Incorporating negative evidence into disease diagnosis: What you know can't hurt you

Tal Friedman
talf301@gmail.com

Erin Grant
erin.grant@mail.utoronto.ca

1 Introduction

Computer-Assisted Diagnosis (CAD) is a useful and desirable tool that has been challenging the AI community for many years. For example, Jaakkola et al. [4] introduced the QMR-DT network, a noisy-OR probabilistic graphical model, constructed from expert knowledge. In particular, CAD is useful in the realm of rare genetic diseases, since clinical geneticists will often be tasked with diagnosing patients that have disorders they may only see a couple of times in their careers. In this regard, having a tool which can make reasonable suggestions for diagnoses a clinician may have never seen before, is invaluable.

In order for a CAD system to be feasible, one requires a standardized dictionary for symptoms and diseases, as well as either a very large training set of diagnosed patients, or some expert information relating symptoms and diseases. While the first option is not possible due to the rarity of the diseases being investigated, the Human Phenotype Ontology (HPO) [5] and Online Mendelian Inheritance in Man (OMIM) [6] together provide the standardized dictionary, as well as the necessary expert information. Specifically, the HPO contains information, for each disease, about empirical frequencies with which related symptoms appear, as well as symptoms that will never appear; we call the latter *negative annotations*.

These negative annotations are conceptually difficult to incorporate into a model, and because of the minor role they play in diagnosis, will typically be ignored in CAD systems. In this work, we attempt to develop a model that makes use of these negative annotations in order to improve CAD.

2 Previous Work

Bauer et al. [1] introduced a disease diagnosis system that uses Bayesian inference to resolve diagnostic queries, entitled the *Bayesian Ontology Query Algorithm* (BOQA). The system comprises a Bayesian network with three layers of Boolean variables: an *item* layer of diseases, a *hidden* layer of phenotypic features, and a *query* layer of phenotypic features. [1] The causal relationships between a disease and its related symptoms are modelled as a set of directed edges from each disease node in the *item* layer to its subset of phenotypic nodes in the *hidden* layer. We may refer to the symptoms caused by a disease as the disease's *phenotype annotations*. Each phenotype node in the hidden layer is paired with exactly one node in the query layer, by a directed edge from the node in the hidden layer to its correspondent in the query layer.

We may interpret this model in a generative fashion as follows: the occurrence of a disease (activity in an item node) causes its symptoms to occur with some probability (activity in a hidden node). Furthermore, the presence of a symptom (activity in a hidden node) may lead to the clinician detecting the symptom (activity in a query node). However, the clinician may fail to recognize the occurrence of a symptom (activity in a hidden node without activity in the corresponding query node), or the clinician could perceive a symptom where none exists (activity in a query node without

[1] In this report, we use the term *phenotype* interchangeably with symptom, since all symptoms of interest to the model are expected to have their root in genetic disorders.

activity in the corresponding hidden node). In this way, the hidden layer allows the system to model uncertainty about the accuracy of a clinician's symptomatic description.

In addition to the edges between layers, BOQA contains within-layer connections to model the ontology of phenotypes described by the Human Phenotype Ontology (HPO) [5]. Within the hidden layer, there is a directed edge from node H_a to H_b if the annotation of the phenotype H_a to a disease implies that the phenotype H_b is also annotated to the disease. Such an edge is present in the network if the phenotype H_a is a child of phenotype H_b in the HPO. These edges serve the purpose of encoding the *annotation propagation rule*: if an item i is annotated to term j then it is implicitly annotated to all ancestors of j.

Lastly, in the case that the edge from H_a to H_b is present, then in the query layer, there is a directed edge from phenotype Q_b to phenotype Q_a (i.e., an edge in the inverse direction of the edge $H_a \rightarrow H_b$). These edges serve to encode the fact that a if a phenotype Q_a is in the search query, then so are all of its ancestors.

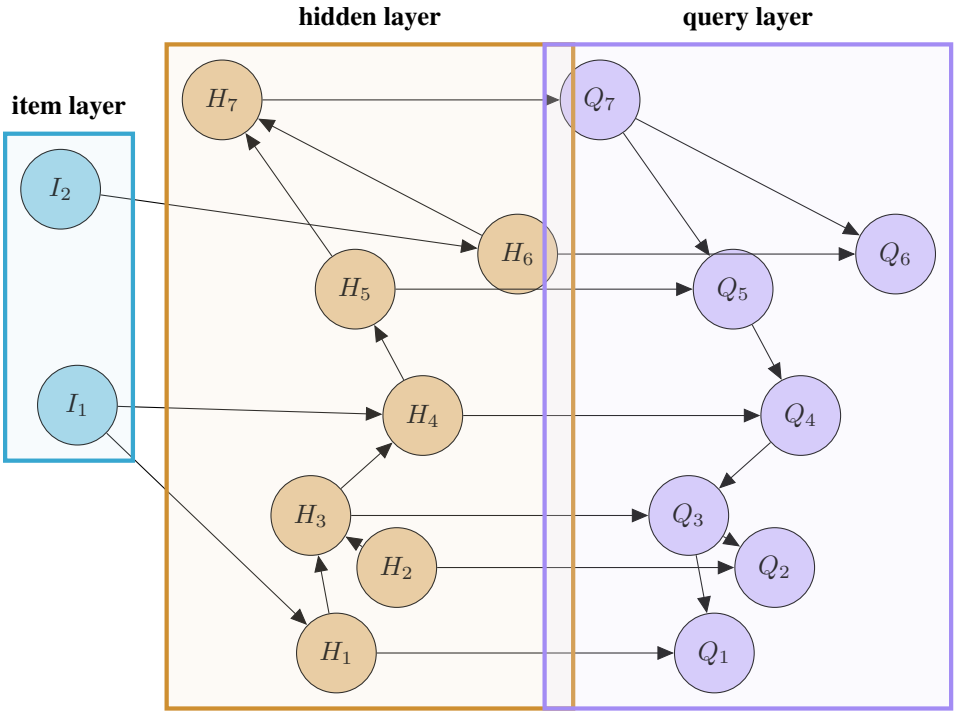


Figure 1: BOQA network structure.[2]

2.1 Network construction[3]

Let M represent the number of terms in the ontology, and let N represent the number of diseases. Let $\{I_i\}_{i=1}^{N}, \{H_j\}_{j=1}^{M}$ and $\{Q_k\}_{k=1}^{M}$ represent the nodes in the *item*, *hidden* and *query* layers, respectively.

