Automatic Thresholding by Sampling Documents and Estimating Recall

ILPS@UVA at TAR Task 2.2

Dan Li[1] and Evangelos Kanoulas[2]

University of Amsterdam, 1098XH, Amsterdam, Netherlands[1,2]
{D.Li, E.Kanoulas}@uva.nl

Abstract. In this paper, we describe the participation of the Information and Language Processing System (ILPS) group at CLEF eHealth 2019 Task 2.2: Technologically Assisted Reviews in Empirical Medicine. This task is targeted to produce an efficient ordering of the documents and to identify a subset of the documents which contains as many of the relevant abstracts for the least effort. Participants are provided with systematic review topics with each including a review title, a boolean query constructed by Cochrane experts, and a set of PubMed Document Identifiers (PID's) returned by running the boolean query in MEDLINE. We handle the problem under the Continuous Active Learning framework by jointly training a ranking model to rank documents, and conducting a "greedy" sampling to estimate the real number of relevant documents in the collection. We finally submitted four runs.

Keywords: Continuous active learning · Active sampling · R estimation.

1 Introduction

Systematic reviews are a type of literature review that uses systematic methods to reliably bring together the findings from multiple studies that address a question and are often used to inform policy and practice, e.g. the development of medical guideline in evidence-based medicine [6]. In order to write a systematic review, researchers have to come up with a Boolean query and conduct a search that will retrieve all the documents that are relevant. This is a difficult task, known in the Information Retrieval (IR) domain as the total recall problem.

The CLEF eHealth Task 2 "Technology Assisted Reviews in Empirical Medicine Introduction" aims to automate this process [3] [4]. It consists of two subtasks. Task 2.1 focuses on the construction of the Boolean query, and Task 2.2 focuses on producing an efficient ordering of the documents, such that all of the relevant abstracts are retrieved as early as possible, and identifying a subset which contains all or as many of the relevant abstracts for the least effort.

We participated Task 2.2 and submitted 4 runs: abs-th-ratio-ilps@uva, abs-hh-ratio-ilps@uva, doc-th-ratio-ilps@uva, doc-hh-ratio-ilps@uva.

2 Task Description

Task 2.2 is Abstract and Title Screening. The participants are given the document collection extracted through the Boolean Search in Task 2.1, and are asked to produce an the efficient ordering of the documents, such that all of the relevant abstracts are retrieved as early as possible, and at the same time to identify a subset of A which contains all or as many of the relevant abstracts for the least effort (i.e. total number of abstracts to be assessed).

3 Method

3.1 The model

In this paper, we propose a novel model for the TAR process inspired by [1] and [5]. The model mainly consists of a ranking module, a sampling module, an assessment module, an estimation module and a stopping module. Given a topic and a document collection $\mathcal{C}$ the reviewer is interested in, together with a target recall level r_{target} that the reviewer wants to achieve, the model iteratively outputs a set of documents until the estimated recall exceeds the target recall. We elaborate the model in Algorithm 1.

Let S denote the set of sampled documents and n denote the number of documents in S, $\mathcal{L}_t$ denote the labelled documents (the training set) at batch t and $\mathcal{U}_t$ denote the unlabelled documents at t. Initially, $\mathcal{L}_t$ is empty and we fill it with a pseudo document d_0 which is made of the description of the topic provided. In `line 3`, k documents are uniformly sampled from $\mathcal{U}_t$, and temporarily labeled non-relevant and added to $\mathcal{L}_t$. In `line 4`, a ranking model is trained on $\mathcal{L}_t$. In `line 5-7`, a sampling distribution $\mathcal{P}_t$ is constructed based on the ranked list of documents produced by the ranking model and a fixed number of b documents are independently and with replacement sampled from $\mathcal{P}_t$. In `line 8`, reviewers assess the relevance of the sampled documents. Note that the sampled b documents may contain duplicates, therefore reviewers only need to assess the unique documents. In `line 10`, $\widehat{R_t}$ and $\widehat{var}(\widehat{R_t})$ are calculated. In `lines 11-15`, the stopping module uses $\widehat{R_t}$ and $\widehat{var}(\widehat{R_t})$ to decide whether to stop or not. In `line 17`, produce the ordering of documents by sampled relevant, sampled non-relevant, un-sampled, with the stopping threshold at the first un-sampled documents.

