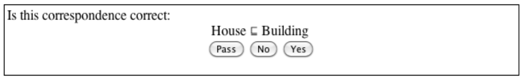
Uncertainty in crowdsourcing ontology matching

Jérôme Euzenat

INRIA & LIG, France

Matching crowdsourcing There may be several motivations for crowdsourcing ontology matching, i.e., relying on a crowd of workers for establishing alignments [2]. It may be for matching itself or for establishing a reference alignment against which matchers are evaluated. It may also be possible to use crowdsourcing as complement to a matcher, either to filter the finally provided alignment or to punctually provide hints to the matcher during its processing.

The ideal way of crowdsourcing ontology matching is to design microtasks (t) around alignment correspondences and to ask workers (w) if they are valid or not.


Is this correspondence correct:

House ⊑ Building

$c_w(t) = \sqsubseteq \quad \equiv$

House ⊑ Building

Uncertainty Most of the crowdsourcing philosophy relies on the idea that microtasks have one single solution that workers are good at finding (even if this requires more skilled workers).

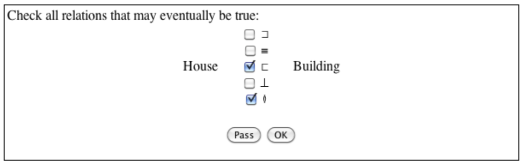
The experience acquired in ontology matching shows that, because concepts are underdefined, there may not be one unique answer to a matching microtask. Moreover, we know that "experts" do not necessary have coinciding opinions [3].

One way to deal with this problem is to take into account uncertainty from the beginning and to know how to deal with uncertainty instead of trying to cast it into certainty.

We base our approach on the principle that workers may not know for certain what the answer is, but they may know for certain that it is among a set of alternatives. Representing this set is a way to deal with uncertainty.

Disjunctive crowdsourcing One first idea is, instead of asking people what the answer is (is this correspondence correct), asking them what could be an answer. For that purpose it is necessary to ask them choosing between several alternative relations.

Using jointly exhaustive and pairwise disjoint (JEPD) relations (R) is a proper way to ask such questions. Moreover, uncertainty may be represented within alignments through algebras of relations [1].


Check all relations that may eventually be true:

House
□ ⊐
□ ≡
☑ ⊏
□ ⊥
☑ ≬
Building

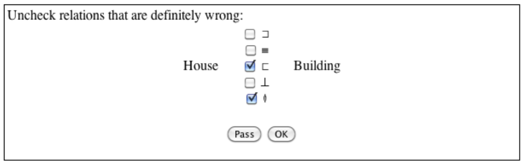
$dc_w(t) = \{\sqsubseteq, \between\} \equiv$

House ⊑ Building

∨ House ≬ Building

This will require slightly more work from workers, but they will not require them to choose between alternatives when they do not see any clear correct one.

Complement crowdsourcing One further possibility, instead of asking people what could be an answer is to ask them what is definitely *not* an answer. In this second setting, it may be easier for people to provide meaningful information without needing to commit to one particular answer.



Uncheck relations that are definitely wrong:

House

□ ⊐
□ ≡
☑ ⊏
□ ⊥
☑ ≬

Building

$$\begin{gathered} cc_w(t) = \{\sqsubseteq, \between\} \equiv \\ \neg(\mathsf{House} \sqsubseteq \mathsf{Building}) \\ \land\neg(\mathsf{House} \between \mathsf{Building}) \end{gathered}$$

Complement crowdsourcing is logically the complement of disjunctive crowdsourcing. However, we conjecture that this will make workers adopt a cautious attitude, discarding only relations that they really think are wrong.

Summary This is related to the consensus between experts [3]. In the initial case, if they do not choose the same relation, they disagree. In the two latter schemes, as long as the intersection between their choices are not disjoint, they do not disagree, but express disjunctive opinions.

With a population W of workers, classical crowdsourcing asks if one relation is true or what is the relation between two entities. So, the result of the task c_w is a single relation. Disjunctive crowdsourcing asks which relations could hold, hence, $dc_w \subseteq R$. Similarly, complement crowdsourcing asks which relations do not hold, hence $cc_w \subseteq R$. We conjecture that:

$$\forall w \in W, \overline{cc_w}(t) \supseteq dc_w(t) \supseteq \{c_w(t)\}$$

This would have the good feature to provide better opportunity for consensus because:

$$\cap_{w\in W}\overline{cc_w}(t) \supseteq \cap_{w\in W} dc_w(t) \supseteq \cap_{w\in W}\{c_w\}(t)$$

It would be an interesting experiment to check if these modalities allow for less conflicts and more accurate alignments. We could test the hypothesis that if it is better to ask users to choose one relation between two entities or to discard nonapplyable relations among all the possible ones.

References

1. Jérôme Euzenat. Algebras of ontology alignment relations. In *Proc. 7th international semantic web conference (ISWC), Karlsruhe (DE)*, pages 387–402, 2008.
2. Cristina Sarasua, Elena Simperl, and Natalya Noy. CrowdMAP: crowdsourcing ontology alignment with microtasks. In *Proc. 11th ISWC*, volume 7649 of *Lecture notes in computer science*, pages 525–541, 2012.
3. Anna Tordai, Jacco van Ossenbruggen, and Bob Wielinga. Let's agree to disagree: on the evaluation of vocabulary alignment. In *Proc. 6th International Conference on Knowledge Capture (K-CAP)*, pages 65–72, Banff (CA), 2011.