Effectiveness Bounds for Non-Exhaustive Schema Matching Systems

Marko Smiljanić° m.smiljanic@utwente.nl

Maurice van Keulen° m.vankeulen@utwente.nl

Willem Jonker•° willem.jonker@philips.com

° Department of EEMCS
University of Twente
P.O.Box 217
7500 AE Enschede
The Netherlands

• Department of Information & System Security
Philips Research
Eindhoven
The Netherlands

Abstract

Semantic validation of the effectiveness of a schema matching system is traditionally performed by comparing system-generated mappings with those of human evaluators. The human effort required for validation quickly becomes huge in large scale environments. The performance of a matching system, however, is not solely determined by the quality of the mappings, but also by the efficiency with which it can produce them. Improving efficiency quickly leads to a trade-off between efficiency and effectiveness. Establishing or obtaining a large test collection for measuring this trade-off is often a severe obstacle. In this paper, we present a technique for determining lower and upper bounds for effectiveness measures for a certain class of schema matching system improvements in order to lower the required validation effort. Effectiveness bounds for a matching system improvement are solely derived from a comparison of answer sets of the improved and original matching system. The technique was developed in the context of improving efficiency in XML schema matching, but we believe it to be more generically applicable in other retrieval systems facing scalability problems.

1 Introduction and related research

Validation of a schema matching system requires large amounts of human effort. The usual measures for reporting effectiveness or quality of a matching system are *precision* and *recall* [6]. These measures, however, are by definition based on a *human* evaluator determining the semantic correctness of a large number of mappings. To construct an appropriate test collection, a human evaluator has to inspect, for each matching problem, the whole search space and identify all correct mappings. To partly overcome the problems of this expensive and error-prone activity and to even out subjective human decisions, it is common to involve many human evaluators in the construction of a test collection. In large scale environments, such an approach would require an insurmountable amount of human effort.

Recent work shows that large scale schema matching systems are gaining importance [2, 9]. At the same time, the availability of large and properly constructed test collections is rather limited in the schema matching domain. Currently, validation of the schema matching system is usually performed using small test collections.

The text document retrieval community has taken the lead in developing techniques to reduce the required amount of effort and in the construction of large properly evaluated test collections. For example, in TREC *pooling* was used [10]: for each keyword query, the top 100 documents produced by each participating system were merged and only these were evaluated by a human. This works under the assumption that it is highly unlikely that a significant number of answers are not found by any participating system. Zobel confirmed that the limit of 100 is adequate [18].

We encountered a large scale validation problem in the context of XML schema matching [15, 16]. In particular, we investigate matching of a small user-given schema against a large repository of XML schemas as part of a personal schema based querying system. In our research, we explicitly focus on the efficiency of schema matching, because the overall performance of a matching system is not solely determined by the quality of the answers (effectiveness), but also by the efficiency with which it can produce them. In XML schema matching, many heuristics have been proposed for similarity between XML schemas [12, 7, 11, 8, 4], but applying these on a large scale, e.g., matching against XML schemas on the Web, is still an open problem. By employing clustering techniques, we attempt to quickly locate parts of schemas in a large repository that are likely to contain a match for a given small personal schema and then focus our search on these parts [16]. The approach is non-exhaustive, because mappings located (partially) outside a cluster or spanning clusters are not considered anymore. To validate scalability and the trade-off between effectiveness and efficiency for this non-exhaustive search approach, we would need much human effort to manually match personal schemas against a sufficiently large repository of schemas.

Reducing human effort and at the same time making sure that effectiveness measures are still reliable is a topic of ongoing research. Sayyadian et al., describe a system for unsupervised tuning of schema matching systems by means of synthetic scenarios [14]. The approach requires that a number of correct mappings is known beforehand. Tuning the system uses transformation rules on these mappings to synthesize a larger number of different schemas, i.e., synthetic schemas, which are used in validating the effectiveness and tuning of schema matching systems. In information retrieval systems, Buckley and Voorhees examined techniques of measuring effectiveness that are robust to massively incomplete relevance judgments [3]. They also suggest that their techniques allow studies of operational efficiency by embedding a small test collection with known judgments in a much larger test collection of similar documents with no judgments. Zobel suggested that a shallow pool of about 30 documents could be evaluated to predict the number of relevant documents further down [18]. More recently, Sanderson and Joho review three methods of test collection construction to see whether or not query and/or system pooling can be avoided to be able to "build a new test collection quickly and with limited resources" [13].

These techniques aim at providing an *estimate* for effectiveness. In this paper, however, we present an approach to determine effectiveness *bounds* of non-exhaustive system improvements, i.e, a lower and upper bound between which precision and recall are guaranteed to lie. Among others, this can be used to (1) provide effectiveness guarantees, (2) get an impression on the efficiency-effectiveness trade-off in an automated way allowing quick evaluation of many different parameter settings and matching system improvements, and (3) assess the accuracy of an effectiveness estimate acquired using other validation techniques.

The effectiveness bounds technique uses solely:

- measured effectiveness of the original system (e.g., on a small test set),
- answer sets of both improved and original system on a large test collection.

The technique can be applied in many unfavorable situations for low-effort scalability and efficiency research. It alleviates the need for manual (human) mapping discovery on large schema matching test collections. In case of research on efficiency improvements of other people's systems, the test collection associated with published effectiveness measures may not be available. Abovementioned techniques [3, 18, 13, 14] can be used to construct new large test collections that are expected to produce roughly the same effectiveness measures. Furthermore, it may happen that an original system for which effectiveness results have been published, is not available. Since the objective function of a system determines the actual ranking, a reconstruction with the same objective function exactly copies its behavior, hence effectiveness measures are expected to carry over to the reconstruction. In our work on applying clustering techniques on the efficiency of XML schema matching [15, 16], we encountered each of these problems in experiments aimed at obtaining an indication of the benefit of our clustering techniques on existing schema matching systems.

The paper is structured as follows. We first define notation and concepts in Section 2. In Section 3, we present the effectiveness bounds approach, which handles uncertainty in the effectiveness of the improved system by strictly adhering to best and worst case analysis. In Section 4, we examine the restrictions one should consider when using interpolation in the effectiveness bounds technique.