Algorithm 1: Automatic thresholding algorithm

```
Input: Target topic; document collection $\mathcal{C}$, target recall $r_{target}$.
Output: A list of retrieved documents with stopping threshold.
$\mathcal{L}_t = \{$pseudo document $d_0$ labelled relevant$\}$
while not stop do
    // Sample
    Temporarily augment $\mathcal{L}_t$ by uniformly sampling $k$ documents from $\mathcal{U}_t$,
      labeled non-relevant.
    Train a ranking model on $\mathcal{L}_t$.
    Rank all the documents in $C$ by the trained over $\mathcal{L}_t$ ranker.
    Construct sampling distribution $\mathcal{P}_t$ over the ranked documents.
    Sample $b$ document from the distribution $\mathcal{P}_t$.
    Render relevance assessments for the sampled documents that belong to $\mathcal{U}_t$.
    Remove the $k$ temporary documents from $\mathcal{L}_t$. Place the $b$ labeled documents
      in $\mathcal{L}_t$, and remove them from $\mathcal{U}_t$.
    // Estimate
    Calculate $\widehat{R_t}$ and $\widehat{var}(\widehat{R_t})$ via (4).
    // Stop condition
    if $\widehat{R}$ and $\widehat{var}(\widehat{R_t})$ satisfy stopping strategy then
        stop = True
    else
        stop = False
    end
end
Produce the ordering of documents by sampled relevant, sampled non-relevant,
  un-sampled, with the stopping threshold at the first un-sampled documents.
```

3.2 Ranking module

We use the TF-IDF vector of a document as its features. Considering effectiveness and efficiency we employ Logistic Regression as the ranking model. We use its implementation in *scikit-learn*[1]. At each batch t, a new model is trained from scratch using the current training data $\mathcal{L}_t$.

3.3 Sampling module

Sampling distribution Note that in Algorithm 1 we need to sample documents from a distribution $\mathcal{P}_t = \left\{p_i^t\right\}$ (for notation simplicity we use $\mathcal{P}$ in this section). Ideally, the selection probability p_i should be positively correlated with the relevance labels, which allows an estimator $\widehat{R}$ with low variance (see Section 3.4). However, the relevance labels are not known until documents are assessed by the reviewers. What we have instead is a ranking model that can predict the

[1] https://scikit-learn.org/

probability of relevance and output a list of ranked documents, which we can use to construct $\mathcal{P}$. We use Power Law distribution which assumes the selection probability of a document is a power function of its position in the ranked list, defined as

$$p_i = \frac{1}{Z}\frac{1}{{r_i}^{\alpha}} \quad r_i \in [1, N], \quad \alpha \in (0, +\infty) \tag{1}$$

where N is the number of documents in $\mathcal{C}$, r_i is the rank position, $Z = \sum_{i=1}^{N} \frac{1}{r_i^{\alpha}}$ is the normalization factor.

Inclusion probability We derive the first-order and second-order inclusion probabilities, which is indispensable to calculate $\widehat{R}$. We adopt *sampling with replacement* as our sampling method. At each batch t and for each draw, a document is sampled independently from one of the aforementioned distributions. Let *selection probability* denote the probability that a document is sampled for a draw, and *inclusion probability* the probability that a document is included in the sample set considering all the draws. Under sampling-with-replacement design, the first-order inclusion probability π_i is given by

$$\pi_i = 1 - \prod_{t=1}^{T} \left(1 - p_i^t\right)^{n_t} \tag{2}$$

The second-order inclusion probability $\pi_{i,j}$ – the probability of any two different document d_i and d_j being included – is given by

$$\pi_{i,j} = \pi_i + \pi_j - \left[1 - \prod_{t=1}^{T} \left(1 - p_i^t - p_j^t\right)^{n_t}\right] \tag{3}$$

3.4 Estimation module

We employ Horvitz-Thompson estimator and Hansen-Hurwitz estimator to estimate R and $var(R)$. Both of them are designed for sampling with unequal probabilities, Hansen-Hurwitz estimator is only restricted for with-replacement sampling, while Horvitz-Thompson estimator is for any design. For more details of the derivation the reader can refer to Chapter 6 in [7].

