Dynamic, Multi-dimensional, and Skillset-specific Reputation Systems for Online Work

Marios Kokkodis
Carroll School of Management, Boston College, Chestnut Hill, MA 02467,
kokkodis@bc.edu

Reputation systems in digital workplaces increase transaction efficiency by building trust and reducing information asymmetry. These systems, however, do not yet capture the dynamic multidimensional nature of online work. By uniformly aggregating reputation scores across worker skills, they ignore skillset-specific heterogeneity (reputation attribution), and they implicitly assume that a worker's quality does not change over time (reputation staticity). Even further, reputation scores tend to be overly positive (reputation inflation), and as a result, they often fail to differentiate workers efficiently.

This work presents a new augmented intelligence reputation framework that combines human input with machine learning to provide dynamic, multi-dimensional, and skillset-specific worker reputation. The framework includes three components: The first component maps skillsets into a latent space of finite competency dimensions (word embedding), and as a result, it directly addresses reputation attribution. The second builds dynamic competency-specific quality assessment models (hidden Markov models) that solve reputation staticity. The final component aggregates these competency-specific assessments to generate skillset-specific reputation scores. Application of this framework on a dataset of 58,459 completed tasks from a major online labor market shows that, compared with alternative reputation systems, the proposed approach (1) yields more appropriate rankings of workers that form a closer-to-normal reputation distribution, (2) better identifies "non-perfect" workers who are more likely to underperform and are harder to predict, and (3) improves the ranking of within-opening choices and yields significantly better outcomes. Additional analysis of 77,044 restaurant reviews shows that the proposed framework successfully generalizes to alternative contexts, where assigned feedback scores are overly positive and service quality is multidimensional and dynamic.

Key words: Reputation frameworks, Reputation inflation, Reputation attribution, Reputation staticity, Online labor markets, Hidden Markov models, Word embedding

Acknowledgments

I thank Sam Ransbotham, Panagiotis Adamopoulos, and Vilma Todri for their guidance and feedback. I also thank Robert Fichman, Gerald Kane, Zhuoxin Li, Xuan Ye, and Mike Teodorescu for their comments and suggestions on improving the paper.

1. Introduction

Online labor markets (Peopleperhour, Freelancer) facilitate global short-term contracts or freelance work (Graham et al. 2017). Buyers purchase services from an abundance of capable online workers that complete diverse tasks, including web development, graphic design, accounting, sales, marketing, and data science. On par with other online platforms, online labor marketplaces grew exponentially during the past decade (Freelancers-union and Upwork 2017, Upwork 2014). This growth will likely continue (if not accelerate) in the future, as automation and the sharing economy structure the future of work (Sundararajan 2016, Institute of Business Value 2019).

One determinant of success for online labor markets is the intermediary trust that they instill between employers and workers (Ba and Pavlou 2002, Pavlou and Gefen 2004, Nica et al. 2017). Reputation systems are a standard mechanism that online labor markets use to increase trust and reduce information asymmetry (Akerlof 1978, Kokkodis and Ipeirotis 2016, Yoganarasimhan 2013, Nica et al. 2017). These systems rely on human input: employers rate workers for the tasks they complete, and these ratings become part of the workers' online resumes. Such reputation mechanisms measure expected service quality (Rahman 2018b, Filippas et al. 2018, Kokkodis and Ipeirotis 2016), and as a result, they increase employers' trust in workers' abilities and facilitate market transactions (Yoganarasimhan 2013, Moreno and Terwiesch 2014, Lin et al. 2016). Besides, workers realize how reputation instills trust and affects employer choices, and tend to readjust their premiums according to their reputation scores (Banker and Hwang 2008, Gandini et al. 2016).

Despite these benefits, the design of current reputation systems does not capture the dynamic and multidimensional nature of online work. In particular, reputation systems in online labor markets implicitly assume (by uniformly averaging all prior feedback scores) that worker quality and

expertise are not evolving (Hendrikx et al. 2015). However, current trends show that new skills are born and old skills die faster than ever (Autor et al. 1998, Autor 2001, Oliver 2015, Kokkodis and Ipeirotis 2016). As a result, to remain marketable, online workers must be diligently and continuously re-educating and reskilling themselves (Kuhn and Skuterud 2004, Stevenson 2009, Oliver 2015). Furthermore, online worker reputation scores are unidimensional and skillset-independent. Digital workplaces, however, are highly heterogeneous in terms of qualifications (Kokkodis and Ipeirotis 2014), while online workers tend to complete tasks that require diverse skillsets (Kokkodis and Ipeirotis 2016). As a result, these unidimensional reputation scores cannot provide accurate estimates of skillset-specific expertise. Finally, on par with other online platforms (Hu et al. 2009, Hu et al. 2017, Zervas et al. 2015), online worker reputation scores tend to be overly positive (Filippas et al. 2018, Abhinav et al. 2017). Such inflated scores do not sufficiently differentiate workers, as most of them are rated as "better than average" (Filippas et al. 2018).

