Leveraging the Transductive Nature of e-Discovery in Cost-Sensitive Technology-Assisted Review

Alessio Molinari

Istituto di Scienza e Tecnologia dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, IT
E-mail: alessiomolinar@gmail.com

Abstract. MINECORE is a recently proposed algorithm for minimizing the expected costs of review for topical relevance (a.k.a. "responsiveness") and sensitivity (a.k.a. "privilege") in e-discovery. Given a set of documents that must be classified by both responsiveness and privilege, for each such document and for both classification criteria MINECORE determines whether the class assigned by an automated classifier should be manually reviewed or not. This determination is heavily dependent on the ("posterior") probabilities of class membership returned by the automated classifiers, on the costs of manually reviewing a document (for responsiveness, for privilege, or for both), and on the costs that different types of misclassification would bring about. We attempt to improve on MINECORE by leveraging the *transductive* nature of e-discovery, i.e., the fact that the set of documents that must be classified is finite and available at training time. This allows us to use EMQ, a well-known algorithm that attempts to improve the quality of the posterior probabilities of unlabelled documents in transductive settings, with the goal of improving the quality (a) of the posterior probabilities that are input to MINECORE, and thus (b) of MINECORE's output. We report experimental results obtained on a large (≈ 800K) dataset of textual documents.

1 What is e-discovery?

When a civil lawsuit is filed in the United States of America, the judge may request an involved party to produce to the other party any evidence relevant to the lawsuit. Upon receiving such a request, the producing party must search in their electronically stored information for any document which might be relevant to the case, in order to disclose it to the other party; this task usually goes under the name of *e-discovery*. In order to do so, it is typically the case that the producing party asks junior lawyers to review (i.e., annotate) for "responsiveness" (i.e., topical relevance to the case) the candidate documents, and then asks senior lawyers to review for "privilege" (i.e., presence of sensitive content, which would allow the party to rightfully withhold the document) the documents that have been deemed responsive.

As a consequence of this two-phase review, each document is classified into one of the three following classes:

- c_P (which stands for "Produce"): The document has been deemed responsive and not privileged; as such, it should be produced to the receiving party;

FDIA 2019, 17-18 July 2019, Milan, Italy.

- c_L (which stands for "Log"): The document has been deemed responsive and privileged; as such, it should be entered into a "privilege log" (i.e., a repository open to inspection by the judge) and should not be disclosed to the receiving party;
- c_W (which stands for "Withhold"): The document has been deemed not responsive, which means that the producing party should not produce it.

In such a scenario, the producing party may incur two types of costs, i.e., *annotation costs* (since lawyers need to be paid for their reviewing work), and *misclassification costs* (that might derive when an inappropriate class – e.g., "Produce" instead of "Log" – is chosen for a document).

Given the enormous amount of digital documents that should undergo review in many practical cases, a need for automation of the review process has arisen. Several techniques for *technology-assisted review* (TAR), that combine techniques from information retrieval and machine learning, have thus been proposed in the last ten years [5]. The aim of TAR algorithms is that of supporting the review process, minimizing the overall costs deriving from annotation efforts and misclassifications.

2 An overview of MINECORE

MINECORE [4] is a recently proposed decision-theoretic algorithm for minimizing the expected costs of review for responsiveness and privilege in e-discovery. Given a set $\mathcal{D}$ of documents that must each be assigned a class in $\mathcal{C} = \{c_P, c_L, c_W\}$ based on whether they belong or not to the class c_r of responsive documents and/or to the class c_p of privileged documents, the goal of MINECORE is to determine, for each document in $\mathcal{D}$, whether manually reviewing d by responsiveness and/or privilege is expected to be cost-effective or not. This determination is based

1. on the ("posterior") probabilities of class membership (written as $\Pr(c_r|d)$ and $\Pr(c_p|d)$, hereafter called the *posteriors*) returned by automated classifiers h_r (that classifies documents by responsiveness) and h_p (that classifies documents by privilege);
2. on the unit costs of manually checking a document for responsiveness (λ^a_r) or for privilege (λ^a_p), where superscript a stands for "annotation";
3. on the costs λ^m_{ij} incurred when mistakenly assigning class c_i to a document which should be assigned class c_j, where $c_i, c_j \in \mathcal{C}$ and superscript m stands for "misclassification".