2 Quality measurement of schema matching systems

2.1 Notation

Let S be a schema matching system used to solve a schema matching problem Q, in which a user-defined schema is matched against a large schema repository. System S will search through the *search space* of all possible schema mappings $SS = \{d_1, \ldots, d_n\}$ and generate a resulting set of *mappings* or *answers* $\{a_1, \ldots, a_n\} \subseteq SS$. A schema mapping maps each element of a user-defined schema onto one element in the repository. In schema matching, elements of the search space are schema mappings, but in other retrieval systems they can in fact be anything such as images, documents, etc. In schema matching systems, it is not an absolute decision whether or not a certain mapping will become an answer. Rather, each mapping has a certain degree of computed quality by which it is ranked. Quality of the mappings is determined by the *objective function* $\Delta : SS \rightarrow \mathbb{R}$ which, in this paper, computes how different two schemas are. If $\Delta(a_1) < \Delta(a_2)$, a_1 is said to be a *better* mapping, i.e., it is higher ranked. Since users are only interested in the most relevant mappings, we define a *threshold* δ. The *answer set* A_S^δ is the set of mappings for which holds that $\forall a \in A_S^\delta \bullet \Delta(a) \leq \delta$. A system S is called *exhaustive* if it returns all possible mappings for

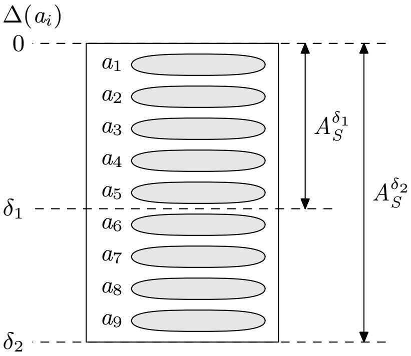


Figure 1. Thresholds in an answer set.

$$P_S^\delta = \frac{|T_S^\delta|}{|A_S^\delta|} \quad R_S^\delta = \frac{|T_S^\delta|}{|H|}$$

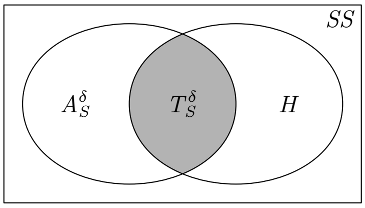


Figure 2. Precision and recall.

a certain threshold, i.e., $A_S^\delta = \{a \in SS \mid \Delta(a) \leq \delta\}$. Consequently, $\delta_1 \leq \delta_2 \Rightarrow A_S^{\delta_1} \subseteq A_S^{\delta_2}$ (see Figure 1). Therefore, by increasing the threshold, we can increase the number of answers S produces. Note that we do not exclude a situation where $\Delta(a_1) = \Delta(a_2)$ in which S is indecisive.

2.2 Precision and recall

The quality of a schema matching system S is expressed in terms of *precision* P_S^δ an *recall* R_S^δ. These measures give an indication of how well S is able to choose the same mappings a human would have chosen for a schema matching problem Q. Let H be the set of correct solutions manually determined by a human evaluator. With this set, we can evaluate the quality of S by distinguishing between correct and incorrect answers in A_S^δ. Let $T_S^\delta = H \cap A_S^\delta$ be the set of *correct answers*, also called *true positives*. Precision and recall are then defined as in Figure 2. Recall is the percentage of correct answers found by the system. Precision is the percentage of correct answers among the answers produced.

2.3 Precision and recall of a non-exhaustive system

Exhaustive search of schema mappings needs exponential time [15]. Efficient techniques are often based on heuristics to quickly, but roughly, restrict the search space.

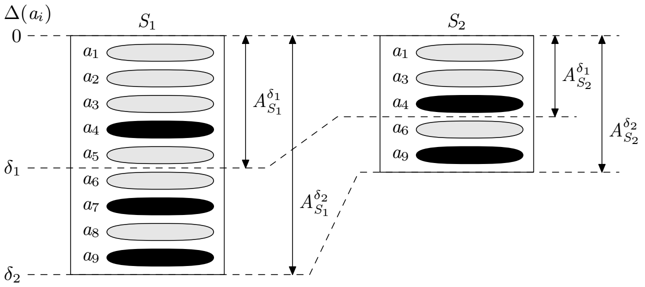


Figure 3. Exhaustive (S_1) and improved, but non-exhaustive, system (S_2).

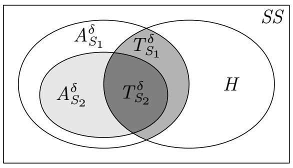


Figure 4. Precision and recall for a non-exhaustive system.

Mostly, there are no guarantees that heuristics do not exclude valid answers, hence the improved system becomes non-exhaustive. We assume here that the improved system S_2 uses the same objective function, i.e., answers produced by the improved system are ranked in the same way as by the original exhaustive system S_1 (see Figure 3). The beam search used in iMap [5], and the probabilistic guarantees approach of [17] are examples of a non-exhaustive system improvements which do not change the objective function.

Since S_2 uses the same objective function as S_1, it is guaranteed that $A_{S_2}^\delta \subseteq A_{S_1}^\delta$ (see Figure 4). For example, Figure 3 depicts correct answers (i.e., answers in H) in grey and incorrect answers in black. Improved system S_2 apparently misses answers a_2, a_5, a_7, and a_8. Note that this 'improved' system exhibits rather bad quality: it misses three correct answers and only one incorrect answer.

2.4 P/R curve

The recall of a schema matching system can be influenced by taking the top-N of highest ranking mappings or by setting some threshold δ. The natural behavior of a schema matching system is to loose precision with rising recall. In producing more mappings, the chance of delivering wrong mappings increases. The characteristics of the precision/recall trade-off is captured by a P/R curve. The intended way of constructing a P/R curve is by determining the precision at 11 fixed recall levels $0, 0.1, \ldots, 1$.

Since it is hard to find the right parameters for obtaining

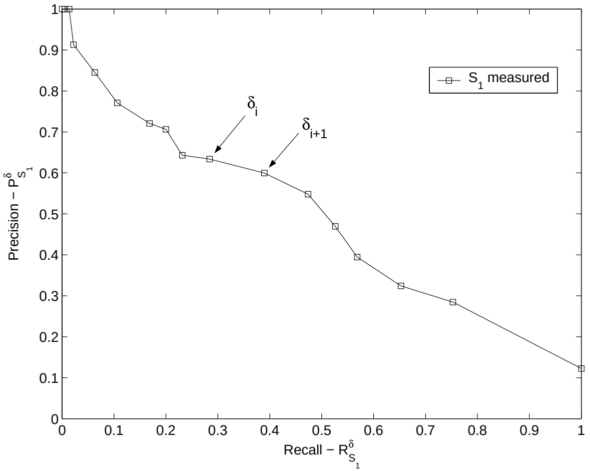


Figure 5. Measured P/R curve.

