Reach for Gold: An Annealing Standard to Evaluate Duplicate Detection Results

TOBIAS VOGEL, ARVID HEISE, UWE DRAISBACH, DUSTIN LANGE, and FELIX NAUMANN, Hasso Plattner Institute

Duplicates in a database are one of the prime causes of poor data quality and are at the same time among the most difficult data quality problems to alleviate. To detect and remove such duplicates, many commercial and academic products and methods have been developed. The evaluation of such systems is usually in need of pre-classified results. Such gold standards are often expensive to come by (much manual classification is necessary), not representative (too small or too synthetic), and proprietary and thus preclude repetition (company-internal data). This lament has been uttered in many papers and even more paper reviews.

The proposed *annealing standard* is a structured set of duplicate detection results, some of which are manually verified and some of which are merely validated by many classifiers. As more and more classifiers are evaluated against the annealing standard, more and more results are verified and validation becomes more and more confident. We formally define gold, silver, and the annealing standard and their maintenance. Experiments show how quickly an annealing standard converges to a gold standard. Finally, we provide an annealing standard for 750,000 CDs to the duplicate detection community.

Categories and Subject Descriptors: H.2.7 [**Database Management**]: Database Administration

General Terms: Algorithms, Management

Additional Key Words and Phrases: Annealing standard, gold standard, silver standard, duplicate detection, classification

ACM Reference Format:
Vogel, T., Heise, A., Draisbach, U., Lange, D., and Naumann, F. 2014. Reach for gold: An annealing standard to evaluate duplicate detection results. *ACM J. Data Inform. Quality* 5, 1–2, Article 5 (August 2014), 25 pages.
DOI: http://dx.doi.org/10.1145/2629687

1. THE LACK OF GOLD STANDARDS FOR DATA QUALITY

Duplicates in a database table are multiple, different records representing the same real-world entity. A prime example for duplicity are customer relationship management systems in which customers are represented multiple times, for instance, with different addresses, typos in their family names, incorrect dates-of-birth, etc. Effects of duplicates range from the harmless poor customer satisfaction to more aggravated problems, such as giving the same customer multiple lines of credit, all the way to low key performance indicators resulting in poor business decisions. Among the many factors of data quality, duplicity is among the most important and the most difficult problems. Many methods have been proposed and/or implemented in commercial data quality applications [Elmagarmid et al. 2007; Naumann and Herschel 2010].

Authors' addresses: T. Vogel (corresponding author), A. Heise, U. Draisbach, D. Lange, and F. Naumann, Hasso Plattner Institute, Postfach 900460, 14440 Potsdam, Germany; email: {tobias.vogel, arvid.heise, uwe.draisbach, dustin.lange, felix.naumann}@hpi.uni-potsdam.de.

DOI: http://dx.doi.org/10.1145/2629687

To find all duplicates within a dataset, the naive approach is to compare each record with all the other records and use a classifier to decide whether a record pair is a duplicate or not. This approach faces two challenges: First, the complexity is quadratic, and second, the decision of whether a pair of records represents the same real-world entity is not trivial. To tackle these challenges, a variety of algorithms have been proposed, such as partitioning methods to reduce the number of comparisons [Baxter et al. 2003] and similarity measures to calculate the similarity of record pairs [Elmagarmid et al. 2007].

All these algorithms have in common that they cannot guarantee finding all duplicates and that declared duplicates might be incorrect. Performance measures are necessary to evaluate these algorithms. There is a variety of these measures, and they all require a gold standard to determine the correctness and completeness of duplicate detection results. A generally accepted dataset and a corresponding gold standard result in a duplicate detection benchmark that makes the repeatability of experiments and the comparability of different methods possible. Unfortunately, there is no single large, available, and nonsynthetic dataset with a corresponding gold standard in the duplicate detection community; this makes it difficult both to evaluate and to compare different results.

To reduce costs associated with creating gold standards, we propose the novel *annealing standard*. "Annealing" means that the corresponding standard iteratively gets better and better and thus "converges" against the not available, yet desirable gold standard.[1] The annealing standard exploits inter-classifier agreement and requires only manual work in cases of doubt. In this article, we consider the classification algorithms and manual decision process as black boxes and focus on describing the workflow to generate a standard for a dataset with the results of several classifiers. It is worth mentioning that all concepts and definitions in the article can also be applied to other classification tasks. Because the duplicate detection problem is a good example for a classification problem with a strong need for an annealing standard, we focus on duplicate detection in the remainder.

1.1. Available Gold Standards

For duplicate detection, there is no single dataset that is used for benchmarking. In Draisbach and Naumann [2010], we describe three datasets that are often used for evaluation and which all have a gold standard.

The *CORA Citation Matching dataset* contains 1,879 references representing different papers and is used in several approaches, to evaluate duplicate detection [Bilenko and Mooney 2003a; Dong et al. 2005; Singla and Domingos 2005]. As described in Draisbach and Naumann [2010], the reference ID (the BibTeX key) is not always faultless, but a manually verified gold standard can be downloaded from the DuDe toolkit website.[2]

The *restaurant dataset* comprises only 864 records, which makes it difficult to evaluate partitioning algorithms. Additionally, it contains only clusters with a maximum size of 2, and this makes it not useful for algorithms that, for example, rely on transitivity to reduce the number of comparisons.

The third dataset comprises 9,763 randomly extracted *CD records from freeDB*.[3] The gold standard contains 299 duplicates which were detected in a manual inspection.

[1]We use the term "annealing" in the same sense as the well-known "simulated annealing" optimization method, namely, cooling-down or solidifying.

[2]`http://www.hpi.uni-potsdam.de/naumann/projekte/dude_duplicate_detection.html`.

[3]`http://www.freedb.org/`.

All three datasets have in common that they comprise only a small number of records. The reason being that a manual inspection of all possible record pairs is very time consuming. An alternative to the manual inspection is using a dataset generator, such as the UIS Database Generator[4] or the FEBRL Generator.[5] Such artificially generated data seems to be an attractive alternative to the manual inspection of real-world data, as the number of duplicates and the error types, such as missing values or typographical errors, can be controlled. On the other hand, generated data needs to reflect issues of real-world data, including the frequency distribution of attribute values and error types. Only real-world data contain the surprising types of errors that one cannot foresee but that one hopes to detect anyway. Synthetically inserting errors into data and then re-discovering them is not sufficiently convincing, therefore, real-world data is generally preferred.

An overview about data generation for deduplication and record linkage is given by Christen [2005]. For all generated datasets, there is always the risk that they do not contain uncommon errors or that classifiers are overfitted regarding the generated error types.

1.2. An Ever-Improving Standard

The core idea of the annealing standard is to create a standard that comprises all duplicates and nonduplicates that can be detected with state-of-the-art algorithms. With any of these algorithms, a first baseline is created and with more algorithms, the standard is refined. This refinement is based on a manual inspection of the differences between the current annealing standard and the newer results. It is not as perfect as a gold standard, but due to the iterative improvement, it becomes nearly as good as a gold standard after enough iterations. This makes it possible to create a standard even for large datasets with limited manual effort, because obvious duplicates or nonduplicates are classified correctly by all state-of-the-art algorithms, and therefore manual inspection is mainly necessary in the gray and particularly difficult area of possible matches.

The annealing standard aims to reduce the manual work needed from the domain expert similarly to active learning in the machine learning community. Here, the two principle approaches either exploit a confidence score of one classifier [Sarawagi and Bhamidipaty 2002] or employ the disagreement of a committee of classifiers [Freund et al. 1997; Seung et al. 1992] to present a small number of pairs with a high uncertainty to a domain expert and feed the labeled pairs back to the classifiers in several iterations. Since especially difficult pairs with a high uncertainty are in the training set, the classifiers achieve good performance with comparably few pairs. In contrast, the annealing standard operates on classifier *results*. The main goal is to directly improve the standard (not the classifiers) and to eliminate all uncertainties regarding the results. Consequently, while they both reduce the manual effort by avoiding manual inspection of trivial pairs, the metric and the goal to find difficult pairs are different in both approaches.

The contributions of this article are as follows.

— We present 2 formal definition of silver and annealing standard for classification problems.
— We provide evaluation metrics for silver and annealing standard.
— We present a workflow for creating an annealing standard using a sequence or batches of classifiers.

[4] http://www.cs.utexas.edu/users/ml/riddle/data.html.
[5] http://sourceforge.net/projects/febrl/.

— We provide an evaluation on a duplicate detection task with 35 different classifiers showing the convergence of the annealing to the gold standard.
— We give an annealing standard for a dataset comprising 750,000 records of audio CDs.

The article is structured as follows. The next section covers different directions of related work. In Section 3, we define gold, silver, and annealing standard and explain their usefulness for evaluating a classifier. Then, Section 4 describes the workflow to create an annealing standard, and Section 5 evaluates the annealing standard using a real-world scenario. Finally, Section 6 concludes the article and gives an outlook on interesting research directions for the future.

2. RELATED WORK

Five areas are related to our proposal of a classification standard: (i) the area of (database) *benchmarking* in general, (ii) classification *frameworks*, which comprise usually multiple algorithms and datasets and are thus useful to perform benchmarking, (iii) *iterative* approaches for classification, (iv) *ensemble* learning techniques to incorporate results from several classifiers, and (v) duplicate detection *measures*, which evaluate the quality of a duplicate detection result.

Benchmarking. Benchmarks are domain-specific and they should be relevant, portable, scalable, and simple [Gray 1991]. The Transaction Processing Performance Council[6] has published several benchmarks for databases, such as TPC-C and TPC-E, as online transactional processing benchmarks, as well as TPC-H as an ad-hoc, decision support (OLAP) benchmark. For the XML data model, there are benchmarks, such as XOO7 [Bressan et al. 2002], XMark [Schmidt et al. 2002], and XMach [Rahm and Böhme 2002]. Benchmarks usually comprise a dataset or dataset generator, a query workload, and some concrete and objective comparison measures, such as transactions per second (tps), price/tps, or Watts/tps. Because these measures do not depend on the semantics of the generated data or the queries, it is fairly simple to generate appropriate datasets and some corresponding query workload. In addition, the queries follow a well-defined and widely accepted semantics, so the query results are predefined and can be verified with ease.

