IST501 Critique

November 9, 2014

1 Human-powered Sorts and Joins

1.1 Summary
This paper focused on unique presentations of data in crowdsourcing platforms to improve task efficiency and reliability. They developed a tool to automatically translate a task with certain parameters into a cost-effective implementation on the Amazon Mechanical Turk platform. The two tasks they focused on were sorts (ordering a set of items i.e. least people to most people) and joins (combining items into a set i.e. photos of Justin Timberlake). They relied on methods like batching (putting multiple subtasks as part of a single subtask) in order to increase cost effectiveness. Additionally, they found that comparisons achieve better results for sorts over numerical ratings (at the cost of computational complexity and cost). They further attempted to combine them to optimize cost and efficiency.

1.2 Pros
The paper is interesting and relatively easy to read. The graphics are very helpful and help to explain their ideas and results very well.

1.3 Cons
The main con with the paper is their decision to stick with database-specific language in a human-centric task. While people may know (or could probably figure out) what a "join" is, there’s little reason to stick to that terminology, though I understand it is because the data itself is coming from a database. The interface they provide itself is somewhat hard to follow. While the SQL itself makes sense, the way they define the tasks in HTML seems a bit confusing.

1.4 Recommendations
I think the biggest improvement might be divorcing themselves from database terminology and syntax. I understand that there’s something cute about it, but it limits the audience of the paper needlessly. This is also true of their interface: using an HTML renderer would make creating a GUI pretty easy, and having an API does not seem in some way better than a GUI for their application.

2 Max Algorithms in Crowdsourcing Environments

2.1 Summary
This paper focuses on the general problem of computing max problems (i.e. finding the "best" something) by using crowdsourcing. To accomplish this, they look at several variables: types of workers, degree of error, length of time, and monetary cost. Then, they look at ways to tune algorithms based on these parameters. The main algorithms they looked at were BubbleMax and Tournament Max. They also looked at various
ways to model human error in performing these tasks, but note that has no effect on what algorithm / parameterization is optimal and just changes the accuracy values.

2.2 Pros
The results were fairly interesting, where they note that it’s much more expensive to find the top 1 than the top 3.

2.3 Cons
The organization of the paper is not great. The graphics are not great. The paper is very information dense and is filled with too many things that draw your eyes in a poor way.

2.4 Recommendations
The result boxes really aren’t a great method for organizing their results, especially since they rely on the preceding text to make very much sense. To improve this, I would recommend a section that summarizes the results and is clear without reading it, as a good place to start the reading. Additionally, the generality of their work made it unclear exactly what the main argument of their paper was, so narrowing it and focusing might have made it more clear.