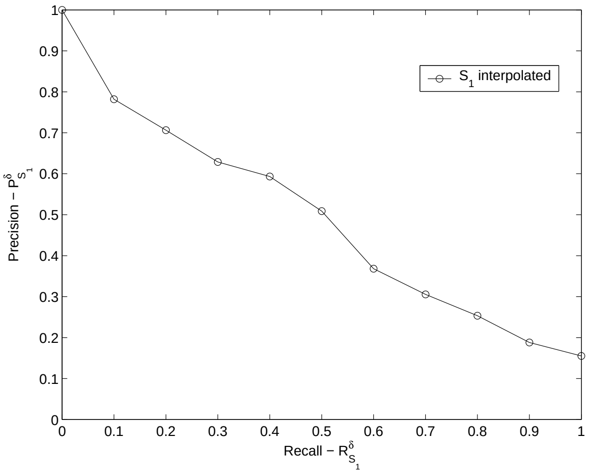
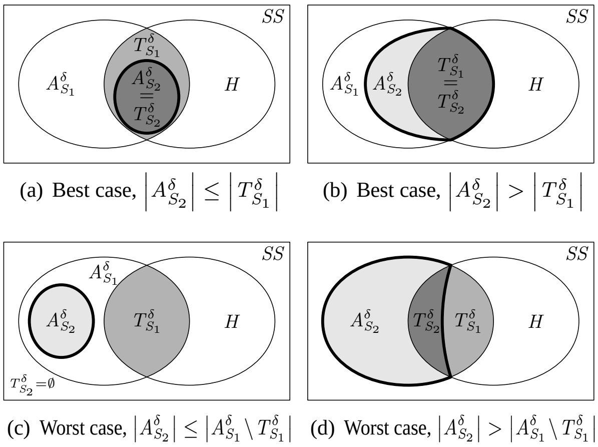


Figure 6. Interpolated P/R curve

these exact recall levels, the P/R curve is often constructed by varying the threshold and then measuring precision and recall. We call this a *measured P/R curve*. See Figure 5 for an illustration of such a curve.

If required, the 11-point P/R curve can be constructed from the measured P/R curve by interpolating the precision at the 11 recall levels. Different interpolation methods can be used. Figure 6 shows an interpolated P/R curve constructed from the measured one in Figure 5.

In this paper, we assume that the effectiveness of a system S is defined in terms of either the measured or the interpolated P/R curve. The curve is the starting point for computing the bounds of the effectiveness of an improved non-exhaustive matching system. We do, however, make an important assumption that the thus measured effectiveness is independent of the size of the search space. In other words, regardless of the size of the schema repository, we assume S will show the same P/R curve. In schema matching so far no research has been done to prove this assumption, but in text retrieval community, this appears to be a reasonable assumption [1].



(a) Best case, $\left|A^{\delta}_{S_2}\right| \leq \left|T^{\delta}_{S_1}\right|$ (b) Best case, $\left|A^{\delta}_{S_2}\right| > \left|T^{\delta}_{S_1}\right|$

(c) Worst case, $\left|A^{\delta}_{S_2}\right| \leq \left|A^{\delta}_{S_1} \setminus T^{\delta}_{S_1}\right|$ (d) Worst case, $\left|A^{\delta}_{S_2}\right| > \left|A^{\delta}_{S_1} \setminus T^{\delta}_{S_1}\right|$

Figure 7. Best and worst case quality

We investigate efficiency improvements using large scale schema repositories. As explained in Section 1, full semantic validation by a human is impractical in such scenarios. Therefore, H is unknown to us as well as all quality measures derived from it. In the sequel, we show ways to establish effectiveness bounds for semantic quality of an improved system that is independent of H.

3 Size-based best and worst-case bounds

3.1 Best and worst-case sizes

With H unknown, the set of true positives T^{δ}_{S} for any system S, is also unknown. Hence, it is not possible to determine the quality of an improved system. What is often available from literature, however, are precision and recall figures in a P/R curve and a formulation of the objective function Δ. As explained in Section 2.3, we assume that a system that only improves the efficiency of S, uses the same objective function. Consequently, the improved system assigns the same scores to answers; it will only examine and produce less answers (see Figure 3). Therefore, quality measures of S can be used to estimate the quality of the improved system. In this section, we examine an approach that is solely based on unsupervised analysis of answer sets A^{δ}_{S} produced by both S and its non-exhaustive improvement.

Let S_1 be an exhaustive matching system and S_2 a non-exhaustive improvement thereof both using objective function Δ. For a given matching problem Q, we know that $A^{\delta}_{S_2} \subseteq A^{\delta}_{S_1}$. Whether the answers S_2 misses are correct or incorrect is, however, unknown. In the best case, S_2 misses only incorrect mappings, in the worst case the most correct ones.

Best case scenario. Let $\widehat{A}^\delta_{S_2/S_1} = \frac{A^\delta_{S_2}}{A^\delta_{S_1}}$ be the size ratio of the answer sizes of systems S_2 and S_1. Two situations can be distinguished: If $A^\delta_{S_2}$ is small enough, it is fully included in $T^\delta_{S_1}$, i.e., $A^\delta_{S_2} \subseteq T^\delta_{S_1}$, hence $T^\delta_{S_2} = A^\delta_{S_2}$ (see Figure 7(a)). Otherwise, $T^\delta_{S_1} \subseteq A^\delta_{S_2}$, hence $T^\delta_{S_2} = T^\delta_{S_1}$ (see Figure 7(b)). Consequently, we derive the following equations for precision and recall of S_2 solely in terms of precision and recall of S_1 and the size ratio.