When creating a benchmark for less well-defined tasks, such as duplicate detection or information retrieval tasks, query results follow a less well-defined semantics. Even among human experts, there is usually some disagreement whether some record pair is in fact a duplicate or whether some webpage is in fact relevant to a search query [Kim and Wilbur 2010]. It is far more costly to create an appropriate dataset, corresponding query results, and expected query results. Each query result must be carefully crafted, preferably double-checked by further human experts. In the domain of information retrieval, the TREC conference and its specific tracks and tasks are well accepted as standard evaluation procedures.[7] For duplicate detection however, there is no such well-accepted benchmark or evaluation set. The proposed annealing standard is a means to fill this gap.

Classification Frameworks. There are various tasks that can be addressed by classification, including spam detection, news article categorization, and part-of-speech tagging. A popular framework for classification in general is Weka [Hall et al. 2009], which offers implementations of the most relevant classification algorithms.

[6] http://www.tpc.org/.
[7] http://trec.nist.gov/.

Since duplicate detection serves as our main target, we discuss frameworks developed specifically for this task in more detail. Köpcke and Rahm have compared different frameworks for entity matching [Köpcke and Rahm 2010]. In their summary, they criticize the frameworks for using different methodologies, measures, and datasets, which makes it difficult to assess the performance of each single system. Furthermore, they mention that the used datasets were mostly quite small, making it impossible to make predictions of the scalability of the approaches. For the future, they see a strong need for standardized benchmarks for entity matching. This observation agrees with Neiling et al. who discuss the properties of an object identification test database and recommend quality criteria [Neiling et al. 2003]. A duplicate detection benchmark for XML (and potentially relational) data is proposed by Weis et al. [2006]. All three papers have in common that they emphasize the necessity of publicly available datasets that can be used for evaluation and thus make the comparison of results possible. Hassanzadeh et al. use the Stringer framework to compare different duplicate-clustering algorithms, and they use generated datasets, because for a thorough evaluation, it is necessary to have datasets for which the actual truth is known [Hassanzadeh et al. 2009]. An annealing standard meets this requirement even for real-world datasets.

Iterative Classification. There is a variety of techniques and systems that manage changes in classification and disagreement among annotators. These systems share traits of the approach presented here. Supervised information retrieval and machine learning algorithms rely on a feedback loop [Salton and Buckley 1997]. The classification result in one stage is evaluated and influences classification in further stages. In the annealing standard, feedback (manual inspection) is also used, but it is not employed to improve further classification (a classifier is assumed as given and fixed) but to increase the quality of the annealing standard itself.

Learn++ is an algorithm that allows the introduction of new classes during classification without the need for *catastrophic forgetting* of the model built up to this point [Polikar et al. 2001]. In the field of ontology annotation, the classification of more and more items from a corpus implies/requires the change of the ontology [Erdmann et al. 2000; Simov et al. 2007]. Some concepts are left out, others are refined, and new subconcepts are introduced. The result is a "hardened" ontology.

In terms of minimizing the (costly) manual effort, Forman proposes incremental retraining after each manual inspection [Forman 2002]. This procedure is hoped to ensure that only the most promising elements are classified. However, in our approach, all classification is already done when it comes to evaluating the results and constructing the annealing standard.

All mentioned contributions have in common that they improve classification efficiency, effectiveness, and capabilities. This article aims to efficiently create a near gold standard that can be used for benchmarking existing classifiers. Of course, a benchmark and gold standard may indirectly improve new classifiers by serving as a training set for humans and computers.

Ensemble Learning. In the case of using several classifiers for a static dataset—in contrast to iterative classification—ensemble techniques combine the classifiers/their models to create a new, improved classifier.

With bootstrap aggregation (bagging), several classification models are trained on different subsets of the data [Breiman 1996]. These models are then combined to create a model that is not prone to overfitting on the dataset. Boosting is a supervised technique to run another classifier on the items misclassified by a former classifier [Freund and Schapire 1996]. In contrast, we let several classifiers run on the entire dataset at

the same time; misclassifications are identified afterwards. RISE is a rule generalization algorithm that takes a set of rules and subsequently merges them as long as these merges do not reduce the overall accuracy [Domingos 1996]. To perform these generalization operations, RISE relies on a training set with correct classification, that is, it is a supervised approach.

We, however, do not aim for manipulating the classifiers or their models. We treat them as black boxes and do not make any assumptions on which algorithm was used; the classifiers might even employ unsupervised methods. In particular, we need not know their precision or recall; boosting is thus not applicable. Instead, we solely operate on the classification result. Consequently, we do not have any models to merge and we cannot rerun the classifiers on subsets of the dataset. Finally, our overall goal is not to build or improve classifiers, but to create a standard to benchmark these classifiers.

Duplicate Detection Evaluation Measures. Christen and Goiser give an overview of quality measures for data linkage [Christen and Goiser 2007]. The measures, for example, *precision*, *recall*, and F_1*-measure* (in the following just called F-measure), are calculated based on classified record pairs that are compared with the real world. Besides the pairwise comparison approach, there is also the cluster-level approach, which uses the similarity of clusters to evaluate duplicate detection results. *Cluster* F_1 (cF_1) is the harmonic mean of cluster precision cP (ratio of completely correct clusters and the total number of retrieved clusters) and cluster recall cR (portion of true clusters retrieved) [Huang et al. 2006]. Another metric is the K *measure*, which is the geometric mean between the *average cluster purity* (i.e., purity of the generated clusters with respect to the reference clusters) and the *average author purity*[8] (i.e., reflects how fragmented the generated clusters are in comparison to the reference clusters) [Cota et al. 2007]. Another measure, proposed by Menestrina et al. [2010], is the *generalized merge distance* (GMD) that can be configured with different cost functions for split and merge steps.

All these measures, regardless of whether they are a pairwise comparison or cluster-level approach, have in common the necessity for a gold standard that defines which records represent same real-world entities.

3. DIFFERENT TYPES OF STANDARDS

In this section, we give an overview of the different standards to evaluate duplicate detection results and additionally define the new annealing standard. The standards differ regarding the completeness and correctness of the duplicates and the required manual effort.

3.1. Gold Standard

In a gold standard, all duplicates are known, and thus, we also know all real-world entities that are only represented by a single record.

Definition 3.1 (*Duplicates*). A duplicate is a pair of distinct records that represent the same real-world entity. All other pairs of distinct records are nonduplicates.

We assume for all duplicates and nonduplicates $\langle r_j, r_k \rangle$ that $j < k$. This serves two purposes: First, we do not want to reward algorithms for finding the tautology $\langle r_j, r_j \rangle$. Second, we do not want to reward algorithms for finding both $\langle r_j, r_k \rangle$ and $\langle r_j, r_k \rangle$. In

[8]In this case, the authors complies with a cluster in the gold standard.

Table I. Acronyms and Abbreviations

Acronym	Meaning
$\langle r_j, r_k \rangle$	A pair of records that might be either a declared duplicate or nonduplicate.
$\mathcal{G}$	A gold standard consists of duplicates $\mathcal{D}_\mathcal{G}$ and nonduplicates $\mathcal{N}_\mathcal{G}$ and is correct and complete.
$\mathcal{S}$	A silver standard is a subset of a gold standard and consists of duplicates $\mathcal{D}_\mathcal{S}$ and nonduplicates $\mathcal{N}_\mathcal{S}$. Therefore, is correct but maybe not complete. In our case, those pairs are manually inspected.
$\mathcal{A}$	An annealing standard consists of undisputed duplicates $\mathcal{D}_\mathcal{A}$ and nonduplicates $\mathcal{N}_\mathcal{A}$ and a silver standard $\mathcal{S}$.
$\mathcal{D}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	All duplicates that are in the gold/silver/annealing standard.
$\mathcal{N}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	All nonduplicates that are in the gold/silver/annealing standard.
$\mathcal{TP}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	A declared duplicate that is correctly classified (according to the gold/silver/annealing standard).
$\mathcal{TN}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	A declared nonduplicate that is correctly classified (according to the gold/silver/annealing standard).
$\mathcal{FN}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	A declared nonduplicate that is actually a duplicate (according to the gold/silver/annealing standard).
$\mathcal{FP}_{\{\mathcal{G}\|\mathcal{S}\|\mathcal{A}\}}$	A declared duplicate that is actually a nonduplicate (according to the gold/silver/annealing standard).

addition, this constraint reduces the size of the gold and silver standards and serves notational simplicity.

Given a set of duplicates, we can calculate the transitive closure to create clusters with records that represent the same real-world entity.

Definition 3.2 (*Cluster*). A cluster c is a set of records $r_j \in \mathcal{R}$ that are pairwise duplicates, that is, all records in c represent the same real-world entity.

With these definitions, a set of records $\mathcal{R}$ can be clustered into a set of disjoint clusters $\mathcal{C} = \{c_1, \ldots, c_m\}$. Note that a cluster resulting from an actual classifier does not necessarily contain all records that represent a particular real-world entity (duplicates might be missing in the set of duplicates). In particular, several separate clusters could contain records that represent the same real-world entity, but the algorithm was unable to find the connecting duplicate relations $\langle r_j, r_k \rangle$ $(r_j \in c_x \neq c_y \ni r_k)$ between them.

We define gold and silver standards using sets of duplicates and nonduplicates. In general, a set $\mathcal{D}$ contains all (known) duplicates, and a set $\mathcal{N}$ contains all (known) nonduplicates. $\mathcal{D}$ and $\mathcal{N}$ are always disjoint and together contain all possible pairs of records in $\mathcal{R}$. Usually, $\mathcal{N}$ is much larger than $\mathcal{D}$.

Definition 3.3 (*Gold Standard*). A gold standard $\mathcal{G}$ for a set $\mathcal{R}$ of records is defined as $\mathcal{G} = \{\mathcal{D}_\mathcal{G}, \mathcal{N}_\mathcal{G}\}$, where the set $\mathcal{D}_\mathcal{G}$ contains all duplicates and the set $\mathcal{N}_\mathcal{G}$ contains all nonduplicates. The sets $\mathcal{D}_\mathcal{G}$ and $\mathcal{N}_\mathcal{G}$ are disjoint, and each pair of records in $\mathcal{R}$ appears in one of the sets.