Horvitz-Thompson estimator The Horvitz-Thompson estimator provides an unbiased estimator of population total under a general sampling theory [2]. Let $\tau = \sum_{i \in S} y_i$ denote the population total. With any sampling design, with or without replacement, the unbiased Horvitz-Thompson estimator of the population total is $\widehat{\tau} = \sum_{i \in S'} \frac{y_i}{\pi_i}$, where S' is the subset of S only containing unique documents, and π_i is the inclusion probability for document i.

In our case, the population total is the total number of relevant documents for a target topic, denoted as $R = \sum_{i=1}^{N} y_i$. The Horvitz-Thompson estimator of R is

$$\widehat{R}_t^{HT} = \sum_{i \in \widetilde{S}_t'} \frac{y_i}{\pi_i} \tag{4}$$

where $\widetilde{S}_t = \cup_{k=1}^{t} S_k$ denote the accumulated sample set till batch t, y_i^t is relevance of document d_i^t. We use $'$ to denotes the operation to remove duplicated documents, and$\,\widetilde{}\,$to denote the operation to cumulate documents in all previous batches.

Hansen-Hurwitz estimator Hansen-Hurwitz estimator provides an unbiased estimator of population total under sampling with replacement from the same distribution [7].

In our case, the sampling distribution changes at each batch and converges to the ultimate distribution produced by the ranking model trained on the whole documents. The Hansen-Hurwitz estimator of R on S_t is

$$\widehat{R}_t^{HH} = \frac{1}{\widetilde{n}_t} \sum_{k \in \{1,2,\ldots,t\}, i \in S_k} \frac{y_i}{p_i^k} \tag{5}$$

3.5 Stopping module

We propose a stopping strategy based on $\widehat{R}$. With sampling continuing, the strategy repeatedly examines whether $\widetilde{r}_t' > \widehat{R} \cdot r_{target}$, and if so stop TAR process. The intuition is straight forward, if we have collected more relevant than the target number we estimated, we should feel confident to stop.

4 Dataset

The dataset consists of 72 topics for training and 31 topics for testing. For each topic, participants will be provided with

1. Topic-ID
2. The title of the review, written by Cochrane experts;
3. The Boolean query manually constructed by Cochrane experts;
4. The set of PubMed Document Identifiers (PID's) returned by running the query in MEDLINE.

5 Runs

The proposed method is topic-wise in the sense that it repeatedly trains a new ranker based on the current assessed documents. It doesn't need extra training topics. Our runs are directly run on test data.

We submitted four runs: abs-th-ratio-ilps@uva, abs-hh-ratio-ilps@uva, doc-th-ratio-ilps@uva, doc-hh-ratio-ilps@uva. *abs* and *doc* denote whether qrels at abstract level or at content level is used for the relevance feedback in assessment module. *th* and *hh* denote whether Horvitz-Thompson estimator or Hansen-Hurwitz estimator is used to estimate R. A description of each run is presented below.

1. **abs-th-ratio-ilps@uva** *abs* qrels and Horvitz-Thompson estimator
2. **abs-hh-ratio-ilps@uva** *abs* qrels and Hansen-Hurwitz estimator
3. **doc-th-ratio-ilps@uva** *doc* qrels and Horvitz-Thompson estimator
4. **doc-hh-ratio-ilps@uva** *doc* qrels and Hansen-Hurwitz estimator

For all the four runs, we set $\alpha = 0.8$, $b = 100$, $k = 100$, target recall $r_{target} = 0.8$.

6 Results

Our method targets on finding a stopping threshold given a target recall. We re-rank all the sampled relevant documents on the top, followed by all the non-relevant documents and all the un-sampled documents. The stopping threshold is set at the position of the first un-sampled documents. As all the sampled documents are before the stopping threshold, the stopping threshold also indicates the cost. As a consequence it is not valid to apply ranking metrics such as Average Precision, we report thresholding based metrics instead.

Table 1 shows the thresholding result on the test set. First, on both *abs* and *content* level, the Horvitz-Thompson estimator has recall_threshold closer to the target recall 0.8 than the Hansen-Hurwitz estimator, which indicates a more accurate estimation of R. Second, both estimators stop at early stage when sampled documents are less than 50% of the complete documents.

