulty* (in yellow). Its five uncle concepts are *Acquired hearing loss*, *Audiogram abnormal*, *Hearing symptoms*, *Perception of hearing loss,* and *Hearing disorder*. The first two concepts are the siblings of the concept *Decreased hearing*, which is a parent of *Hearing difficulty,* in the same area. The other three uncle concepts are siblings of the concept *Decreased hearing*, because they are all children of *Hearing finding* in different areas from *Hearing Difficulty*.

As Fig. 9 shows, the first two uncle concepts (in green) are from the area {*Finding site, Interprets*} that contains *Hearing difficulty*. In contrast, two uncle concepts *Hearing sympotoms* and *Perception of hearing loss* (in red) are in the area {*Finding site*, *Interprets, Finding informer*}, while the other uncle concept *Hearing disorder* is in the area {*Finding site*, *Interprets*}. Both are in different areas than the nephew concept *Hearing difficulty*. *Hearing difficulty* is semantically more similar to *Acquired hearing loss,* and *Audiogram abnormal* from the same area as they are all various kinds of hearing findings similar to *Hearing difficulty*. *Hearing difficulty* is less similar to *Hearing sympotoms* and *Perception of hearing loss* in a different area, since they represent symptoms and the perception of hearing. Similarly, *Hearing difficulty* is also less similar to *Hearing disorder* in another area, which is a more general concept that is the root of a subhierarchy consisting of hearing disease concepts (not shown in the diagram). The training data including all IS-A pairs and nephew-uncle pairs from the same area is refered as **Area Taxonomy data**, in contrast to the previously introduced Hierarchy data.

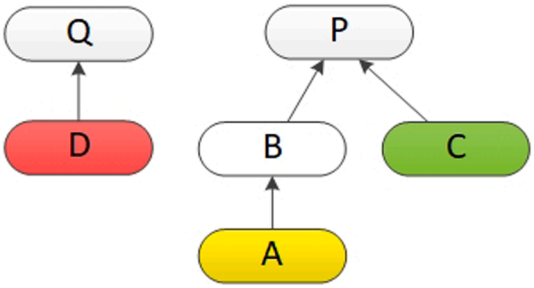



**Fig. 8.** The nephew-uncle pair (A, C).

*3.2.2. Training an IS-A relationship classifier*

In each experiment, we trained two independent classifiers (models) using different data sets prepared with the above two techniques for performance comparison. In an ontology, there are more non-IS-A concept pairs (negative pairs) than IS-A concept pairs (positive pairs). In case of an extreme imbalance between the number of positive pairs and the number of negative pairs, the balancing of the dataset is a common practice to prevent a bias of the model's prediction [44]. We followed this practice, since in a hierarchy the number of non-IS-A concept pairs is of a higher magnitude than the number of IS-A concept pairs. To avoid an imbalanced training data issue, after we extracted the positive and negative pairs, we randomly downsampled the collection of negative pairs to the size of the set of positive pairs in each training round for both models. Then the dataset was divided according to a 90:10 ratio for training and validation, respectively.

Thus, the $\text{BERT}_{\text{BASE+SNO}}$ model was fine-tuned in the training phase to predict the correct labels for the IS-A concept pairs and the non-IS-A concept pairs, utilizing the NSP binary sentence-pair classification task. We have used the sentence prediction capability of $\text{BERT}_{\text{BASE+SNO}}$, and added a *softmax* layer with categorical cross-entropy on top of it. The obtained model is denoted as $\text{BERT}_{\text{BASE+SNO+CLF}}$ (CLF = classifier), the model after fine-tuning.

To achieve this, the model and the classifier were trained at the same time to predict IS-A links between concept pairs of a ontology concepts, i.e., the parameters of $\text{BERT}_{\text{BASE+SNO}}$ and the classifier were fine-tuned to maximize the log-probability of the correct label (IS-A or non-IS-A). We illustrate this process with the concept *Edema of wrist* as an example in Fig. 10. The input "1 *Finding of wrist region (\t) Edema of wrist (\t)*" was converted as one training instance to "[CLS] finding of wrist region [SEP] ed ##ema of wrist [SEP]" with Class label = 1. Class 1 means that there should be an IS-A link between the two concepts, and Class 0 means that there shouldn't be such a link. The $\text{BERT}_{\text{BASE+SNO+CLF}}$ model computes the probabilities for Class 0 and Class 1, and records the result as a 2 element vector. The label of the class with the higher probability is reported as the prediction output. The error between the predicted label and the true label was backpropagated through the model to improve the model's parameters. For this we used the default model hyperparameters in pre-trained $\text{BERT}_{\text{BASE+SNO}}$, with one exception, the number of training epochs (=6).

*3.3. Test with new data*

To evaluate the $\text{BERT}_{\text{BASE+SNO+CLF}}$ models on previously unseen data, we created separate test tasks, using new concepts from the *Clinical finding* and *Procedure* hierarchies of the January 2018 release. For each new concept that was added to the *Clinical finding/Procedure* hierarchy in this release, we prepared both positive and negative samples for testing. To obtain positive testing samples, we extracted each new concept and its parents as IS-A concept pairs. For example, *Lesion of left ear* has two parents *Disorder of left ear* and *Ear lesion.* The corresponding positive testing samples are "*Disorder of left ear* (\t) *Lesion of left ear*" and "*Ear lesion* (\t) *Lesion of left ear*" with the class label = 1 (true).

