Example Journal, XXXX, XX: 1–14

Fine-tuning Gold Questions in Crowdsourcing Tasks using Probabilistic and Siamese Neural Network Models

José María González Pinto[1], Kinda El-Maary[1] and Wolf-Tilo Balke[1]

[1]*Institute for Information Systems Technische Universität Braunschweig Braunschweig, Germany, pinto,elmaarry,balke@ifis.cs.tu-bs.de*

ABSTRACT

The economic benefits of crowdsourcing have furthered its widespread use over the past decade. However, increasing numbers of fraudulent workers threaten to undermine the emerging crowdsourcing economy: requestors face the choice of either risking low-quality results or having to pay extra money for quality safeguards such as gold questions or majority voting. The more safeguards injected into the workload, the lower are the risks imposed by fraudulent workers, yet the higher are the costs. So, how many of them are actually needed? Is there a generally applicable number or percentage? This paper uses deep learning techniques to identify *custom-tailored numbers of gold questions per worker* for individually managing the cost/quality balance. Our new method follows real-life experiences: the more we know about workers before assigning a task, the clearer our belief or disbelief in this worker's reliability gets. Employing probabilistic models, namely Bayesian belief networks and certainty factor models, our method creates worker profiles reflecting different a-priori belief values, and we prove that the actual number of gold questions per worker can indeed be assessed. Our evaluation on real-world crowdsourcing datasets demonstrates our method's efficiency in saving money while maintaining high-quality results.

Keywords: Quality Control, Crowdsourcing, Gold Questions, Probabilistic models, Deep Learning

ISSN 2332-4031; DOI 10.1561/106.XXXXXXXX

1 Introduction

In recent years, several hybrid solutions for augmenting and extending traditional database capabilities with intelligent human steering have been developed. For example 1) processing queries that cannot be adequately answered by database systems (Franklin *et al.*, 2011), such as skyline queries (Lofi *et al.*, 2013), top-k and group-by queries (Davidson *et al.*, 2013; Zheng *et al.*, 2015), and 2) dealing with missing information (Nieke *et al.*, 2014). The benefits for data science come as no surprise, since crowdsourcing is relatively cheap, agile, and offers an intelligent and global 24/7 online labor pool. However, the anonymity of workers on the platforms and the short-term nature of the work contracts also invite *fraudulent misuse* threatening to cancel the benefits. In contrast to traditional workplaces, crowdsourcing requestors do not know much about the workers they are hiring: there is no interview process, no CVs, no personal impressions. In the best case for crowdsourcing, platforms offer *reputation scores* from previous work. Unfortunately, it has been shown that reputation systems only work for *long-standing* relationships. As an analogy, remember peer-to-peer networks: also here reputation systems have been proposed to combat malicious peer behavior, see, e.g., (Aberer and Despotovic, 2001; Kamvar *et al.*, 2003). However, meaningful scores were difficult to construct since reputations suffered from the cold start problem (Daltayanni *et al.*, 2015) and were easy to fake (Yu and Singh, 2003). These problems become even more pronounced in crowdsourcing due to the high attrition rates of workers (Ross *et al.*, 2010) reports about 70%, i.e., relationships tend to be *even more short-termed* than in peer-to-peer systems. Hence, crowdsourcing requestors usually favor on-the-run methods to instantly judge workers leveraging the limited amount of information provided. For the rest of this paper and without loss of generality we will focus only on *gold questions* as a quality mechanism, i.e., questions whose correct answers are known to the requestor and where failing to answer a preset number of them indicates fraud. Until now the problem of "how many gold questions to use?" has no definitive answer (Liu *et al.*, 2013). Obviously, there is a cost/quality trade-off: the more gold questions are used, the better the output's quality will be, yet the higher the costs are. If too many *gold* questions are posed after a certain number of questions, the returned benefit becomes minimal, and the costs become unjustifiably high. On the other hand, if too little *gold* questions are posed, the returned result remains inconclusive in determining the worker's reliability or fraudulence.

In this paper, we develop a *worker-aware ad hoc method that exploits the limited information known about the short-term hired workers*. We transform this information into a digital personal impression indicating whether a worker matches the profiles of fraudulent or reliable workers. Profiles of fraudulent workers come at a higher risk, and should thus be tested more rigorously, whereas profiles of reliable workers come at a lower risk and need only be loosely tested. Our goal with our proposed approach is not to discriminate workers but rather to adjust the number of gold questions to assess more realistically the quality of the work that he or she is delivering. In other words, our method does not attempt to exclude workers, but

rather to *instantiate* more quality controls in case of fraudulent behavior. However, in the case of false positives task providers will have to decide on the acceptance rate on the gold questions to minimize the impact of exclusions. Accordingly, for our method's technical implementation we turn to well-established approaches for uncertainty management. Namely, we investigate the usage of both: 1) probabilistic certainty factor models (Stadler, 2004) pioneered for uncertain deduction of diagnoses by medical expert systems.

The model reflects the *relative change of belief and/or disbelief* in some hypothesis given new observations, i.e., in our case the relative change in the high/low risk associated with a worker. 2) Bayesian Belief Networks, which express uncertainty through probabilities. Using either of these models allows building a system of decision rules, which create digital impressions of each worker and what we refer to as *enhanced worker profiles*. These worker profiles do not only encode what is known about a worker but also assess the risks involved with a worker by either 1) relative apriori belief or disbelief measures as given by the certainty factor model or 2) probabilities as given by the Bayesian belief network. Our underlying assumption is that the higher the belief value or the probability in a worker's reliability, the lower the risk, and the less gold questions may be used. The higher the disbelief value or, the lower the probability in a worker's reliability, the higher the risk, and the more gold questions should be used. By personalizing the number of questions to be asked based on these enhanced profiles, we can then save some money, while maintaining a high level of result quality. To realize this underlying idea, we identify the exact number of gold questions to be asked (the *gold par*), by fitting an exponential distribution of number of questions to be asked on top of the values of belief/disbelief and probabilities.

In this paper, we extend our work presented in (Maarry and Balke, 2018) to explore the potential and limitations of *latent representations* based on neural networks in our quest to distinguish between fraudulent and reliable workers. Indeed, distinguishing between reliable and fraudulent workers is a fundamental component of our proposed approach to adjust the number of gold questions dynamically. Thus, given the success of Deep Learning models in fields such as computer vision and natural language processing, we test in this work their applicability to our particular setting. In particular, given the limited amount of both data and attributes, we restricted our investigation to a particular neural network architecture called 'Siamese Networks' (Section 4.3) that we hypothesized could be a good fit for our problem. We demonstrate the applicability of our method on real-world crowdsourcing test data of 200 workers and compare how well it performs regarding overall result quality, the effectiveness of the algorithm, fraud detection's failure rates, and discrimination rates against reliable workers.

However, further work could be done, if data is available, to account for a different type of spammers: *colluding spammers*. As introduced in (Checco *et al.*, 2018), colluding spammers can undermine the use of gold questions. In particular, as the authors demonstrated in their paper, it is feasible to build and deploy a system that can detect which parts of a crowdsourcing job are more likely to be gold questions (Checco *et al.*, 2018). Given that it is unlikely to get more attributes for building our profiles, such an attack in gold questions will be for us relevant to investigate if we can get access to training data with such "colluding spammers". Otherwise, our main assumption high-performance on gold questions leads to low-risk workers will not work as expected.

2 Safeguards in Practice

Many safeguards for quality issues in crowdsourcing systems have been investigated. We identify four families of safeguards:

Pessimistic safeguards ensure high quality by directly identifying fraudulent workers and excluding them. The most common approach in this family are *gold questions* randomly injected into the workload. Failing to answer a preset number of these questions (whose correct answers are known by the system), declares the corresponding worker as fraudulent, in turn leading to exclusion. These safeguards are typically *worker-oblivious*, i.e., no distinction in the underlying testing mechanism is made for different workers. A notable exception is skill-adapted gold questions (Maarry and Balke, 2015) and adaptive gold questions (Maarry *et al.*, 2015), which aim at adapting gold questions to the underlying skills of workers for a fairer judgment of workers in alignment with the vision of impact sourcing.

Optimistic safeguards ensure high quality by aggregating the results of multiple workers on a given task. The best-known aggregation method here is majority voting. Other weighted aggregation methods in the literature include the expectation maximization (EM) algorithm (Dawid and Skene, 1979), a Bayesian version of the EM algorithm (Pearl, 1985), and a probabilistic approach in (Whitehill *et al.*, 2009). This family is more *worker-aware*, as it tries to identify the workers' reliability and may distinguish different levels of skills, which can then be incorporated as weights in the final step of aggregation.

Feedback-based safeguards ensure high quality by monitoring the history of workers and their outputs' feedback Ignjatovic *et al.*, 2008, thus making it also a *worker-aware* family of safeguards. A typical example of this family is reputation-based systems, whether based on a reputation model (Krieg, 2001; Lofi *et al.*, 2013) or deterministic approaches (Noorian and Ulieru, 2010).

Incentive-based safeguards ensure high quality by motivating the workers either intrinsically or extrinsically (Hossain, 2012). Intrinsic refers to motivations inherent to the task itself, e.g., Zooniverse[1]. Extrinsic refers to external motivations that offer some reward, e.g., monetary rewards (Kazai, 2011).