Bullet 3 is due to the fact that in e-discovery not all misclassifications are equally serious; for instance, inadvertently disclosing a privileged document is typically a very serious mistake, while inadvertently disclosing a nonresponsive nonprivileged document is usually a less serious one.

MINECORE consists of three phases, which we summarize below. In **Phase 1** we train the two automated classifiers h_r and h_p, and use them to generate, for each document $d \in \mathcal{D}$, the two posteriors $\Pr(c_r|d)$ and $\Pr(c_p|d)$ mentioned in Bullet 1. We can reasonably assume c_r and c_p to be stochastically independent, which implies that we may assume $\Pr(c_P|d) = \Pr(c_r|d)\Pr(\overline{c}_p|d)$, $\Pr(c_L|d) = \Pr(c_r|d)\Pr(c_p|d)$, and $\Pr(c_W|d) = \Pr(\overline{c}_r|d)$. MINECORE takes a *risk minimization* approach, i.e., it classifies each document d in the class

$$\begin{aligned} h(d) &= \arg\min_{c_i} R(d, c_i) \\ &= \arg\min_{c_i} \sum_{j\in\{P,L,W\}} \lambda^m_{ij} \Pr(c_j|d) \end{aligned} \tag{1}$$

where $R(d, c_i)$ is the *risk* associated with assigning d to class c_i. In other words, MINECORE assigns to each document d the class that brings about the minimum misclassification risk, i.e., the minimum expected misclassification cost, thus avoiding courses of actions for which a combination of probability of class membership and misclassification cost is high. The function for measuring the global misclassification cost is thus

$$K^m(\mathcal{D}) = \sum_{i,j \in \{P,L,W\}} \lambda^m_{ij} D_{ij} \tag{2}$$

where D_{ij} is the number of documents $d \in \mathcal{D}$ whose predicted class $h(d)$ is c_i and whose true class (which we denote by $y(d)$) is c_j.

Phase 2 and **Phase 3** are essentially identical to each other. The only difference is that, while Phase 2 determines the subset of documents which are expected to be cost-effective to manually review by responsiveness, Phase 3 determines (once these documents have been indeed manually reviewed by responsiveness) the same for privilege. We will thus only describe Phase 2, leaving it to the reader to work out the details of Phase 3.

Note that, if τ_r documents are reviewed by responsiveness and τ_p documents are reviewed by privilege, the overall cost of the entire process may be described as

$$\begin{aligned} K^o(\mathcal{D}) &= K^m(\mathcal{D}) + K^a(\mathcal{D}) \\ &= K^m(\mathcal{D}) + \lambda^a_r \tau_r + \lambda^a_p \tau_p \end{aligned} \tag{3}$$

If document d is reviewed by responsiveness, this has the effect of removing (assuming infallible reviewers) any uncertainty about whether $d \in c_r$ or not. In other words, if by subscript $n \in \{1, 2, 3\}$ we indicate the value of a given quantity after Phase n has been carried out, reviewing d by responsiveness means that $\Pr_2(c_r|d)$ will be either 0 or 1. As a result, if d is reviewed by responsiveness it will in general hold that $\Pr_1(c_r|d) \neq \Pr_2(c_r|d)$, $h_1(d) \neq h_2(d)$, and $K^m_1(\mathcal{D}) \neq K^m_2(\mathcal{D})$. Since reviewing d by responsiveness brings about an additional λ^a_r cost, it is worthwhile to annotate d only if, as a result of the annotation, $K^o_2(\mathcal{D}) \leq K^o_1(\mathcal{D})$, i.e., $K^m_2(\mathcal{D}) + \lambda^a_r \leq K^m_1(\mathcal{D})$; in other words, the additional annotation cost must be offset by a reduction in overall misclassification cost of greater or equal magnitude. Of course, computing precisely whether there is going to be such a reduction at all is not possible, because at the time of deciding whether d should be annotated or not we do not know the value of $y_r(d)$ (a binary variable that indicates whether the reviewer will annotate d as responsive or not), and we do not know the true label $y(d)$ of d. However, it is possible to compute an expectation of this reduction over the $y_r(d)$ and $y(d)$ variables; when this expected value exceeds λ^a_r, MINECORE decrees that d should be annotated by responsiveness. We refer the reader to [4, §3] for details on how the above expected value is computed, and for a full mathematical specification.