Given these shortcomings of current reputation systems in online labor markets, *how can we design dynamic, multidimensional, skillset-specific reputation frameworks*? To address this question, I propose an intelligence augmentation (IA) system that relies on three design principles: (1) Mapping of any combination of arbitrary skills into a latent space of finite competency dimensions (Word Embedding), (2) dynamic competency-specific quality assessment (hidden Markov models), and (3) aggregation of these competency-specific assessments. Decomposition of skills to competencies facilitates skillset-specific reputation. Dynamic quality assessment explains the evolution of workers as they learn new skills and gain expertise. Aggregation of the competency-specific quality assessments for any given combination of skills results in representative (normal-like) skillset-specific reputation distributions (Schmidt and Hunter 1983) that catalyze worker differentiation.

Analysis of 58,459 completed tasks from a major online labor market shows that the proposed approach significantly outperforms ten alternative advanced reputation systems (including the market's current reputation system, systems that rely on link analysis, gradient boosting, neural networks, and adaptations of recommender systems). In particular, compared with these systems,

the proposed approach (1) yields more appropriate rankings of workers that form a closer-to-normal reputation distribution, (2) better identifies "non-perfect" workers who are more likely to underperform and are harder to predict, and (3) improves the ranking of within-opening choices and yields significantly better outcomes. Additional analysis of 77,044 restaurant reviews shows that the proposed framework successfully generalizes to alternative contexts, where assigned feedback scores are overly positive and service quality is multidimensional and dynamic.

This work is the first to identify shortcomings of current reputation systems of online labor markets and to present design principles that future reputation systems should have in order to estimate a worker's dynamic and multidimensional reputation. By solidifying these principles into different components, the proposed IA framework combines human input with machine intelligence to result in accurate, skillset-specific reputation scores. Such accurate scores (1) help workers to differentiate, (2) guide employers to make informed and fast (reduced search cost; see Bakos 1997) decisions, and (3) enable the market to improve its recommendation algorithms but also to understand the supply distributions across latent competencies. By predicting underperforming workers, the framework preemptively informs employers, an intervention that could reduce the number of adverse outcomes. Positive outcomes increase participation in the marketplace, thereby generating a continuous stream of revenue for the platform (Tripp and Grégoire 2011).

This IA framework also highlights how combining human input with advanced machine learning techniques can augment intelligence by creating the necessary conditions for humans to make informed decisions. Such systems have the potential to increase efficiency and outcome quality precisely because they intelligently differentiate workers (i.e., identify each individual's latent qualities). Efficient differentiation can further guide labor supply redistribution (e.g., by motivating workers to re-educate) and inform career path advisers (Kokkodis and Ipeirotis 2020). As a result, the deployment of the proposed IA framework in different types of online platforms could have implications for workers, employers, businesses, and the future of work.

2. Research context

Digital markets increase trust and signal the quality of their services and products through online reputation systems (Ba and Pavlou 2002, Pavlou and Gefen 2004, Tadelis 2016, Dellarocas 2003, Zervas et al. 2015, Dellarocas 2006). These systems facilitate product selection across a wide range of domains, including movies (Duan et al. 2008), books (Chevalier and Mayzlin 2006), music (Kokkodis and Ransbotham 2020), electronics (Cui et al. 2012, Ghose and Ipeirotis 2011), hotels (Ye et al. 2009), local businesses (Luca 2016, Lu et al. 2013), and mobile apps (Lee and Raghu T. 2014).

2.1. Overview of current reputation systems designs

Given this established impact of online reputation systems, prior research has focused on improving their design and increasing their performance. Researchers have proposed reputation systems that have context-specific objectives and serve alternative domains such as e-commerce platforms, online communities, crowdsourcing platforms, and peer-to-peer networks. Based on their architecture, I cluster existing reputation systems into *human-based*, *machine-based*, and *hybrid* (human and machine).