Our proposed method falls under the *pessimistic safeguard* family but is a *worker-aware* method. Moreover, in contrast to the pessimistic safeguards, our method is designed to adapt to each new worker, by re-computing the sufficient number of gold questions needed to distinguish between reliable and fraudulent workers.

The most relevant work on adaptive quality control is (Liu *et al.*, 2013) who investigated the universal number of gold questions needed. Although they concluded that the problem

[1] https://www.zooniverse.org/

is unlikely to reach a definitive answer, their work provides a *rule of thumb* for the optimal number of gold questions to be used: either *linearly* with or following the *order of the square root* of the total size of the task given to a worker. The choice of either rule of thumb depends on the corresponding level of aggregation used: two-stage or joint inference. Since this is closest to our work, we designated it as a baseline and compared our results against the order of the square root scaling rule for the optimal number of questions to be used. We compare among others the overall number of gold questions needed, the overall accuracy rate of the gold questions as a safeguard for distinguishing between reliable and fraudulent workers (see Section 5).

3 Creating Worker Profiles

We can formally concisely define the problem as follows

Problem Definition. *Given a crowdsourcing task T comprising n questions, we want to find the minimal number of gold questions m, such that 0<m<n needed to determine the reliability of a given worker w. A reliable worker is defined as a worker, whose accuracy rate on a crowdsourcing task >0.75*

Our method can be divided into three necessary steps:

1. Creating a system of *decision rules*: the enhanced worker profiles. For building these profiles and their corresponding encoding of associated risks, we experiment with two approaches: a) Bayesian belief networks, which encode risks with probabilities and b) certainty factor models, which encode risks with relative apriori measures of belief and disbelief. We turned to both of these models, as they are well-established approaches for uncertainty management. The uncertainty in our problem materializes in our attempt to reason whether a new incoming worker is reliable or not based on uncertain knowledge.
2. Identifying the minimum required number of gold questions for each enhanced worker profile, the *gold par*.
3. Mapping the workers to their corresponding enhanced profile.

In this section, we focus on the first step of creating enhanced worker profiles; Section 4 covers the second and third step. Starting with certainty factor models and Bayesian belief networks, we map and redefine both models' parameters to our crowdsourcing setup.

3.1 The Certainty Factor Model (CF)

The probabilistic certainty factor model (CFM) was first developed by (Shortliffe and Buchanan, 1975) for MYCIN, a medical expert system employing certainty factors (CF) for uncertain deduction within heuristic systems. In essence, CFs do not correspond to probabilities, but rather depict the relative change of belief and/or disbelief in some hypothesis H given a certain observation E. The combinations of these **M**easures of **B**elief $MB(H|E)$ and **D**isbelief $MD(H|E)$ constitutes the CFs. These measures are relatives and are not to be confused with probabilities. Nevertheless, their values are normalized to span between $[0, 1]$, with 1 representing the highest belief or disbelief with respect to a certain hypothesis H , and 0 representing the lowest belief or disbelief, again with respect to a certain hypothesis H. Moreover, these measures are individually observed, i.e. for $MB(H|E) = x$ and $MD(H|E) = y \implies x + y = 1$

Definition 1 MB, MD and CF. *Given an observation E and a Hypothesis H, we can compute the $MB(H|E), MD(H|E)$, and the $CF(H|E)$*

$$MB(H|E) = \begin{cases} \frac{max[P(H|E),P(H)]-P(H)}{1-P(H)} & \text{if } P(H) \text{ is} \neq 1 \\ 1 & \text{otherwise} \end{cases}$$

$$MD(H|E) = \begin{cases} \frac{P(H)-min[P(H|E),P(H)]}{P(H)} & \text{if } P(H) \text{ is} \neq 0 \\ 1 & \text{otherwise} \end{cases}$$

$$CF(H|E) = MB(H|E) - MD(H|E)$$

In other words, $CF(H|E)$ can also be formulated as follows:

$$CF(H|E) = \begin{cases} \frac{P(H|E)-P(H)}{1-P(H)} & \text{if } P(H|E) \geq P(H), P(H) \text{ is} \neq 1 \\ \frac{P(H|E)-P(H)}{P(H)} & \text{if } P(H) \geq P(H|E), P(H) \text{ is} \neq 0 \end{cases}$$

The CF rules' value span between $[-1.0, 1.0]$. Subsequently, we distinguish two types of rules:

1. *Confirming CF rules*: are those having a high measure of belief, i.e., positive certainty factor value $CF(H|E) \geq 0$.
2. *Disconfirming CF rules*: are those having a high measure of disbelief, i.e., negative certainty factor value $CF(H|E) < 0$.

Given a set of *CF* rules, new rules/deductions can be automatically drawn: 1) chaining and 2) parallel combination. The latter is of particular interest: it consolidates different observations leading to the same hypothesis. This allows us to create more complex *CF* rules, which combine several independent observations. Parallel combination can be efficiently computed from the rules directly; that is, there is no need to go back to the data for computations.

Definition 2 Deduction by Parallel Combination. *Given two CF rules: $CF_{E_1}(H)$ and $CF_{E_2}(H)$ where two observations E_1 and E_2 lead to the same Hypothesis H. A new $CF_{E_1E2}(H)$ can be deduced by parallel combination as follows:*

$$CF_{E_1E2}(H) = \begin{cases} x + y - x * y & \text{for } x \geq 0, y \geq 0 \\ x + y + x * y & \text{for } x \leq 0, y \leq 0 \\ \frac{x+y}{1-min(|x|,|y|)} & \text{for } -1 < x * y < 0 \end{cases}$$

where $x = CF_{E_1}(H)$ and $y = CF_{E_2}(H)$

In case of combining more than two *CF* rules with different observations, the above definition applies by taking the result of the first two combined *CF* rules and designating it as x when combining it with the next *CF* rule and so on (Mellouli, 2014).

3.2 Bayesian Belief Network (BN)

Bayes nets belong to the family of probabilistic graphical models and are used to represent/infer knowledge about an uncertain domain (Pearl, 1985; Pearl, 1988). Somewhat similar to our crowdsourcing problem, Bayes nets have been applied to the target recognition problems, where transponders should be identified as Friend or Foe (Krieg, 2001).

Bayes nets encode a directed acyclic graph G. The directed property of the graph complies with our problem, since our underlying idea is that observing certain attributes of the workers lead to a certain belief value in the reliability of a worker. Accordingly, this directed relationship from the observations to the hypothesis can be represented in Bayes Nets with a directed graph. Formally, Bayes nets B is given by the pair $B =< G, \theta >$. The graph G is made up of a set of random Variables V, depicted by n nodes $x_1, x_2, ..., x_n$, and directed edges $\overrightarrow{E} : x_i \rightarrow x_j$. The underlying semantics of Bayes nets, namely, the *local markov property* defines an ordering of the nodes such that only the nodes indexed lower than i can have a directed path to x_i. The nodes are the worker's observations and the hypothesis to be inferred, i.e., the reliability of the worker. There are different types of nodes: Root, Parent and Child nodes. The edges E represent the probabilistic dependency between the nodes. For discrete variables, the relationship between them is given by the conditional probability distribution. The second parameter θ depicts the full joint distribution as follows:

Definition 3 Full Joint distribution of BN. *The full joint distribution for a Bayes net B having n nodes $x_1, x_2, ..., x_n$, can be defined by the product of the local conditional distributions*

$$P(x_1, x_2, ..., x_n) = \prod_{1 \leq i \leq n} P(x_i | Parents(x_i))$$

There are two types of reasoning: *Predictive support* and *diagnostic support*. Predictive support is top-down reasoning starting from the parents' node to the child node, while diagnostic reasoning is bottom-up reasoning starting from the child node. Since we aim to infer whether a worker is reliable, i.e., inferring the child node/hypothesis, we follow the predictive support inference (see Definition 4). In our crowdsourcing setup, the random variables V are the workers' observations and are depicted by the parent nodes. The parent nodes in our case also happen to be the root nodes, since they have no predecessor nodes, while the hypothesis is the child node. Each root node has a prior probability distribution.

Definition 4 Inferencing in BN. *We can infer the strength of our hypothesis H having seen a worker's observation E using the Bayes nets conditional probability formula:*

$$P(child|parent) = \frac{P(child, parent)}{P(parent)}$$

i.e.

$$P(H|E) = \frac{P(H, E)}{P(E)}$$

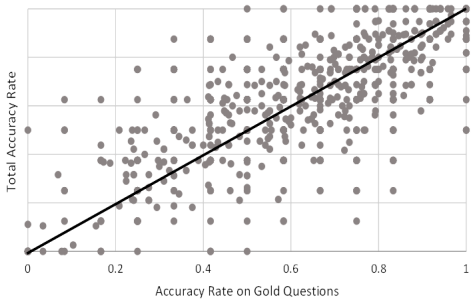

Figure 1: Correlation between workers' gold question's accuracy and overall accuracy rate

3.3 Formulating the Hypothesis: H

For our crowdsourcing setup, the hypothesis is always the same. Namely, given a pool of workers W, a worker $w \in W$ is a reliable worker. It follows then that workers fitting low risk profiles (i.e., profiles with a positive CF value, where $MB(H|E) > MD(H|E)$ or profiles with $P(H|E) \geq 0.5$ come at lower risk, while those fitting high risk profiles (i.e. profiles with a negative CF value, where $MB(H|E) < MD(H|E)$ or profiles with $P(H|E) < 0.5$ come at higher risk (more on how to map a worker to a worker profile is explained in Section 4.2). But the question of how to define such a reliable worker instantly arises. The difficulty of this question lies within the scarcity of the data the requestor has on a particular worker, who is more often than not, a new worker. Currently, the only ad hoc quantitative metric available to the requestors is the accuracy rate of a worker on the gold questions.