Indices of phenotypic nodes in this network correspond to indices of terms in the HPO; i.e., H_i and Q_i together correspond to the ith node in the ontology. We identify parent-child relations from the ontology in the following manner: let $\text{pa}(i) = \{\text{pa}(i)_1, \ldots, \text{pa}(i)_J\}$ denote the J indices of the

[2]Reproduced from Sebastian Bauer, Sebastian Köhler, Marcel H. Schulz, and Peter N. Robinson. Bayesian ontology querying for accurate and noise-tolerant semantic searches. *Bioinformatics*, 28(19):2502–2508, 2012.

[3]Equations in Section 2.1 are reproduced from Sebastian Bauer, Sebastian Köhler, Marcel H. Schulz, and Peter N. Robinson. Bayesian ontology querying for accurate and noise-tolerant semantic searches. *Bioinformatics*, 28(19):2502–2508, 2012.

direct parents of term i in the ontology, and similarly let $\text{chi}(i) = \{\text{chi}(i)_1, \ldots, \text{chi}(i)_K\}$ denote the K indices of the direct children of term i in the ontology.

Furthermore, indices of disease nodes in the network correspond to indices of diseases that are annotated by terms in the HPO. Identify the annotations for a disease as follows: let $\text{ea}(j) = \{\text{ea}(j)_1, \ldots, \text{ea}(j)_L\}$ denote the indices of the L terms for which the jth disease is explicitly annotated.

The local probability distributions for the hidden nodes can then be written as

$$P\left(H_i = 1 \mid I_{\text{ea}(i)}, \bigvee H_{\text{chi}(i)}\right) = \left(1 - \prod_{j=\text{ea}(i)_1}^{\text{ea}(i)_L} (1 - I_j\, f_{ji})\right)^{1-\bigvee H_{\text{chi}(i)}} \tag{1}$$

where $\bigvee$ represents the logical disjunction, and f_{ji} represents the empirical frequency of the occurrence of phenotype i with disease j. This formulation captures the annotation propagation rule within the hidden layer, since it evaluates to 1 if $\bigvee H_{\text{chi}(i)} = 1$; that is, if any child annotations are active. As well, it captures that the hidden node is inactive with probability 1 if all diseases are inactive (i.e., $I_i = 0, \forall i$), and otherwise the probability of activity in the hidden state is a function of the empirical association of disease and symptom.

The local probability distribution for the query nodes, given the states of H_i and $Q_{\text{pa}(i)}$, can be written as

$$\begin{aligned}
&P\left(Q_i = 0 \mid H_i = 1, \bigwedge Q_{\text{pa}(i)} = 0\right) = 1 && P\left(Q_i = 0 \mid H_i = 0, \bigwedge Q_{\text{pa}(i)} = 0\right) = 1 \\
&P\left(Q_i = 1 \mid H_i = 1, \bigwedge Q_{\text{pa}(i)} = 0\right) = 0 && P\left(Q_i = 1 \mid H_i = 0, \bigwedge Q_{\text{pa}(i)} = 0\right) = 0 \\
&P\left(Q_i = 0 \mid H_i = 1, \bigwedge Q_{\text{pa}(i)} = 1\right) = \beta && P\left(Q_i = 0 \mid H_i = 0, \bigwedge Q_{\text{pa}(i)} = 1\right) = 1 - \alpha \\
&P\left(Q_i = 1 \mid H_i = 1, \bigwedge Q_{\text{pa}(i)} = 1\right) = 1 - \beta && P\left(Q_i = 1 \mid H_i = 0, \bigwedge Q_{\text{pa}(i)} = 1\right) = \alpha
\end{aligned}$$

where $\bigwedge$ represents the logical conjunction, β represents the probability of a false negative (i.e., $H_i = 1$ but $Q_i = 0$), and α represents the probability of a false positive (i.e., $Q_i = 1$ but $H_i = 0$).

Now, defining a variable $m_{xyz|QH}$ by

$$m_{xyz|QH} = \left|\left\{k \mid (Q_k = x) \wedge (H_k = y) \wedge \left(\bigwedge Q_{\text{pa}(k)} = z\right)\right\}\right|,$$

the joint probability of the query nodes Q_i may be written as

$$\prod_{i=1}^{M} P\left(Q_i \mid H_i, \bigwedge Q_{\text{pa}(i)}\right) = \beta^{\,m_{011|QH}}\ (1-\beta)^{m_{111|QH}}\ \alpha^{m_{001|QH}}\ (1-\alpha)^{m_{101|QH}},$$

under the assumption that the invalid configurations $m_{110|QH}$ and $m_{100|QH}$ occur with probability zero, and using the simplification that the configurations $m_{010|QH}$ and $m_{000|QH}$ have probability one.