For the negative samples, we are not limited to nephew-uncle pairs, but can use any combination of non-IS-A concept pairs. However, it is not practical to test each new concept by pairing it with all existing 111,081 (or 57,806) concepts in the *Clinical Finding* (or *Procedure*) hierarchy, because computing would take an inordinate amount of time. As mentioned above, it is a common practice in Machine Learning (ML), in case of extreme imbalance between two categories, to extract a smaller random sample for testing [45]. The benefit of sampling an equal number of positive and negative testing instances is that the calculations of Precision, Recall, and F1 scores is straightforward. Otherwise, additional metrics may be needed for evaluation measures [46]. Thus, we have chosen an equal number of positive and negative

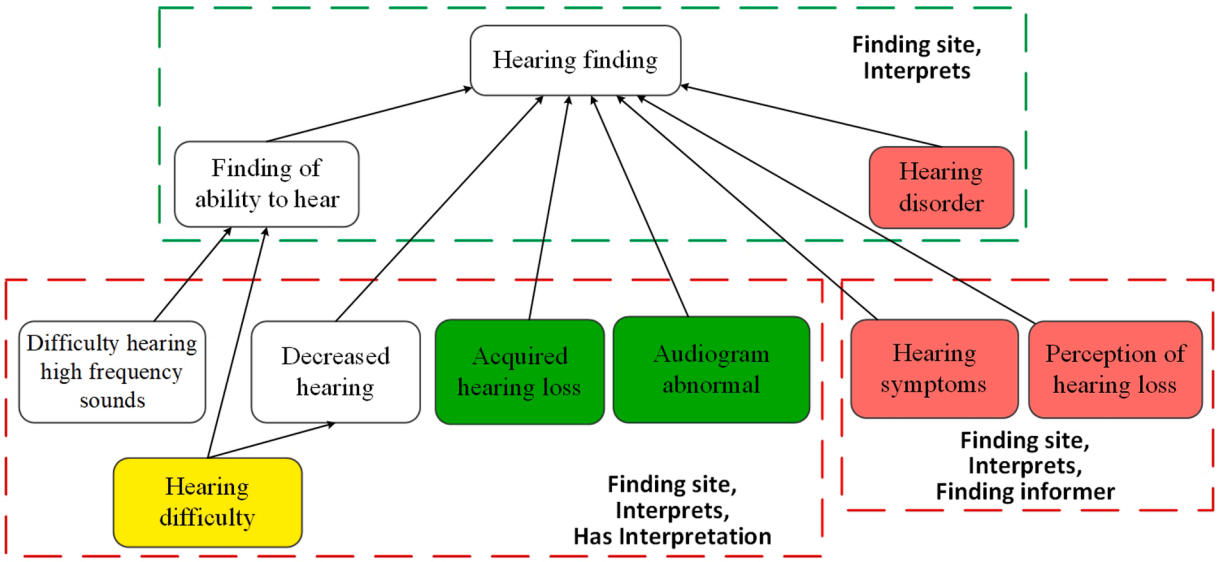



**Fig. 9.** Uncle-nephew pairs within/without the same area.

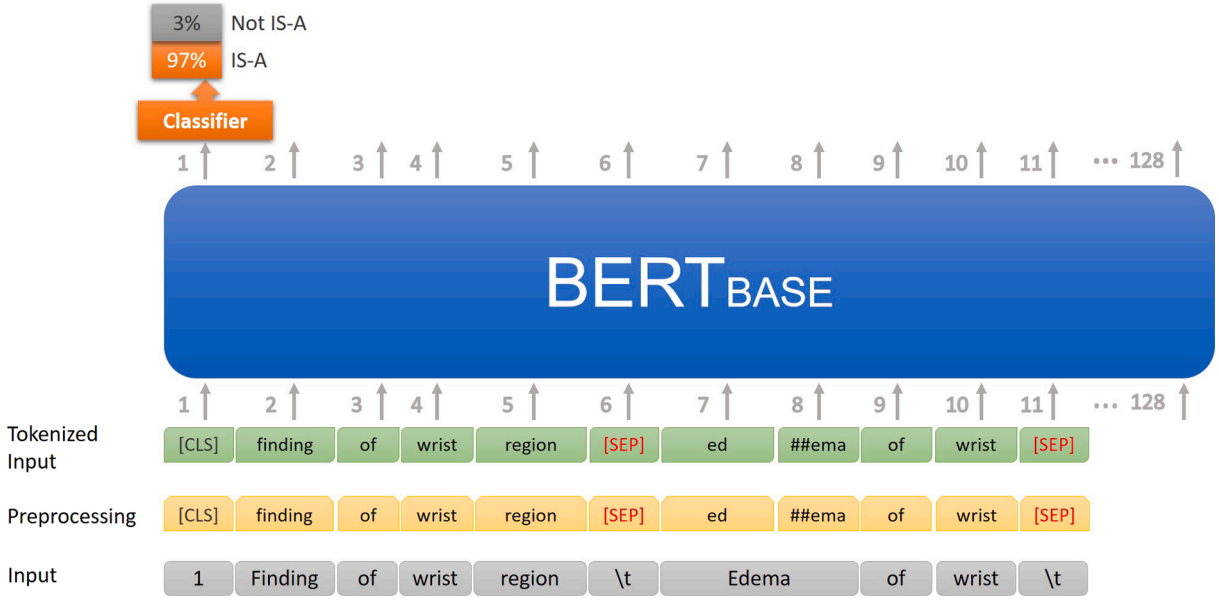



**Fig. 10.** Fine-tuning the $BERT_{BASE+SNO}$ model with concept pairs to obtain $BERT_{BASE+SNO+CLF}$ model.

instances. To obtain negative testing samples, we paired each new concept with a randomly chosen concept from the other new concepts' parents as non-IS-A concept pairs. For example, we randomly selected *Disorder of soft tissue of upper limb*, which is the parent of *Congenital trigger finger of right hand*, and paired it with *Lesion of left ear* to form a testing instance "*Disorder of soft tissue of upper limb* (\t) *Lesion of left ear*" with the expected label = 0.