In general, we seek workers whose overall accuracy rate is higher than 75%. An initial correlation investigation between the accuracy rate on the gold questions and the overall accuracy rate on the whole task shows, as expected, a high positive correlation of 0.7 (see Figure 1). Outliers can also be observed, which is attributed to 1) strategic spammer schemes, where they always submit the frequent answer label, 2) inherently small crowdsourcing tasks, e.g., five tasks and one gold question. We ran this experiment on real crowdsourcing datasets comprised of 1006 workers, with 40% gold questions (see Section 5.1).

Accordingly, we generally define the hypothesis that a worker is reliable if he/she attains at least 75% accuracy rate on the gold questions. Eventually, however, workers fitting low-risk profiles are assigned less gold questions, while workers fitting high-risk profiles are assigned more gold questions. Consequently, we vary the expected quality thresholds such that workers fitting the disconfirming profile with the lowest value, i.e., $CF(H|E) \rightarrow -1.0$ or $P(H|E) = 0$, should attain at least 75% quality rate, while workers fitting profiles with higher values should attain higher quality rates. The idea is to decrease the discrimination rate against workers fitting high risk profiles, while still maintaining the threshold quality. On the other hand, workers fitting low risk profiles should prove their reliability even more so by scoring higher quality thresholds. Thus, we uniformly fit the quality thresholds to be attained on the belief/disbelief values and probabilities, such that the thresholds range between 75% - 100%, where the high risk worker

profile: $CF(H|E) \rightarrow -1.0$ or $P(H|E) = 0$ should attain at least 75% accuracy rate on gold questions to be considered reliable, while the low risk worker profile: $CF(H|E) \rightarrow 1.0$ or $P(H|E) = 1$ should attain a perfect 100% accuracy rate to be considered reliable. In practice, such a perfect profile does not exist.

3.4 Formulating the Observations: E

Observations capture the limited information we know about workers. Of course, all attributes available in crowdsourcing platforms can and should be exploited for best discrimination accuracy. As all crowdsourcing tasks were run on the CrowdFlower[2] platform (see Section 5.2), below we list the publicly available attributes offered by CrowdFlower. For each attribute, we also investigated its domain to find out which of its instances should be considered as an observation. An attribute's instance is a valid observation if it is *frequent* since both CF rules and Bayesian conditional probabilities become unreliable if based on sparse observations.

1. *Channel*: There are 30 different crowdsourcing channels, from which workers are hired. Only eight channels, however, are dominating our labor quota, leaving 22 channels providing only around 4.3% of the total workforce (see Figure 2). Accordingly, we only use the top 8 channels as observations, which constitute 95.6% of the data.
2. *Country*: In total, we have workers from 75 different countries. Figure 3 shows a clear Zipfian distribution, with 85% of workers coming from only 24 countries. We limit our observation to these 24 countries ignoring the distribution's tail.
3. *Started_at*: this attribute marks the time (GMT) at which a worker started working. On its own, this attribute would not make much sense, but rather in combination with the country, since it would indicate whether working in the morning, evening or night is more reliable. As seen in Figure 4, about 88.7% of the workers worked between 08 am - 6 pm. We used these hours for our observations.
4. *City*: the city attribute proved too sparse, as it is only available for 71% of the workers (i.e., 716 workers). Moreover, it exhibited an extremely long-tailed distribution of 462 cities. The head of the distribution, on the other hand, had two cities: Caracas and Belgrade, comprising 7% of the workers. Accordingly, we chose to disregard the city attribute all together as a discriminating observation.
5. *Trust*: the trust attribute ranges between 0.0 and 1.0 and is computed by the platform based on the last task a worker performed. For this attribute, all values proved sensible to be taken as observations. We aggregated the values by grouping them into intervals of 0.1, thus yielding ten values of trust.

Overall, our models' observations comprise eight channels, eight different hours to work within, 24 countries and ten levels of trust.

[2] https://www.crowdflower.com/

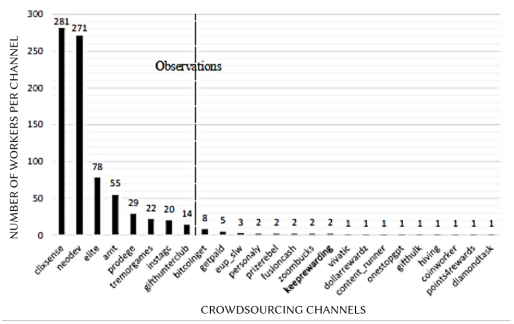

Figure 2: Channel attribute domain analysis

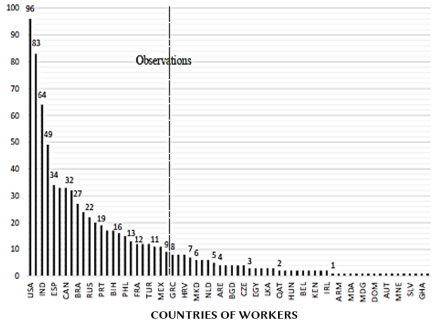

Figure 3: Country attribute domain analysis

4 Towards the Gold Par

After computing the enhanced worker profiles, the next step is to determine how many gold questions are sufficient per profile, depending on the profiles' encoded risk.

4.1 Mapping Profiles to the Gold Par

Following our notion that low-risk workers should be asked less gold questions than high-risk workers, both the uniform and the exponential distribution could mimic this notion when fitted on top of the enhanced worker profiles, such that at the worst case scenario, e.g., $CF(H|E) \rightarrow -1.0$, 50% gold questions should be asked, and at the best case scenario, e.g. $CF(H|E) \rightarrow 1.0$, only 1 gold question needs to be asked. Note that such a perfect profile does not exist in practice.

The exponential distribution, however, has a lower discrimination rate than the uniform distribution, since low-risk workers are given more gold questions, thus more chances, to break away from their high-risk profile. Moreover, the exponential distribution also takes into account, that gold questions could be imbalanced and that some of them might be more difficult i.e. honest workers fitting high-risk profiles might end up getting the short end of the stick. Whereas, workers fitting low-risk profiles get exponentially less number of questions, which also decreases the overall costs of utilizing safeguards.

Experimenting with various exponential distributions having different rate parameters yielded the best results with the exponential distribution $f(x, \lambda) = \lambda e^{-x\lambda}$, where $\lambda = 2$ (see Figure 5). As discussed in Section in 3.3, although workers fitting low risk profiles get a less number of gold questions,

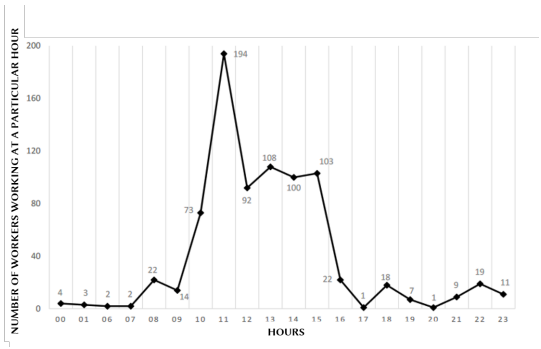

Figure 4: Started_at attribute domain analysis

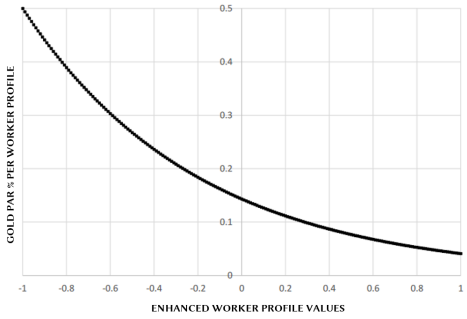

Figure 5: Exponentially Fitting the Gold Par to the enhanced worker profiles

they are expected to score higher accuracy rates. This is similar to real world situations, where workers compete with front runners in their field.

4.2 Mapping Workers to their Profile

After computing the enhanced worker profiles for all the different observations and all the different combination of observations, these profiles could be stored in a database, against which new incoming workers can be mapped to. The mapping of a new worker to a profile is based on the matching of observations. Since we combine observations to create more complex profiles, a worker may fit multiple profiles which could be low risk or high risk. There are multiple strategies here to choose which profile to use:

- Optimistic mapping, where the worker is mapped to the enhanced worker profile with the highest belief value or probability. Here, workers are given the benefit of the doubt, and they are assigned less gold questions.
- Pessimistic mapping, where the worker is mapped to the enhanced worker profile with the highest disbelief value or lowest probability. Here, a more skeptical approach is taken, and the workers are subject to more gold questions.

In our evaluation section, we tested both the optimistic and pessimistic mapping. As to be expected, the pessimistic mapping is more expensive, since more safeguards are used.