3 Improving our posteriors via EMQ

The goal of MINECORE is to identify the τ_r (resp., τ_p) documents that, when reviewed for responsiveness (resp., privilege), will each bring about a reduction in the expected overall cost of the review process. In order to do this, MINECORE uses as input the data listed in the three bullets at the beginning of §2.

In this work we attempt to improve this process not by modifying MINECORE, but by improving the quality of the input that MINECORE receives, with the goal of having MINECORE bring about a higher reduction in expected costs. Since the input data mentioned in Bullets 2 and 3 are user-defined parameters, and are hence not under our control, we focus on the input data mentioned in Bullet 1, i.e., the posteriors $\Pr(c_r|d)$ and $\Pr(c_p|d)$.

Our contribution is based on the observation that e-discovery has, from the standpoint of machine learning, a *transductive* nature, i.e., the unlabelled set $\mathcal{D}$ is finite and available at training time. This fact can be exploited in order to improve the quality of our posteriors.[1] In fact, Saerens and colleagues [6] have presented an instance of the well-known EM ("expectation maximization") algorithm (an instance which, following [2], we will call EMQ) that iteratively improves the quality of such probabilities in transductive contexts. EMQ is based on a mutually recursive algorithm that alternates between (a) recomputing the prior probabilities $\Pr(c)$ (hereafter: the *priors*) as the average of the posteriors over the entire set, and (b) recomputing the posteriors by multiplying the old posteriors by the ratio between the new priors and the old priors. The iteration is carried out until convergence, and has been shown to deliver both improved posteriors [6] and improved priors [2].

Therefore, (a) we obtain calibrated posteriors $\Pr(c_r|d)$ and $\Pr(c_p|d)$, for each document $d \in \mathcal{D}$, from our classifiers h_r and h_p (which, in the experiments of §4, we obtain via a linear SVM), (b) we update them via EMQ, and (c) we feed them to MINECORE.

4 Experiments

We run experiments in which we compare two versions of MINECORE, one that uses the original posteriors (here dubbed MINECORE$^{\text{MLE}}$, since the probabilities are obtained from a *Maximum Likelihood Estimator*) and one that uses the EMQ-enhanced posteriors (here dubbed MINECORE$^{\text{EMQ}}$). We use the same dataset as [4], which consists of $\approx$ 20,000 training documents, $\approx$ 780,000 test documents, and 120 instantiations of the (c_r, c_p) pair of classes; the values of λ^a_r, λ^a_p, λ^m_{ij}, are from "CostStructure1" in [4, §4.3]. We refer to [4, §4] for other details about the experimental setup. The evaluation function we use is $K^o(\mathcal{D})$ from Equation 3. We measure the ratio $\Delta_{K^o}(\mathcal{D}) = \frac{K^o_{\text{MLE}}(\mathcal{D})}{K^o_{\text{EMQ}}(\mathcal{D})}$ between $K^o(\mathcal{D})$ as deriving from MINECORE$^{\text{MLE}}$ and $K^o(\mathcal{D})$ as deriving from MINECORE$^{\text{EMQ}}$; if this ratio is > 1, this means that EMQ delivers an improvement, and that our attempt has been successful.

We show the outcome of our experiment in Figure 1, where we display the class pairs on the X axis (sorted by their $\Delta_{K^o}(\mathcal{D})$ value in order to improve the readability of the plot) and $\Delta_{K^o}(\mathcal{D})$ on the Y axis; the red line indicates the value $\Delta_{K^o}(\mathcal{D}) = 1$. A dot above the red line indicates, for the corresponding class pair, an improvement in overall cost with respect to the MINECORE$^{\text{MLE}}$ baseline, whereas a dot below the red line indicates a deterioration. Figure 1 shows that for 60% of the pairs, using EMQ brings about a deterioration. The average value of $\Delta_{K^o}(\mathcal{D})$ across the 120 class pairs is 0.99, which indicates a 1% average deterioration.