Figure 1 and 2 shows the topic-wise recall_threshold v.s. norm_threshold. Horvitz-Thompson estimator stops at various recall for different topics, while Hansen-Hurwitz estimator stops between 0.8 - 1.0 for most topics. It indicates the estimation of R can help to stop viewing documents, but the variance of the estimated R is large for different topics.

Table 1: Thresholding results on the test set

RUN	norm_threshold	recall_threshold
abs-th-ratio-ilps@uva	0.423	0.838
abs-hh-ratio-ilps@uva	0.47	0.89
doc-th-ratio-ilps@uva	0.392	0.894
doc-hh-ratio-ilps@uva	0.426	0.95

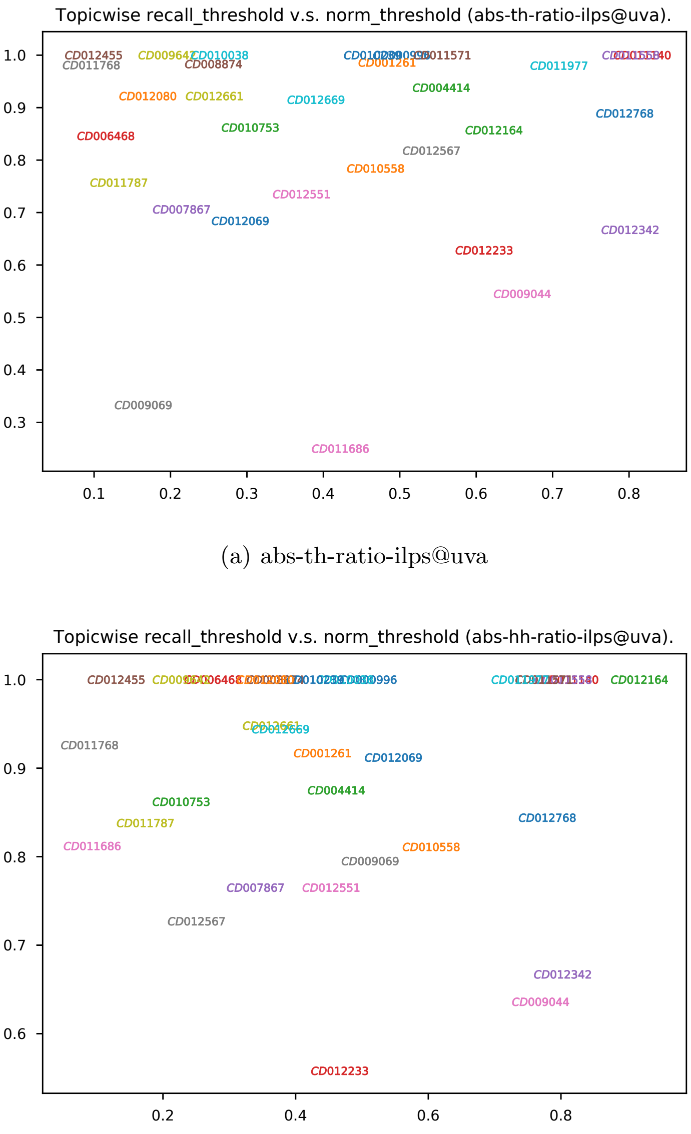
7 Conclusion

In this paper, we presented the runs we submitted to the CLEF 2019 eHealth Task 2.2. We handle the problem under the Continuous Active Learning framework by jointly training a ranking model to rank documents, and conducting a "greedy" sampling to estimate the real number of relevant documents in the collection. We finally submitted four runs.

The result indicates the method can retrieve most relevant documents (around 80% to 90%) with the cost viewing less than 50% of the complete documents. The estimation of R can help to stop viewing documents, but the variance of the estimated R is large for different topics. Further work needs to be done to reduce the variance of the estimated R.