[**best case**]

$$
\begin{aligned}
|T^\delta_{S_2}| &= \min(|T^\delta_{S_1}|, |A^\delta_{S_2}|) && (1)\\
P^\delta_{S_2} &= \frac{|T^\delta_{S_2}|}{|A^\delta_{S_2}|} = \frac{\min(|T^\delta_{S_1}|, |A^\delta_{S_2}|)}{|A^\delta_{S_2}|}\\
&= \min(\frac{|T^\delta_{S_1}|}{|A^\delta_{S_2}|}, 1)\\
&= P^\delta_{S_1} \cdot \min(\frac{1}{\widehat{A}^\delta_{S_2/S_1}}, \frac{1}{P^\delta_{S_1}}) && (2)\\
R^\delta_{S_2} &= \frac{|T^\delta_{S_2}|}{|H|} = \frac{\min(|T^\delta_{S_1}|, |A^\delta_{S_2}|)}{|H|}\\
&= \min(R^\delta_{S_1}, \frac{|A^\delta_{S_2}|}{|H|}) = R^\delta_{S_1} \cdot \min(1, \frac{|A^\delta_{S_2}|}{|T^\delta_{S_1}|})\\
&= R^\delta_{S_1} \cdot \min(1, \frac{\widehat{A}^\delta_{S_2/S_1}}{P^\delta_{S_1}}) && (3)
\end{aligned}
$$

Worst case scenario. Again two situations: If $A^\delta_{S_2}$ is small enough, then it may be fully 'detached' from $T^\delta_{S_1}$, i.e., precision and recall are zero (see Figure 7(c)). Otherwise, we get the situation depicted in Figure 7(d).

[**worst case**]

$$
\begin{aligned}
|T^\delta_{S_2}| &= \max(0, |A^\delta_{S_2}| - (|A^\delta_{S_1}| - |T^\delta_{S_1}|)) && (4)\\
P^\delta_{S_2} &= \frac{|T^\delta_{S_2}|}{|A^\delta_{S_2}|}\\
&= \max(0, \frac{|A^\delta_{S_2}| - (|A^\delta_{S_1}| - |T^\delta_{S_1}|)}{|A^\delta_{S_2}|})\\
&= \max(0, 1 - (\frac{|A^\delta_{S_1}|}{|A^\delta_{S_2}|} - \frac{|T^\delta_{S_1}|}{|A^\delta_{S_2}|}))\\
&= \max(0, 1 - (\frac{1}{\widehat{A}^\delta_{S_2/S_1}} - \frac{P^\delta_{S_1}}{\widehat{A}^\delta_{S_2/S_1}}))\\
&= \max(0, 1 - \frac{1 - P^\delta_{S_1}}{\widehat{A}^\delta_{S_2/S_1}}) && (5)
\end{aligned}
$$

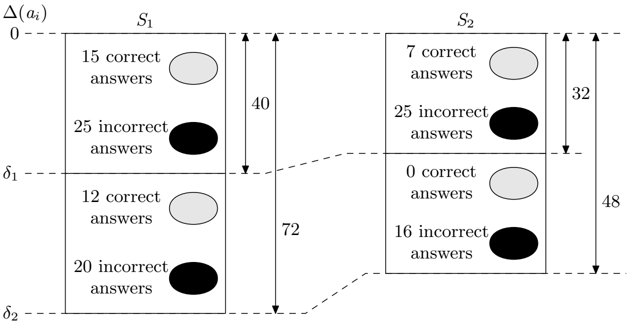


Figure 8. Incremental worst case estimation example

$$
\begin{aligned}
R^\delta_{S_2} &= \frac{|T^\delta_{S_2}|}{|H|}\\
&= \max(0, \frac{|A^\delta_{S_2}| - (|A^\delta_{S_1}| - |T^\delta_{S_1}|)}{|H|})\\
&= \max(0, \frac{|A^\delta_{S_2}|}{|H|} - \frac{|A^\delta_{S_1}|}{|H|} + \frac{|T^\delta_{S_1}|}{|H|})\\
&= \max(0, R^\delta_{S_1}(\frac{|A^\delta_{S_2}|}{|T^\delta_{S_1}|} - \frac{|A^\delta_{S_1}|}{|T^\delta_{S_1}|} + 1))\\
&= \max(0, R^\delta_{S_1}\left(\frac{\frac{|A^\delta_{S_2}|}{|A^\delta_{S_1}|} - \frac{|A^\delta_{S_1}|}{|A^\delta_{S_1}|}}{\frac{|T^\delta_{S_1}|}{|A^\delta_{S_1}|}} + 1\right))\\
&= \max(0, R^\delta_{S_1}(\frac{\widehat{A}^\delta_{S_2/S_1} - 1}{P^\delta_{S_1}} + 1)) && (6)
\end{aligned}
$$

Note that all formulas for best/worst case precision and recall for S_2 are defined in terms of precision and recall of S_1, which we assumed is known, and the size ratio of the answer sets of both systems, which is acquired through experiments.

3.2 Establishing best and worst case bounds incrementally

The bounds for recall and precision given in the previous section, can be used for any threshold δ. In this section, we first show by means of an example that the bounds are unnessarily pessimistic, and then describe a more accurate incremental approach.

Figure 8 shows concrete numbers for two hypothetical systems S_1 and S_2. S_1 is known from literature to have stable precision $P^{\delta_1}_{S_1} = P^{\delta_2}_{S_1} = 3/8$ (37.5%) for two given thresholds $\delta_1 \leq \delta_2$. We rebuilt S_1 with the published objective function Δ. For thresholds δ_1 and δ_2, it produces

40 and 72 answers, respectively. Suppose, we build an improved system (S_2) that uses the same objective function Δ, but has a more efficient algorithm that possibly misses answers in an early search space restricting phase.

Let us look at the worst case for the precision of S_2. We experimentally determine that for thresholds δ_1 and δ_2, S_2 produces 32 and 48 answers. Hence, $\widehat{A}^{\delta_1}_{S_1/S_2}$ and $\widehat{A}^{\delta_2}_{S_1/S_2}$ are $4/5$ and $2/3$. Using the formulas from the previous section, we obtain for each threshold independently, worst case bounds $P^{\delta_1}_{S_2} = 7/32$ and $P^{\delta_2}_{S_2} = 1/16$.

The reasoning behind this is as follows. For δ_1, $P^{\delta_1}_{S_1} = 3/8$, i.e., we know that, 15 of the 40 answers are correct and the remaining 25 are incorrect. Increasing the threshold to δ_2 gives 12 additional correct answers and 20 additional incorrect ones (left part of Figure 8). S_2 only misses answers, so the worst case for δ_1 is that all 8 answers missed, were correct ones. Among the 32 answers, there are at least 7 correct and at most 25 incorrect ones. Worst case bound for $P^{\delta_1}_{S_2} = 7/32$ (21.9%). Similarly, for δ_2, there are among the 48 answers, at most 45 incorrect and at least 3 correct. Therefore, the worst case bound for $P^{\delta_2}_{S_2} = 1/16$ (6.3%) (right part of Figure 8).