According to this definition, all duplicates are known and correct. Thus, using the transitivity property of duplicity finds no additional duplicates. As mentioned in Section 2, some evaluation measures require a gold standard that consists of record pairs, and some require clusters. Both representations are equivalent: clusters can be used to generate record pairs, and vice versa record pairs can be used to generate

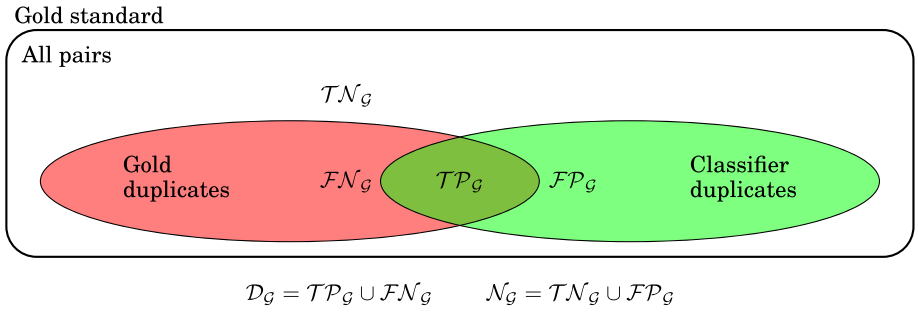


Fig. 1. Evaluation based on a gold standard.

clusters. Thus, it does not matter whether a gold standard is given as sets of record pairs or as sets of clusters. This definition also agrees with Bilenko and Mooney [2003b], who describe a gold standard as a set of equivalence classes, where each equivalence class contains the records of a particular entity and all duplicate records are identified.

Table I shows the acronyms and abbreviations used throughout the article for reference. Some terms are introduced later.

For real-world datasets, a gold standard is created by manually inspecting all possible record pairs. As the complexity for this inspection is quadratic, it is only feasible for smaller datasets. For synthetic data, the duplicates and the gold standard can be generated, often by polluting records. However, this approach raises the problem that two polluted records might be so similar that even domain experts would classify this pair as duplicates although they are not. Thus, the generated gold standard would not be complete. For the evaluation of algorithms that select candidate pairs for comparison, Whang et al. use an exhaustive comparison with a classifier to define a "gold standard" [Whang et al. 2009]. As there is a high probability that some pairs are classified incorrectly, such a "gold standard" does not comply with our definition.

Evaluation with Gold Standard. Having a gold standard, it is possible to measure key figures, such as precision and recall, because we know the duplicates and all of them are correct. *Precision* is defined as the fraction of correctly classified duplicates (true positives, $\mathcal{TP}_\mathcal{G}$) and all classified duplicates (true positives and false positives, $\mathcal{TP}_\mathcal{G} \cup \mathcal{FP}_\mathcal{G}$). Intuitively, precision is a measure for the correctness of the result. *Recall* on the other hand is a measure for the completeness of the result. It is defined as the fraction of correctly classified duplicates and all real duplicates within the gold standard (true positives and false negatives, $\mathcal{TP}_\mathcal{G} \cup \mathcal{FN}_\mathcal{G}$). Thus, we have

$$Precision = \frac{\mathcal{TP}_\mathcal{G}}{\mathcal{TP}_\mathcal{G} \cup \mathcal{FP}_\mathcal{G}}, \tag{1}$$

$$Recall = \frac{\mathcal{TP}_\mathcal{G}}{\mathcal{TP}_\mathcal{G} \cup \mathcal{FN}_\mathcal{G}}. \tag{2}$$

Figure 1 shows the evaluation as a Venn-diagram. Among all pairs, some are known to be duplicates according to the gold standard, and some pairs are declared to be duplicates by a classifier. A gold standard is the best way to evaluate a classifier, because it gives exact numbers for precision and recall and it is also very easy to apply. However, the creation of a gold standard is very costly or even infeasible for larger datasets.

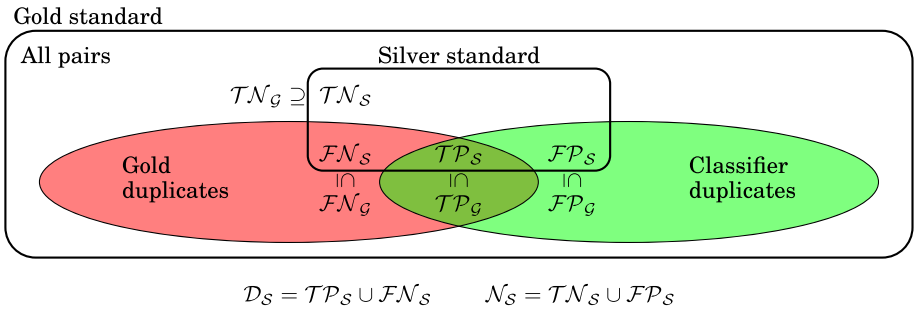


Fig. 2. The silver standard is a subset of the gold standard.

3.2. Silver Standard

A silver standard is a subset of a gold standard. Some duplicates are known and correctly classified, but there might still be additional duplicates that are (yet) unknown. In particular, there might be smaller or fewer clusters of duplicates in a silver standard. Additionally, a silver standard may include correctly classified nonduplicates, which is helpful, for example, for machine learning algorithms that need positive and negative examples.

Definition 3.4 (*Silver Standard*). A silver standard $\mathcal{S}$ for a set $\mathcal{R}$ of records is defined as $\mathcal{S} = \{\mathcal{D}_\mathcal{S}, \mathcal{N}_\mathcal{S}\}$, where $\mathcal{D}_\mathcal{S} \subseteq \mathcal{D}_\mathcal{G}$ and $\mathcal{N}_\mathcal{S} \subseteq \mathcal{N}_\mathcal{G}$.

Hence, a silver standard is correct, but usually not complete. All classified pairs (duplicates and nonduplicates) are in accordance with the gold standard, but for some (most) pairs, a silver standard does not state anything.

A silver standard can be created by a domain expert that manually labels a subset of the record pairs as duplicate or nonduplicate. These pairs can be, for example, randomly sampled or—to find rather hard-to-classify pairs—retrieved by applying any known duplicate detection algorithm to produce a set of candidate pairs. If metadata about the silver standard size in proportion to the expected number of duplicates is available, it is possible to estimate the overall recall of a deduplication process.

Figure 2 shows the relationship between the silver and the gold standard. In absence of a known gold standard, a comparison with a silver standard classifies only a subset of record pairs, because a silver standard is not necessarily complete. If a declared duplicate is within the true duplicates or within the true nonduplicates of the silver standard, then it can be classified as either a true positive ($\mathcal{TP}_\mathcal{S}$) or as a false positive ($\mathcal{FP}_\mathcal{S}$). Vice versa, if a declared nonduplicate is within the true duplicates or within the true nonduplicates of the silver standard, it can be classified to be either a false negative ($\mathcal{FN}_\mathcal{S}$) or true negative ($\mathcal{TN}_\mathcal{S}$). For all declared duplicates and declared nonduplicates that are not within the silver standard, we cannot make a statement whether they are classified correctly. Thus, these record pairs should not be considered to evaluate the duplicate detection results based on this silver standard.

Note that Figure 2 does not state that a silver standard contains false negatives ($\mathcal{FN}_\mathcal{S}$). Instead, some classifier has declared a particular pair as nonduplicate, but it is a duplicate according to the silver standard and thus this pair is a false negative.

Other definitions for a silver standard also exist in the literature: the CALBC initiative [Rebholz-Schuhmann et al. 2010] provides a large-scale biomedical text corpus for tagged named entities. The authors name the corpus itself a silver standard, containing annotations from different automatic annotation systems. The information is added to the silver standard, if at least two annotation systems agree on it, but there is no manual inspection.

Another example is the BioCreative III Gene Normalization task that refers to identifying and linking gene mentions in free text to standard gene database identifiers [Lu and Wilbur 2010]. While the gold standard consists of 50 manually annotated documents, the so-called silver standard comprises 507 documents with automatically detected identifiers. Only identifiers with a probability of at least 50 % were added to the silver standard (no manual inspection). The authors report that the produced results of the task gain better results when evaluated with the silver standard than with the gold standard, but that the relative rankings tend to be largely preserved.

Evaluation with Silver Standard. To evaluate a classifier based only on a silver standard, we try to extrapolate from the silver to the gold standard. We can calculate precision and recall similar to the gold standard if we assume that the distribution of duplicates and nonduplicates in the silver standard is similar to that of the gold standard. Since this assumption does not always hold, we provide a better estimation for differing duplicate distributions in the silver and gold standards. To estimate precision and recall for the silver standard, we need to estimate the following parameters.

— *Overall Amount of Duplicates.* We need an estimation of the assumed amount of duplicates in the complete dataset as the parameter $\pi \approx |\mathcal{D}_\mathcal{G}|$. This parameter can be used to calculate the completeness of the silver standard regarding the amount of duplicates. An estimation needs to take into account knowledge about the creation of the silver standard as well as the overall quality of the dataset.
— *Correctness of Missing (Non)Duplicates.* Since the silver standard may be an arbitrary subset of the overall set of pairs, we cannot infer the correctness of the missing pairs from the silver standard. Thus, we estimate the classifier's correctness regarding missing duplicates with the parameter $\phi_\mathcal{D}$ and the correctness regarding missing nonduplicates with the parameter $\phi_\mathcal{N}$. Usually, we expect $\phi_\mathcal{N}$ to be much higher than $\phi_\mathcal{D}$, since in general, nonduplicates are much more obvious than duplicates. The correctness of the classifier on the silver standard's pairs may be a helpful indicator for estimating $\phi_\mathcal{D}$ and $\phi_\mathcal{N}$.

With these parameters, we can calculate estimated numbers of correctly or wrongly detected duplicates as follows.

$$|\widetilde{\mathcal{TP}}_\mathcal{G}| = |\mathcal{TP}_\mathcal{S}| + \phi_\mathcal{D}(\pi - |\mathcal{D}_\mathcal{S}|), \tag{3}$$

$$|\widetilde{\mathcal{FP}}_\mathcal{G}| = |\mathcal{FP}_\mathcal{S}| + (1 - \phi_\mathcal{N})(|\mathcal{R}| - \pi - |\mathcal{N}_\mathcal{S}|), \tag{4}$$

$$|\widetilde{\mathcal{FN}}_\mathcal{G}| = |\mathcal{FN}_\mathcal{S}| + (1 - \phi_\mathcal{D})(\pi - |\mathcal{D}_\mathcal{S}|). \tag{5}$$

We can use these estimations to calculate precision and recall on the complete dataset using Formulas (1) and (2) in Section 3.1.