We randomly shuffled all the testing concept pairs into batches and sent them to the trained $BERT_{BASE+SNO+CLF}$ models for prediction. The tested models use the previously learned weights to process each input pair and return a class label (0 or 1) as prediction result. Label 1 is correct for a positive testing sample, indicating the existence of an IS-A link in the new SNOMED CT release. In other words, the existence of an IS-A link in the new release of SNOMED CT is correctly predicted. In the Background section, we discussed the possibility of errors among the existing IS-A relationships in SNOMED. Nevertheless, such errors would occur in a very small proportion of the concepts, given the extensive work that has gone into improving the SNOMED CT over the past decade. Thus, we are considering the parents of the new concepts that were added in the new release as gold standard for evaluating the correctness of our predictions. For a negative testing sample, there should be no IS-A link between these two concepts, so label 0 is correct. We calculated the prediction accuracy in terms of Precision, Recall, F1 and F2 scores by comparing the result labels predicted by the tested model with the ground-truth labels.

## 4. Results

We report the prediction results of the two models, one trained with Hierarchy data (referred to as Hierarchy model) and the other one trained with Area Taxonomy data (referred to as Area Taxonomy model). The testing samples (concept pairs) were extracted from the *Clinical Finding* and *Procedure* hierarchies of the SNOMED CT 2018 January release, which were not included in the training. In each experiment, we tested the two trained models using the same testing samples. Besides the typical metrics Precision, Recall and F1, we used

another metric called F2. The F2 score is calculated from the generalized F score $F_\beta$ where

$$F_\beta = (1+\beta^2)*\frac{\text{precision}*\text{recall}}{(\beta^2*\text{precision})+\text{recall}}$$

with $\beta = 2$. We set $\beta = 2$ to emphasize that recall is considered more important than precision in this task. This experiment was repeated ten times and the Precision (P), Recall (R), F1, and F2 scores for the corresponding ten tests with the *Clinical Finding* hierarchy are presented in Table 1. Each model was tested against 8,574 pairs (4,287 positives and 4,287 negatives). For example, in Test 5 for IS-A classification, the Precision is 0.83, Recall is 0.93, F1 score is 0.88, and F2 score is 0.91 for the Hierarchy model. When testing the Area Taxonomy model, Precision is 0.81, Recall is 0.96, F1 score is 0.88, and F2 score is 0.93. The Precision score dropped from 0.83 to 0.81 while the Recall score increased from 0.93 to 0.96, improving by 3%. For Non-IS-A tests, the Recall score dropped from 0.81 to 0.78 while the Precision score increased from 0.93 to 0.95. Comparing the average of ten experiments between the Hierarchy model and the Area Taxonomy model shows that the Area Taxonomy model improves the recall score from 0.94 to 0.96 at the cost of the Precision score dropping from 0.85 to 0.80, and the F1 score drops from 0.89 to 0.87, while the F2 score remains the same as 0.92.

Table 2 shows the Precision (P), Recall (R), F1, and F2 scores of the ten experiments with the *Procedure* hierarchy. Each model was tested against 3,908 pairs (1,954 positives and 1,954 negatives). For example, in Test 9 for IS-A classification, the prediction results for the Hierarchy model are Precision = 0.78, Recall = 0.98, F1 score = 0.87, and F2 score = 0.93. When testing the Area Taxonomy model, the Precision is 0.74, Recall is 0.99, F1 score is 0.85, and F2 score is 0.93. Comparing the two models by averaging ten experiments, the Precision score dropped from 0.776 to 0.703 while the Recall score increased from 0.980 to 0.985. The average F1 scores for the two models are 0.867 and 0.821, respectively, while the F2 scores are 0.931 and 0.912.

Regarding the prediction of IS-A links for new concepts, we show ten examples of our two models' prediction results (Table 3) for ten pairs for which the second concept was newly added to SNOMED CT's *Clinical finding* hierarchy in the 2018 January release. The first five examples are IS-A connected concept pairs, which is indicated by the value 1 in the True label column. The other five examples are synthesized non-IS-A concept pairs, indicated in the True label column by 0.

For each test, we paired one *Test Concept* with one *New Concept* as one test instance, then we let the model predict IS-A links between them. For instance, for Example 3, we chose *Cerebrovascular disease* as the Test concept and paired it with the new concept *Occlusion of left pontine artery*. Then the task became to predict whether there is an IS-A link between the two concepts. Both the Hierarchy model and the Area Taxonomy model returned the correct label (=1). Correct predictions are marked in green. In example 4, the Hierarchy model is wrong, and the Area Taxonomy model is correct that *Congenital conductive hearing loss* IS-A *Decreased hearing*. Both models are wrong about *Bone cyst of right foot,* because it is not an *Osteomyelitis of right ankle* (Example 9), thus they are marked in red.

After downsampling the training data from the majority class (negative training samples), we also experimented with combining upsampling the minority class (positive training samples) and downsampling the majority class to tackle the imbalanced training dataset issue. We refer to this as "mixed sampling." For the *Clinical Finding* hierarchy, we upsampled the positive training data to two times its original size and then downsampled the negative training data to three times of the original size of the positive training data, to obtain a positive to negative data ratio of 2:3. For the *Procedure* hierarchy, we upsampled the positive training data to five times its original size and then downsampled the negative training data to the (same) size of the positive upsampled data, to obtain a positive to negative data ratio of 1:1. For both hierarchies, 20 experiments were conducted, namely ten experiments with summarization and ten without summarization.