The inaccuracy of this reasoning is obvious. If among the first 32 answers, there are already 7 correct ones, it is not possible that for δ_2 we have in total 3 correct answers, if the only thing S_2 does is *add* 16 more answers. In the second *increment*, i.e., the answers a_i with $\delta_1 < \Delta(a_i) \leq \delta_2$, S_2 can in the worst case miss all 12 correct answers and 4 incorrect ones. In other words, the second increment for S_2 contains 41 incorrect answers and no correct ones (right part of Figure 8). Hence, a more accurate worst case bound for $P^{\delta_2}_{S_2} = 7/48$ (14.6%).

Consequently, a gain in accuracy is obtained if the bounds for precision and recall are computed increment-by-increment. An increment is defined by two threshold values $\delta_1 - \delta_2$. The answer set of the increment $\widehat{A}^{\delta_1-\delta_2}_S$ contains all answers with a ranking between these thresholds, i.e., answers a_i with $\delta_1 < \Delta(a_i) \leq \delta_2$. The answer set is defined by $\widehat{A}^{\delta_1-\delta_2}_S = A^{\delta_2}_S \setminus A^{\delta_1}_S$. Since an increment contains correct and incorrect answers, it makes sense to speak about precision and recall of an increment. Let $\widehat{T}^{\delta_1-\delta_2}_S = \widehat{A}^{\delta_1-\delta_2}_S \cap H = T^{\delta_2}_S \setminus T^{\delta_1}_S$ be the set of correct answers of increment $\delta_1 - \delta_2$. Precision and recall of an increment can now be defined as follows

$$\begin{aligned}
\widehat{P}^{\delta_1-\delta_2}_S &= \frac{\left|\widehat{T}^{\delta_1-\delta_2}_S\right|}{\left|\widehat{A}^{\delta_1-\delta_2}_S\right|} = \frac{\left|T^{\delta_2}_S\right| - \left|T^{\delta_1}_S\right|}{\left|A^{\delta_2}_S\right| - \left|A^{\delta_1}_S\right|} \\
&= \frac{\frac{\left|T^{\delta_2}_S\right|}{|H|} - \frac{\left|T^{\delta_1}_S\right|}{|H|}}{\frac{\left|A^{\delta_2}_S\right|}{|H|} - \frac{\left|A^{\delta_1}_S\right|}{|H|}} = \frac{R^{\delta_2}_S - R^{\delta_1}_S}{\frac{R^{\delta_2}_S}{P^{\delta_2}_S} - \frac{R^{\delta_1}_S}{P^{\delta_1}_S}} \qquad (7) \\
\widehat{R}^{\delta_1-\delta_2}_S &= \frac{\left|\widehat{T}^{\delta_1-\delta_2}_S\right|}{|H|} = \frac{\left|T^{\delta_2}_S\right| - \left|T^{\delta_1}_S\right|}{|H|} \\
&= R^{\delta_2}_S - R^{\delta_1}_S \qquad (8)
\end{aligned}$$

In four steps, we can establish the accurate effectiveness bounds using the above formulas.

1. Determine for which sequence of thresholds $0, \delta_1, \ldots, \delta_n$ the original measurements were made. These determine the increments $0 - \delta_1, \delta_1 - \delta_2, \ldots, \delta_{n-1} - \delta_n$.

2. For a given exhaustive system S_1, we have precision and recall for any two thresholds δ_1 and δ_2. Using the above formulas, we can calculate the precision and recall for any increment $\delta_i - \delta_j$ $(j = i + 1)$ ($\widehat{P}^{\delta_i-\delta_j}_{S_1}$ and $\widehat{R}^{\delta_i-\delta_j}_{S_1}$, respectively).

3. Using formulas 2, 3, 5, and 6 of Section 3.1, we can then calculate the best and worst case precision and recall for an improved system S_2 for each increment $\delta_i - \delta_j$ ($\widehat{P}^{\delta_i-\delta_j}_{S_2}$ and $\widehat{R}^{\delta_i-\delta_j}_{S_2}$, respectively).

4. Finally, Equations 7 and 8 can be used again, but now to calculate best and worst case precision and recall for S_2 at each threshold δ_j based on the precision and recall bounds at threshold δ_i. Since the bound of the first increment $0 - \delta_1$ can be established directly using Equation 5, this allows an incremental derivation of bounds at all thresholds. In the special case where an increment $\delta_i - \delta_j$ does not contain any correct answers, precision and recall are zero for the increment, which does not allow for calculating precision and recall at δ_2. Instead, note that recall is the same as in δ_1 and precision can be calculated directly from Figure 2.

To illustrate this incremental approach, let us look at the worst case for S_2 in Figure 8 again. (1) This example uses two increments $0 - \delta_1$ and $\delta_1 - \delta_2$. (2) $P^{\delta_1}_{S_1} = P^{\delta_2}_{S_1} = 3/8$ is given. Note that Equation 7 is actually independent of $|H|$, i.e., we calculate $\widehat{P}^{\delta_1-\delta_2}_{S_1} = 3/8$. (3) Using Equation 5, we obtain a worst case bound $\widehat{P}^{\delta_1-\delta_2}_{S_2} = 0$. (4) The bound of a first increment $0 - \delta_1$ can be established directly using Equation 5 giving $\widehat{P}^{0-\delta_1}_{S_2} = P^{\delta_1}_{S_2} = 7/32$. Using the second formula in Equation 7, we finally derive $P^{\delta_2}_{S_2} = 7/48$.

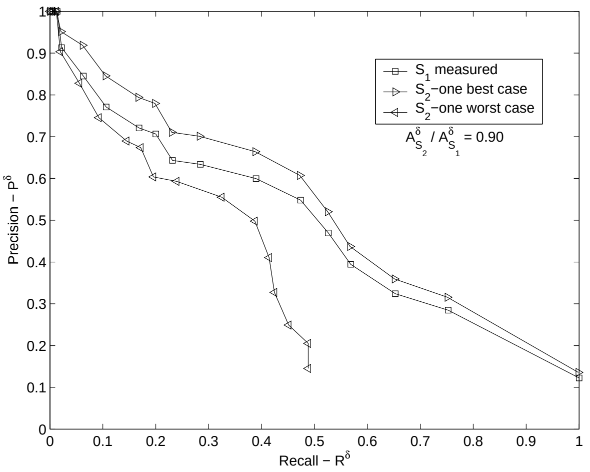


Figure 9. Best/worst case P/R curve for fixed $\widehat{A}^{\delta}_{S_2/S_1}$=0.9