The joint probability distribution over the network is then realized as

$$\begin{aligned}
&P(I_1, \ldots, I_N, H_1, \ldots, H_M, Q_1, \ldots, Q_M) \\
&= P(I_1, \ldots, I_N) \prod_{i=1}^{M} P\left(H_i \mid \bigvee I_{\text{ea}(i)}, \bigvee H_{\text{chi}(i)}\right)\ P\left(Q_i \mid H_i, \bigwedge Q_{\text{pa}(i)}\right).
\end{aligned} \tag{2}$$

2.2 Inference

The marginal probability of a configuration of the disease nodes $I_1, \ldots, I_N$ in the item layer, given some phenotypic evidence $Q_1, \ldots, Q_M$, is given by marginalizing over $\vec{H}$ as

$$\begin{aligned} P(I_1, \ldots, I_N \mid Q_1, \ldots, Q_M) &= \frac{\sum_{\vec{H}} P(I_1, \ldots, I_N, H_1, \ldots, H_M, Q_1, \ldots, Q_M)}{P(Q_1, \ldots, Q_M)} \\ &= \frac{\sum_{\vec{H} \in \{0,1\}^M} P(I_1, \ldots, I_N, H_1, \ldots, H_M, Q_1, \ldots, Q_M)}{P(Q_1, \ldots, Q_M)} \end{aligned}$$

The goal of this model is to find the configuration of diseases with highest posterior probability, given some phenotypic evidence. This is a MAP inference problem, and so may be solved by maximizing the product of the likelihood and the prior; i.e., the solution is given by

$$\begin{aligned} &\operatorname*{argmax}_{(I_1,\ldots,I_N)} P(I_1, \ldots, I_N \mid Q_1, \ldots, Q_M) \\ &= \operatorname*{argmax}_{(I_1,\ldots,I_N)} \frac{P(Q_1, \ldots, Q_M \mid I_1, \ldots, I_N)\, P(I_1, \ldots, I_N)}{P(Q_1, \ldots, Q_M)} \\ &= \operatorname*{argmax}_{(I_1,\ldots,I_N)} P(Q_1, \ldots, Q_M \mid I_1, \ldots, I_N)\, P(I_1, \ldots, I_N) \\ &= \operatorname*{argmax}_{(I_1,\ldots,I_N)} P(I_1, \ldots, I_N) \sum_{\vec{H} \in \{0,1\}^M} \prod_{i=1}^{M} P\left(H_i \mid \bigvee I_{\text{ea}(i)}, \bigvee H_{\text{chi}(i)}\right)\, P\left(Q_i \mid H_i, \bigwedge Q_{\text{pa}(i)}\right). \end{aligned} \tag{3}$$

2.3 Complexity restrictions

The computation in Eqn. (3) is intractable, since there are 2^N configurations of the item nodes over which to maximize the posterior, and there are 2^M configurations of the hidden nodes $H_1, \ldots, H_M$ over which to marginalize. Bauer et al. make two simplifications to make inference tractable; we describe them in Section 2.3.1 and Section 2.3.2.

2.3.1 One-disease constraint

To avoid maximizing over the 2^N configurations of the item nodes, Bauer et al. impose the constraint

$$\sum_{i=1}^{N} I_i = 1. \tag{4}$$

Eqn. (4) enforces the restriction that only one disease node may be active given any query, and so the maximum is taken over N one-hot configurations, in order to compute the MAP estimate. This is equivalent to assuming that a patient can have only a single disease; since this is a diagnosis system for rare genetic diseases, this is a reasonable assumption. Therefore, we adopt the one-disease constraint into all models that we describe in Section 4.

2.3.2 k-least frequency annotations constraint

To avoid marginalizing over the 2^M configurations of the hidden nodes, Bauer et al. simplify the local probability distribution of each hidden node that is given by Eqn. (1). In particular, they restrict the number of frequency annotations f_{ji}, for each disease node I_i, that are not exactly zero or exactly one, to the annotations with the k least frequency values. [4] This has the effect of enforcing all other hidden node likelihoods to be either active or inactive with probability one, conditional on a disease node. Therefore, to compute the sum over in $\vec{H} \in \{0,1\}^M$ in Eqn. (3), the model need only take into account the subset of summands representing configurations in which the likelihood of all

[4] Bauer et al. use $k = 10$ for their final experimental results.

hidden nodes with deterministic activity, is one, since the remaining terms are zero. This makes the marginalization computationally tractable. In Section 4, we describe models that either adopt or relax this constraint.

3 Objectives

The main contribution of this report is to incorporate the negative phenotype annotations given by OMIM [3] into Bauer et al.'s [1] BOQA network. By doing this, we wish to improve classification accuracy for diseases that have negative annotations, while not diminishing performance for cases in which the disease is only positively annotated. As well, we require that the complexity we add to the model does not cause the runtime of the algorithm to become intractable. Also, since the structure of the network is such that the local conditional probability distributions are directly interpretable, we can make interpretable requirements about the posterior probabilities of each disease. The following paragraphs summaries the requirements.

Consider two diseases, I_1 and I_2, that have identical phenotype annotations, except a single symptom Q_1 that is negatively annotated to I_1. In order for the system to make the best diagnosis, if a patient exhibits all symptoms annotated to both diseases, yet does not exhibit phenotype Q_1, we require that the system assign higher posterior probability to I_1. Furthermore, we need the system to assign higher posterior probability to I_2 if the patient displays phenotype Q_1.

Now, consider a single disease I_1 and two patients who share the same symptoms except for a phenotype, Q_1, for which I_1 is negatively annotated. We require that the system assign higher posterior probability to the patient who does not exhibit phenotype Q_1.

Formally: Let $(Q_1, ..., Q_M)$ be query variables. Furthermore, let I_1 be a disease that is explicitly negatively annotated to Q_1, and let I_2 be a disease that is explicitly positively annotated to Q_1.

Then, we require that the following relationships hold:

$$P(I_1 \mid Q_1 = 0, \ldots, Q_M) > P(I_2 \mid Q_1 = 0, \ldots, Q_M);$$

$$P(I_1 \mid Q_1 = 0, \ldots, Q_M) > P(I_1 \mid Q_1 = 1, \ldots, Q_M).$$

In Section 4, we formulate our model so as to satisfy these requirements.