While the creation of a smaller silver standard requires less resources than the gold standard, the parameter estimations make the application of the silver standard nontrivial. Thus, in the next section, we describe our novel annealing standard that is inexpensive to create and can be applied almost as easily as a gold standard.

3.3. The New Annealing Standard

In many cases, neither a silver nor a gold standard are available. What is known are the best effort results of a duplicate detection experiment. We call this the baseline. It consists of pairs of records where each pair is declared either as duplicate or as nonduplicate. All other pairs that are not explicitly classified are implicitly nonduplicates. Most likely, precision and recall are not perfect. Yet the idea of the annealing standard is to establish those results as a baseline against which other experiments (different

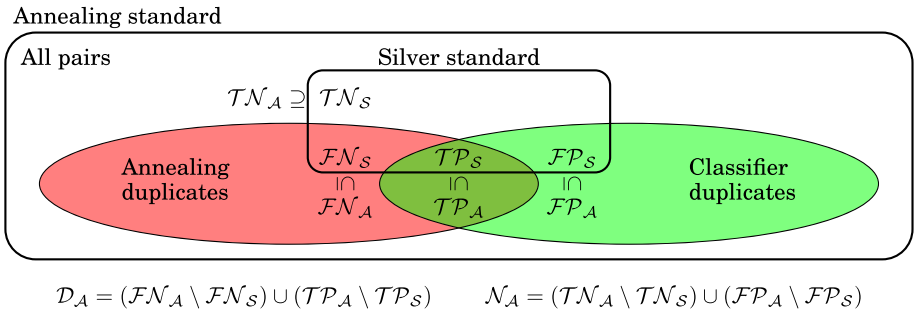


Fig. 3. In absence of a gold standard, the annealing standard takes its role.

algorithm/different similarity-measure) can evaluate and which other experiments can improve upon.

Definition 3.5 (*Annealing Standard*). The annealing standard $\mathcal{A}$ for a set $\mathcal{R}$ of records is defined as $\mathcal{A} = \mathcal{S} \cup \{\mathcal{D}_\mathcal{A}, \mathcal{N}_\mathcal{A}\}$, where $\mathcal{S}$ is the silver standard just defined, $\mathcal{D}_\mathcal{A}$ is a set of potential duplicates, and $\mathcal{N}_\mathcal{A}$ is a set of potential nonduplicates. All four sets $(\mathcal{D}_\mathcal{S}, \mathcal{N}_\mathcal{S}, \mathcal{D}_\mathcal{A}, \mathcal{N}_\mathcal{A})$ of $\mathcal{A}$ are mutually disjoint.

The set $\mathcal{D}_\mathcal{A}$ of potential duplicates contains all pairs that are classified as duplicates within a duplicate detection experiment but have not yet been manually inspected. Vice versa, the set $\mathcal{N}_\mathcal{A}$ of potential nonduplicates contains all pairs that are classified as nonduplicates within a duplicate detection experiment. Pairs that underwent a manual inspection are contained either in $\mathcal{D}_\mathcal{S}$ or $\mathcal{N}_\mathcal{S}$ if the expert labeled them as duplicates or as nonduplicates, respectively. Figure 3 shows the role of the annealing standard as a replacement of the gold standard.

Example. Let a dataset $\mathcal{R} = \{a,b,c,d,e,f,g,h\}$ and two classification results with the declared duplicates $\{\langle a,b\rangle, \langle c,d\rangle\}$ and $\{\langle a,b\rangle, \langle e,f\rangle\}$, where manual inspection reveals that $\langle c,d\rangle$ is a duplicate and $\langle e,f\rangle$ is a nonduplicate. The pair $\langle a,b\rangle$ is undisputed among the two classifiers and thus located in $\mathcal{D}_\mathcal{A}$. $\langle c,d\rangle$ is member of $\mathcal{D}_\mathcal{S}$ and $\langle e,f\rangle$ is contained in $\mathcal{N}_\mathcal{S}$. See Section 4 for an extended explanation of this example, including the devised workflow.

Note that $\mathcal{D}_\mathcal{A}$ is transitively closed, because it is directly derived from the undisputed decisions within the (transitively closed) classification results. $\mathcal{D}_\mathcal{S}$ is not transitively closed, because it contains only genuinely manually inspected pairs. With respect to the files we provide (see Section 5.4) we leave it to the user to create transitive closures and to tag inferred edges. All inferred edges have then neither been manually inspected nor did all classifiers agree on them being duplicates.

Evaluation with Annealing Standard. To calculate precision and recall with the annealing standard, we assume that the sets $\mathcal{D}_\mathcal{A}$ and $\mathcal{N}_\mathcal{A}$ contain correctly classified duplicates and nonduplicates, respectively. Thus, we can use the Formulas 1 and 2 in Section 3.1 using the following estimations, which estimate not only the size of the three estimates, but also their prospective contents.

$$\widetilde{\mathcal{TP}}_G = \mathcal{TP}_\mathcal{S} \cup \mathcal{TP}_\mathcal{A}, \tag{6}$$

$$\widetilde{\mathcal{FP}}_G = \mathcal{FP}_\mathcal{S} \cup \mathcal{FP}_\mathcal{A}, \tag{7}$$

$$\widetilde{\mathcal{FN}}_G = \mathcal{FN}_\mathcal{S} \cup \mathcal{FN}_\mathcal{A}. \tag{8}$$

Table II. Classified Record Pairs

ID1	ID2	Duplicate	Version
1	2	True	1
7	10	False	2
...	...	...	...

Table III. Data Model Metadata

Version	Date	Author	#Added pairs	#Inspected pairs	#Changed pairs
1	2013-05-30	John	100	0	0
2	2014-05-10	Peter	30	50	15
...	...	...	...	...	...

Data Model. An annealing standard is incrementally improved in the course of time. With each new classification result, a new version of the annealing standard is created. The differences between the previous annealing standard and the results have to be inspected manually. Thus, each version is an improvement of the previous one until all possible pairs are inspected manually. In this case, the annealing standard has been converted into a gold standard. To make differences between different annealing standard traceable, all pairs in the annealing standard are tagged with a version label, indicating the annealing standard version of the last change for this pair.

Tables II and III describe the data model of the annealing standard. Table II shows the classified record pairs with {*id1*, *id2*} as primary key. Additionally, we need a constraint $id1 < id2$ to ensure that a record pair is not inserted twice with swapped IDs. All record pairs are classified as duplicate or nonduplicate, and the attribute *version* contains information when a record pair was inserted or its duplicity information was updated the last time.

In the beginning, we have a probably high number of nonduplicates that have not yet been inspected, and thus to save storage space, we do not save the pairs $\mathcal{N}_\mathcal{A}$ explicitly. All record pairs with $version = 1$ are potential duplicates of the baseline classifier. In the following iterations, the annealing standard is refined (see Section 4). In each iteration, the differences between the current classification result and the baseline (all inserted records and updated records) are manually checked. Thus, all records with $version \geq 2$ are the silver standard, with $duplicate = true$ for $\mathcal{D}_\mathcal{S}$ and $duplicate = false$ for $\mathcal{N}_\mathcal{S}$. As mentioned before, the potential duplicates $\mathcal{D}_\mathcal{A}$ are all records with $version = 1 \wedge duplicate = true$, and the potential nonduplicates $\mathcal{N}_\mathcal{A}$ are all not included record pairs.

Next to the table with the record pairs, there is optionally also a table with metadata (see Table III). This table contains the change history of an annealing standard, with the creation date and the responsible person for each version. Furthermore, it contains the number of explicitly added record pairs, the number of manually inspected record pairs, and the number of changed record pairs (only for changes in a pair's duplicity, not in its version). This metadata helps to explain differences between duplicate detection experiments conducted with different versions of the same annealing standard. In the example (Table III), version 2 could have been created by merging in a 60-pair classification result (already transitively closed), where 10 pairs confirm the current annealing standard. The remaining 50 pairs create conflicts (are inspected), from which 30 pairs are new (added) and 20 pairs are already known, but with the opposite duplicity statement. From these 20 pairs, 15 pairs were correct in the classification result (according to the manual inspection and in contradiction to the previous annealing standard) and hence, their duplicity was changed.

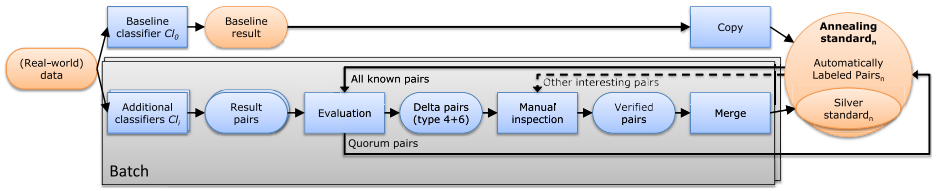


Fig. 4. Workflow to create silver and annealing standards.

4. WORKFLOW FOR THE ANNEALING STANDARD

Figure 4 shows the proposed workflow for the creation and maintenance of the annealing standard. Given a dataset, preferably from a real-world setting, a baseline classifier Cl_0 creates an initial set of duplicate pairs. Of course, this result set may harbor false positives and false negatives; nevertheless after copying its transitive closure it constitutes the initial annealing standard $\mathcal{A}_0$.

Following the idea of annealing, results from additional classifiers are added either sequentially or in batches to improve the current standard. Each new classifier Cl_i, which can be a different configuration of a previous classifier or an entirely new classifier, produces a new set of result pairs (see lower path of Figure 4). The transitive closure is created for these pairs, too.

4.1. Incorporating New Results into the Annealing Standard

For now we focus on sequential addition of classification results and discuss batches in Section 4.3. The declared duplicate pairs $\langle r_j, r_k \rangle$ can be distinguished into four types with respect to their membership in different parts of the annealing standard $\mathcal{A}_{i-1}$. Definition 3.5 defines an annealing standard as $\mathcal{A} = \{\mathcal{D}_\mathcal{S}, \mathcal{N}_\mathcal{S}, \mathcal{D}_\mathcal{A}, \mathcal{N}_\mathcal{A}\}$.