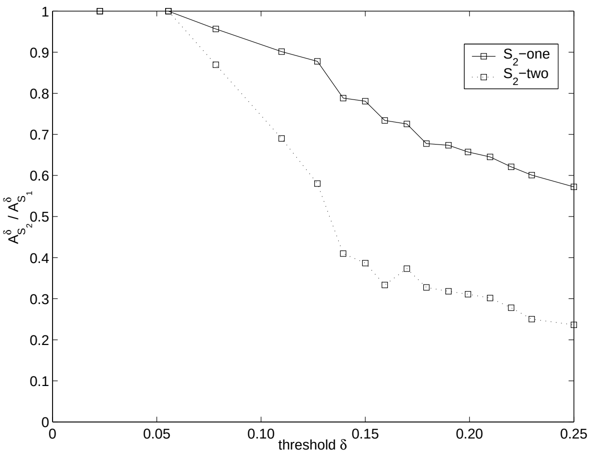


Figure 10. Measured $\widehat{A}^{\delta}_{S_2/S_1}$ for two rather different system improvements

3.3 Best and worse case P/R curve

Using the formulas of the previous sections, we can derive a best and worst case P/R curve by varying the threshold. At each threshold value, the matching is performed by both the improved and the original system to obtain the result sizes. The obtained sizes are used to compute both best and worst case precision and recall values, in this way establishing effectiveness bounds for that threshold value. The curves for best and worst case demarcate the area within which the actual P/R curve should lie. In this section, we will show some examples of thusly established P/R curves, to give some insight in the behavior of the process.

The approach is ultimately based on answer sizes, more concretely on $\widehat{A}^{\delta}_{S_2/S_1}$. Observe that for $\widehat{A}^{\delta}_{S_2/S_1} = 1$, the best and worst case bounds are exactly the same and equal to the original P/R curve for S_1. This is because of our assumptions. If an improved system produces the same number of answers, then it necessarily produces the same answers, hence has the same precision and recall characteristics.

Figure 9 shows the resulting effectiveness bounds for a hypothetical system S_2 that behaves with a fixed answer size ratio $\widehat{A}^{\delta}_{S_2/S_1} = 0.9$ for each threshold δ. In other words, it misses the same fraction of answers for all increments.

To give insight into the behavior of the process for real life systems, we chose two actual improved systems S_2-one and S_2-two with different behavior (taken from our work in XML schema matching). See Figure 10 for the measured $\widehat{A}^{\delta}_{S_2/S_1}$ for both systems. S_2-one shows a smoothly declining ratio of retrieved answers, with an increasing threshold. At $\delta = 0.25$ about 60% of the answers are still retained. S_2-two is more rigorous in missing answers. Of the answers with a score higher than 0.13, only about 25–30% is retained. The answers with the best score still have a high chance of being retained tough.

The result of determining the best and worst case P/R curves for both systems can be found in Figure 11 (the 'random case' is explained in Section 3.4). Notice, that for both systems, the best and worst case curves are far away from each other especially at higher recall levels. This does not mean that the systems are bad, only that we could not establish tighter effectiveness bounds. For all we know, S_2-one may in fact behave close to its worst case, while S_2-two behaves close to its best case, or vice versa. What we can see, for example, is that for recall levels up to 0.15, S_2-one guarantees a worst case precision of 0.5.

This analysis also shows that the answer size ratio $\widehat{A}^{\delta}_{S_2/S_1}$ significantly influences the accuracy, especially the worst case. The bigger the answer size $A^{\delta}_{S_2}$, the better the chances to acquire narrow bounds. And, as we already mentioned, with $\widehat{A}^{\delta}_{S_2/S_1} = 1$, we have is absolute certainty. On the other hand, in general the approach provides rather wide bounds unfortunately.

3.4 Comparing with a 'random' system

The curves for the best and worst case are rather far apart. Another interpretation of what a *worst case* is, may improve this. If we assume that any non-exhaustive improvement that we construct, produces a better set of answers than simply picking them randomly, then we may use the P/R curve of this hypothetical random system as worst case bound. In this section, we explore this idea.

Let S_1 be an original schema matching system and S_2 a

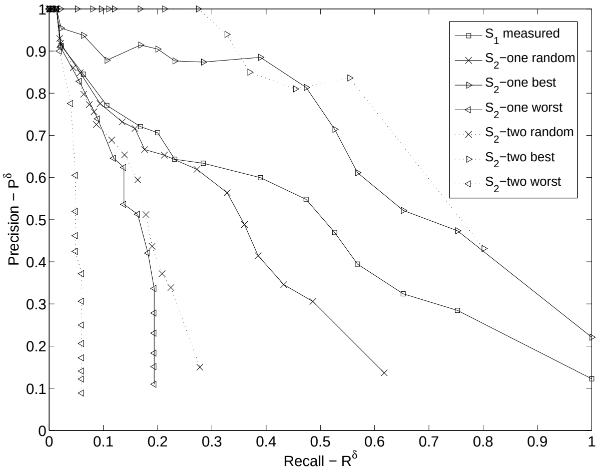


Figure 11. Best/worst case P/R curve for the two systems

non-exhaustive improvement of S_1. Let S_{random} be a random system that simply executes S_1 and for each increment selects a certain percentage of answers randomly. Since we are using the random system to compare with S_2, we need it to produce the same number of answers as S_2. In other words, S_{random} has the same answer size ratio curve as S_2 (see Figure 10).

The random P/R curve is computed using the incremental computation described in Sec. 3.2 with the following difference. In *Step 2*, the best and the worst case formulas for precision and recall of an increment are not used. Instead, the precision and recall of the increment of the random system are computed using the following formulas.