References

1. Cormack, G.V., Grossman, M.R.: Autonomy and reliability of continuous active learning for technology-assisted review. CoRR **abs/1504.06868** (2015), http://arxiv.org/abs/1504.06868
2. Horvitz, D.G., Thompson, D.J.: A generalization of sampling without replacement from a finite universe. Journal of the American statistical Association **47**(260), 663–685 (1952)
3. Kanoulas, E., Li, D., Azzopardi, L., Spijker, R.: Clef 2019 technologically assisted reviews in empirical medicine overview. In: CLEF 2019 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS (2019)
4. Kelly, L., Suominen, H., Goeuriot, L., Neves, M., Kanoulas, E., Li, D., Azzopardi, L., Spijker, R., Zuccon, G., Jimmy, Palotti, J.: Overview of the clef ehealth evaluation lab 2019. In: CLEF 2019 - 10th Conference and Labs of the Evaluation Forum, Lecture Notes in Computer Science (LNCS). Springer (2019)
5. Li, D., Kanoulas, E.: Active sampling for large-scale information retrieval evaluation. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. pp. 49–58. ACM (2017)
6. O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., Ananiadou, S.: Using text mining for study identification in systematic reviews: a systematic review of current approaches. Systematic reviews **4**(1), 5 (2015)
7. Thompson, S.K.: Sampling. John Wiley & Sons, Inc., Hoboken, New Jersey, 3 edn. (2012)



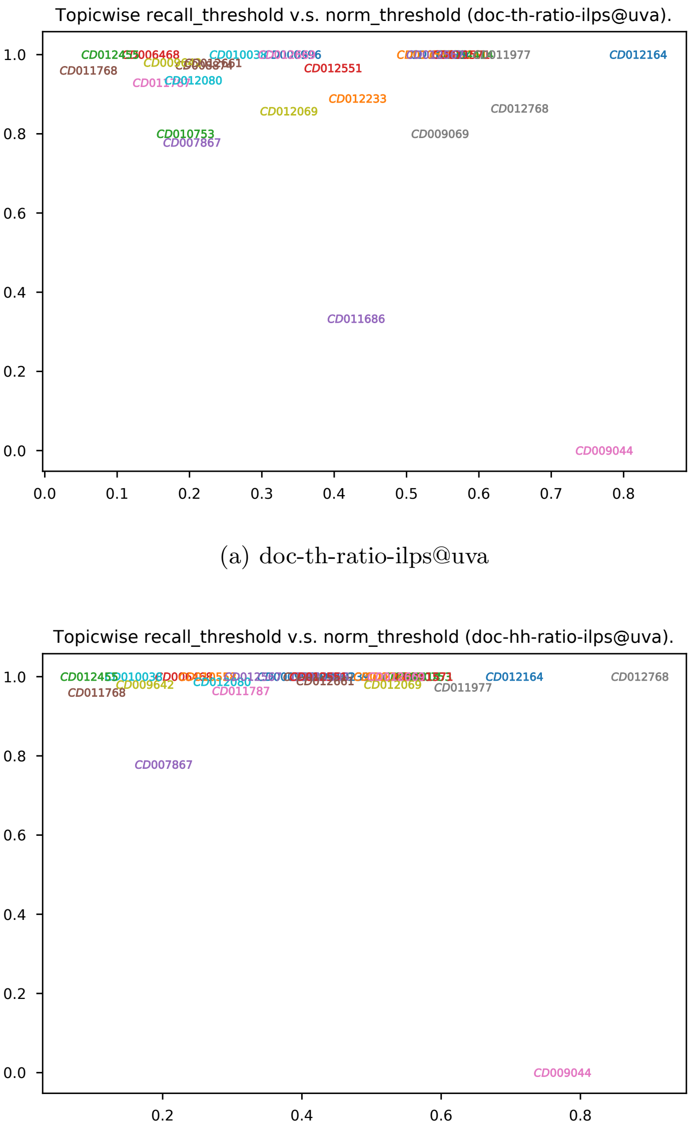
Topicwise recall_threshold v.s. norm_threshold (abs-th-ratio-ilps@uva).

(a) abs-th-ratio-ilps@uva

Topicwise recall_threshold v.s. norm_threshold (abs-hh-ratio-ilps@uva).

(b) abs-hh-ratio-ilps@uva

Fig. 1: Topicwise recall_threshold v.s. norm_threshold at *abs* level.



(a) doc-th-ratio-ilps@uva

(b) doc-hh-ratio-ilps@uva

Fig. 2: Topicwise recall_threshold v.s. norm_threshold at *content* level.