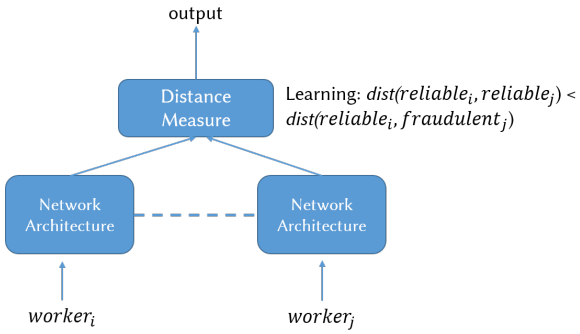

Figure 6: The Siamese Architecture

4.3 Siamese Neural Network Architecture

In this section, we introduce the Siamese Neural Network Architecture that we use in this work. We will follow the definition provided by (Bromley *et al.*, 1993) where researchers introduced this neural network architecture. In (Bromley *et al.*, 1993) researchers established the Siamese network as "a neural network architecture that consists of two identical subnetworks joined at their outputs. The Siamese network has two input fields to compare two patterns and one output whose state value corresponds to the similarity between the two patterns". In short, the Siamese network tries to learn to differentiate between two inputs. Herein, we try to learn to differentiate between reliable and fraudulent workers (Figure 6).

Since its inception in (Bromley *et al.*, 1993) different tailored Siamese networks have been deployed to target different problems successfully. Two major applications of the Siamese network can be found in semantic tasks in Natural Language Processing and of course in Computer Vision. In the following discussion, we review relevant work that motivated our design decisions. For instance, in (Mueller and Thyagarajan, 2016) researchers presented a Siamese architecture using Long Short-Term Memory networks (LSTMs) as the sub-networks to learn a semantic distance between pair of sentences. The Siamese architecture outperformed strong baselines in the semantic task. The model used the Manhattan distance on top of the learned shared representations to make predictions. The success of the model in (Mueller and Thyagarajan, 2016) inspired one of our tailored models that uses the same distance measure but with a different sub-network architecture. In a similar line of work a Siamese architecture described in (Das *et al.*, 2016) tackled the challenge of finding similar semantic questions in online community question-answering systems. Instead of the Manhattan distance, researchers in (Das *et al.*, 2016) used the contrastive loss function first introduced in (Hadsell *et al.*, 2006). Moreover, the Siamese architecture used as a sub-network a Convolutional Neural Network (CNN). Researchers have also applied variants of the Siamese architecture in computer vision. For instance, in (Koch *et al.*, 2015) researchers explored the idea of using Siamese based architecture in a one-shot learning setting for image recognition. As mentioned by the researchers one-shot learning implies that "we may only observe a single example of each possible class before predicting a test instance"(Koch *et al.*, 2015). The strategy adopted in the paper is to train a model that learns to distinguish between same and different

pairs for the n categories given in the data. This means that a testing time the model is evaluated by a verification task to determine to which category a given test image belongs to out of all the possible categories that exist. What is different from the previous approaches that we mentioned above is that instead of trying to learn a distance between the outputs, the researchers concatenated the feature vector learned in the sub-network and the pair-wise difference between them as the final layer of a sub-network that uses CNN-based approach. Some other challenging task in computer vision such as person re-identification from multiple camera views have also used Siamese network-based models such as the work of (Shen *et al.*, 2017). Yet another domain where Siamese networks have been applied is as part of an information system pipeline in non-factoid question-answering problems such as in (Tran and Niederée, 2018).

Regardless of the specific problem, all the research efforts share the challenge of finding a specific sub-network architecture with specific hyperparameters to accurately deter-mine how different or similar the two inputs are. In our work, we want to show the potential of such architecture to the problem of distinguishing accurately between reliable and fraudulent workers. The primary technical challenge that we face here is the limited set of attributes available.

We can formally define the problem that we want to solve using the Siamese architecture as follows:

Problem Definition. *Given a training set of vector representation of pairs of workers $< w_i, w_j >$ comprising n positive pairs and m negative pairs, we want to find the sub-network architecture with its hyperparameters to determine the reliability of a new unseen worker w_k from a test set. Here positive pairs refers to pairs of reliable workers and negative pairs refers to pairs where one worker is not reliable.*

To solve the problem, we focus in this work on an exploration of Feedforward Networks also known as multilayer perceptrons (MLPs) as the building block of our proposed subnetwork architecture.

Once we have trained a model, testing involves the following procedure: for each unseen vector representation of a worker of the test set, we select a reliable random worker from our training set to generate the pair that the trained model will assess.

For self-containment, we include in the following paragraphs the definition of MLPs that one can find in (Goodfellow *et al.*, 2016), in particular, the terminology discussed in Chapter 6. Thus, readers already familiar with MLPs can skip the following subsection, or those interested in a depth discussion of the topic should read (Goodfellow *et al.*, 2016).

4.3.1 Feedforward Networks

Feedforward networks also are known as multilayer perceptrons (MLPs), are learning models whose goal is to approximate some function f. For instance, for a classifier that aims at learning if an email is spam or not, $y = f(x)$ maps an input x to a category y. A feedforward network defines a mapping $y = f(x; \theta)$ and learns the value of the parameters θ that results in the best function approximation. These models are called feedforward because information flows through the function being evaluated from x, through the intermediate computations used to define f, and finally to the output y. There are no feedback connections in which outputs of the model are fed back into itself. These networks are represented by composing together many different functions. The model is associated with a directed acyclic graph describing how the functions are compose together. The overall length of the change gives the depth of the model. People refer to the functions as the layers of the network. The final layer of a feedforward network is called the output layer.

These networks are trained in a supervised fashion using labeled training data. Training a specific network involves specifying a loss function that measures the performance on the training data and an optimizer which the network will use to update itself based on the data it sees and its loss function using a specific variant of stochastic gradient descent (SGD).

5 Evaluation

We now demonstrate the applicability/efficiency of our method in saving costly safeguards while maintaining high quality in real crowdsourcing tasks. We compare our method against the baseline in (Liu *et al.*, 2013), to which we refer henceforth as the 'Optimal K' method.

5.1 Data and Crowdsourcing tasks' Overview

For six different datasets, we designed a crowdsourcing task and posted a total of 25 jobs on the CrowdFlower crowdsourcing platform. We chose quite a heterogeneous set of crowdsourcing tasks to generate a universal set of enhanced worker profiles:

1. *Sharpness Image dataset*, comprising 192 in-house high-quality images. In total six jobs were submitted to the crowd, each job had 48 questions. The crowd was given five versions of the same picture and were asked to order them according to their level of sharpness. A total of 184 workers were hired.
2. *Definition datase*t, crawled from the verbal practice questions section of the Graduate Record Examination (GRE) dataset[3] 2015. 176 questions were assigned to 70 workers over five jobs. The crowd was given multiple-choice questions, where correct corresponding definitions of words had to be chosen.
3. *Cars dataset*, crawled from Heise.de[4] in 2011. 125 questions were assigned to 87 workers over seven jobs. The crowd was asked to look up missing data for a particular car model.
4. *The open source "Image descriptions"* dataset[5], comprising 225,000 tuples. 1,320 questions were assigned to 482 workers over three jobs. The workers were shown a large variety of images with a corresponding word. Their task was to identify whether the word matched and described the image.

[3] http://www.graduateshotline.com/

[4] http://www.heise.de/autos/neuwagenkatalog

[5] http://dbgroup.cs.tsinghua.edu.cn/ligl/crowddata/

5. *The open "Semantic relationships between two concepts dataset"*, comprising 3,536 tuples. 50 questions were assigned to 39 workers over one job. Workers were asked to judge whether the semantic equivalence in sentences was correct or not.
6. *The open source "Decide whether two English sentences are related dataset"* , comprising 555 tuples. A total of 730 questions were assigned to 404 workers over three jobs. Given two sentences: a fact and a deduction sentence, the workers had to judge if the deduction sentence were correct.

Throughout the 25 jobs we ran, a total of 1,266 workers were hired. While processing the workers' data, we found that about 35% (i.e., 445) of workers had worked in more than one job. After removing duplicate workers by merging their data, we ended up with 1,006 workers. For evaluation, we split our database of workers into two datasets: A training dataset was used to create the enhanced worker profiles (806 workers), and a test dataset of 200 workers was used for evaluation. For the test dataset, we created five datasets with different percentages of spammers and reliable workers in order to observe their impact on the overall accuracy of our method. To do so, we selected at random from a large pool of workers spammers and no-spammers until we reached the desired ratio. Following our problem definition in Section 3, reliable workers are those workers achieving an accuracy rate higher than 75%.

- Spammers75 (S75): 75:25 ratio of spammers/reliable workers.
- Spammers66 (S66): 66:34 ratio of spammers/reliable workers.
- Balanced (B): 50:50 ratio of spammers/reliable workers.
- Reliable66 (R66): 34:66 ratio of spammers/reliable workers.
- Reliable75 (R75): 25:75 ratio of spammers/reliable workers.

5.2 Populating the worker profiles database

To generate the set of enhanced worker profiles, we used the training dataset comprising 806 workers and used the following attributes as observations: Channel, Country, Started_at, and Trust.

5.2.1 CFM-generated enhanced worker profiles

In total, 16,199 enhanced worker profiles of different granularities were generated. Namely, 47 single observation pro-files (in total we had 50 different observations, but for three trust values, no profiles were generated since the values never occurred given the hypothesis). Moreover, parallel combinations generated the following combined-observation profiles: 728 2-Set Observation profiles, 4,672 3-Set observation profiles, and 10,752 4-Set observation profiles.