2.1.1. Human-based reputation systems: Most commercial reputation systems rely solely on human ratings (e.g., in online marketplaces; see Tadelis 2016, Luca 2017, Einav et al. 2016). For instance, Amazon users post reviews, rate products, and rate other reviews (Amazon 2018). Similarly, eBay sellers and buyers leave feedback for each other (eBay 2018). This feedback reflects both an overall rating (good, neutral, and negative), but also numerical ratings for accuracy, communication, shipping time, and shipping charges. A similar reputation system appears in many online question-and-answer communities (e.g., Stackoverflow and discourse communities), where users up-vote or down-vote responses (Stackoverflow 2018, Kokkodis et al. 2020b). Third-party reputation platforms that allow users to review and rate are also available for multiple products and services, such as restaurants and hotels (e.g., TripAdvisor, Yelp; see Kokkodis and Lappas 2020) and Amazon Mechanical Turk (AMT) users (Turkopticon 2018).

2.1.2. Machine-based reputation systems: Machine-based approaches do not require human raters. Instead, they often rely on network analysis to identify user quality. In online and question-answering communities, such methods focus on identifying expert (or helpful) users (Jurczyk and Agichtein 2007, Zhang et al. 2007, Bouguessa et al. 2008). In large organizations, proposed approaches combine information retrieval and graph-based techniques to analyze user social (and other) profiles and identify areas of expertise (Balog and De Rijke 2007). Similarly, peer-to-peer network reputation systems use network analysis to estimate the sharing quality of each participating node (Kamvar et al. 2003).

2.1.3. Hybrid reputation systems: Many reputation systems combine information from human raters with machine learning and network analysis. Such hybrid approaches are examples of IA systems (Jain et al. 2018), as they enhance human judgment by combining artificial and human intelligence. The most basic ones combine ratings with information from social and other online sources (Sabater and Sierra 2001a,b, Hendrikx and Bubendorfer 2013). These systems are tailored for e-commerce platforms (Hendrikx and Bubendorfer 2013). For online communities, hybrid approaches use human cognitive traits along with subjective logic to identify experts (Pelechrinis et al. 2015). At the same time, peer-to-peer networks require different types of hybrid reputation systems that combine network analysis along with ratings and the personal histories of each node to estimate node trustworthiness (Damiani et al. 2002, Curtis et al. 2004, Xiong and Liu 2004, Tian and Yang 2011).

Hybrid reputation systems also appear in crowdsourcing settings (e.g., AMT; see Allahbakhsh et al. 2012, Jagabathula et al. 2014). For instance, some reputation management models include a rater's credibility along with information from each worker's set of completed tasks to estimate worker quality (Allahbakhsh et al. 2012). Similarly, and in order to filter out adversarial workers (Jagabathula et al. 2014), proposed reputation systems in crowdsourcing settings penalize workers with a poor reputation (Xie et al. 2015).

2.2. Reputation systems in online labor markets

Similar to most e-commerce platforms, online labor markets offer reputation systems that allow workers to receive feedback for the tasks they complete (Filippas et al. 2018, Wood-Doughty 2018). Over a series of completed tasks, these feedback scores accumulate to generate a worker's reputation on the platform (Rahman 2018a). Worker reputation "institutes trust among quasi-strangers" (Nica et al. 2017), and as a result, increases marketplace efficiency by reducing information asymmetry (Kokkodis et al. 2015). In particular, worker reputation is a major driving force in hiring choices (Yoganarasimhan 2013), and it correlates positively with worker earnings (Banker and Hwang 2008, Gandini et al. 2016, Moreno and Terwiesch 2014). Even having a reputation (compared with being new in the market) improves a worker's current (Lin et al. 2016) and subsequent hiring chances (Pallais 2014). These reputation effects are not uniform, as positive verified information appears to disproportionately benefit workers from less developed countries (Agrawal et al. 2013, Kanat et al. 2018). Finally, reputation is not necessarily category-specific; it transfers across multiple task categories that require diverse skillsets (Kokkodis and Ipeirotis 2016).

2.2.1. Shortcomings of reputation systems in online labor markets: Despite these multidimensional effects, reputation systems in online labor markets are not perfect (Filippas et al. 2018), as they experience: (1) reputation inflation, (2) reputation attribution, and (3) reputation staticity.