4 Modifications

In this section we describe the various modifications to the network structure and inference procedure made in order to incorporate the desired functionality. [5]

4.1 Modification to the network structure

Recall that in the BOQA network, the probability of activity in a hidden node is realized as

$$P\left(H_i = 1 \mid I_{\text{ea}(i)}, \bigvee H_{\text{chi}(i)}\right) = \left(1 - \prod_{j=\text{ea}(i)_1}^{\text{ea}(i)_L} (1 - I_j\, f_{ji})\right)^{1-\bigvee H_{\text{chi}(i)}} \tag{5}$$

where f_{ji} represents the empirical frequency of the occurrence of phenotype i with disease j. Under this formulation, if disease i is not annotated to phenotype j, and none of the children of j are either (i.e., $\bigvee H_{\text{chi}(i)} = 0$), then the probability of activity in the hidden layer is simply zero. In other words, the likelihood of observing symptom j, given that the patient has disease i, is zero.

[5] Our code can be found at `https://github.com/talf301/boqa-negative`. We reimplemented the BOQA baseline codebase in Python, and manually coded all modifications described in this section.

However, since the symptom is not negatively annotated to the disease, we would like the model instead to assign some likelihood to the occurrence of this event, encoding the expectation that the symptom may occur together with the disease by chance. In particular, a likelihood of zero should be assigned only to those symptoms that are negatively annotated to a disease. We modify the local probability distribution in (5) to capture this specification as described in the following paragraphs.

Let $\text{pos}(i) = \{\text{pos}(i)_1, \ldots, \text{pos}(i)_S\}$, for each H_i, index the S diseases for which H_i is explicitly positively annotated and let $\text{neg}(i) = \{\text{neg}(i)_1, \ldots, \text{neg}(i)_T\}$, the T diseases for which H_i is explicitly negatively annotated.

Then if we let the probability of activity in a hidden node H_j be given by

$$\begin{aligned} &P\left(H_i = 1 \mid I_{\text{pos}(i)_1}, \ldots, I_{\text{pos}(i)_S}, \bigvee I_{\text{neg}(i)}, \bigwedge H_{\text{chi}(i)}\right) \\ &= \left(1 - \left(\prod_{j=\text{pos}(i)_1}^{\text{pos}(i)_S} (1 - I_j\, f_{ji})\right)^{1-\bigvee I_{\text{neg}(i)}}\right)^{1-\bigvee H_{\text{chi}(i)}} \end{aligned} \tag{6}$$

where if the empirical frequency is not available but the symptom j is not negatively annotated to disease i, then frequency f_{ij} is set to some small value, p. [6]

However, the result of this modification to the network is that exact inference is now intractable, since marginalizing over all binary assignments to the hidden nodes is exponential in the number of phenotypes, as noted in Section 2.3.2. Furthermore, we may not apply the k-least frequencies restriction described in Section 2.3.2. to simplify the marginalization, since that would force some hidden nodes to deterministically be inactive, conditional on activity in some item node, even though the corresponding phenotypes may not be negatively annotated to the active disease. Therefore, we consider several methods to approximate computation of the marginals, which we describe in the next sections.

4.2 Modifications to the inference procedure

We test two different methods of sampling to approximate inference in the network.

4.2.1 p-sampling

As described for Eqn. (6), we assume that for each hidden node without an annotation (positive or negative), the frequency has a fixed value of p. More details are given in Section 5.2.4.

4.2.2 Information-content sensitive p-sampling

Here, rather than using the value p directly, we weight p by the inverse information content of the phenotype associated with the hidden node. Eqn. (7) shows how we compute the information content for each phenotype.

$$-\log_2 \frac{\text{\# of times phenotype or its descendants is annotated to a disease}}{N}, \tag{7}$$

where N is the number of known diseases. More details are given in Section 5.2.5.

5 Experiments

5.1 Data & data generation

The HPO [5] is a directed acyclic graph that depicts an ontology of 10 476 phenotypes. It specifies annotations for 6575 diseases, 471 of which possess negative annotations. In addition to using the ontology as expert information in the model construction and inference procedures, we exploit it to

[6] It should be noted that activity of a child annotation of a node H_j entails activity in H_j under the formulation in (6), even if phenotype j is negatively annotated to the disease of interest. However, this situation does not occur in the HPO, and so we need not treat special cases.

generate patients for testing. [7] We spawned 100 patients infected with diseases that possess positive annotations but not necessarily negative annotations, and then generated a further 100 patients each infected with a negatively-annotated disease.

We generated and infected patients in the following manner:

1. For a given disease, we sampled each of the disease's annotated symptoms with probability equal to the empirical association of symptom and disease, as specified by the HPO annotations.
2. To simulate noise in the patient queries, we added a number of unrelated symptoms to the patient query by sampling with uniform probability over all symptoms. [8]
3. To simulate imprecision in the patient queries, for each symptom generated in Steps 1 and 2, with some probability we replaced the symptom with one of its ancestors in the HPO. [9]

We will refer to the data generated in this way as the *artificial patient data*.

In addition to this artificially generated data, we tested all models on 101 patient queries obtained from real-life clinician-patient encounters, which we refer to as the *naturalistic patient data*. [10]

5.2 Models to test

In order to test our modifications as outlined in Section 4.1, we performed experiments on various models which differ in their structure and inference methods. Section 5.2.1 and Section 5.2.2 describe our benchmark, and Section 5.2.3, Section 5.2.4 and Section 5.2.5 describe the models introduced by this report.

5.2.1 No frequency annotations model

This model is a simplification of the model of Bauer et al. [1]; specifically, all stochastic state propagations are made deterministic; in other words, it is assumed that the empirical frequency in Eqn. (1) is either zero or one. Exact inference is performed. We use this model to benchmark our results.