The average Precision, Recall, F1 and F2 results for the *Clinical Finding* hierarchy with downsampling only are compared with mixed sampling in Table 4. For IS-A classification, the averaged F1 score of the Hierarchy model with downsampling-only is 0.89, which is marginally better than the F1 score of 0.88 for mixed sampling. Similarly, for the Area Taxonomy model, the average F1 scores are 0.87 vs. 0.86. These differences are practically not significant. We observed a similar trend for the non-IS-A classification, where the models with downsampling-only achieved slightly higher averaged F1 scores.

The averaged Precision, Recall, F1 and F2 results of ten experiments with downsampling-only for the *Procedure* hierarchy are compared with mixed sampling in Table 5. For IS-A classification, the averaged F1 scores of the Area Taxonomy model using downsampling-only is 0.82, which is better than for mixed sampling, where the F1 score of 0.81. For the Hierarchy model, the average F1 scores are the same: 0.87. We observed a similar trend for the non-IS-A classification, where the model with downsampling-only achieved higher averaged F1 scores for the Area Taxonomy model, while having the same F1 scores for the Hierarchy model.

## 5. Discussion

In this paper, we have moved from the CNN model in previous work [7,9] to using the BERT model. We investigated a hybrid technique, testing whether ontology summarization can be used with BERT, a model targeting NLP applications, to improve the recall for concept placement. The model differences lie in the neural network structures. BERT is composed of transformers, while CNN is a neural network performing layer-wise convolutions. Transformers in BERT are attention-based, which means BERT can learn the underlying

**Table 1**
Precision (P), Recall (R), F1, and F2 scores for ten experiments of *Clinical Finding* hierarchy.

| Clinical Finding | IS-A Classification | | | | | | | | Non-IS-A Classification | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Hierarchy | | | | Area Taxonomy | | | | Hierarchy | | | | Area Taxonomy | | | |
| No. | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 |
| 1 | 0.83 | 0.94 | 0.88 | 0.92 | 0.79 | 0.95 | 0.87 | 0.91 | 0.93 | 0.81 | 0.87 | 0.83 | 0.94 | 0.75 | 0.84 | 0.78 |
| 2 | 0.84 | 0.93 | 0.88 | 0.91 | 0.8 | 0.96 | 0.87 | 0.92 | 0.93 | 0.82 | 0.87 | 0.84 | 0.95 | 0.77 | 0.85 | 0.80 |
| 3 | 0.85 | 0.94 | 0.9 | 0.92 | 0.8 | 0.96 | 0.87 | 0.92 | 0.94 | 0.84 | 0.88 | 0.86 | 0.95 | 0.76 | 0.84 | 0.79 |
| 4 | 0.87 | 0.94 | 0.9 | 0.93 | 0.8 | 0.96 | 0.87 | 0.92 | 0.93 | 0.86 | 0.9 | 0.87 | 0.95 | 0.76 | 0.85 | 0.79 |
| 5 | 0.83 | 0.93 | 0.88 | 0.91 | 0.81 | 0.96 | 0.88 | 0.93 | 0.93 | 0.81 | 0.87 | 0.83 | 0.95 | 0.78 | 0.85 | 0.81 |
| 6 | 0.86 | 0.94 | 0.9 | 0.92 | 0.81 | 0.96 | 0.88 | 0.93 | 0.93 | 0.85 | 0.89 | 0.86 | 0.95 | 0.78 | 0.86 | 0.81 |
| 7 | 0.85 | 0.94 | 0.89 | 0.92 | 0.79 | 0.96 | 0.87 | 0.92 | 0.93 | 0.84 | 0.88 | 0.86 | 0.95 | 0.75 | 0.84 | 0.78 |
| 8 | 0.84 | 0.94 | 0.89 | 0.92 | 0.8 | 0.96 | 0.87 | 0.92 | 0.93 | 0.83 | 0.88 | 0.85 | 0.95 | 0.76 | 0.84 | 0.79 |
| 9 | 0.83 | 0.94 | 0.89 | 0.92 | 0.8 | 0.96 | 0.87 | 0.92 | 0.94 | 0.81 | 0.87 | 0.83 | 0.95 | 0.76 | 0.84 | 0.79 |
| 10 | 0.87 | 0.94 | 0.9 | 0.93 | 0.8 | 0.96 | 0.87 | 0.92 | 0.93 | 0.85 | 0.89 | 0.86 | 0.95 | 0.75 | 0.84 | 0.78 |
| Average | 0.85 | **0.94** | 0.89 | 0.92 | 0.80 | **0.96** | 0.87 | 0.92 | 0.93 | 0.83 | 0.88 | 0.85 | 0.95 | 0.76 | 0.85 | 0.79 |
| Standard Deviation | 0.02 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.01 | 0.02 | 0.00 | 0.01 | 0.01 | 0.01 |