Reputation inflation: Similar to other online marketplaces (Zervas et al. 2015, Hu et al. 2017), reputation scores in online labor markets are highly inflated (Filippas et al. 2018, Abhinav et al. 2017). This inflation happens mainly for two reasons. First, users who receive low feedback scores cannot get hired, and as a result, they abandon the marketplace (Jerath et al. 2011, Jøsang and Golbeck 2009, Jøsang et al. 2007). Second, employers feel peer pressure to assign positive ratings (Filippas et al. 2018). The combination of the two yields reputation distributions that are positively skewed, where every worker is assumed to be "better than average." Such inflated reputation scores do not sufficiently differentiate workers, as they form noisy estimates of service quality (Hendrikx et al. 2015).

Reputation attribution: At the same time, current reputation systems in online labor markets provide unidimensional reputation scores that describe overall service quality. However, these digital workplaces are highly heterogeneous in terms of qualifications (Kokkodis and Ipeirotis 2014), as they offer tasks that require a diverse range of skills (e.g., logo design, software development, data analytics, marketing skills). Besides, workers often do not focus on specific types of tasks, but instead, they complete tasks that require diverse skillsets (Kokkodis and Ipeirotis 2016). The existence of this highly heterogeneous environment in terms of skills suggests that unidimensional reputation scores cannot capture skill-specific qualities. For instance, consider a worker who provides an IT service and completes a task that requires `networking`, `C`, and `Python`. When the task is over, this worker receives a feedback score of 0.9. Does this score capture the worker's service quality on `networking`, on `C`, on `Python`, or on any combination of these skills?

Reputation staticity: Finally, the rapid evolution of skills and worker expertise in online labor markets further limits current reputation systems. Because new skills are born and old skills die faster than ever before (Autor et al. 1998, Autor 2001, Kokkodis and Ipeirotis 2016, Oliver 2015), workers need to continuously keep re-educating themselves (Kuhn and Skuterud 2004, Stevenson 2009, Oliver 2015, Kokkodis 2020). Workers are therefore dynamic entities that evolve by either gaining expertise on skills they know, or by investing in learning new skills (Kokkodis and Ipeirotis 2020). Current reputation systems assume that the quality of a service does not change over time (Jøsang et al. 2007), as they uniformly average received ratings to provide an aggregate quality score (Hendrikx et al. 2015). This assumption is valid for a product in an e-commerce platform (e.g., a book or a camera), but it is misleading in representing the quality of a worker who gains expertise and acquires new skills over time.

These shortcomings of reputation systems in online labor markets result in service quality estimates that are often not predictive of future worker performance (Filippas et al. 2018). Consequently, decisions based on such reputation scores could yield unsuccessful collaborations that hurt the marketplace (Tripp and Grégoire 2011). As a result, there is a need for exploring alternative reputation systems that could potentially address these shortcomings and provide more representative reputation scores.

2.2.2. Do existing reputation systems address these shortcomings? Section 2.1 classifies current reputation systems into human-based, machine-based, and hybrid. The presented commercial applications of human-based reputation systems experience reputation inflation, and to a certain degree, reputation attribution and reputation staticity. In particular, e-commerce reputation scores are inflated (Chevalier and Mayzlin 2006, Hu et al. 2009, Hu et al. 2017, Zervas et al. 2015) due to response bias—i.e., who chooses to rate a service (Moe and Schweidel 2012)—and due to acquisition bias—i.e., buyers typically choose services that they expect to like (Hu et al. 2017). Reputation attribution appears when products receive unidimensional ratings describing multiple dimensions (e.g., value for money, appearance, durability). TripAdvisor acknowledges the issue of reputation attribution and offers reputation scores in four secondary dimensions (i.e., location, cleanliness, service, value). Even though such multidimensional systems better describe service quality, they do not generalize to an arbitrary set of dimensions, and they usually rely on a few dimensions that humans can efficiently rate. Finally, because some of the rated products or services are dynamic (e.g., venues evolve on TripAdvisor), their respective reputation systems likely experience reputation staticity.

Machine-based link analysis approaches require a network of interactions related to service quality in order to work (e.g., Jurczyk and Agichtein 2007). At the same time, they do not rely on any evaluation measure (either objective through testing or subjective through human raters). Hence, their applicability to contexts that require human-perceived service quality, such as online work, is limited. Applicability is also an issue for hybrid-based reputation systems (Section 2.1.3) that either focus on different objectives (e.g., node quality in peer-to-peer networks; see Damiani et al. 2002) or require information that is not freely available (e.g., social behavior; see Sabater and Sierra 2001b) in online labor markets. Overall, these machine-based and hybrid approaches do not focus on addressing reputation inflation, reputation attribution, and reputation staticity, as these shortcomings do not pose significant limitations in their respective contexts (i.e., peer-to-peer networks and question-answering communities).