Single-Observation Profiles: Figures 7–10 plot CF values for single observation profiles. In Figure 7, highest quality work tends to be done around 12 GMT, i.e., $CF(reliableworker|12) \rightarrow 0.35$. Further analysis uncovered that this work was mostly done by German workers, probably during lunch breaks. In Figure 8, workers hired from Amazon Mechanical Turk seem more reliable than those from gifthunterclub:

$$CF(reliableworker|gifthunterclub) \rightarrow -0.27$$

compare to $CF(reliableworker|AMT) \rightarrow 0.6$

German workers show a high confirming profile:

$$CF(reliableworker|GER) \rightarrow 0.58,$$

where workers from Pakistan have the lowest disconfirming profile: $CF(reliableworker|PAK) \rightarrow -0.77$ (Figure 9). Lastly, in Figure 10 it comes as no surprise that workers with highest trust value 1.0 show confirming profiles:

$$CF(reliableworker|1.0) \rightarrow 0.5$$

while those having the lowest trust value of 0.4 have highest disconfirming profiles:

$$CF(reliableworker|0.4) \rightarrow -0.65$$

Combined-Observation Profiles: Below we show the top generated confirming/disconfirming 3-Set and 4-set observation profiles. The hours encoded in the rules for the Started_at attribute have been converted from GMT to the local time of the corresponding country within the same profile. For profiles without a corresponding country, the time is indicated in GMT format. The decimal numbers are trust values; the whole numbers refer to Started_at observation.

Top Ten Confirming 3-SET Observation Profiles:

1. $CF(reliable\ worker|amt, DEU, 1.0) \rightarrow 0.921$
2. $CF(reliable\ worker|instagc, DEU, 1.0) \rightarrow 0.899$
3. $CF(reliable\ worker|amt, DEU, 16) \rightarrow 0.896$
4. $CF(reliable\ worker|amt, 14, 1.0) \rightarrow 0.876$
5. $CF(reliable\ worker|amt, DEU, 0.9) \rightarrow 0.873$
6. $CF(reliable\ worker|DEU, 16, 1.0) \rightarrow 0.869$
7. $CF(reliable\ worker|instagc, DEU, 16) \rightarrow 0.868$
8. $CF(reliable\ worker|amt, GBR, 1.0) \rightarrow 0.864$
9. $CF(reliable\ worker|amt, DEU, 0.8) \rightarrow 0.858$
10. $CF(reliable\ worker|amt, ITA, 1.0) \rightarrow 0.856$

Top Ten Disconfirming 3-SET Observation Profiles:

1. $CF(reliable\ worker|PAK, 16, 0.4) \rightarrow -0.943$
2. $CF(reliable\ worker|PAK, 19, 0.4) \rightarrow -0.942$
3. $CF(reliable\ worker|gifthunter, PAK, 0.4) \rightarrow -0.941$
4. $CF(reliable\ worker|neodev, PAK, 0.4) \rightarrow -0.935$
5. $CF(reliable\ worker|PAK, 15, 0.4) \rightarrow -0.935$
6. $CF(reliable\ worker|clixsense, PAK, 0.4) \rightarrow -0.928$
7. $CF(reliable\ worker|elite, PAK, 0.4) \rightarrow -0.928$
8. $CF(reliable\ worker|PAK, 13, 0.4) \rightarrow -0.925$
9. $CF(reliable\ worker|PAK, 18, 0.4) \rightarrow -0.92$
10. $CF(reliable\ worker|PAK, 21, 0.4) \rightarrow -0.892$

Top Ten Confirming 4-SET Observation Profiles:

1. $CF(reliable\ worker|amt, DEU, 14, 1.0) \rightarrow 0.949$
2. $CF(reliable\ worker|instagc, DEU, 14, 1.0) \rightarrow 0.935$
3. $CF(reliable\ worker|amt, DEU, 15, 1.0) \rightarrow 0.929$
4. $CF(reliable\ worker|amt, DEU, 16, 1.0) \rightarrow 0.923$
5. $CF(reliable\ worker|amt, DEU, 12, 0.9) \rightarrow 0.918$
6. $CF(reliable\ worker|Prodege, DEU, 12, 1.0) \rightarrow 0.918$
7. $CF(reliable\ worker|amt, GBR, 12, 1.0) \rightarrow 0.9123$
8. $CF(reliable\ worker|instagc, DEU, 15, 1.0) \rightarrow 0.910$
9. $CF(reliable\ worker|amt, DEU, 12, 0.8) \rightarrow 0.908$

10. $CF(reliable\ worker|amt, ITA, 12, 1.0) \rightarrow 0.907$

Top Ten Disconfirming 4-SET Observation Profiles:

1. $CF(reliable\ worker|gifthunter, PAK, 16, 0.4) \rightarrow -0.959$
2. $CF(reliable\ worker|gifthunter, PAK, 19, 0.4) \rightarrow -0.958$
3. $CF(reliable\ worker|neodev, PAK, 16, 0.4) \rightarrow -0.955$
4. $CF(reliable\ worker|neodev, PAK, 19, 0.4) \rightarrow -0.954$
5. $CF(reliable\ worker|gifthunter, PAK, 15, 0.4) \rightarrow -0.953$
6. $CF(reliable\ worker|clixsense, PAK, 16, 0.4) \rightarrow -0.95$
7. $CF(reliable\ worker|elite, PAK, 16, 0.4) \rightarrow -0.95$
8. $CF(reliable\ worker|clixsense, PAK, 19, 0.4) \rightarrow -0.949$
9. $CF(reliable\ worker|elite, PAK, 19, 0.4) \rightarrow -0.949$
10. $CF(reliable\ worker|neodev, PAK, 15, 0.4) \rightarrow -0.948$

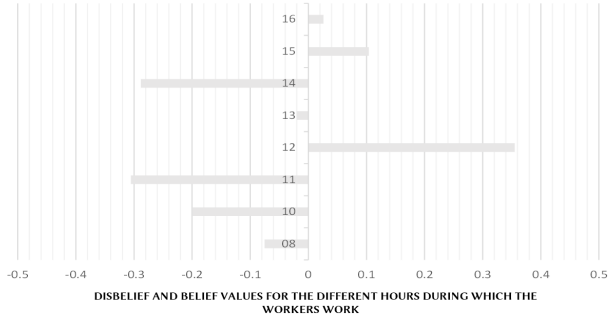

Figure 7: Single Started_at CF Rules

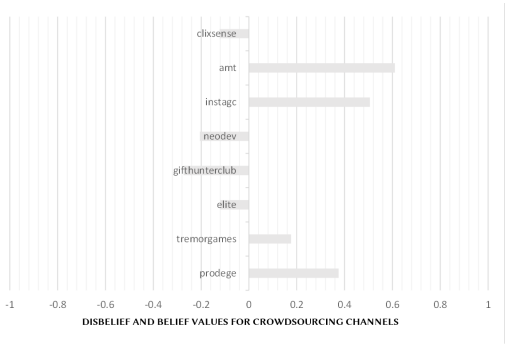

Figure 8: Single Started_at CF Rules

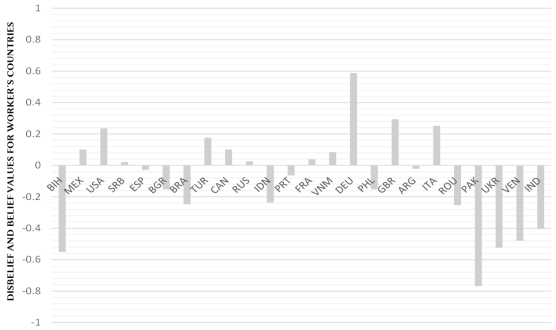

Figure 9: Single Country CF Rules

The combined-observation profiles show similar insights to the single-observation profiles: Work done at 12 GMT by German workers showing trust values higher than 0.8 and hired from the AMT channel, have high belief values in the hypothesis: a low-risk profile. In contrast, work done at 11 GMT by Pakistani workers showing trust values lower than 0.7 and hired from the gifthunterclub channel, tend to have high disbelief values, a high-risk profile.

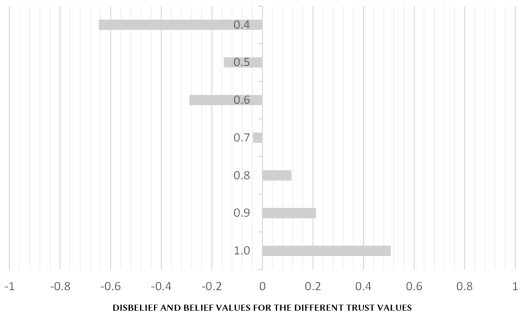

Figure 10: Single Trust CF Rules

5.2.2 BN-generated enhanced worker profiles

In total, 1,454 enhanced worker profiles of different granularities were generated, in particular, 47 single observation profiles. For the Bayesian belief network, these single observation profiles are nothing but prior distributions, i.e., world probabilities that need to be estimated. For the country attribute, we turned to the population statistics of the world[6] to estimate the countries' prior estimations. The other priors, namely Channel, Started_at and Trust, proved difficult to estimate from the data we had. This is one of BN's drawbacks: biases in estimations may easily be introduced.

Moreover, unlike CFM's computation of more complex profiles by parallel combination, BN repeatedly needs to scan the data for generating these profiles. The following combined-observation profiles were deduced: 1169 2-Set Observation profiles, 213 3-Set observation profiles, and 25 4-Set observation profiles. Compared to the CF database of profiles, the BN database is significantly smaller. This comes as no surprise since BN computes conditional probabilities based on actual occurrences of combined-observation: unseen combined observations are not generated. Single-observation profiles for the BN are prior probabilities. For the country observation, we quickly computed it based on real-world data, with P (DEU) = 0.011. Other priors were challenging to get and were accordingly estimated from the dataset. In other words, we used biased empirical priors calculated using the training data. The combined-observation profiles for the BN are on the other hand simple conditional probabilities (see Definition 4).