5.2.2 k-least frequency annotations model

This model is identical to the final model of Bauer et al. [1]. Specifically, the number of non-deterministic activity state propagations from the item layer to the hidden layer is capped at k for each item node, and exact inference is performed. We use this model as well to benchmark our results.

5.2.3 Sampling model

We take the model of Bauer et al. [1] with all frequency annotations; however, we do not perform exact inference. Instead, for each item node, we independently sample each edge that has frequency information, with probability equal to the frequency, creating n[11] instances of the network. We then compute the item node marginals for each of these models, and take the average as the value of the posterior probability of each disease.

This process of sampling edges based on frequencies is equivalent to fixing the stochastic activity of the hidden layer based on frequency information; therefore we are sampling over models that

[7] Rare genetic diseases is, true to its name, a domain in which there are few examples, so we have to generate more test data to fill the void.

[8] For all experimental results, we chose the number of noise symptoms to be half the number of symptoms generated in Step 1.

[9] Each symptom is replaced by another symptom chosen uniformly from a set containing all ancestors of the symptom and the symptom itself. In this way, there is some small probability that the symptom is retained, and thus the query is made no less precise.

[10] We take the naturalistic patients data from the PhenoTips patients repository [2]. The data is not publicly available.

[11] For p-sampling and information-content sensitive p-sampling, we chose $n = 1000$.

have a fixed activity configuration in the hidden layer. In short, this is Bauer et al.'s model without the constraint in Section 2.3.2, and with approximated inference for the marginals, accomplished by sampling.

5.2.4 p-sampling model with negative annotations

This model is the BOQA model, with the modified hidden layer local probability distribution described in Eqn. (6), and the specification for the sampling procedure described in Section 4.2.1, with the sampling itself being done as described in Section 5.2.3. It therefore utilizes negative annotations.

5.2.5 Information-content sensitive p-sampling model with negative annotations

Again, this is the BOQA model with the modified hidden layer local probability distribution described in Eqn. (6). The specification for the sampling procedure used is that described in Section 4.2.2, with the sampling itself being done as described in Section 5.2.3. It also therefore utilizes negative annotations.

6 Results

6.1 Discriminative threshold metrics of evaluation

Section 6.1.1 and Section 6.1.2 discuss methods that Bauer et al. [1] use to evaluate their model; these metrics are related to counting true- / false-positive / negative cases and varying a discriminative threshold.

6.1.1 ROC curve

A receiver operating characteristic (ROC) curve requires a variable *threshold value* that specifies the true- / false-positive / negative counts. The threshold value that we use is rank. More specifically, all test items at or above rank r are treated as positives, and the rest as negatives, where r is the threshold variable. Therefore, the $(0, 0)$ classifier is the classifier with a threshold of rank zero (i.e., no ranked items are positives). Similarly, the $(1, 1)$ classifier is the classifier with a threshold rank of the number of test examples (i.e., all ranked items are positives).

We plot ROC curves for all models, evaluated on the artificial patient data, aggregating all diseases (Figure 2a), on the artificial patient data, for only the negatively annotated diseases (Figure 2b), and for the naturalistic patient data, aggregating all diseases (Figure 2c).

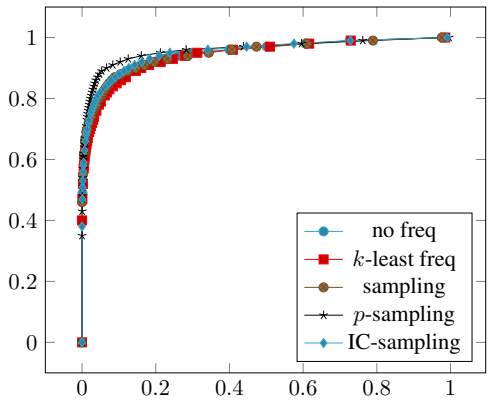
From the results in Figure 2, we see that all models performed almost equally well according to the ROC metric, for both naturalistic and artificial data, and for general diseases as well as negatively annotated diseases in particular. However, while we see that the p-sampling and IC-sensitive p-sampling models perform better than the other models on the pooled diseases (Figure 2a), they perform worse on the negatively-annotated diseases (Figure 2b). It is difficult to tell apart performance on the naturalistic data (Figure 2c).

6.1.2 Precision / recall curve

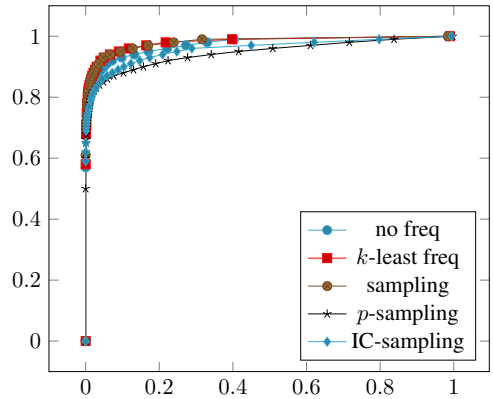
We plot precision-recall (PR) curves for all models, evaluated on the artificial patient data, aggregating all diseases (Figure 3a), on the artificial patient data, for only the negatively annotated diseases (Figure 3b), and for the naturalistic patient data, aggregating all diseases (Figure 3c). The PR curves also require definition of a threshold value r; we defined it as described in Section 6.1.1.

The trend is clear for artificial patients from the results in Figure 3a and Figure 3b: the IC-sensitive p-sampling model performs worst, with the p-sampling model performing next best; both are worse than the benchmarks. However, we see that the sampling model is consistently better than the benchmarks.