**Table 2**
Precision (P), Recall (R), F1, and F2 scores for ten experiments of *Procedure* hierarchy.

| *Procedure* | IS-A Classification | | | | | | | | Non-IS-A Classification | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Hierarchy | | | | Area Taxonomy | | | | Hierarchy | | | | Area Taxonomy | | | |
| No. | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 |
| 1 | 0.78 | 0.98 | 0.87 | 0.93 | 0.7 | 0.99 | 0.82 | 0.91 | 0.97 | 0.72 | 0.83 | 0.76 | 0.98 | 0.58 | 0.73 | 0.63 |
| 2 | 0.78 | 0.98 | 0.87 | 0.93 | 0.69 | 0.98 | 0.81 | 0.90 | 0.98 | 0.72 | 0.83 | 0.76 | 0.97 | 0.56 | 0.71 | 0.61 |
| 3 | 0.77 | 0.98 | 0.86 | 0.93 | 0.7 | 0.98 | 0.82 | 0.91 | 0.98 | 0.71 | 0.82 | 0.75 | 0.97 | 0.58 | 0.73 | 0.63 |
| 4 | 0.77 | 0.98 | 0.86 | 0.93 | 0.69 | 0.98 | 0.81 | 0.90 | 0.98 | 0.7 | 0.82 | 0.74 | 0.97 | 0.56 | 0.71 | 0.61 |
| 5 | 0.77 | 0.98 | 0.86 | 0.93 | 0.7 | 0.99 | 0.82 | 0.91 | 0.97 | 0.71 | 0.82 | 0.75 | 0.98 | 0.57 | 0.72 | 0.62 |
| 6 | 0.77 | 0.98 | 0.87 | 0.93 | 0.69 | 0.98 | 0.81 | 0.90 | 0.98 | 0.71 | 0.82 | 0.75 | 0.97 | 0.56 | 0.71 | 0.61 |
| 7 | 0.77 | 0.98 | 0.87 | 0.93 | 0.73 | 0.99 | 0.84 | 0.92 | 0.98 | 0.71 | 0.82 | 0.75 | 0.98 | 0.64 | 0.77 | 0.69 |
| 8 | 0.8 | 0.98 | 0.88 | 0.94 | 0.71 | 0.98 | 0.82 | 0.91 | 0.98 | 0.75 | 0.85 | 0.79 | 0.97 | 0.59 | 0.73 | 0.64 |
| 9 | 0.78 | 0.98 | 0.87 | 0.93 | 0.74 | 0.99 | 0.85 | 0.93 | 0.98 | 0.71 | 0.83 | 0.75 | 0.98 | 0.66 | 0.79 | 0.71 |
| 10 | 0.77 | 0.98 | 0.86 | 0.93 | 0.68 | 0.99 | 0.81 | 0.91 | 0.97 | 0.7 | 0.81 | 0.74 | 0.97 | 0.54 | 0.7 | 0.59 |
| Average | 0.776 | **0.980** | 0.867 | 0.931 | 0.703 | **0.985** | 0.821 | 0.912 | 0.977 | 0.714 | 0.825 | 0.755 | 0.974 | 0.584 | 0.730 | 0.635 |
| Standard Deviation | 0.01 | 0.00 | 0.01 | 0.00 | 0.02 | 0.01 | 0.01 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.01 | 0.04 | 0.03 | 0.04 |

**Table 3**
Prediction results of two models on five IS-A & five non-IS-A examples from *Clinical Finding* hierarchy. Green fill indicates that the model correctly predicted the True Label.

| Index | Test Concept | New Concept | True Label | Hierarchy model prediction | Area Taxonomy prediction |
|---|---|---|---|---|---|
| 1 | *Visual cortex injury* | *Injury of right visual cortex* | 1 | 1 | 1 |
| 2 | *Drug therapy finding* | *Has supply of rescue medication* | 1 | 1 | 1 |
| 3 | *Cerebrovascular disease* | *Occlusion of left pontine artery* | 1 | 1 | 1 |
| 4 | *Decreased hearing* | *Congenital conductive hearing loss* | 1 | 0 | 1 |
| 5 | *Congenital anomaly of fetus* | *Malformation of central nervous system of fetus* | 1 | 1 | 1 |
| 6 | *Disorder of bilateral ulnar nerves* | *Loss of tissue of right eye co-occurrent with laceration* | 0 | 0 | 0 |
| 7 | *Gastric ulcer* | *Complex burn of wrist* | 0 | 0 | 0 |
| 8 | *Occlusion of left cerebellar artery* | *Dissection of basilar artery* | 0 | 0 | 0 |
| 9 | *Osteomyelitis of right ankle* | *Bone cyst of right foot* | 0 | 1 | 1 |
| 10 | *Injury of toe* | *Open wound of left foot* | 0 | 1 | 0 |

"attention" of the input text and each word's relationships to its context. This makes BERT more appropriate for this concept placement task (which is text-based relationship identification) as CNN is merely functioning as a classifier.

One question is whether an ML model should be better trained with the whole SNOMED or just with the hierarchy to which we are adding this concept. The ontological modeling differences among the 19 hierarchies of SNOMED CT will inevitably propagate to the downstream ML model. This may impair the performance of the trained ML model in distinguishing IS-A links in different hierarchies, because different hierarchies are often modeled by different curators and are modeled following different modeling principles. The features learned from the *Clinical Finding* hierarchy are most likely less useful for distinguishing IS-A links in other hierarchies, e.g., *Specimen*, that are covering a different subject. Verifying this plausible conjecture remains a task for future work.

To assess the performance of traditional Machine Learning methods for classification or prediction problems, the common approach is to evaluate a model/system's performance with the F1 score, the harmonic average of recall and precision. However, there are applications, where recall is more important than precision. In some areas, such as web search, the precision is almost impossible to ascertain, because many web searches report tens of thousands of hits that cannot be evaluated manually. When recall is more important than precision, researchers traditionally switch to a higher order F-measure such as F2, as was done in this paper, contrasting it with F1. Recall is considered more important whenever the penalty for missing a positive instance is much higher than the penalty of getting a negative instance falsely reported as positive. Thus, in medicine/medical informatics, we want all cancer cases to be discovered by a (cheap) test, at the risk of getting false positives that can be disproved by a subsequent (more expensive) test.

A similar situation exists in NLP, when a two-step process of a fast and simple tagger and a more complex parser is used. The tagger is expected to have high recall, which will improve the accuracy of the parsing step [47]. The task of finding one or several parents for 2005 new concepts is difficult and time-consuming for a human curator. Thus, we view our model-based prediction as corresponding to the NLP tagger, while the human curator takes on the role of the parser. This makes the recall of the model-based prediction more important than the precision.