5.3 Evaluating the Gold Par

Using our test dataset of 200 workers for evaluation, the relatively small database of enhanced worker profiles on average profiled 96.5% of new incoming workers (i.e., 193 workers fit at least one of the enhanced worker profiles) when using the CF

[6] http://www.worldometers.info/world-population/population-by-country/

database, and 94% of new incoming workers (i.e. 188 workers) when using BN.

Next, we evaluate our method with both types of profile mapping (*optimistic* and *pessimistic*) and compare it against *optimal k* (see Section 2), in terms of overall quality, effectiveness of the algorithm, number of gold questions used, i.e. incurred costs, fraud detection failure rates, and discrimination against reliable workers.

5.3.1 Method's Effectiveness and the Gold Par

We evaluate two trading-off parameters: 1) the method's effectiveness, that is, how effective the method is in including reliable workers, while simultaneously removing spammers. 2) The gold par percentage that is the overall percentage of gold questions posed in the crowdsourcing task. The more questions are posed, the more information the method gets, which directs it to the correct decision. Nevertheless, more gold questions, incur higher costs, e.g., BN-pessimistic method is overall the most effective (90.04%), yet comes at the highest cost of gold questions usage (19.31%).

In general, all methods seem to be more effective on datasets with higher spammers' percentage, with effectiveness values decreasing as more reliable workers are present (see Figure 11(a)), i.e., the discrimination rate against reliable workers is high, as can be seen in Figure 11(d). However, our CF-based and BN-based methods, whether utilizing the pessimistic or optimistic mapping, are more effective with datasets having higher spammers' percentage: S75 and S66, while optimal k seems more effective with datasets including more reliable workers: R66 and R75 (see Figure 11(a)). This implies that our method can reliably detect spammers, while optimal k is better at detecting reliable workers. Whereas the pessimistic methods have the highest percentage of gold par cost, and thus the highest cost. The CF-based optimistic method seems to score the trade-off balance by having the lowest percentage of gold par, and thus the lowest cost (see Figure 11 (a)), while still being as effective as Optimal K for S75, S66 and B datasets. For our test dataset, we have a total of 6631 tasks. If we, for in-stance, compute 5 cents per question, then on average for the CF-based optimistic method, 10.8% gold par costs around 36$, while optimal k costs 42.5$ at 12.8% gold questions. This means a cost reduction of about 18%. More gold questions should be asked in datasets having higher levels of spammers since more workers will be mapped to high-risk profiles, which consequently leads to a higher percentage of gold par usage. Surprisingly, we experienced a relatively similar percentage of gold par regardless of the composition of the dataset. Looking into the data, we attribute that to the inherent size of our tasks, i.e., most are relatively small ($\sim$ 20 questions per job).

5.3.2 Failure Rate

Next, we evaluated the failure rate of the methods in overseeing spammers and letting them through to work on the tasks (i.e., the False Positives). The lower the failure rate, the better. Figure 11(c), supports the previous results, where optimal k has the worst failure rate. Naturally, the failure rate is more pronounced in S75 and S66 and becomes less noticeable in R66 and R75 due to their inherent nature of having fewer spammers. On the other hand our methods have lower failure rates, which again adheres to their effectivity when handling datasets with high percentages of spammers, e.g., for dataset S66, the following failure rates were experienced: Optimal K had 24.69%, CF-based methods had on average 16.3%, and BN-based methods had on average 5% failure rate. Here the BN-based methods have lower failure rates than the CF-based methods, since they utilize more gold questions and are thus more informed.

5.3.3 Reliable Worker Pool and Discrimination Rate

After looking at failure rates, we test the second parameter impacting the methods' effectiveness: discrimination rates against reliable workers (i.e., the False Negatives). The lower the discrimination rate, the better. Optimal k shows the lowest discrimination rates. Naturally, this becomes more pronounced for the R66 and R75 datasets and is alike for all the other methods. The CF-based and BN-based optimistic methods are slightly more discriminating than their pessimistic counterparts, which would indicate, that perhaps the small number of gold questions that were given to the reliable workers might have been inherently difficult. This is also attributed to the inherent design of our methods, which enforces a higher quality threshold on workers with low-risk profiles. Looking at the overall actual number of reliable workers hired from the available pool (see Figure 11 (e)), backs up the discrimination rates in Figure 11(d).

5.3.4 Overall Resulting Quality

Regardless of the discrimination rates, or the effectiveness of the method, for a requestor or a data scientist, the ultimate quality measure for any safeguard is the resulting quality plus the costs it incurs. We can see that throughout our experiments that indeed all the methods achieved about the same quality level as illustrated in Figure 11(f); although it came as a pleasant surprise that despite needing less gold questions the CF-based optimistic method achieves even slightly better quality levels. For example for the dataset: B, the CF-based pessimistic method achieves 90%, the CF-based optimistic method achieved 89%, while optimal k achieves only 87% quality rate. Measuring the standard error for the resulting overall quality yielded a small rate of 0.0007. Posing crowdsourcing tasks of bigger size might reflect savings in gold questions even better, yet most of our crowdsourcing tasks had only 20 questions (thus 5% gold questions vs. 10% gold questions amounts to only one more gold question to be used).

5.4 Evaluating the Siamese Architecture

In this section, we provide details of the experimental setting we devised to evaluate the different implementations of the Siamese architecture. Firstly, we carry out a preprocessing step on each of the five datasets in order to attain the Siamese network's two input, which should be in a vector form of the attributes that we have previously discussed. Out of the four

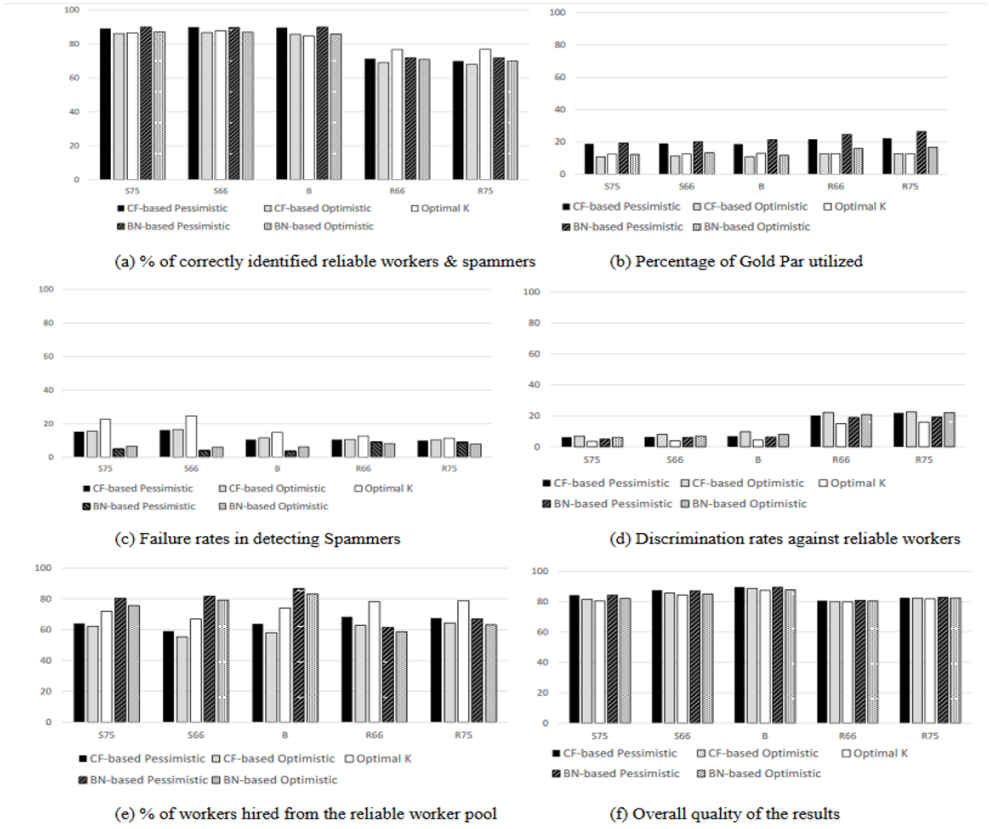

Figure 11: Evaluating Optimal k vs. CF-based/BN-based (Pessimistic and Optimistic) methods on S75, S66, B, R66, and R75

available attributes, three of them is categorical, and one is numerical. Accordingly, for each of the categorical variables that can take on K different values, we represent each data point with a K-dimensional vector x in which one element of x equals 1 and all the remaining elements equal zero. Secondly, in order to derive the hyperparameters, e.g., number of layers, number of neurons in each layer, regularization to avoid overfitting for the different models, we reserve part of the training data for validation purposes. For all the datasets, we reserved 20% of the original training data as validation data. The models evaluated here used hyperas[7] to search for the optimal hyperparameters mentioned above.

Next, with the remaining training data, we generated the pairs which are to be used as inputs for the Siamese networks that we test. Thus, we proceeded as follows: for each reliable worker, we generate positive pairs and negative pairs. A positive pair comprises of two reliable workers and, a negative pair comprises of a reliable worker and a non-reliable worker. Following one of the empirical findings, which states that having many examples, e.g., above 5000 (Goodfellow *et al.*, 2016) can lead to robust deep learning models, we generated all possible positive pairs and all possible negative pairs for each dataset. In Table 1, we can indeed see that the number of generated samples seems to be enough for our particular setting. Our later experiments, however, would prove otherwise (see Section 5.4.1).