One could argue that given the contextual similarities between crowdsourcing settings and online labor markets, their reputation systems (Section 2.1.3) could be applicable in both contexts. Digital workplaces, however, differ from many crowdsourcing settings in that workers are highly paid and highly skilled (Paolacci et al. 2010, Kokkodis and Ipeirotis 2014). As a result, the objective of crowdsourcing systems to filter out adversarial workers does not apply to the focal context, as none of the high-skilled workers in online labor markets will purposely perform subpar work (e.g., by mindlessly labeling images, which is a typical task on AMT). Furthermore, low-skilled AMT workers are not susceptible to supply trends that often require reskilling; hence, skillset diversity and heterogeneity are not evident in crowdsourcing settings. As a result, crowdsourcing approaches do not focus on and do not solve reputation staticity,[1] reputation attribution, and reputation inflation.

Prior research has also proposed machine-based and hybrid reputation systems specifically for online labor markets (Christoforaki and Ipeirotis 2015, Daltayanni et al. 2015). Machine-based approaches rely on item response theory (Hambleton et al. 1991) to continuously generate test questions and evaluate the expertise of a user on a given skill (Christoforaki and Ipeirotis 2015). Many online labor markets already use such tests to certify workers on specific skills (Upwork 2018). In theory, these approaches could address reputation attribution, as they can evaluate the expertise of each worker on a given skill. In practice, however, they are costly (in terms of money and, most importantly, time), while they do not scale to multiple skills since they test one skill at a time (Upwork 2018). Even further, workers who take these tests have the option to either reveal or hide their scores, which results in the disclosure of positive-only certifications (Ipeirotis 2013). Finally, from a platform's perspective, creating and maintaining tests for hundreds of skills could incur additional costs.

A hybrid approach that is relevant to this work uses link-analysis and implicit reputation signals (e.g., "shortlisted," "hired," "ignored") to estimate a reputation score for each worker (WorkerRank;

[1] Allahbakhsh et al. (2012) weights the timing of each rating so in theory it has the potential to address reputation staticity. Table 1 clarifies this point.

Table 1 Comparison of relevant literature on reputation systems

Paper or commercial implementation	Context	Applicable to OLMs	Reputation inflation	Reputation attribution	Reputation staticity	Requires testing	Methodology
Commercial reputation systems	OLMs	✓	✗	✗	✗	✗	Human-based, rating assignment
Multidimensional reputation systems	TripAdvisor	✓	✗	✤	✗	✗	Human-based, rating assignment
Christoforaki and Ipeirotis (2015)	OLMs	✓	✠	✠	✠	✓	Machine-based, skill-specific testing (IRT)
Jurczyk and Agichtein (2007)	Q&A forums	✗	✗	✗	✗	✗	Machine-based, network analysis
Zhang et al. (2007)	Q&A forums	✗	✗	✗	✗	✗	Machine-based, network analysis
Bouguessa et al. (2008)	Q&A forums	✗	✗	✗	✗	✗	Machine-based, network analysis
Kamvar et al. (2003)	Peer-to-peer sharing networks	✗	✗	✗	✗	✗	Machine-based, network analysis
Sabater and Sierra (2001b)	E-commerce platforms, social settings	✗	✗	✗	✗	✗	Hybrid, combines social, individual and ontological dimensions
Allahbakhsh et al. (2012)	Croudsourcing (AMT)	✗	✗	✗	✢	✗	Hybrid, graph-based analysis
Jagabathula et al. (2014)	Croudsourcing (AMT)	✗	✗	✗	✗	✗	Hybrid, graph-based analysis
Xie et al. (2015)	Croudsourcing (AMT)	✗	✗	✗	✗	✗	Hybrid, Bayesian game framework
Daltayanni et al. (2015)	OLMs	✓	?	✗	✗	✗	Link analysis, implicit feedback
This research	OLMs	✓	✓	✓	✓	✗	Deep learning, Hidden Markov Models