(1) $\langle r_j, r_k \rangle \in \mathcal{D}_\mathcal{S}$. These pairs are certain true positives; their duplicity has been manually confirmed in the past.
(2) $\langle r_j, r_k \rangle \in \mathcal{D}_\mathcal{A}$. These pairs are probable true positives; they serve as further confirmation that they in fact are duplicates, but a manual check has not been performed.
(3) $\langle r_j, r_k \rangle \in \mathcal{N}_\mathcal{S}$. These pairs are certain false positives; they are clear errors, because their nonduplicate status has been manually confirmed in the past.
(4) $\langle r_j, r_k \rangle \in \mathcal{N}_\mathcal{A}$. These pairs are probably false positives; no previous classifier has yet declared this pair to be a duplicate.

The same distinction can be made for pairs that were not declared to be duplicates by Cl_i, that is, were declared nonduplicates.

(5) $\langle r_j, r_k \rangle \in \mathcal{D}_\mathcal{S}$. These pairs are certain false negatives; they are clear errors and should have been declared as duplicates by classifier Cl_i.
(6) $\langle r_j, r_k \rangle \in \mathcal{D}_\mathcal{A}$. These pairs are probable false negatives; all previous classifiers have declared this pair to be a duplicate.
(7) $\langle r_j, r_k \rangle \in \mathcal{N}_\mathcal{S}$. These pairs are certain true negatives; their nonduplicity has been manually confirmed in the past.
(8) $\langle r_j, r_k \rangle \in \mathcal{N}_\mathcal{A}$. These pairs are probable true negatives; they serve as further confirmation that they in fact are not duplicates, but a manual check has not been performed.

Pairs of types 1, 2, 7, and 8 can be ignored for now. Either they have been manually verified as being correct (1 and 7) or as more and more classifiers are tested against the annealing standard, the certainty of their correctness increases (2 and 8). All other

pairs (3, 4, 5, 6) represent a *conflict* between the annealing standard so far and the last classifier. Pairs of type 3 and 5 are certain classifications that reside in the silver standard. They can also be ignored for now—they constitute certain errors of the classifier. Finally, pairs of types 4 and 6 constitute supposed errors, labeled as "Delta pairs" in Figure 4, and shall be manually inspected. These are the pairs that contradict previous automated classifications and that have not yet been manually checked. The expectation is that for a good current annealing standard and a good classifier the amount of work for manual inspection is reasonable.

Any pair from $\mathcal{D}_\mathcal{A}$ or $\mathcal{N}_\mathcal{A}$ that is manually inspected and classified as duplicate or nonduplicate is "promoted" to the silver standard, that is, either to $\mathcal{D}_\mathcal{S}$ or $\mathcal{N}_\mathcal{S}$ depending on the expert decision. The result of this process is a new annealing standard $\mathcal{A}_i$, which typically contains a slightly expanded silver standard. The result of Cl_i is finally evaluated against $\mathcal{A}_i$ in terms of precision, recall, and other measures.

As more and more experiments are performed, the set of manually inspected pairs grows. It is the nature of this workflow that precisely the difficult-to-classify pairs are those that at some point undergo a manual inspection. Those pairs that are never manually inspected but survive their initial classification, whether as duplicates or as nonduplicates, even after many experiments can be considered stable. In the worst case, all pairs are manually inspected at some point, for instance when two classifiers label every pair complementarily.

Continued Example from Section 3.3. We can now discuss a potential workflow that leads to the creation of the exemplary annealing standard of Section 3.3. Let a dataset $\mathcal{R} = \{a, b, c, d, e, f, g, h\}$. In total, there are $\frac{|\mathcal{R}|\cdot(|\mathcal{R}|-1)}{2} = 28$ pairs, each of them ending up in one of the four sets ($\mathcal{D}_\mathcal{S}$, $\mathcal{N}_\mathcal{S}$, $\mathcal{D}_\mathcal{A}$, or $\mathcal{N}_\mathcal{A}$) at the end of the workflow.

A first classifier declares $\langle a, b\rangle$ and $\langle c, d\rangle$ as duplicates. This is the baseline, and since there cannot be any disputes at this point, both pairs are inserted into the duplicates of the annealing standard $\mathcal{D}_\mathcal{A}$. All the other 26 possible pairs, for example, $\langle e, f\rangle$, are (implicitly) declared nonduplicates and reside in the nonduplicates of the annealing standard $\mathcal{N}_\mathcal{A}$ until further review is performed.

Subsequently, another classifier declares $\langle a, b\rangle$ and $\langle e, f\rangle$ as duplicates. While $\langle a, b\rangle$ is confirmed (regarding the current annealing standard) and remains in $\mathcal{D}_\mathcal{A}$, $\langle c, d\rangle$ and $\langle e, f\rangle$ are not supported by all (two) classifiers and undergo a manual inspection. In this example, manual inspection of both pairs reveals that $\langle c, d\rangle$ is actually a duplicate and $\langle e, f\rangle$ is actually a nonduplicate. Thus, $\langle c, d\rangle$ is "promoted" from $\mathcal{D}_\mathcal{A}$ to $\mathcal{D}_\mathcal{S}$, because its duplicity has just been confirmed. In contrast, $\langle e, f\rangle$ is moved to $\mathcal{N}_\mathcal{S}$.

Finally, $\mathcal{D}_\mathcal{A}$, $\mathcal{D}_\mathcal{S}$, and $\mathcal{N}_\mathcal{S}$ contain one pair each, whereas all the other 25 pairs are in $\mathcal{N}_\mathcal{A}$. In total, only two manual inspections were performed instead of 28.

4.2. Convergence and Manual Inspections

With each new experiment, the annealing standard converges to a gold standard, in the meantime providing an ever-growing silver standard. Solely pairs that are so difficult to classify that no classifier has yet performed correctly remain as errors in the annealing standard. Please note that each classifier result has to be transitively closed before further processing (as previously described).

The manual inspection of duplicates is a key step to creating the annealing standard, in particular because we assume manual classifications to always be correct. Thus, only experts should perform this classification or at least clear instructions should help the users in their classification. In general, manual classification is performed only for the cases with differing classifier results (to reduce manual effort), but it can

also be performed when the classifiers agree, but the objects still seem interesting or relevant ("other interesting pairs" in Figure 4).

In some cases, manual decisions may differ depending on the interviewed expert. For example, a news article on economic crisis may be considered as politics article or as business article; two records from a person table with differing family names can be regarded as duplicate or nonduplicate—both with good reasons. To resolve these problems, two- or more-fold validation can be employed. There are elaborate approaches to determine the needed number of classifiers and how to combine manual decisions [Sheng et al. 2008]. In this article, we consider the manual decision process as black box, that is, it is irrelevant how many manual classifiers have been employed and how the decision process works. We consider only the decision at the end of this process and store it in the silver standard part of the annealing standard (see also Figure 3 and Table II). Similarly, we treat both the first classifier as well as the consecutive classifiers as black boxes and process their results only.

4.3. Saving Manual Work with Quorums and Batches

The basic workflow can be extended by adding pairs meeting a given *duplicate quorum* and *nonduplicate quorum* directly to the annealing standard to defer and possibly save some manual inspections. Each automatically declared duplicate is further annotated by the number of classifiers that agree on the declaration. If the duplicate quorum is met, the pair is considered a duplicate, even if not all classifiers agree. Analogously, if a pair is not labeled as duplicate by enough classifiers to meet the nonduplicate quorum, the pair is considered as nonduplicate.

The quorum can be absolute or relative and is trivially met for nonconflicting pairs. Obviously, a pair that meets a quorum at any given time, may later still require manual inspection. For example, consider a duplicate quorum of 80 %, where it is sufficient that four out of five classifiers label a pair as duplicate. During the integration of the first four classifiers, this quorum can be met only trivially if all classifiers agree. When integrating the fifth classifier, no manual assessments for duplicate candidates are necessary, because either a pair has already been manually assessed before or all four previous classifier agreed. Thus, independent of whether the fifth classifier has declared the pair as a duplicate or not, it is still considered a duplicate without further manual inspection. Nevertheless, the addition of the sixth classifier might further decrease the support of the declared duplicate, so that only four out of six classifiers agree and thus trigger a manual inspection.

Up to this point, we only considered the sequential addition of new classification results. However, when multiple new classification results are to be integrated at once, new opportunities to reduce the manual effort arise. In a batch, the results of the classifiers and the current annealing standard are first merged to count each declared duplicate and then the quorums are applied. In contrast to the sequential addition, we can defer manual inspections from pairs of all new classification results likewise and not only from the last classification result. For instance, if we use a nonduplicate quorum of 80 % and integrate the first five classification results in a batch, we do not need to manually assess any pair that was found by only one classifier. As can be seen in the evaluation in Section 5.3 especially, the nonduplicate quorum helps greatly to reduce the number of manual inspections.

5. IMPLEMENTATION AND EVALUATION

In this section, we evaluate our method with respect to two important questions: (1) How well do evaluation results against the annealing standard converge to results against the gold standard, that is, is an annealing standard a suitable substitute for a gold standard? (2) How expensive is it to create a good annealing standard, that is,

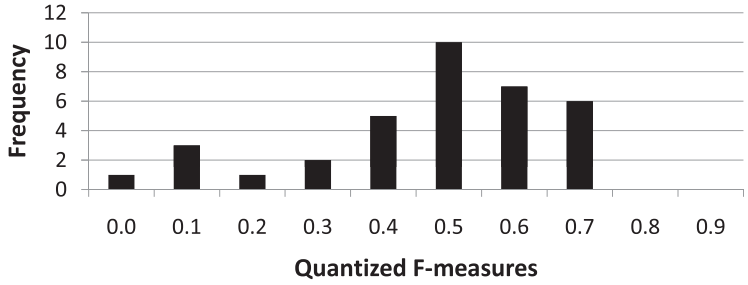


Fig. 5. Histogram of the quantized F-measures of the 35 different duplicate detection classifiers (truncated to one decimal place).

how many manual classifications are needed? Section 5.1 describes the overall experimental setup, the used dataset, and explains how our annealing standard was created. The experimental results to answer the two questions are shown and interpreted in Section 5.2. Section 5.3 reveals the potential to save manual inspections when using quorums and finally Section 5.4 describes the creation of an annealing standard for a real-world dataset.