Our objective is to increase the recall when identifying the parent(s),

**Table 4**
The average Precision (P), Recall (R), F1, and F2 scores for ten experiments of the *Clinical Finding* hierarchy with two sampling approaches.

| *Clinical Finding* | | IS-A Classification | | | | | | | | Non-IS-A Classification | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Hierarchy | | | | Area Taxonomy | | | | Hierarchy | | | | Area Taxonomy | | | |
| | | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 |
| Downsampling-only | Average | 0.85 | 0.94 | **0.89** | 0.92 | 0.80 | 0.96 | **0.87** | 0.92 | 0.93 | 0.83 | 0.88 | 0.85 | 0.95 | 0.76 | 0.85 | 0.79 |
| | Standard Deviation | 0.02 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.01 | 0.02 | 0.00 | 0.01 | 0.01 | 0.01 |
| Upsampling positive sample and downsampling negative sample to achieve 2:3 | Average | 0.83 | 0.92 | **0.88** | 0.90 | 0.79 | 0.95 | **0.86** | 0.91 | 0.91 | 0.81 | 0.86 | 0.83 | 0.93 | 0.74 | 0.83 | 0.77 |
| | Standard Deviation | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.01 | 0.00 | 0.02 | 0.01 | 0.02 |

**Table 5**
The averaged Precision (P), Recall (R), F1, and F2 scores for ten experiments of the *Procedure* hierarchy with two sampling approaches.

| *Procedure* | | IS-A Classification | | | | | | | | Non-IS-A Classification | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Hierarchy | | | | Area Taxonomy | | | | Hierarchy | | | | Area Taxonomy | | | |
| | | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 | P | R | F1 | F2 |
| Downsampling-only | Average | 0.78 | 0.98 | **0.87** | 0.93 | 0.70 | 0.99 | **0.82** | 0.91 | 0.98 | 0.71 | 0.83 | 0.75 | 0.97 | 0.58 | 0.73 | 0.63 |
| | Standard Deviation | 0.01 | 0.00 | 0.01 | 0.00 | 0.02 | 0.01 | 0.01 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.01 | 0.04 | 0.03 | 0.04 |
| Upsampling positive sample 5 times and downsampling negative sample to 1:1 | Average | 0.79 | 0.97 | **0.87** | 0.93 | 0.70 | 0.98 | **0.81** | 0.90 | 0.96 | 0.74 | 0.83 | 0.77 | 0.96 | 0.58 | 0.72 | 0.63 |
| | Standard Deviation | 0.02 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.00 | 0.03 | 0.02 | 0.02 | 0.00 | 0.02 | 0.01 | 0.02 |

thus placing more concepts algorithmically in their proper positions in the hierarchy. High recall and low precision mean that some parents proposed by the system will be wrong, but most or all real parents will be identified. A manual review by a curator will easily expose false parents. For a curator it is a much easier task to verify a proposed placement and reject false parents, than to find the proper placement without algorithmic help. Hence, we chose to increase the recall at the expense of lowering the precision.

To illustrate this point, consider an example with 100 true IS-A links to correct parents. Method 1 finds 120 IS-A links, of which 90 are correct. Method 2 finds only 100 links, of which 80 are correct. For Method 1 the numbers are (R = 0.9, P = 0.75); for Method 2 (R = 0.8, P = 0.8). For the human curator it is easier to reject the 30 false positives of Method 1 than to find the 10 links that Method 2 is missing relative to Method 1, because she would potentially need to review *the complete hierarchy*. For example, in our testing there are five parent candidates for concept *Laceration of adductor muscle of thigh*: *Traumatic injury of skeletal muscle*, *Laceration of lower limb*, *Weakness of extremities as sequela of stroke*, *Paresthesia*, *Subretinal lesion*. As *Laceration of adductor muscle of thigh* is essentially a muscle laceration of the thigh, a domain expert can easily accept the first two concepts *Traumatic injury of skeletal muscle* and *Laceration of lower limb* as the parents, and reject the other concepts because they are out of scope of muscle laceration.

Consider our novel transformation from the immediate hierarchical neighborhood (Fig. 4(a)) to a text format fitting for training BERT (Fig. 4 (c)). In Fig. 4(a) there are $m + n$ IS-A relationships, which are all captured in the text, where *n-m* IS-A relationships from the focus concept F to parents are repeated. In our earlier work [17], the modeling was different. It was composed of three sentences. In the first sentence, all the parents' names were concatenated. The second sentence was the focus concept's name. The third sentence was obtained by concatenating all the children's names. In that configuration, only two IS-A relationships of the $m + n$ IS-A relationships in the immediate hierarchical neighborhood of F were captured: from F to the last parent $P_m$ and from the first child $C_1$ to F. Those were the only two occurrences of consecutive pairs of phrases, reflecting IS-A relationships in the resultant text. The name of F is not the next sentence for the phrase with the name of any of the other parents, as the name of $P_m$ is in the middle.

A similar situation exists for the children. Thus the previously reported modeling [17] was inferior to the one presented in this paper. Indeed, comparing with the testing result for the *Clinical Finding* hierarchy [17], while the average recall here is about the same, the precision is improved from 0.79 to 0.85, reflecting a better training of the BERT model to distinguish between IS-A links and non-IS-A links. This improvement shows the importance of accurate modeling of such a transformation from the immediate hierarchical neighborhood to a text format. The test data in both studies has been the same, reflecting the upgrade of SNOMED from the July 2017 release to the January 2018 release. However, the previous research [17] was important as the first work showing that it is possible to harness the power of the BERT model to differentiate between IS-A pairs and pairs not linked by IS-A relationships in an ontology.

