Healthcare Transformation with Informatics and Artificial Intelligence
J. Mantas et al. (Eds.)

doi:10.3233/SHTI230545

Implementing Informative-Based Active Learning in Biomedical Record Linkage for the Splink Package in Python

Marko MILETIC[a] and Murat SARIYAR[a,1]
[a] *Bern University of Appl. Sciences, Switzerland*

Abstract. In biomedical record linkage, efficient determination of a threshold to decide at which level of similarity two records should be classified as belonging to the same patient is frequently still an open issue. Here, we describe how to implement an efficient active learning strategy that puts into practice a measure of usefulness of training sets for such a task. Our results show that active learning should always be considered when training data is to be produced via manual labeling. In addition to that, active learning gives a quick indication how complex a problem is by looking into the label frequencies: If the most difficult entities are always stemming from the same class, then the classifier will probably have less problems in distinguishing the classes. In big data applications, these two properties are essential, as the problems of under- and overfitting are exacerbated in such contexts.

Keywords. Active learning, record linkage, entropy, splink

1. Introduction

Biomedical record linkage is well-known in medical informatics but is still associated with unresolved issues in practical applications [1]. For instance, when trying to link data of patients existing in different repositories, it is important to decide at which level of similarity two records should be classified as belonging to the same patient. In previous investigations, we proposed to rely on extreme value statistics [2]. Even though, this works well for an initial guess when no training data is available, there is a potential for improvements, as the empirical distributions of matches and non-matches can vary a lot due to their dependence on the data quality as well as on hand-tailored data reducing steps such as blocking [3, 4]. If there are resources for manual labeling, a more promising alternative is to use representative training data for determining decision thresholds. In order to minimize the number of training data to be labeled, active learning should be considered in such cases rather than simple random sampling [5].

Even though, there are different active learning strategies, e.g., characterized by the fact of being the classification method applied or not, there is one central characteristics for all such methods, namely, the implementation of a measure of usefulness for training sets. Frequently, the Shannon entropy metric is used as a basis. In informal terms, a pair of records (or the related comparison pattern) is deemed useful if most of the similarity

[1] Corresponding Author: Murat Sariyar, Bern University of Applied Sciences, Quellgasse 21, CH2502 Biel/Bienne, Switzerland; E-mail: murat.sariyar@bfh.ch.

values are in the middle of the overall value range. With the same justification as for empirical probability distributions, the frequency of the record pairs should be considered as well. If a complex case is not frequent enough, its impact on the result may be too insignificant to justify spending resources on labeling it. Several extensions of this simple idea exist in the literature, and we have decided to implement and extend one proposal that does not depend on the classifying method and accounts for the fact, that usefulness of additional examples do not only depend on the example and its frequency, but also on the relation with the record pairs already included in the training set produced thus far [6]. For this purpose, the Shannon entropy and an uncertainty measure are used.

In the following, we will first describe our use case, the package used for record linkage and the active learning strategy we implemented. In the Result section, the main properties of resulting training set, the classification results, and the main challenges in applying active learning are described. We conclude with the main insights of our work for further projects, especially when linking large datasets.

2. Methods

The use case is linking patient information of different registries with a population registry using first name, last name, address, postcode, state, and date of birth. Since we have provided similar data to the community and our primary objective is to present our active learning implementation, we utilized a modified version of our simulated data from the Python RecordLinkage package. This modified version includes a shift of the string variables from a German-based to an English-based value range and the inclusion of the variable "suburb" [7]. In the real-world scenario, both registries have a few million entries, in the simulated data this was reduced to 5000 entries each. Producing results in a few seconds in contrast to hours or even days is very important in calibrating the record linkage approach in view of the different parameters that have to be set.