This is surprising, given the positive results that have been reported in past literature for EMQ. As a result, we have further tried to compare the quality of the $\Pr^{\text{MLE}}(c|d)$ and $\Pr^{\text{EMQ}}(c|d)$ sets of posteriors via an application-independent measure of quality. For this, we have chosen "soft accuracy" (a variant of what has been called the *Brier score*), which corresponds to the standard accuracy measure $A = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{FP}+\text{TN}}$ as evaluated on a "soft" contingency table where cells are (instead of counts) sums of probabilities of class membership.[2] Essentially no

[1] At a first approximation, we may say that the quality of a posterior $\Pr(c|d)$ is high when $|h_c(d) - \Pr(c|d)|$ is low; see §4 for more details.

[2] For example, for a document d that actually belongs to c: (1) if c is actually assigned, for a standard contingency table this contributes a value of 1 to the TP cell; (2) if a

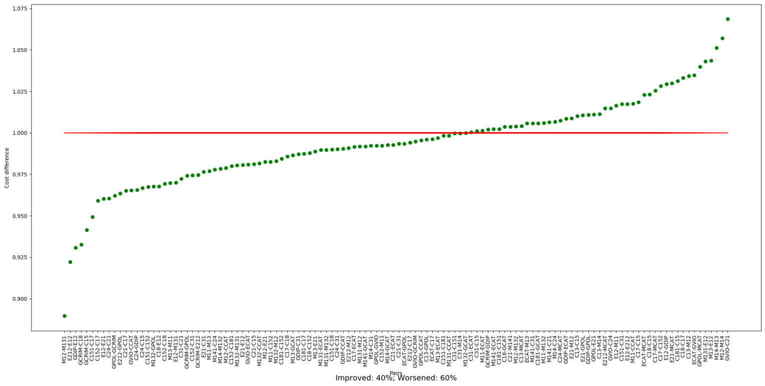

Fig. 1. Values of $\Delta_{K^o}(\mathcal{D})$ for the 120 class pairs; values above (resp., below) the red line indicate that the use of EMQ has generated an improvement (resp., a deterioration).

difference in soft accuracy was detected, with MLE posteriors obtaining $A = 0.9743$ and EMQ posteriors obtaining $A = 0.9742$. Note that, in our case, Δ_{K^o} is a more appropriate evaluation measure than A, since it is application-dependent. In fact, while A measures the quality of our posteriors in a somehow abstract way, Δ_{K^o} directly measures the impact that these posteriors have on MINECORE; this adds to the disappointment that the results returned by EMQ have brought about.

We have also checked if the application of EMQ has improved the class priors. In order to do so we have measured, for each estimation method $M \in \{\text{MLE}, \text{EMQ}\}$ and for each class c that shows up in at least one of the 120 class pairs, the *absolute estimation error* (AE), i.e., the absolute value $\text{AE}_M(c) = |\Pr(c) - \hat{\Pr}(c)_M|$ of the difference between the true class prior $\Pr(c)$ and the estimated class prior $\hat{\Pr}(c)_M$; for each $M \in \{\text{MLE}, \text{EMQ}\}$ we have then averaged these values across all such classes. The results show that $\text{AE}_{\text{MLE}} = 0.191$ and $\text{AE}_{\text{EMQ}} = 0.082$, i.e., the use of EMQ brings about a reduction of approximately 57% in the absolute estimation error of the class priors.

Overall, this result is surprising: the use of EMQ improved (as expected, and by a very large margin) the quality of the priors, but did not bring about any improvement (actually: brought about a small deterioration) in the quality of the posteriors. The reason this is surprising is that the quality of the posteriors and the quality of the priors should go hand in hand; indeed, the very reason why EMQ is expected to improve the priors is that it computes it as the sum of the supposedly better-quality posteriors that it also computes.

One of the reasons we are witnessing such an unexpected behaviour might lie beneath the fact that the "distribution drift" in our data is very low (the average difference between the prevalence of a class in the training and in the test sets is $\leq 0.6\%$). Indeed, the results of [1] indicate that EMQ tends to work better in high-drift conditions that in low-drift ones. In order to check whether, in a dataset characterized by higher drift, EMQ would deliver a better performance, we have generated artificial drift in the dataset by removing increasingly high quantities of negative examples from the training set. This should allow EMQ to improve the quality of both priors and posteriors, thus improving the performance of MINECORE. Since example removal is completely random, we have repeated the experiment several times, first removing up to 60% (and eventually up to 80%) of the negatives. MINECORE

posterior $\Pr(c|d)$ is returned, for a "soft" contingency table this contributes a value of $\Pr(c|d)$ to the TP cell and a value $(1 - \Pr(c|d))$ to the FN cell.