For training, we selected negative instances that are close to positive instances to better train the model to distinguish between IS-A and non-IS-A concept pairs. Better training should yield better performance in testing. However, during testing, we need to test with a sample taken from all concepts in the hierarchy, since we do not know where the parents will be.

For both hierarchies, the new transformation from the immediate hierarchical neighborhood of the ontology to the text format yields a very high recall without ontology summarization, 0.94 and 0.98 for the *Clinical finding* and *Procedure* hierarchies, respectively. This high recall result leaves only little room (0.06 and 0.02) for improvement by the taxonomy summarization. For the *Clinical finding* hierarchy, the improvement of 0.02 in the Recall was one third of the potential improvement (0.06). For the *Procedure* hierarchy, the improvement of 0.005 in the Recall was one quarter of the potential improvement (0.02). When the Recall is already high, it is harder to achieve an improvement. The relative improvement (one third vs. one quarter) is similar.

To address the issue of imbalanced training data, we repeated our experiments with mixing the use of downsampling for the majority class and upsampling for the minority class. In addition, we tested the impact of using equal sizes of training data (ratio of 1:1) versus using different sizes (2:3). In these experiments, we observed the same phenomenon that the recall for the model with summarization data is improved versus the recall for the model without summarization data.

**Future work:** The essence of transfer learning is to exploit knowledge gained from a pre-trained model and apply it to solve another problem. Hence, the quality, richness, and bias of the pre-trained model determines its compatibility with a downstream task. In this paper, we fine-tuned the $BERT_{BASE}$ model (due to hardware limitations) for an IS-A relationship classification task. In the future, we will experiment with the $BERT_{LARGE}$ model, which has more parameters than the $BERT_{BASE}$ model and was proven to be more powerful in various NLP benchmark tests. Another possibility to improve performance is to fine-tune a pre-trained model embedded with rich medical knowledge. Hence, we will employ BERT's variants in the biomedical or clinical domains, for instance, BioBert [23], ClinicalBERT [24] and NCBI BlueBERT [25], to deal with the IS-A relationship classification task in a medical ontology. In addition, our proposed method has the potential to be modified for use as a quality assurance tool, e.g., recommending missing IS-A relationships for primitive concepts, or identifying intermediate IS-A relationships to uncover concepts that are too general and should be ancestors rather than parents.

## 6. Conclusions

We demonstrated that the language representation model BERT can be fine-tuned to predict IS-A relationships between new concepts and pre-existing concepts in SNOMED CT. This model can not only identify potential parents of a new concept, but also filter out irrelevant concepts, reducing the number of improper placement choices for a concept. We showed that the trained BERT model achieved an average F1 (F2) score of 0.87 (0.92) in testing with 8,574 concept pairs containing 2005 new concepts in the *Clinical Finding* hierarchy of SNOMED CT. The average F1 (F2) score in testing with 3,908 concept pairs containing 911

new concepts for the *Procedure* hierarchy was 0.821 (0.912). Furthermore, we employed the Area Taxonomy ontology summarization technique to refine the training data, which resulted in a higher Recall. Ontology curators can benefit from this high Recall, since it indicates that the trained model will propose a higher ratio of proper parents for a given concept. Therefore, the proposed method can save curators time and effort that would be needed to search for those parent candidates manually.

## CRediT authorship contribution statement

**Hao Liu:** Conceptualization, Methodology, Software, Writing - original draft. **Yehoshua Perl:** Supervision, Methodology, Writing - review & editing. **James Geller:** Writing - review & editing.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

Research reported in this publication was supported by the National Center for Advancing Translational Sciences (NCATS), a component of the National Institute of Health (NIH) under award number UL1TR003017. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