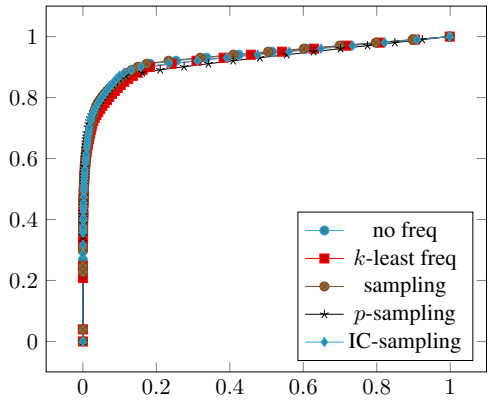
Once again, the performance of different models on the naturalistic patient data is hard to distinguish in Figure 3c.



(a) Artificial patient data; general disease.



(b) Artificial patient data; negatively-annotated disease.



(c) Naturalistic patient data; general disease.

Figure 2: Receiver operating characteristic (ROC) curves.

6.2 Other metrics of evaluation

Section 6.1.1 and Section 6.1.2 describe metrics that are sensitive to the false positive count. However, the way that we defined our threshold variable (rank) means that there are a large number of false positives. As well, our model is not a binary classifier in the strict sense, which ROC and PR curve evaluation metrics expect; instead it is a ranking system. Therefore, it seems necessary to evaluate our model by other metrics. Section 6.2.1 and Section 6.2.2 discuss methods that rely on things other than the true- / false-positive / negative counts for evaluation.

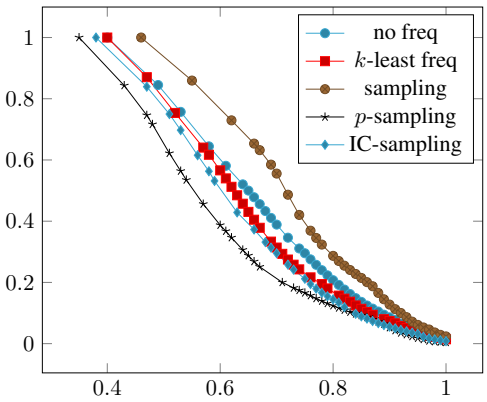
6.2.1 Mean reciprocal rank

Define the *mean reciprocal rank* metric by the following computation:

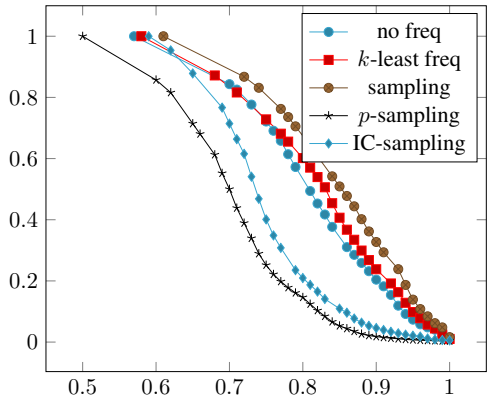
$$\text{mean reciprocal rank} = \frac{1}{|S|} \sum_{k=1}^{|S|} \frac{1}{\text{rank}_k},$$

where S is the set of test cases, and rank_k is the rank of the gold-standard disease in the kth test case. Then a large value of the mean reciprocal rank is indicative of good performance.

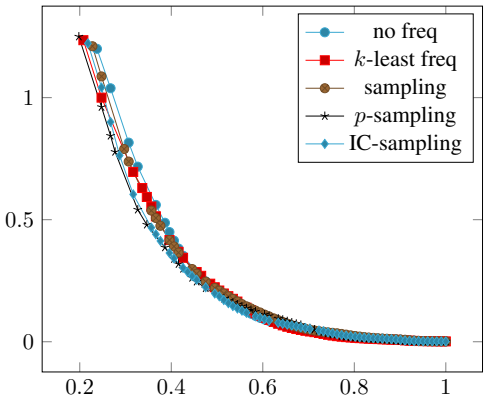
From the results in Figure 4, we see an interesting pattern: the models that perform relatively badly on the artificial data tend to perform well on the naturalistic data, according to this metric. For example, the IC-sampling model performs worst on the artificial data, but performs best on the naturalistic data. The exception is the no-frequency model, which consistently performs badly, as to be expected because it is too simplistic.



(a) Artificial patient data; general disease.



(b) Artificial patient data; negatively-annotated disease.



(c) Naturalistic patient data; general disease.

Figure 3: Precision-recall (PR) curves.

6.2.2 Binned rank counts

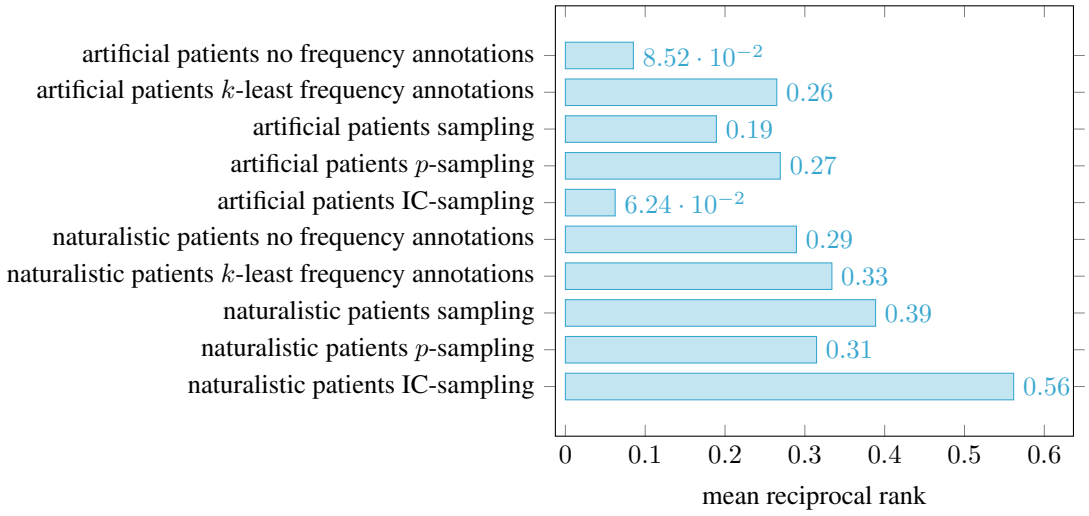
To construct the binned rank plots, we count the number of test cases for which the gold-standard disease falls into the ranking bin of interest.