[**random case**]

$$\widehat{P}^{\delta_i-\delta_{i+1}}_{S_{random}} = \widehat{P}^{\delta_i-\delta_{i+1}}_{S_1} \tag{9}$$

$$\widehat{R}^{\delta_i-\delta_{i+1}}_{S_{random}} = \widehat{R}^{\delta_i-\delta_{i+1}}_{S_1} \cdot \frac{\widehat{A}^{\delta_i-\delta_{i+1}}_{S_{random}}}{\widehat{A}^{\delta_i-\delta_{i+1}}_{S_1}} \tag{10}$$

These formulas are acquired as follows. When randomly selecting answers from $\widehat{A}^{\delta_i-\delta_{i+1}}_{S_1}$ in order to form $\widehat{A}^{\delta_i-\delta_{i+1}}_{S_{random}}$ the ratio of *correct* and *incorrect* answers in these two sets remains the same, thus $\frac{\left|\widehat{T}^{\delta_i-\delta_{i+1}}_{S_{random}}\right|}{\left|\widehat{A}^{\delta_i-\delta_{i+1}}_{S_{random}}\right|} = \frac{\left|\widehat{T}^{\delta_i-\delta_{i+1}}_{S_1}\right|}{\left|\widehat{A}^{\delta_i-\delta_{i+1}}_{S_1}\right|}$. When combining this equation with the ones given in Figure 2 the result are Equations 9 and 10: precision of the random system increment does not change, but the recall is reduced proportional to its size.

Figure 11 also shows the curves of the random system corresponding to S_2-one and S_2-two. Given the expectation that an improved system performs better than the random system, this gives a more useful lower bound, since it produces a narrower interval. For example, Figure 11 shows that precision of 0.5 is maintained up to a recall of 0.35 for S_2-one

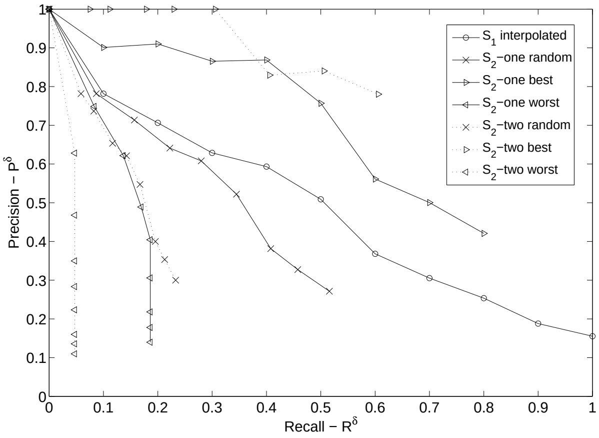


Figure 12. Best/worst case based on an interpolated P/R curve (guess $|H| = 15000$)

4 Effects of interpolation on establishing effectiveness bounds

4.1 An interpolated P/R curve as input

The best/worst case analysis presented above is based on a measured P/R curve (see Section 2.4) as input. A published 11-point P/R curve from literature (called interpolated P/R curve in Section 2.4) seems to be equally suitable, but in fact, it lacks one kind of information: the specific threshold points. Using the equations from Figure 2, we can derive that $\left|A^\delta_S\right| = \frac{R^\delta_S\cdot|H|}{P^\delta_S}$. Observe that from an interpolated P/R curve, it is not possible to determine at which δ-value a certain precision and recall was measured, because $|H|$ is unknown to us. Without this information, it is not possible to correlate the published precision and recall measures with the answer sets acquired on a different different, large scale, test collection.

The only missing parameter is the size of H. Given a value for $|H|$, it is possible to establish the correspondence between the δ-values and the 11 points of the given interpolated P/R curve. In other words, with a given value for $|H|$ one can transform an interpolated P/R curve into a measured one.

We performed initial experiments investigating the impact of varying $|H|$ on the resulting measured P/R curve. Figure 12 shows a P/R curve comparable to the one in Figure 11, but this one was obtained by using the interpolated P/R curve of Figure 6 with $|H| = 15000$. It shows that the impact of varying $|H|$ is that the effectiveness bounds

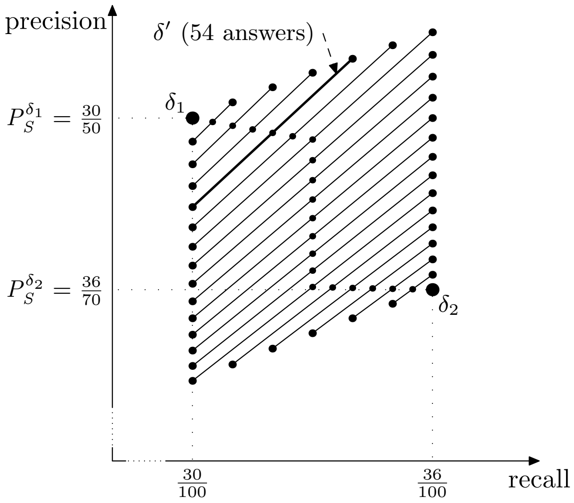


Figure 13. Boundaries for interpolation on sub-increment level ($|H| = 100$).

become a little bit less accurate. We suspect that in cases where $|H|$ is unknown, a rough estimate suffices to obtain a reasonably accurate measured P/R curve. Further research is needed to confirm this suspicion.

4.2 Sub-increment level bounds

In the previous section, we have seen the need to use interpolation in order to determine the precision and recall when the 11-point P/R curve is used as input. A similar situation occurs when experiments on large collections use more (finer) threshold points than the measured P/R curve; the need arises to compute the *P/R* values for thresholds not directly specified in the measured P/R curve. This can only be done by interpolation. Without going into detail about the effects of different interpolation approaches, we give a characterization of the boundaries between which interpolated points on the P/R curve are guaranteed to lie. We analyze this issue by means of an example.

Say, we have obtained from literature results from an experiment with a certain system S and we rebuilt this system using the same objective function. Suppose, at two thresholds δ_1 and δ_2, literature reports $|H| = 100, R_S^{\delta_1} = \frac{30}{100}, R_S^{\delta_2} = \frac{36}{100}, P_S^{\delta_1} = \frac{30}{50}, P_S^{\delta_2} = \frac{36}{70}$. This is illustrated with the two big points in Figure 13. Our rebuilt system produces 50 and 70 answers for these thresholds, respectively.

Let us examine some intermediary threshold $\delta_1 \leq \delta' \leq \delta_2$. We observe that our rebuilt system produces 54 answers. Since there is no quality measurement available, precision and recall for this threshold are unknown, i.e., the location of the point on the P/R curve corresponding with δ' is unknown. We do know, however, that at δ_1, there were 30 correct answers among the 50. At δ', there are 4 more answers of which it is unknown whether or not they are correct. In the worst case, they are all incorrect ($R_S^{\delta'} = 30, P_S^{\delta'} = 30/54$), in the best case they are all correct ($R_S^{\delta'} = 34, P_S^{\delta'} = 34/54$). In other words, an interpolated point for δ' should lie on the line between $(30/100, 30/54)$ and $(34/100, 34/54)$. In Figure 13, this is depicted with the thick line marked 'δ' (54 answers)'.