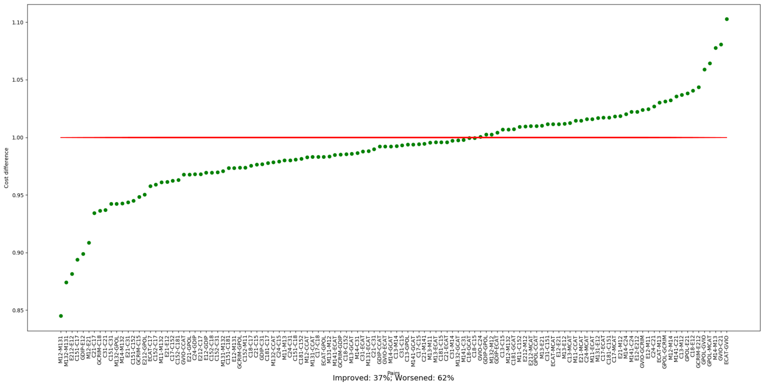

Fig. 2. Values of $\varDelta_{K^o}(\mathcal{D})$ for the 120 class pairs after a random 80% of the negative training examples have been removed.

was then run first with MLE-generated posteriors and then with EMQ-generated posteriors. However, once again the results did not support our conjecture: even after removing 80% of the negatives from the training set (see Figure 2), $\varDelta_{K^o}(\mathcal{D})$ does not get higher than 1, and gets slightly smaller than 1 in most cases.

5 Conclusion

Although our idea of improving the posteriors input to MINECORE by leveraging the transductive nature of our dataset seemed (and still seems to us) reasonable and promising, the experiments we have ran so far indicate the opposite: EMQ has not improved our posteriors, while it has improved the priors. This is still work in progress, and one hypothesis that we are going to test is that the EMQ algorithm might be well-suited to datasets that exhibit some type of drift but not others [3].

We still believe that exploiting the transductive nature of e-discovery could prove a key intuition for improving the results of MINECORE. Indeed, our future work on this will focus on new ways to leverage this aspect.

Acknowledgments. The research described in this paper is part of the work carried out by the author for his MSc thesis in Digital Humanities at the University of Pisa (supervisors: Andrea Esuli and Fabrizio Sebastiani), to be discussed in 2019.

References

1. Caelen, O.: Quantification and learning algorithms to manage prior probability shift. Master's thesis, Institut de Statistique, Biostatistique et Sciences Actuarielles, Université Catholique de Louvain, Louvain-la-Neuve, BE (2017)
2. Gao, W., Sebastiani, F.: Tweet sentiment: From classification to quantification. In: Proceedings of the 7th International Conference on Advances in Social Network Analysis and Mining (ASONAM 2015). pp. 97–104. Paris, FR (2015). https://doi.org/10.1145/2808797.2809327
3. Moreno-Torres, J.G., Raeder, T., Alaiz-Rodríguez, R., Chawla, N.V., Herrera, F.: A unifying view on dataset shift in classification. Pattern Recognition **45**(1), 521–530 (2012). https://doi.org/10.1016/j.patcog.2011.06.019
4. Oard, D.W., Sebastiani, F., Vinjumur, J.K.: Jointly minimizing the expected costs of review for responsiveness and privilege in e-discovery. ACM Transactions on Information Systems **37**(1), 11:1–11:35 (2019). https://doi.org/10.1145/3268928

5. Oard, D.W., Webber, W.: Information retrieval for e-discovery. Foundations and Trends in Information Retrieval **7**(2/3), 99–237 (2013). https://doi.org/http://dx.doi.org/10.1561/1500000025
6. Saerens, M., Latinne, P., Decaestecker, C.: Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. Neural Computation **14**(1), 21–41 (2002). https://doi.org/10.1162/089976602753284446