## References

[1] İ. Pembeci, Using Word Embeddings for Ontology Enrichment, Int. J. Intelligent Syst. Appl. Eng. 4 (3) (2016) 49–56.
[2] E. Alfonseca, S. Manandhar, An unsupervised method for general named entity recognition and automated concept discovery. Proceedings of the 1st international conference on general WordNet, 2002.
[3] Maedche A, Staab S. Mining ontologies from text. International conference on knowledge engineering and knowledge management; 2000: Springer.
[4] A.M. Jimenez, M.J. Lawley, Snorocket 2.0: Concrete Domains and Concurrent Classification, OWL Reasoner Evaluation Workshop (ORE) (2013).
[5] R. Shearer, B. Motik, H.I. HermiT, A Highly-Efficient OWL Reasoner, Owled (2008).
[6] SNOMED CT. 11/17/2019]. Available from: https://www.snomed.org/.
[7] H. Liu, J. Geller, M. Halper, Y. Perl, Using Convolutional Neural Networks to Support Insertion of New Concepts into SNOMED CT, Proc AMIA Symp. 750 (2018).
[8] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Info. Process. Syst. (2012).
[9] L. Zheng, H. Liu, Y. Perl, J. Geller, Training a Convolutional Neural Network with Terminology Summarization Data Improves SNOMED CT Enrichment, Proc. AMIA Symp. (2019).
[10] H. Min, Y. Perl, Y. Chen, M. Halper, J. Geller, Y. Wang, Auditing as part of the terminology design life cycle, J. Am. Med. Inform. Assoc. 13 (6) (2006) 676–690.
[11] M. Halper, H. Gu, Y. Perl, C. Ochs, Abstraction Networks for Terminologies: Supporting Management of "Big Knowledge", Artif. Intell. Med. 64 (1) (2015) 1–16.
[12] Y. Wang, M. Halper, H. Min, Y. Perl, Y. Chen, K.A. Spackman, Structural methodologies for auditing SNOMED, J. Biomed. Inform. 40 (5) (2007) 561–581.
[13] Devlin J, Chang M-W, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:181004805. 2018.
[14] Sang EF, De Meulder F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. arXiv preprint cs/0306050. 2003.
[15] Rajpurkar P, Zhang J, Lopyrev K, Liang P. Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:160605250. 2016.
[16] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A. Ng, et al., Recursive deep models for semantic compositionality over a sentiment treebank. Proceedings of the 2013 conference on empirical methods in natural language processing, 2013.
[17] H. Liu, Y. Perl, J. Geller, Transfer Learning from BERT to Support Insertion of New Concepts into SNOMED CT, Proc. AMIA Symp. (2019).
[18] G. Elhanan, Y. Perl, J. Geller, A survey of SNOMED CT direct users, 2010: impressions and preferences regarding content and quality, J. Am. Med. Inform. Assoc. 18 (2011) i36–i44 (Supplement_1).
[19] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Adv. Neural Info. Process. Syst. (2013).
[20] J. Pennington, R. Socher, Manning C. Glove, Global vectors for word representation. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014.
[21] Joulin A, Grave E, Bojanowski P, Douze M, Jégou H, Mikolov T. Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:161203651. 2016.
[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, et al., Attention is all you need, Adv. Neural Info. Process. Syst. (2017) 5998–6008.
[23] Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, et al. Biobert: pre-trained biomedical language representation model for biomedical text mining. arXiv preprint arXiv: 190108746. 2019.
[24] Huang K, Altosaar J, Ranganath R. ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission. arXiv preprint arXiv:190405342. 2019.
[25] Peng Y, Yan S, Lu Z. Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets. arXiv preprint arXiv:190605474. 2019.
[26] Elhanan G, Perl Y, Geller J. A survey of direct users and uses of SNOMED CT: 2010 status. AMIA Annual Symposium Proceedings; 2010: American Medical Informatics Association.
[27] L. Cui, W. Zhu, S. Tao, J.T. Case, O. Bodenreider, G.-Q. Zhang, Mining non-lattice subgraphs for detecting missing hierarchical relations and concepts in SNOMED CT, J. Am. Med. Inform. Assoc. 24 (4) (2017) 788–798.
[28] M. Halper, Y. Wang, H. Min, Y. Chen, G. Hripcsak, Y. Perl, et al., Analysis of error concentrations in SNOMED, AMIA Annu. Symp. Proc. 314–8 (2007).
[29] C. Ochs, J. Geller, Y. Perl, Y. Chen, J. Xu, H. Min, et al., Scalable Quality Assurance for Large SNOMED CT Hierarchies Using Subject-based Subtaxonomies, J. Am. Med. Inform. Assoc. 22 (3) (2014) 507–518.
[30] Y. Wang, M. Halper, D. Wei, H. Gu, Y. Perl, J. Xu, et al., Auditing complex concepts of SNOMED using a refined hierarchical abstraction network, J. Biomed. Inform. 45 (1) (2012) 1–14.
[31] Y. Wang, M. Halper, D. Wei, Y. Perl, J. Geller, Abstraction of complex concepts with a refined partial-area taxonomy of SNOMED, J. Biomed. Inform. 45 (1) (2012) 15–29.
[32] A. Agrawal, Z. He, Y. Perl, D. Wei, M. Halper, G. Elhanan, et al., The readiness of SNOMED problem list concepts for meaningful use of electronic health records, Artif. Intell. Med. 58 (2) (2013) 73–80.
[33] C. Wang, X. He, A. Zhou, A short survey on taxonomy learning from text corpora: Issues, resources and recent advances. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017.
[34] Nguyen KA, Köper M, Walde SSi, Vu NT. Hierarchical embeddings for hypernymy detection and directionality. arXiv preprint arXiv:170707273. 2017.
[35] Ivan Sanchez Carmona V, Riedel S. How well can we predict hypernyms from word embeddings? a dataset-centric analysis. 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017-Proceedings of Conference; 2017: Association for Computational Linguistics.
[36] C. Wang, X. He, A. Zhou, Improving Hypernymy Prediction via Taxonomy Enhanced Adversarial Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 2019.
[37] C. Wang, Y. Fan, X. He, A. Zhou, Predicting hypernym–hyponym relations for Chinese taxonomy learning, Knowl. Inf. Syst. 58 (3) (2019) 585–610.
[38] C. Wang, Y. Fan, X. He, A. Zhou, A family of fuzzy orthogonal projection models for monolingual and cross-lingual hypernymy prediction, The World Wide Web Conference (2019).
[39] Wang C, Yan J, Zhou A, He X. Transductive non-linear learning for chinese hypernym prediction. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); 2017.
[40] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, et al., TensorFlow: A System for Large-Scale Machine Learning, OSDI (2016).
[41] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, et al., Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. Proceedings of the IEEE international conference on computer vision, 2015.
[42] C.P. Morrey, J. Geller, M. Halper, Y. Perl, The Neighborhood Auditing Tool: a hybrid interface for auditing the UMLS, J. Biomed. Inform. 42 (3) (2009) 468–489.
[43] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. arXiv preprint arXiv:150807909. 2015.
[44] N.V. Chawla, Data mining for imbalanced datasets: An overview, Springer, Data mining and knowledge discovery handbook, 2009, pp. 875–886.
[45] Wang S, Liu W, Wu J, Cao L, Meng Q, Kennedy PJ. Training deep neural networks on imbalanced data sets. 2016 international joint conference on neural networks (IJCNN); 2016: IEEE.
[46] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (4) (2009) 427–437.
[47] Eisner J. In what NLP (Natural Language Processing) applications is recall more important than precision? 2014 02/14/2020]. Available from: https://www.quora.com/In-what-NLP-Natural-Language-Processing-applications-is-recall-more-important-than-precision.