5.1. Data and Settings

To evaluate the idea of growing an annealing standard and creating a silver standard as a by-product, we use a customer dataset. It contains about 1 million address records with 12 attributes. The data was artificially polluted with duplicates by a large industry partner who uses this dataset as internal duplicate detection benchmark. This gives us reason to believe that the degree and form of those duplicates is realistic. The dataset contains about 90,000 pairwise duplicates. The gold standard is known, so our "manual inspection" was in fact a look-up in the gold standard.

Over the past few years, this dataset was used several times for a three-day data cleansing and duplicate detection workshop with different student teams. The task of the student teams was to competitively find duplicates within the dataset. Using the gold standard, precision, recall, and F-measure were calculated and compared among the different teams. We ran this workshop several times, yielding 35 classification results in total. The results are very precision-oriented in general with an average precision of 83 % and an average recall of only 40 %. The resulting average F-measure is 52 % and the best F-measure is 76 % (see Figure 5 for the distribution). Thus, the quality of the classifiers are below typical classification results in the duplicate detection area. Nevertheless, we believe they are sufficient to evaluate the feasibility and usefulness of the annealing standard.

We use these 35 independently created results as our classifiers. Since in real-life the order of the duplicate detection runs is unpredictable with regard to the monotonicity of the F-measure, we used a random order of the 35 classifiers. Note that the order of the classification results does not influence the number of conflicts, but just the behavior of the F-measure. To bypass the effects of accidentally selecting a poor order, we took 1,000 distinct random permutations of the 35 classifiers and present the average in the following figures and descriptions. The following paragraphs distinguish individual classifiers—in reality these are the averages over the 1,000 permutations. Referring to Figure 4, we consider the first classification result as the baseline and the 34 following as additional classification results.

There are several different metrics for evaluating such a process for creating an annealing standard. The *precision* and *recall* of the annealing standard compared to

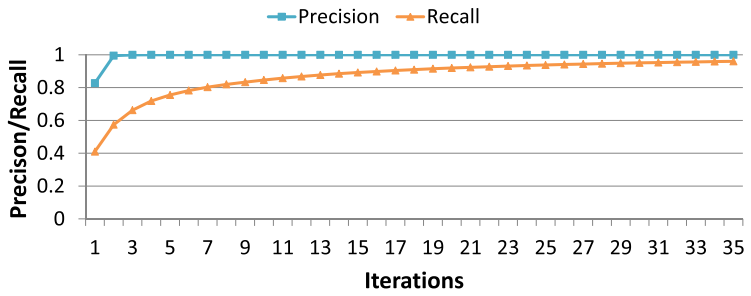


Fig. 6. Precision and recall in the annealing standard.

the gold standard describe whether and how the annealing standard evolves towards the gold standard over time.

The number of manually inspected pairs determines the amount of manual effort, directly derived from the size of the delta between the current result and the annealing standard so far (i.e., how many pairs have to be manually inspected). We also show the silver and annealing standards with regard to their respective number of duplicates and nonduplicates.

5.2. Evaluation Results

Convergence of Precision and Recall. Figure 6 shows that both precision and recall of the annealing standard converge. In the second iteration, the precision already achieves a value of nearly 1.0: all classified duplicates are true duplicates with regard to the gold standard. After the first iteration, the annealing standard's precision is necessarily the precision of the baseline classification result. The figure further shows that any combination of two classification results is enough to make the annealing standard's precision nearly perfect.

This rapid convergence comes with the price of a relatively high number of pairs that have to be manually inspected as described later. In scenarios where only precision needs to be evaluated and where the annealing standard is created with precision-oriented classifiers, a few iterations suffice.

The recall continuously grows much slower and does not reach a level of 1.0 within the 35 iterations. This is because the particular classifiers were all quite conservative and found (over all classification results) only 86,000 of the 90,000 duplicates in the gold standard. Obviously, the missing 4,000 duplicates are especially hard to find, not a single classifier succeeded.

Figure 7 shows the absolute number of pairs contained in the annealing standard classified as true/false positives as well as true/false negatives with regard to the gold standard. The baseline (the first iteration) is successively improved towards the gold standard with more and more manually verified pairs and a decreasing delta. The last bar in the figure shows the convergence's target: the gold standard.

Furthermore, the growth of the recall in Figure 6 corresponds to the growth of the number of true positives and the reduction of the false negatives in Figure 7. Precision in Figure 6 reaches 1.0, as soon as all false positives (red) are removed after the second iteration.

Number of Manually Inspected Pairs. Figure 8 shows the delta size for the iterations, representing the number of manually inspected pairs per iteration in linear and in logarithmic scale, separated in pairs that would be manually classified as duplicates and nonduplicates. In the first iterations, the classifiers find different duplicates, causing a large number of manual inspections. After a few iterations, no new duplicates

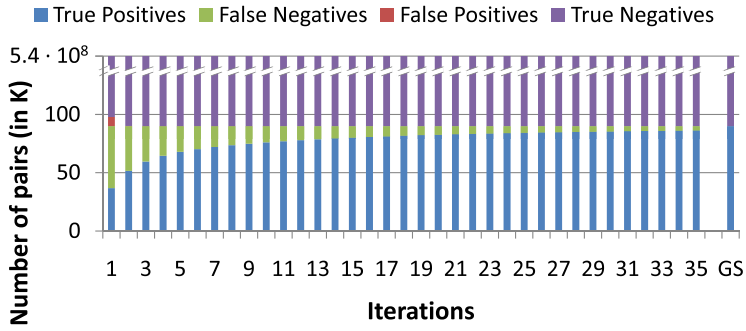
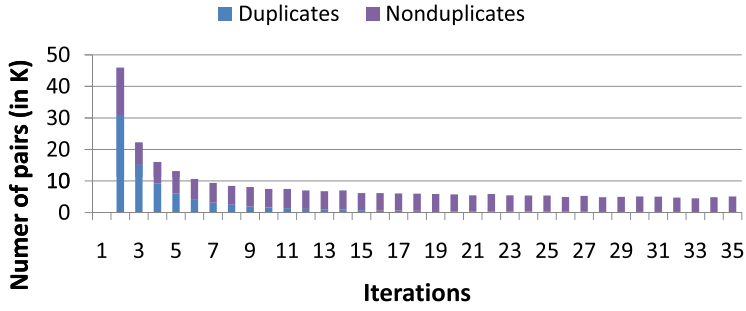
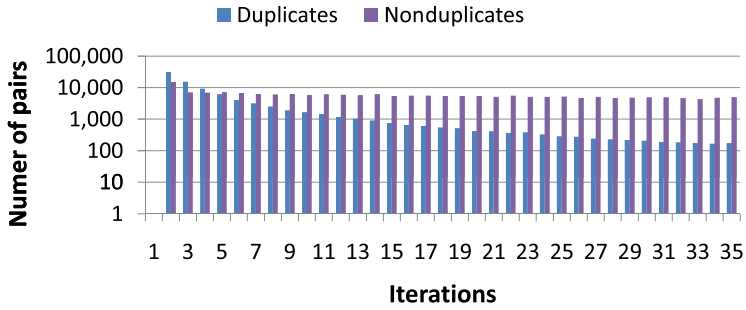


Fig. 7. Absolute numbers of pairs in the annealing standard with regard to the gold standard. The last bar shows the gold standard.



(a) Delta size in linear scale.



(b) Delta size in logarithmic scale.

Fig. 8. Delta sizes in linear/logarithmic scale.

are declared, but the amount of nonduplicates remains large, compared to the number of duplicates. Every classifier generates a delta of at least 4,500 new pairs that have never been manually inspected before. The ratio of nonduplicates to duplicates is strongly skewed towards the nonduplicates over time.

The absolute number of necessary manual inspections is quite high for this experiment: The first two classifiers disagree on about 45,000 pairs (on average); over the course of the experiment altogether 283,000 manual inspections were needed. There are two reasons for these large numbers. First, note that the set of classifiers are the result of a three-day workshop with students—not those of experienced research or industry teams. Second, the number is dwarfed by the overall number of candidate pairs, which is $\frac{n \cdot (n-1)}{2} \approx 5.4 \cdot 10^{11}$.

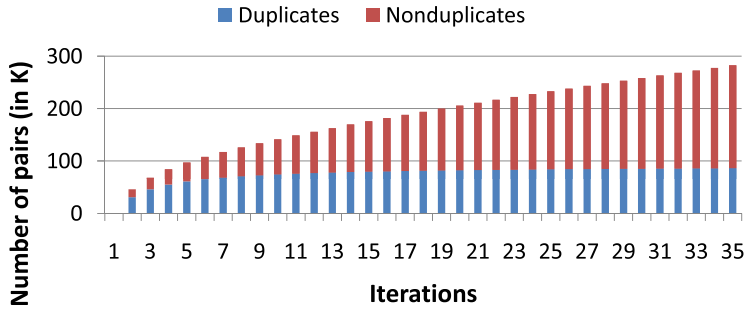


Fig. 9. Size of the silver standard over the iterations.

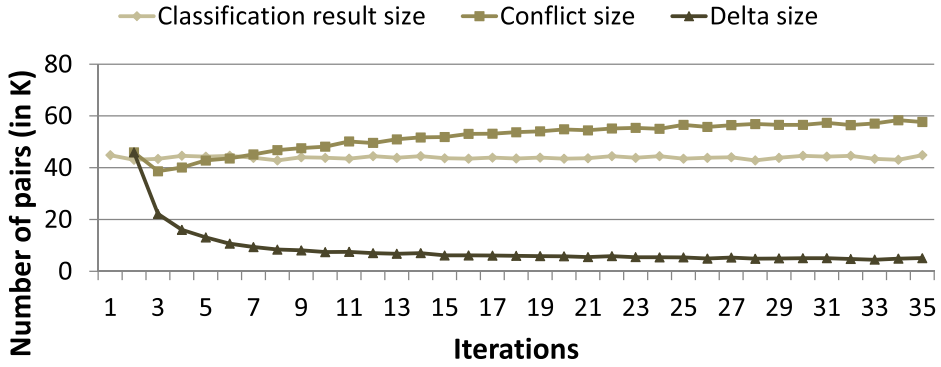


Fig. 10. Number of pairs in the classification result vs. conflict vs. delta.