[7] https://github.com/maxpumperla/hyperas

Table 1: Statistics of the Training Data

Dataset	Positive pairs	Negative pairs
Spammers75 (S75)	87,912	103,059
Spammers66 (S66)	78,680	102,003
Balanced (B)	75,350	101,475
Reliable66 (R66)	68,382	100,084
Reliable75 (R75)	56,882	96,795

5.4.1 Results of the Siamese Experiments

We evaluate three models of the Siamese architecture that use MLPs as sub-network. The main difference between the models is how, at the final layer, the model processes the internal representation of the inputs to assess the differences between reliable and fraudulent workers. Following what we found in the literature discussed in Section 4.3, we test the Manhattan distance, the Euclidean distance and a model that concatenates the internal representations before making the final prediction. We will refer to each model as Siamese Euclidean, Siamese Manhattan, and Siamese Concatenation respectively. For each model and each dataset we perform a grid search of the specific architecture (number of layers, neurons per layer, optimizer), and two regularization strategies dropout and early stopping (Hinton *et al.*, 2012). In Figure 12 we show the results of the performance of the models measured in the percentage of correctly identified reliable workers and spammers.

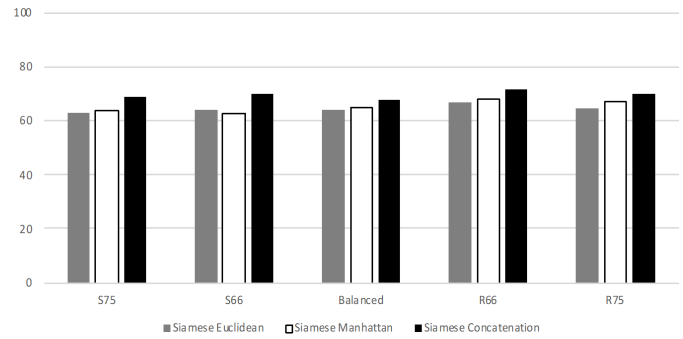

Figure 12: Percentage of correctly identified reliable workers and spammers

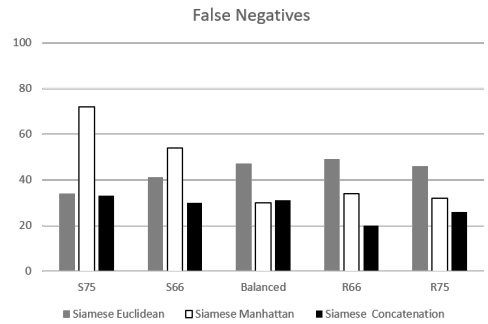

Figure 14: Discrimination rates against reliable workers.

In general, the three models' performance in each dataset is somewhat stable. In other words, the models generalize equally well due to a large number of available positive and negative pairs that they use to learn. To our surprise, we can observe that none of the Siamese architectures outperform our probabilistic-based approaches. We can observe that the Siamese Concatenation model outperforms in each data set all the other Siamese architectures by a small but significant margin. To better understand the differences between the models, we analyze the false positives and false negatives. In Figure 13 we can observe the failure rates against spammers (false positives). The Siamese Euclidean seems to be better to spot reliable workers. However, this difference vanishes with the datasets that have more spammers. We can also observe that the Siamese Manhattan seems to be better to detect spammers. We can see for instance that in the dataset with more spammers (S75) it achieves the lowest score (zero), e.g., it catches all the spammers. This comes with a high price: it has the highest discrimination ratio (false negatives see Figure 14).

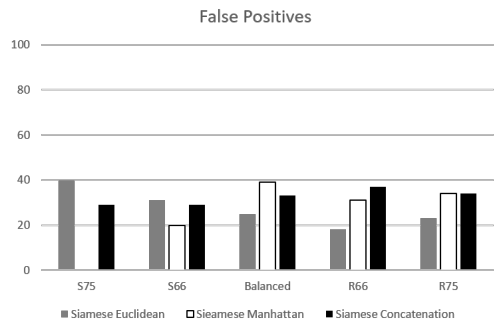

Figure 13: Failure rates in detecting spammers.

Moreover, the behavior of the Siamese Euclidean is the opposite when we look at Figure 14 concerning false negatives. We can see also that the model Siamese Concatenation behaves better than the other two models in all the datasets except the balanced dataset. In the latter, it gives comparable results concerning the Siamese Manhattan. These observations lead us to the conclusion that an ensemble of the models could lead to better performance. We show in Figure 15 the accuracy of these ensembles, where Siamese Concat+Euclidean refers to the model that uses the concatenation model and the Euclidean distance. Siamese Concat+Manh refers to the model that uses the concatenation model and the Manhattan distance, whereas Siamese Concat+Both refers to the model that uses the Concatenation with Euclidean and the Manhattan distance. The gains that we can observe confirms our general finding after experimenting with these models: there is no latent representation that any of the models can build using the four attributes available. In summary, these models show some potential and an intuitive interpretation, but without more attributes, they cannot improve on the results obtained through the Certainty factor model or Belief networks.

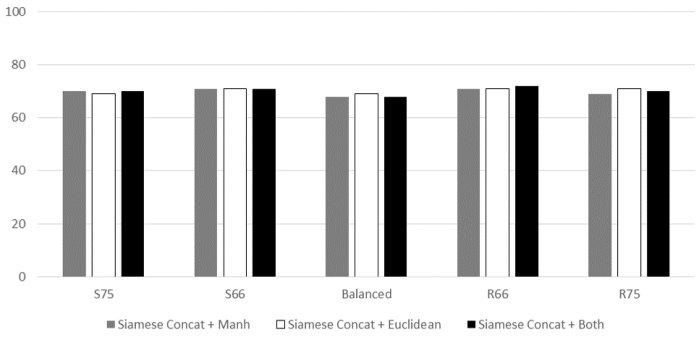

Figure 15: Percentage of correctly identified reliable workers and spammers

We also measure the impact of our best model regarding the training data available. To do so, we started with 5000 examples given the empirical observations of Goodfellow *et al.*, 2016. If with 5000 our model was not able to achieve the results obtained with all the data, then we started to sample 10%, 50% and 75% of the rest of the available training data. Results vary depending on the dataset. For instance, the model trained on the S75 dataset achieved the best results using only 5000 samples. For the S66 dataset, we needed 10% of the data. For the balanced dataset, we needed 50% of the data. For the last two datasets when we have more reliable workers than spammers we needed up 75% of the data. In summary, the fewer spammers the models are aware of, the less effective they are and thus need more training data to reach the best results.

6 Conclusion

We designed a method for using individualized numbers of gold questions that leverages the limited amount of information known about short-term hired anonymous workers. Surprisingly, with as little as four publicly available attributes, we were already able to create meaningfully enhanced worker profiles. The generated enhanced profiles encode the risk associated with each worker regarding a particular task by exposing relative a-priori belief /disbelief measures or probabilities. In a nutshell, the higher the belief value or probability of a worker's reliability, the lower the risk, and the lesser the number of gold questions to be used. Moreover, the higher the disbelief value or, the lower the probability of a worker's reliability, the higher the risk, and the more gold questions need to be used. For generating this profile database, we experimented with certainty factor models and Bayesian belief networks. Interestingly, the Bayesian belief network did not outperform simple certainty factors. This might be caused by the inherently small amount of training data, in addition to biases introduced while estimating priors. We also designed a Siamese based neural network architecture in an attempt to find a latent representation to differentiate between reliable and fraudulent workers. To our surprise, none of the models that we implemented could outperform our previously proposed probabilistic-based approaches. Our results indicate that neural network models are incapable of learning a latent representation likely due to the limited number of attributes available. In short, our findings indicate Certainty factors fit better the problem addressed in this paper. We illustrated the applicability of our method on practical crowdsourcing tasks and demonstrated its potential in saving money while maintaining high-quality results. We tested our method against five different datasets with different compositions of spammers and reliable workers. Our method works best when there are more spammers, and always achieved at least comparable quality results to the 'optimal k' baseline. Moreover, our CF-based optimistic method achieved higher quality rates at a lower number of gold questions: only 12.8% gold questions were used in contrast to 14.8% with *optimal k*.

References

Aberer, K. and Z. Despotovic. "Managing Trust in a Peer-2-peer Information System". In: *Proceedings of the Tenth International Conference on Information and Knowledge Management. CIKM '01*. Atlanta, Georgia, USA: ACM. 310–317. ISBN: 1-58113-436-3. DOI: 10.1145/502585.502638. URL: http://doi.acm.org/10.1145/502585.502638.

Bromley, J., I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. "Signature Verification Using a "Siamese" Time Delay Neural Network". In: *Proceedings of the 6th International Conference on Neural Information Processing Systems. NIPS'93*. Denver, Colorado: Morgan Kaufmann Publishers Inc. 737–744. URL: http://dl.acm.org/citation.cfm?id=2987189.2987282.

Checco, A., J. Bates, and G. Demartini. "All That Glitters Is Gold - An Attack Scheme on Gold Questions in Crowdsourcing". In: *Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2018, Zürich, Switzerland, July 5-8, 2018*. 2–11. URL: https://aaai.org/ocs/index.php/HCOMP/HCOMP18/paper/view/17925.