By varying thresholds between δ_1 and δ_2, one obtains many lines that demarcate boundaries for interpolating points on the P/R curve. Note that taking the point halfway between worst and best case (small dots in the figure) is not the same as linear interpolation between δ_1 and δ_2.

The shape of the boundary can be explained as follows. Close to the measured points, there are only a few answers for which it is unknown whether or not they are correct or incorrect. This establishes restrictions on how good the best case and how bad the worst case can be. In the figure, this becomes apparent by the three sections observable in the halfway-points. The fact that precision can go up in a P/R curve was also observed in the appendix of [10]. Without further analysis, the figure shows, that the safest, i.e., with smallest error, interpolation choice is made by taking the mid points in the lines.

Finally, because several answers may have the same score, best and worst case points may not be as evenly distributed in practice as in the figure.

5 Conclusion

Validating a schema matching system in a large scale environment is an expensive, if even possible, task because of the human effort required. In this paper we have shown, that in certain circumstances, lower and upper bounds can be given for the effectiveness of a system *improvement* without the need for human evaluators. Such effectiveness bounds can be used, for example, to claim that the trade-off in effectiveness for an efficiency improvement is at most $x\%$, or to get a quick impression on the trade-off to allow for quickly evaluating many parameter settings, algorithms, and systems in a less costly way.

Summarizing, we determine a best and worst case P/R curve of a non-exhaustive improvement of a certain original system that uses the same objective function. The actual P/R curve of the improvement is unknown, but should lie between these bounds. Many techniques are known to give estimates, but the aim of this paper is to give best and worst case bounds for such estimates. The accuracy of such effectiveness bounds can be increased by using an incremental approach. We furthermore suggest that our worst case analysis is perhaps too pessimistic: it can be argued that any realistic improvement will perform better than a hypothetical 'improvement' that simply selects answers randomly from the original system. Finally, we examine what restrictions to consider when trying to apply interpolation to our technique. Note that the technique is an analytical and exact result, not an estimate for which experimental validation is necessary. Moreover, if experimental validation were possible, the technique would not be needed in the first place.

System improvements that increase query execution performance significantly, need to drastically restrict the search space by disregarding objects that are unlikely to be answers with considerable relevance. In other words, efficient but still qualitatively good performing systems show answer size ratios like S_2-two of Figure 10. With this approach, it often appears to be impossible to obtain narrow effectiveness bounds for the entire spectrum, because at higher recall levels, there is inadequate accuracy for making useful worst case claims. But, for schema matching systems as well as information retrieval systems in general, the top-N is usually the most interesting and for such recall levels, we can give useful, i.e., narrow effectiveness bounds.

References

[1] M. Beigbeder and A. Imafouo. An experimental methodology to study collections size impact on retrieval effectiveness. In *DIR '05: Proc. of 5th Dutch-Belgian Information Retrieval Workshop*, Jan. 2005.

[2] P. A. Bernstein, S. Melnik, M. Petropoulos, and C. Quix. Industrial-Strength Schema Matching. *SIGMOD Rec.*, 33(4):38–43, 2004.

[3] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *Proc. of the 27th Int. Conf. on Research and Development in Information Retrieval SIGIR)*, pages 25–32. ACM Press, 2004.

[4] C. Delobel, C. Reynaud, M.-C. Rousset, J.-P. Sirot, and D. Vodislav. Semantic integration in xyleme: a uniform tree-based approach. *Data Knowledge Engineering*, 44(3):267–298, 2003.

[5] R. Dhamankar, Y. Lee, A. Doan, A. Halevy, and P. Domingos. iMAP: discovering complex semantic matches between database schemas. In *SIGMOD '04: Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 383–394. ACM Press, 2004.

[6] H. Do, S. Melnik, and E. Rahm. Comparison of schema matching evaluations. In *Proceedings of the 2nd Int. Workshop on Web Databases*, 2002.

[7] H. H. Do, S. Melnik, and E. Rahm. Comparison of schema matching evaluations. In *Web, Web-Services, and Database Systems, NODe 2002 Web and Database-Related Workshops, October 7–10, Revised Papers*, volume 2593 of *Lecture Notes in Computer Science*, pages 221–237. Springer, 2003.

[8] H. H. Do and E. Rahm. Coma - a system for flexible combination of schema matching approaches. In *Proc. of 28th Int. Conf. on Very Large Data Bases (VLDB)*, pages 610–621, 2002.

[9] Erhard Rahm and Hong-Hai Do and Sabine Mamann. Matching Large XML Schemas. *SIGMOD Rec.*, 33(4):26–31, 2004.

[10] D. Harman. Overview of the first text retrieval conference. In *Proc. of the 16th Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 36–47. ACM Press, June 1993.

[11] J. Madhavan, P. A. Bernstein, and E. Rahm. Generic schema matching with cupid. In *Proc. of 27th Int. Conf. on Very Large Data Bases (VLDB)*, pages 49–58. Morgan Kaufmann, Sept. 2001.

[12] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10(4):334–350, 2001.

[13] M. Sanderson and H. Joho. Forming test collections with no system pooling. In *Proc. of the 27th Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 33–40. ACM Press, 2004.

[14] M. Sayyadian, Y. Lee, A. Doan, and A. S. Rosenthal. Tuning schema matching software using synthetic scenarios. In *VLDB '05: Proceedings of the 31st international conference on Very large data bases*, pages 994–1005. VLDB Endowment, 2005.

[15] M. Smiljanic, M. van Keulen, and W. Jonker. Formalizing the XML Schema Matching Problem as a Constraint Optimization Problem. In *Proceedings of the 16th International Conference on Database and Expert Systems Applications (DEXA), 22-26 August 2005, Copenhagen, Denmark*, volume 3588 of *LNCS*, pages 333–342. Springer, Aug. 2005.

[16] M. Smiljanic, M. van Keulen, and W. Jonker. Using Element Clustering to Increase the Efficiency of XML Schema Matching. In *Proceedings of the 2nd International Workshop on Challenges in Web Information Retrieval and Integration (WIRI'06)*, Apr. 2006.

[17] M. Theobald, G. Weikum, and R. Schenkel. Top-k query evaluation with probabilistic guarantees. In *Proc. of the 30th Int. Conf. on Very Large Data Bases (VLDB)*, pages 648–659. Morgan Kaufmann, Sept. 2004.

[18] J. Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proc. of the 21st Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 307–314. ACM Press, Aug. 1998.