The results in Figure 5b and Figure 5a show a similar trend to those described in Section 6.2.1: models that tend to do well on the artificial data, do worse on the naturalistic data and vice versa. In particular, the IC-sampling model does best by a large margin on the naturalistic data, but has very low performance on the artificial data. The exception to the analogy to Section 6.2.1 is that the sampling method performs well in both artificial and naturalistic data scenarios.

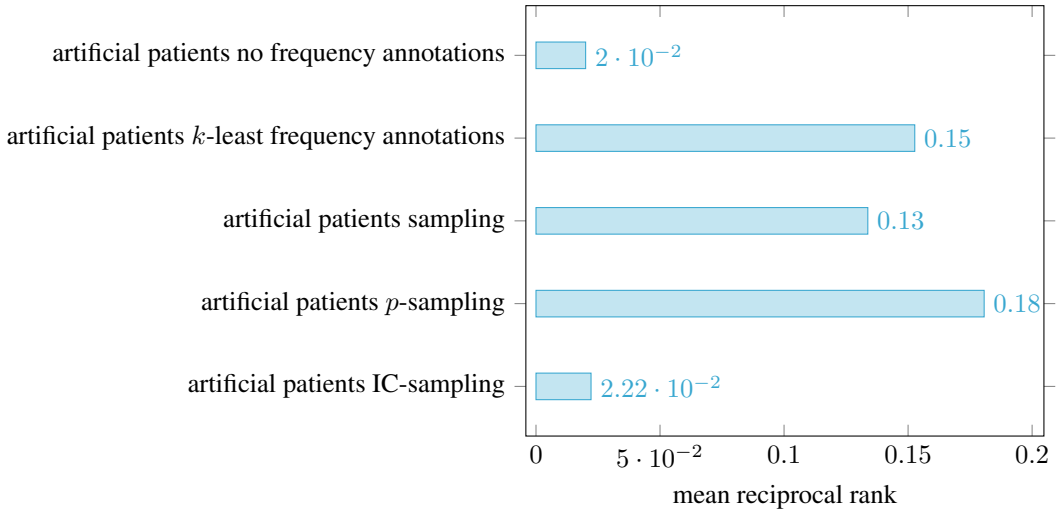
7 Discussion & conclusion

We have made three modifications to the structure of the BOQA model and how inference is done with it: sampling, p-sampling, and Information Content (IC) sampling. The results show, through most experiments and metrics, that sampling provides an improvement over the current k-least frequency annotation method. Additionally, it appears that both p-sampling and IC-sampling perform rather poorly on artificial patients, but reasonably well on naturalistic patients, especially IC-sampling.

Comparing p-sampling and IC-sampling to other methods is tricky, as it must be noted that the former two both require the hyperparameter p to be tuned, which we did not have the computational resources to do. In contrast, sampling and k-least frequency annotation methods have no such hyperparameter, and thus cannot be fine-tuned any further. While this may leave room for improvement,



(a) All diseases pooled.



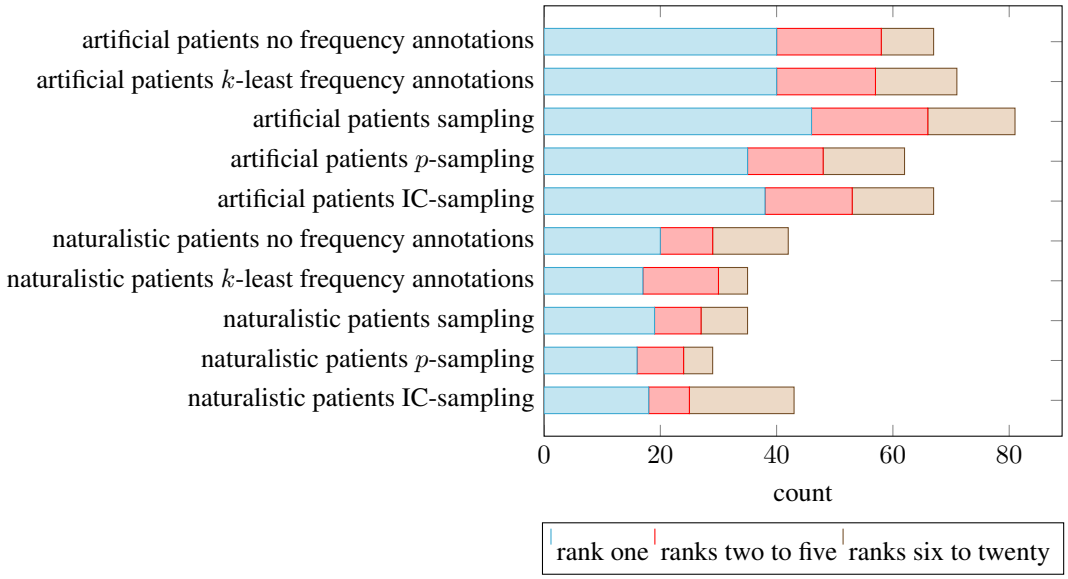
(b) Only diseases with negative annotations.

Figure 4: Mean reciprocal rank plots.