Daltayanni, M., L. de Alfaro, and P. Papadimitriou. "WorkerRank: Using Employer Implicit Judgements to Infer Worker Reputation". In: *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. WSDM '15*. Shanghai, China: ACM. 263–272. ISBN: 978-1-4503-3317-7. DOI: 10.1145/2684822.2685286. URL: http://doi.acm.org/10.1145/2684822.2685286.

Das, A., H. Yenala, M. Chinnakotla, and M. Shrivastava. "Together we stand: Siamese Networks for Similar Question Retrieval". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics. 378–387. DOI: 10.18653/v1/P16-1036. URL: https://www.aclweb.org/anthology/P16-1036.

Davidson, S. B., S. Khanna, T. Milo, and S. Roy. "Using the Crowd for Top-k and Group-by Queries". In: *Proceedings of the 16th International Conference on Database Theory. ICDT '13*. Genoa, Italy: ACM. 225–236. ISBN: 978-1-4503-1598-2. DOI: 10.1145/2448496.2448524. URL: http://doi.acm.org/10.1145/2448496.2448524.

Dawid, A. P. and A. M. Skene. "Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm". *Journal of the Royal Statistical Society. Series C (Applied Statistics)*. 28(1): 20–28. ISSN: 00359254, 14679876. URL: http://www.jstor.org/stable/2346806.

Franklin, M. J., D. Kossmann, T. Kraska, S. Ramesh, and R. Xin. "CrowdDB: Answering Queries with Crowdsourcing". In: *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data. SIGMOD '11*. Athens, Greece: ACM. 61–72. ISBN: 978-1-4503-0661-4. DOI: 10.1145/1989323.1989331. URL: http://doi.acm.org/10.1145/1989323.1989331.

Goodfellow, I., Y. Bengio, and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press.

Hadsell, R., S. Chopra, and Y. LeCun. "Dimensionality Reduction by Learning an Invariant Mapping". *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. 2: 1735–1742.

Hinton, G. E., N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. "Improving neural networks by preventing co-adaptation of feature detectors". *ArXiv*. abs/1207.0580.

Hossain, M. "Users' motivation to participate in online crowdsourcing platforms". *ICIMTR 2012 - 2012 International Conference on Innovation, Management and Technology Research*. May. DOI: 10.1109/ICIMTR.2012.6236409.

Ignjatovic, A., N. Foo, and C. T. Lee. "An Analytic Approach to Reputation Ranking of Participants in Online Transactions". In: *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*. IEEE Computer Society. 587–590.

Kamvar, S. D., M. T. Schlosser, and H. Garcia-Molina. "The Eigentrust Algorithm for Reputation Management in P2P Networks". In: *Proceedings of the 12th International Conference on World Wide Web. WWW '03*. Budapest, Hungary: ACM. 640–651. ISBN: 1-58113-680-3. DOI: 10.1145/775152.775242. URL: http://doi.acm.org/10.1145/775152.775242.

Kazai, G. "In Search of Quality in Crowdsourcing for Search Engine Evaluation". In: *Advances in Information Retrieval*. Ed. by P. Clough, C. Foley, C. Gurrin, G. J. F. Jones, W. Kraaij, H. Lee, and V. Mudoch. Berlin, Heidelberg: Springer Berlin Heidelberg. 165–176. ISBN: 978-3-642-20161-5.

Koch, G., R. Zemel, and R. Salakhutdinov. "Siamese neural networks for one-shot image recognition". In: *International Conference on Machine Learning (ICML)*. Vol. 2.

Krieg, M. L. "A tutorial on Bayesian belief networks". *Tech. rep.* Defence Science and Technology Organisation Salisbury. Australia.

Liu, Q., M. Steyvers, and A. Ihler. "Scoring Workers in Crowdsourcing: How Many Control Questions Are Enough?" In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. NIPS'13*. Lake Tahoe, Nevada: Curran Associates Inc. 1914–1922. URL: http://dl.acm.org/citation.cfm?id=2999792.2999826.

Lofi, C., K. El Maarry, and W.-T. Balke. "Skyline Queries in Crowd-enabled Databases". In: *Proceedings of the 16th International Conference on Extending Database Technology. EDBT '13*. Genoa, Italy: ACM. 465–476. ISBN: 978-1-4503-1597-5. DOI: 10.1145/2452376.2452431. URL: http://doi.acm.org/10.1145/2452376.2452431.

Maarry, K. E. and W.-T. Balke. "Retaining Rough Diamonds: Towards a Fairer Elimination of Low-Skilled Workers". In: *Database Systems for Advanced Applications*. Ed. by M. Renz, C. Shahabi, X. Zhou, and M. A. Cheema. Cham: Springer International Publishing. 169–185.

Maarry, K. E. and W.-T. Balke. "Quest for the Gold Par: Minimizing the Number of Gold Questions to distinguish between the Good and the Bad". In: *10th ACM Conference on Web Science*. ACM. Amsterdam, The Netherlands: ACM.

Maarry, K. E., U. Güntzer, and W.-T. Balke. "Realizing Impact Sourcing by Adaptive Gold Questions: A Socially Responsible Measure for Workers' Trustworthiness". In: *Web-Age Information Management*. Ed. by X. L. Dong, X. Yu, J. Li, and Y. Sun. Cham: Springer International Publishing. 17–29. ISBN: 978-3-319-21042-1.

Mellouli, T. "Complex certainty factors for rule based systems - Detecting inconsistent argumentations". *CEUR Workshop Proceedings*. 1335(Jan.): 81–102.

Mueller, J. and A. Thyagarajan. "Siamese Recurrent Architectures for Learning Sentence Similarity". In: *Proceedings of the 30th Conference on Artificial Intelligence (AAAI 2016)*.

Nieke, C., U. Güntzer, and W.-T. Balke. "TopCrowd Efficient Crowd-enabled Top-k Retrieval on Incomplete Data". In: *Conceptual Modeling*. Ed. by E. Yu, G. Dobbie, M. Jarke, and S. Purao. Cham: Springer International Publishing. 122–135. ISBN: 978-3-319-12206-9.

Noorian, Z. and M. Ulieru. "The State of the Art in Trust and Reputation Systems: A Framework for Comparison". *J. Theor. Appl. Electron. Commer. Res.* 5(2): 97–117. ISSN: 0718-1876. DOI: 10.4067/S0718-18762010000200007. URL: http://dx.doi.org/10.4067/S0718-18762010000200007.

Pearl, J. "Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning". In: *Proc. of Cognitive Science Society (CSS-7)*.

Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN: 0-934613-73-7.

Ross, J., L. Irani, M. S. Silberman, A. Zaldivar, and B. Tomlinson. "Who Are the Crowdworkers?: Shifting Demographics in Mechanical Turk". In: *CHI '10 Extended Abstracts on Human Factors in Computing Systems. CHI EA '10*. Atlanta, Georgia, USA: ACM. 2863–2872. ISBN: 978-1-60558-930-5. DOI: 10.1145/1753846.1753873. URL: http://doi.acm.org/10.1145/1753846.1753873.

Shen, C., Z. Jin, Y. Zhao, Z. Fu, R. Jiang, Y. Chen, and X.-S. Hua. "Deep Siamese Network with Multi-level Similarity Perception for Person Re-identification". In: *Proceedings of the 25th ACM International Conference on Multimedia. MM '17*. Mountain View, California, USA: ACM. 1942–1950. ISBN: 978-1-4503-4906-2. DOI: 10.1145/3123266.3123452. URL: http://doi.acm.org/10.1145/3123266.3123452.

Shortliffe, E. and B. Buchanan. "A Model of Inexact Reasoning in Medicine". *Mathematical Biosciences*. 23(Apr.): 351–379. DOI: 10.1016/0025-5564(75)90047-4.

Stadler, F. *Induction and Deduction in the Sciences*. Springer.

Tran, N. K. and C. Niederée. "A Neural Network-based Framework for Non-factoid Question Answering". In: *Companion Proceedings of the The Web Conference 2018. WWW '18*. Lyon, France: International World Wide Web Conferences Steering Committee. 1979–1983. ISBN: 978-1-4503-5640-4. DOI: 10.1145/3184558.3191830. URL: https://doi.org/10.1145/3184558.3191830.

Whitehill, J., T.-f. Wu, J. Bergsma, J. R. Movellan, and P. L. Ruvolo. "Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise". In: *Advances in Neural Information Processing Systems 22*. Ed. by Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta. Curran Associates, Inc. 2035–2043. URL: http://papers.nips.cc/paper/3644-whose-vote-should-count-more-optimal-integration-of-labels-from-labelers-of-unknown-expertise.pdf.

Yu, B. and M. P. Singh. "Detecting Deception in Reputation Management". In: *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems. AAMAS '03*. Melbourne, Australia: ACM. 73–80. ISBN: 1-58113-683-8. DOI: 10.1145/860575.860588. URL: http://doi.acm.org/10.1145/860575.860588.

Zheng, Y., J. Wang, G. Li, R. Cheng, and J. Feng. "QASCA: A Quality-Aware Task Assignment System for Crowdsourcing Applications". In: *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. SIGMOD '15*. Melbourne, Victoria, Australia: ACM. 1031–1046. ISBN: 978-1-4503-2758-9. DOI: 10.1145/2723372.2749430. URL: http://doi.acm.org/10.1145/2723372.2749430.