We reran the evaluation process with the ten best classification results in terms of precision. The number of manual inspections significantly decreases up to 50% for classification results with a high precision. Nevertheless, the second iteration still required 30,000 inspections on average, because the classifiers with a high precision are mostly quite conservative with a small recall and found very different pairs. However, the number of nonduplicates that need to be inspected in each iteration is 3 to 4 times lower compared to the evaluation with all classifiers.

The silver standard is an accumulation of the manually inspected pairs (i.e., the delta) and thus, grows monotonically (Figure 9). It continuously grows while the number of found true duplicates does not change much (Figure 8). Thus, after a while mostly nonduplicates are inserted into the silver standard and one could stop the iterations earlier and save manual inspections.

Figure 10 shows the size of the conflicts (the difference between the current classifier and the annealing standard so far) against the size of the deltas. The size of the classification results is also included for comparison. The set of pairs in the delta is a subset of the set of pairs in the conflict. Due to the random order and their independence, the classification result size fluctuates and no trend can be observed.

The number of conflicting pairs slowly increases, because the silver standard incorporates more and more hard-to-classify pairs over time. These pairs are misclassified by most of the classifiers and can only be detected as soon as one classifier decides correctly and the manual inspection confirms the new classification. This manually confirmed classification improves the quality of the silver standard: from now on, this common misclassification is detected and thus, the conflict size of the following classifiers is increased.

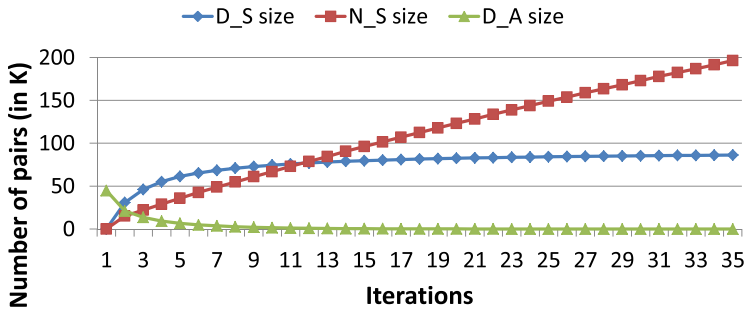


Fig. 11. Number of duplicates and nonduplicates for silver standard and number of duplicates for annealing standard.

Since the silver standard is empty in the beginning, the conflict size equals the delta size in the second iteration. Beginning with the third iteration, only misclassifications of the new classifier and misclassifications that all previous classifiers have done, are in the set of conflicts.

The delta is smaller, since it does not comprise those pairs (type 3 and 5) that contradict with the silver standard but only those that contradict the baseline prediction (type 4 and 6) and have to be classified manually, subsequently.

Figure 11 shows the changes of the sizes of the three sets $\mathcal{D}_\mathcal{S}$, $\mathcal{N}_\mathcal{S}$, and $\mathcal{D}_\mathcal{A}$. The size of $\mathcal{D}_\mathcal{S}$ and $\mathcal{N}_\mathcal{S}$ after the first iteration is zero, because at this point no pairs can have been manually checked.

The number of duplicates in the annealing standard ($\mathcal{D}_\mathcal{A}$) starts with the number of duplicates declared by the baseline classifier. Consecutively, some of these decisions are revoked by further classifiers and thus, pairs move into the silver standard ($\mathcal{D}_\mathcal{S}$ or $\mathcal{N}_\mathcal{S}$). Only a few pairs in $\mathcal{D}_\mathcal{A}$ survive all iterations and are never questioned. These duplicates seem to be found very easily. Note that no statement is made about whether those pairs actually are duplicates.

The nonduplicates in the annealing standard ($\mathcal{N}_\mathcal{A}$) initially start with about 540 billion, but some pairs are actually duplicates or different classifiers disagree upon their correct classification. They are consequently removed from $\mathcal{N}_\mathcal{A}$ and fed into $\mathcal{D}_\mathcal{S}$ or $\mathcal{N}_\mathcal{S}$. Nevertheless, the size of $\mathcal{N}_\mathcal{A}$ remains almost constant in respect to its size and would be out of range of Figure 11 and is therefore omitted.

As a large portion of duplicates according to the gold standard are found, $\mathcal{D}_\mathcal{S}$ converges against the total number of duplicates. $\mathcal{N}_\mathcal{S}$ also steadily increases but achieves a larger momentum than $\mathcal{D}_\mathcal{S}$ in the end as the deltas of the classification results contain more and more nonduplicates (Figure 8).

As a conclusion, an annealing standard can be created, but the manual effort is still large, because even a single poor classifier can boost the amount of manual work. In the following, we alleviated such effects with quorums.

5.3. Quorums and Batch Updates

In an additional experiment, we first examined how many classifiers declared the same duplicates and how many of these duplicates are indeed true positives. Second, we evaluated how duplicate and nonduplicate quorums (see Section 4.3) impact the amount of manual work needed.

Figure 12 shows that most declared duplicates have very little support: 57 % of the overall 283,000 declared duplicates have been found by only one classifier. Similarly, most of the remaining declared duplicates were found by two and three classifiers.

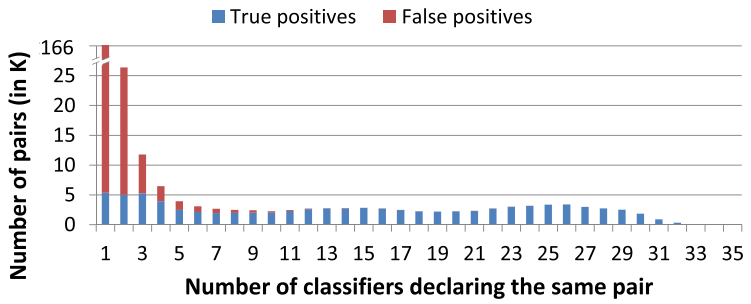


Fig. 12. Number of duplicate declarations for the same pair and classification correctness.

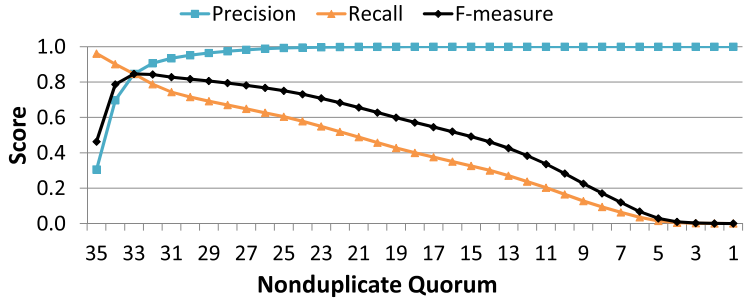


Fig. 13. Effect of nonduplicate quorum on recall, precision, and F-measure.

Additionally, there are only ≈5,000 real duplicates among these seldom found declared duplicates; most of them were false positives.

Consequently, with a nonduplicate quorum, we can greatly reduce the number of comparisons. A quorum of 35 represents the baseline and results in 283,000 comparisons. By reducing the quorum by 1 to 34, we already save 166,000 comparisons (57 %). A quorum of 33 yields in a total of 90,000 and a quorum of 32 in 78,000 comparisons. The majority of these comparisons are false positives; however, we also lose roughly 5,000 true positives for each step that we decrease the quorum.

We thus measured the impact of absolute nonduplicate quorums on the overall quality of the annealing standard in Figure 13. Obviously, the recall monotonously drops if we increase the nonduplicate quorum, because fewer pairs are even considered to be duplicates altogether. However, at the same time, fewer false positives need to be manually assessed and corrected. Thus, depending on the domain, recall and precision need to be traded, which we did with the well-established F-measure.

A nonduplicate quorum of 35 represents the baseline: the declaration of one classifier is already enough for a pair to be a potential duplicate causing manual assessment. However, only 33 % of the declared duplicates are true positives with a recall of 96 %. A nonduplicate quorum of 34 would ignore all declared duplicates with a support of one and significantly improve the precision to 71 % and yield a recall of 90 %. Using F-measure, the sweet spot for this dataset and the given classifiers is a nonduplicate quorum of 2 (precision 85 %, recall 84 %).

Similarly, a duplicate quorum could help save some manual work. All 18,072 pairs that were labeled by at least 26 classifiers as duplicates have indeed been true positives. Even if only 17 of the 35 classifiers agreed, only 30 of the 41,487 declared positives were false, yielding a high precision of 99.93 %.

Finally, this mechanism could also be employed to tell poor classifiers apart: For each pair that bypasses the manually inspection due to only few disagreeing classifiers, we can note the classifiers that disagree and eventually identify poor-performing classifiers.

5.4. Real-World Annealing Standard

We ran the annealing standard workflow in a real-world setting using a larger sample of CD information from the freeDB project (see Section 1.1): 750,000 CD entries comprising information about artist, title, genre, release year, track lists, etc. Four of the authors each developed a classifier to identify duplicates.

Together, the classifiers found about 134,000 duplicate clusters with 366,000 nodes. Next to the manual inspections to decide on disputed edges, we additionally manually falsified the clusters that contained "unknown artist" or "unknown title" CDs, making use of the "other interesting pairs" feature in Figure 4. In total, we performed 1,648 manual inspections. Eventually, we present a consistent, reasonably-sized set of files: an annealing standard (containing all agreed pairs), a silver standard (containing all manually inspected pairs), the dataset, and the four classifications. You can find them on our webpage at `http://www.hpi.uni-potsdam.de/naumann/projekte/annealing_standard.html`.

6. CONCLUSIONS & OUTLOOK

With the proposed annealing standard, we provide an approach for creating a valuable, high quality standard for even large datasets that can be used as a classification benchmark. We have discussed and experimentally evaluated this approach for the duplicate detection domain.

With each new evaluated classifier, a new version of the annealing standard is created. Thus, it is not possible to compare results like precision, recall, and F-measure with classifiers that used a previous version of the annealing standard. To use an annealing standard as a benchmark dataset, it has to be frozen at some point in time. As we could see from the experimental evaluation, the annealing standard is highly developed after a certain number of iterations. Freezing an annealing standard does not mean that there is no more manual inspection necessary. While the frozen annealing standard can be used as benchmarking dataset, the annealing standard can be further improved to obtain a better benchmarking dataset at a later point in time.