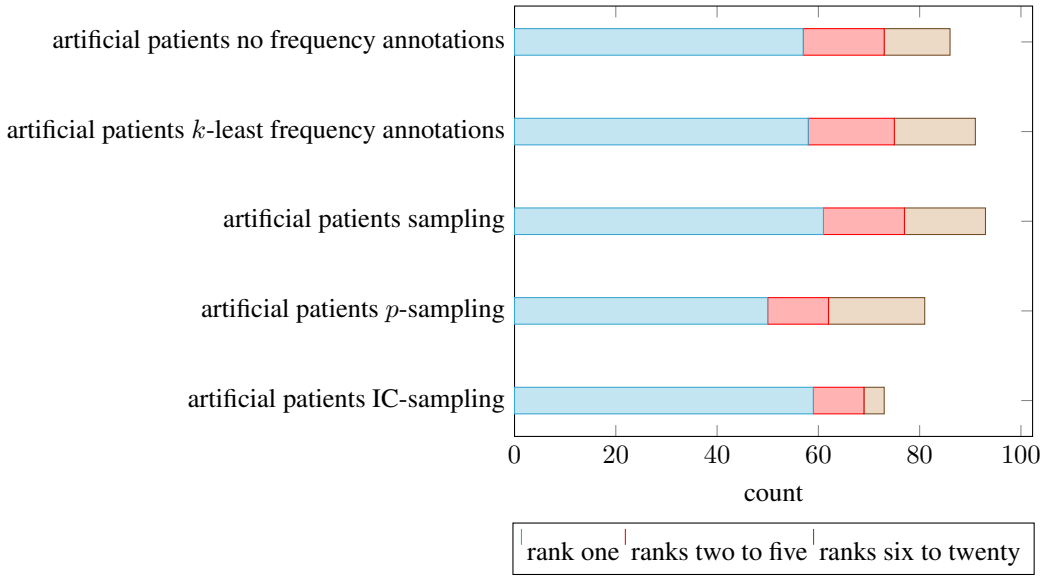
there is no guarantee that fine-tuning p would provide a large increase in performance to the models. In this case of this domain problem, the size of the ontology coupled with the possibility of any number of phenotypes being annotated to a disease may mean that p-sampling, and hence IC-sampling, introduces too much noise into the model to be an effective method.

As for our data, it is difficult to judge whether artificial or naturalistic results should be weighed more heavily, as they often appear to be very different. By nature, artificial patients are guaranteed to contain at least some phenotypic information that is relevant to the true diagnosis. This is not necessarily true for naturalistic data, as the clinician seeing the patient may not spend enough time to properly enter information, and also has access to other information about the patient when making the correct diagnosis, meaning that the clinician's phenotypic description may be incomplete.

In addition, the available naturalistic data had only a small proportion of diseases with negative annotations. Therefore, in this report we were unable to investigate whether the change in perfor-



(a) Artificial and naturalistic patients; general disease.



(b) Artificial patients; disease with negative annotations.

Figure 5: Binned ranking plots.

mance among models, between the general diseases and the negatively-annotated diseases, would occur similarly for the naturalistic patient data. This, in conjunction with the fact that the results differed to a great extent between the artificial and naturalistic patient dataset, implies we cannot make a strong prediction either way; therefore, explicit testing of the models on naturalistic patient cases with negatively annotated data is necessary in another study. Furthermore, such data would also enable us to evaluate our test data generation procedure, for if the performance did not differ in a similar manner to the difference in performance for the artificial patient data, then the quality of our artificially generated data would come into question.

In terms of evaluation metrics, presently ours do not take into account the relative severity of ranking particular diseases above the gold-standard disease. However, in reality it may be the case that the

model ranks a close variant above the gold-standard disease, in which case the misclassification cost incurred should be lesser than if an unrelated disease were higher ranked. A further investigation is necessary to determine whether the relative performance of the considered methods would change under a cost metric that takes this into account.

In conclusion, it is difficult to be certain of the performance of the methods we have introduced. There is the potential to improve them via tuning of hyper parameters, or by tuning more quickly, and perhaps precisely, using a method other than sampling. Additionally, a more thorough investigation of different α and β values could be performed: for computational reasons we only used fixed values of $\alpha = 0.001,\ \beta = 0.1$ throughout this report. It is a possibility that using different values or integrating over a distribution for these values will improve results.

As for further steps, one possibility is to take a variational approach. This would allow one to consider probabilistic states for the hidden nodes rather than just sampling over many deterministic ones, meaning that complex modifications to the model would still be computationally tractable.

References

[1] Sebastian Bauer, Sebastian Köhler, Marcel H. Schulz, and Peter N. Robinson. Bayesian ontology querying for accurate and noise-tolerant semantic searches. *Bioinformatics*, 28(19):2502–2508, 2012.

[2] M. Girdea, S. Dumitriu, M. Fiume, S. Bowdin, K. M. Boycott, S. Chénier, D. Chitayat, H. Faghfoury, M. S. Meyn, P. N. Ray, J. So, D. J. Stavropoulos, and M. Brudno. PhenoTips: Patient phenotyping software for clinical and research use. *Human Mutation*, 34:1057–1065, 2013.

[3] Ada Hamosh, Alan F. Scott, Joanna S. Amberger, Carol A. Bocchini, and Victor A. McKusick. Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Research*, 33(D1):D514–D517, 2005.

[4] Tommi S. Jaakkola and Michael I. Jordan. Variational probabilistic inference an d the QMR-DT network. *Journal of Artificial Intelligence Research*, 10:291–322, 1999.

[5] Sebastian Köhler et al. The Human Phenotype Ontology Project: Linking molecular biology and disease through phenotype data. *Nucleic Acids Research*, 42(D1):D966–D974, 2014.

[6] McKusick-Nathans Institute of Genetic Medicine (Johns Hopkins University). Online Mendelian Inheritance in Man (OMIM), April 2015. Available online at `http://omim.org/`.