We use the Python Package Splink [8] as our environment for performing record linkage instead of our own mature R package RecordLinkage [9], because in the real-world setting millions of data sets have to be linked and handling big data was an essential requirement. Splink allows using the PySpark backend in order to apply Fellegi-Sunter model with parameter estimations based on an Expectation-Maximization algorithm. The tool has several noteworthy features, such as the ability to make term-frequency adjustments and customize comparison functions. However, it does not provide any guidance on how to establish the appropriate thresholds for matches.

As a promising basis for efficiently determining thresholds described in the scientific literature, we used and adapted information-based active learning [6]. The algorithm has three parts: (i) initializing the training set by using stratified sampling; (ii) computing usefulness of the labeled record pairs by using following formula:

$$info(\boldsymbol{w}_j, \boldsymbol{T}) = \alpha \cdot entropy(\boldsymbol{w}_j, \boldsymbol{T}) + (1-\alpha) \cdot uncertainty(\boldsymbol{w}_j, \boldsymbol{T}),$$

where $\boldsymbol{w}_j$ is a comparison pattern (representing a concrete record pair) j, $\boldsymbol{T}$ is the training set comprising all comparison patterns (record pairs) labeled thus far, $entropy(\boldsymbol{w}_j, \boldsymbol{T})$ is the sum of entropies with respect to $\sum_{\boldsymbol{w_k} \in \boldsymbol{T_S}} sim(\boldsymbol{w}_j, \boldsymbol{w}_k)$ and $\sum_{\boldsymbol{w_k} \in \boldsymbol{T_O}} sim(\boldsymbol{w}_j, \boldsymbol{w}_k)$,, where $\boldsymbol{T}_S$ is the subset of $\boldsymbol{T}$ comprising comparison patterns having the same label as $\boldsymbol{w}_j$ (match or non-match), and $\boldsymbol{T}_O$ is the subset of comparison patterns having the opposite label as $\boldsymbol{w}_j$. The $uncertainty(\boldsymbol{w}_j, \boldsymbol{T})$ is a measure for the frequency of comparison patterns of the same class as $\boldsymbol{w}_j$ within a certain neighborhood: the higher the frequency,

the lesser the uncertainty. Usually, α is set to 0.5. (iii) In the final step of the algorithm, the neighborhood of the most informative comparison patterns is utilized to select unlabeled comparison patterns that are similar to the informative labeled ones. The farthest-first algorithm is then used to choose which of these patterns should be labeled. We adapted the algorithm at several stages (see next section) and we determined the threshold for matches in the following way: Determine the rate of false-positives δ_m that is allowed (usually 1% or less), then, start from the maximum matching weights, and reduce it until you reach δ_m.

3. Results

Applying information-based active learning on the final record pairs on the data set comprising 26932 non-matches and 5000 matches, which was the results of using different blocking strategies, given by

```
blocking_rules = [
    "l.given_name = r.given_name AND l.surname = r.surname",
    "l.date_of_birth = r.date_of_birth",
    "l.state = r.state AND l.address_1 = r.address_1",
    "l.street_number = r.street_number AND l.address_1 = r.address_1",
    "l.postcode = r.postcode",
],
```

we selected 126 training samples. Thanks to the high-quality data and effective blocking strategy, the majority of the training samples were matches, with only two instances of non-matches. This finding suggests that, in most cases, these non-matches were the most complex ones. Figure 1 shows a histogram of the match weights, illustrating the disproportionate high number of record pairs with high match weights (in most cases matches). To make the results more robust, we randomly sampled different numbers of training items from the entire dataset. This approach added more non-matches to the training set, for example, we added 168 non-matches when we randomly sampled 200 comparison patterns. In all cases, a threshold value of -4 was the best choice, resulting in 2 false positives in the training sets and 29 false positives when applied to the whole dataset. It was important to provide sensitive analyses with respect to the number of training samples in order to show the efficiency as well as the validity of the active learning strategy, which is always an issue if only few samples can be labeled manually.