A critical step in the creation of an annealing standard is the (black-box) manual inspection, which faces two challenges.

Quality of Manual Inspection. Although manual inspection should be conducted by a domain expert, there is still the chance of an incorrectly inspected pair. As the inspected pairs are part of the silver standard and thus no longer part of the delta, they will not be checked again and might indicate incorrect results for future classifiers. This is not only a problem of the annealing or silver standard, but concerns also existing gold standards. A solution might be the requirement that within each iteration the delta has to be inspected by more than one domain expert. This helps to increase the confidence in the inspected pairs. A second solution is to resubmit pairs for manual inspection, if after a manual inspection several classifiers return a contrary result. A straight-forward implementation would treat human inspections as an additional classifier with higher weight and form a feedback loop that resubmits pairs if they do not meet the duplicate and nonduplicate quorums.

Workload for Manual Inspection. Especially for the first iterations, a relatively high number of manual inspections is necessary. For this task crowd-sourcing services,

such as Amazon Mechanical Turk,[9] are an alternative for reducing the required time [Paolacci et al. 2010; Welinder et al. 2010]. This has already been evaluated for annotation tasks, but raises again the question of how trustworthy the results are [Snow et al. 2008]. A high workload is expected if the results of a poor quality classifier are evaluated with the annealing standard. To allow only useful classifiers to contribute to the annealing standard, there could be a restriction that only classifiers with a score $> 95\%$ F-measure with the silver standard (or some other a-priori knowledge of their quality) are accepted to avoid unworthy manual inspections. Finally, a more sophisticated approach beyond our current quorum-technique (possibly weighted by dynamically determined classifier quality) could further reduce the number of manual inspections.

An interesting direction of future research would be to merge the annealing standard with active learning methods. Both approaches aim to reduce manual effort. A unified approach would require manual classification decisions for both objects that were classified with low confidence (active learning) and objects that were classified differently by at least two classifiers (annealing standard). We expect the labeled data to be of higher quality for both training and evaluating classifiers. Note that while the annealing standard is a black-box approach regarding the classifiers, active learning (and thus a unified approach, too) depends on internals of the classifiers to determine which objects to label next.

For the future, we also plan to evaluate our approach with more real-world datasets and for other classification domains, such as classifying spam emails. Another research topic is the determination of parameters $\phi_{\mathcal{D}}$ and $\phi_{\mathcal{N}}$ as described in Section 3.2, to estimate the correctness of missing duplicates and missing nonduplicates within the silver standard. To reduce the workload for the manual inspection for classifier developer, we are planning to evaluate crowd-sourcing possibilities and configurations. Finally, we would like to provide an annealing standard system that allows the administration of different annealing standard versions for different datasets or corpora.

REFERENCES

Rohan Baxter, Peter Christen, and Tim Churches. 2003. A comparison of fast blocking methods for record linkage. In *Proceedings of the KDD Workshop on Data Cleaning, Record Linkage and Object Consolidation*.

Mikhail Bilenko and Raymond J. Mooney. 2003a. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the ACM International Conference of Knowledge Discovery and Data Mining (SIGKDD)*.

Mikhail Bilenko and Raymond J. Mooney. 2003b. On evaluation and training-set construction for duplicate detection. In *Proceedings of the KDD Workshop on Data Cleaning, Record Linkage and Object Consolidation*.

Leo Breiman. 1996. Bagging predictors. *Machine Learn. 24*, 2, 123–140.

Stéphane Bressan, Mong Li Lee, Ying Guang Li, Zoé Lacroix, and Ullas Nambiar. 2002. The XOO7 benchmark. In *Proceedings of the VLDB Workshop on Efficiency and Effectiveness of XML Tools and Techniques (EEXTT)*.

Peter Christen. 2005. Probabilistic data generation for deduplication and data linkage. In *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*.

Peter Christen and Karl Goiser. 2007. Quality and complexity measures for data linkage and deduplication. In *Quality Measures in Data Mining*, Fabrice Guillet and Howard J. Hamilton (Eds.), Springer, 127–151.

Ricardo G. Cota, Marcos Andr Gonçalves, and Alberto H. F. Laender. 2007. A heuristic-based hierarchical clustering method for author name disambiguation in digital libraries. In *Proceedings of the Brazilian Symposium on Databases*.

Pedro Domingos. 1996. Unifying instance-based and rule-based induction. *Machine Learn. 24*, 2 (August 1996), 141–168.

[9] `http://www.mturk.com`.

Xin Dong, Alon Halevy, and Jayant Madhavan. 2005. Reference reconciliation in complex information spaces. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*. 85–96.

Uwe Draisbach and Felix Naumann. 2010. DuDe: The duplicate detection toolkit. In *Proceedings of the International Workshop on Quality in Databases (QDB)*.

Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios. 2007. Duplicates record detection: A survey. *IEEE Trans. Knowl. Data Engi.* (07), 1–16.

Michael Erdmann, Alexander Maedche, Hans-Peter Schnurr, and Steffen Staab. 2000. From manual to semi-automatic semantic annotation: About ontology-based text annotation tools. In *Proceedings of the COLING Workshop on Semantic Annotation and Intelligent Content*.

George Forman. 2002. Incremental machine learning to reduce biochemistry lab costs in the search for drug discovery. In *Proceedings of the International Workshop on Data Mining in Bioinformatics*.

Yoav Freund and Robert E. Schapire. 1996. Experiments with a new boosting algorithm. In *Proceedings of the International Conference on Machine Learning*. Morgan Kaufmann, 148–156.

Yoav Freund, H. Sebastian Seung, Eli Shamir, and Naftali Tishby. 1997. Selective sampling using the query by committee algorithm. *Machine Learn. 28*, 2–3 (August 1997), 133–168.

Jim Gray (Ed.). 1991. *The Benchmark Handbook for Database and Transaction Processing Systems*. Morgan Kaufmann Publishers.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: An update. *SIGKDD Explor. Newsl. 11*, 1 (July 2009), 10–18.

Oktie Hassanzadeh, Fei Chiang, Hyun Chul Lee, and Renée J. Miller. 2009. Framework for evaluating clustering algorithms in duplicate detection. *Proc. VLDB Endow. 2*, 1 (August 2009), 1282–1293.

Jian Huang, Seyda Ertekin, and C. Lee Giles. 2006. Efficient name disambiguation for large-scale databases. In *Proceedings of the European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)*.

Won Kim and W. John Wilbur. 2010. Improving a gold standard: Treating human relevance judgments of MEDLINE document pairs. In *Proceeding of the International Conference on Machine Learning and Applications*.

Hanna Köpcke and Erhard Rahm. 2010. Frameworks for entity matching: A comparison. *Data Knowl. Eng. 69*, 2 (2010), 197–210.

Zhiyong Lu and W. John Wilbur. 2010. Overview of BioCreative III Gene Normalization. In *Proceedings of the BioCreative III Workshop*.

David Menestrina, Steven Euijong Whang, and Hector Garcia-Molina. 2010. Evaluating entity resolution results. *Proc. VLDB Endowm. 3*, 1–2 (September 2010), 208–219.

Felix Naumann and Melanie Herschel. 2010. *An Introduction to Duplicate Detection (Synthesis Lectures on Data Management)*. Morgan and Claypool Publishers.

Mattis Neiling, Steffen Jurk, Hans-J. Lenz, and Felix Naumann. 2003. Object identification quality. In *Proceedings of the nternational Workshop on Data Quality in Cooperative Information Systsems (DQCIS)*.

Gabriele Paolacci, Jesse Chandler, and Panagiotis G. Ipeirotis. 2010. Running experiments on Amazon Mechanical Turk. *Judgment Decision Making 5*, 5 (2010), 411–419.

Robi Polikar, L. Upda, S. S. Upda, and Vasant Honavar. 2001. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE Trans. Syst. Man, Cybernetics, Part C 31*, 4 (2001), 497–508.

Erhard Rahm and Timo Böhme. 2002. XMach-1: A multi-user benchmark for XML data management. In *Proceedings of the VLDB Workshop on Efficiency and Effectiveness of XML Tools and Techniques (EEXTT)*.

Dietrich Rebholz-Schuhmann, Antonio Jimeno-Yepes, Erik M. van Mulligen, Ning Kang, Jan A. Kors, David Milward, Peter Corbett, Ekaterina Buyko, Elena Beisswanger, and Udo Hahn. 2010. CALBC Silver Standard Corpus. *J. Bioinform. Comput. Biol. 8*, 1 (February 2010), 163–179.

Gerard Salton and Chris Buckley. 1997. Improving retrieval performance by relevance feedback. In *Readings in Information Retrieval*, Morgan Kaufmann Publishers Inc.

Sunita Sarawagi and Anuradha Bhamidipaty. 2002. Interactive deduplication using active learning. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*. 269–278.

Albrecht Schmidt, Florian Waas, Martin L. Kersten, Michael J. Carey, Ioana Manolescu, and Ralph Busse. 2002. XMark: A benchmark for XML data management. In *Proceedings of the International Conference on Very Large Databases (VLDB)*. 974–985.

H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *Proceedings of the International Workshop on Computational Learning Theory (COLT)*.

Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. 2008. Get another label? Improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the ACM International Conference of Knowledge Discovery and Data Mining (SIGKDD)*. 614–622.

Kiril Simov, Petya Osenova, Alexander Simov, Anelia Tincheva, and Borislav Kirilov. 2007. A system for a semi-automatic ontology annotation. In *Proceedings of the Workshop on Computer-Aided Language Processing (CALP)*.

Parag Singla and Pedro Domingos. 2005. Object identification with attribute-mediated dependences. In *Proceeding of the European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)*. 297–308.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 254–263.

Melanie Weis, Felix Naumann, and Franziska Brosy. 2006. A duplicate detection benchmark for XML (and relational) data. In *Proceedings of the SIGMOD International Workshop on Information Quality for Information Systems (IQIS)*.

Peter Welinder, Steve Branson, Serge Belongie, and Pietro Perona. 2010. The multidimensional wisdom of crowds. In *Proceedings of the Neural Information Processing Systems Conference (NIPS)*.

Steven Euijong Whang, David Menestrina, Georgia Koutrika, Martin Theobald, and Hector Garcia-Molina. 2009. Entity resolution with iterative blocking. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*.

Received June 2013; revised December 2013; accepted March 2014