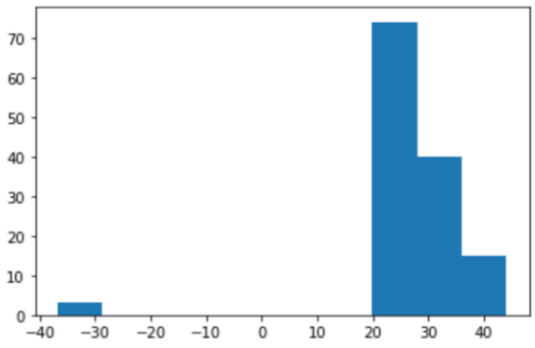

Figure 1. Histogram of the match weights on the training sample produced by our adaptation of the information-based active learning strategy.

Our adaptations of the information-based active learning strategy were related to steps of the algorithm: (i) For the initialization of the training set, we used canopy clustering with a frequency-based determination of the T_1 and T_2 values, guaranteeing that 30-40 canopies were created. (ii) Computing $info(\boldsymbol{w}_j, \boldsymbol{T})$ was modified by including the matching weights as well into the similarity measurement (cosine similarity) and by using the third quartile as δ_m instead of the mean. (iii) The candidate generation was not done iteratively, but by the fixing the number k as a global parameter, which also led to a slight modification of the farthest-first computation.

4. Discussion

Our results show that active learning should always be considered when training data is to be produced via manual labeling. In addition to that, active learning gives a quick indication how complex a problem is by looking into the label frequencies: If the most difficult entities are always stemming from the same class, then the classifier will probably have less problems in distinguishing the classes. In big data applications, these two properties are essential, as the problems of under- and overfitting are exacerbated in such contexts. Further, it is important to consider the manual work in record linkage from a broader perspective than just focusing on generating training samples. The original Fellegi-Sunter model included an area of uncertainty that required manual review. To address this issue, we propose a strategy for allocating the budget for manual labeling. This strategy involves the following steps: first, generate training data through active learning and a few hundred randomly sampled data to determine a threshold for definite matches that is minimizing false positives. Second, the remaining budget should be allocated to define a threshold for definite non-matches. This can be achieved by sorting all the comparison patterns in descending order and selecting the number of rows equal to the number of comparisons reserved for manual review. The match weight of the row in which one ends up is the second threshold for definite non-matches. Our next research project is to provide a framework for all manual tasks within the record linkage.

References

[1] Harron K, Dibben C, Boyd J, et al. Challenges in administrative data linkage for research. Big Data Soc 2017; 4: 2053951717745678.
[2] Sariyar M, Borg A, Pommerening K. Controlling false match rates in record linkage using extreme value theory. J Biomed Inform 2011; 44: 648–654.
[3] Steorts RC, Ventura SL, Sadinle M, et al. A Comparison of Blocking Methods for Record Linkage. In: Domingo-Ferrer J (ed) Privacy in Statistical Databases. Cham: Springer International Publishing, 2014, pp. 253–268.
[4] O'Hare K, Jurek-Loughrey A, de Campos C. An unsupervised blocking technique for more efficient record linkage. Data Knowl Eng 2019; 122: 181–195.
[5] Dasgupta SD. Two faces of active learning. Theor Comput Sci 2011; 412: 1767–1781.
[6] Christen V, Christen P, Rahm E. Informativeness-Based Active Learning for Entity Resolution. In: Cellier P, Driessens K (eds) Machine Learning and Knowledge Discovery in Databases. Cham: Springer International Publishing, 2020, pp. 125–141.
[7] De Bruin J. Python Record Linkage Toolkit: A toolkit for record linkage and duplicate detection in Python. Epub ahead of print December 2019. DOI: 10.5281/zenodo.3559043.
[8] Enamorado T, Fifield B, Imai K. Using a Probabilistic Model to Assist Merging of Large-Scale Administrative Records. Am Polit Sci Rev 2019; 113: 353–371.
[9] Sariyar M, Borg A. The RecordLinkage Package: Detecting Errors in Data. R J 2010; 2: 61–67.