OLMs: Online labor markets. IRT: Item response theory. Q&A: Question-answering communities. AMT: Amazon Mechanical Turk
The column "Applicable to OLMs" identifies whether the approach could be deployed in an online labor market. Columns "Reputation inflation," "Reputation attribution," and "Reputation staticity" identify whether the research addresses these shortcomings of current reputation systems in online labor markets. Column "Requires testing" identifies whether the approach requires workers (users) to take tests.
✠: Testing has the potential to address reputation inflation, reputation attribution, and reputation staticity, but it requires investment in time (and money) and it has scalability and cost constraints (Section 2.2.2). ✢: Approach allows reputation in four dimensions; hence, to a certain degree, it addresses reputation attribution. Appendix E shows that the proposed approach outperforms such multidimensional reputation systems. ✢: Approach incorporates the timing of each feedback in estimating reputation, so, in theory it addresses reputation staticity. ?: Approach could potentially address reputation inflation.

see Daltayanni et al. 2015). Specifically, by creating a historical graph between jobs and workers, the WorkerRank algorithm compares how different employers rank workers through their hiring processes. By construction, this approach does not focus and does not solve reputation attribution and reputation staticity; however, because it has the potential to provide more representative reputation scores, it implicitly addresses reputation inflation. (Section 5 empirically shows the superiority of the proposed approach over the WorkerRank algorithm across multiple dimensions.)

2.2.3. Designing dynamic reputation systems: Table 1 compares these relevant reputation systems found in academia and industry and identifies that they do not explicitly address reputation inflation, reputation attribution, and reputation staticity. This paper fills this gap by presenting

the design principles that a reputation system needs in order to successfully attack reputation attribution, reputation staticity, and reputation inflation. Specifically, these principles are (1) decomposition of any arbitrary combination of skills into a set of finite competency dimensions, (2) dynamic estimation of competency-specific reputation, and (3) on-demand aggregation of these competency-specific reputations to estimate skillset-specific reputation scores. The first principle solves reputation attribution, as it allows for the efficient decomposition of reputation to a finite set of competency dimensions. The second principle allows for competency-specific dynamic estimation of quality and directly addresses reputation staticity. The third principle creates skillset-specific estimates of quality. Since human abilities follow normal-like distributions (Schmidt and Hunter 1983), accurate skillset-specific reputation scores address reputation inflation and facilitate worker differentiation.

2.3. Connection with recommender systems

Table 1 demonstrates the gap within the reputation systems literature that this study fills. Yet, and similar to reputation systems, recommender systems also resolve information asymmetries and allow customers to make better-informed decisions (Dellarocas 2006, Adomavicius and Tuzhilin 2005, Adamopoulos 2013). *Can such recommender systems resolve reputation staticity, reputation attribution, and reputation inflation and provide current skillset-specific worker-reputation scores in online labor markets?*

2.3.1. Differences between reputation and recommender systems: Reputation and recommender systems serve conceptually different objectives (Jøsang et al. 2013). Overall, reputation frameworks are broader in scope. For instance, reputation scores can generate product rankings (Amazon 2020), and, at the same time, they can provide quality estimates that affect product valuations (Moreno and Terwiesch 2014) and shape expectations (Ho et al. 2017). Even further, reputation systems can provide product managers with constructive feedback on how to improve their products (Proserpio and Zervas 2017). Besides, and specifically to online labor markets, employers can use reputation scores to invite workers to apply to their job openings (Rahman

2018b). Once workers apply to a job opening, employers can rank and choose applicants according to their reputation scores (Kokkodis et al. 2015, Abhinav et al. 2017, Kokkodis 2018).

On the other hand, recommender systems often serve a single objective. For instance, they recommend products that customers usually buy together (basket analysis; see Agrawal et al. 1994). Or, they use customers' observed actions to recommend items that users will choose next (next-item recommendations; see Quadrana et al. 2018, Rendle et al. 2010). Or, they provide product recommendations based on what similar users with the targeted user have liked in the past (Billsus and Pazzani 1998, Breese et al. 1998, Adomavicius and Tuzhilin 2005, Breese et al. 1998, Delgado and Ishii 1999). Specifically to online labor markets, existing recommender systems (1) rank job applicants within a given opening (Kokkodis et al. 2015, Abhinav et al. 2017), (2) recommend tasks for workers to apply (Baba et al. 2016, Goswami et al. 2014, Horton 2017), and (3) provide career path recommendations (Patel et al. 2017, Kokkodis and Ipeirotis 2020).

In many cases, reputation and recommender systems work together to increase trust and reduce uncertainty (Jøsang et al. 2013). In online labor markets, in particular, reputation scores are often predictors in probabilistic recommender systems of different objectives. For instance, job-applicant recommenders use worker reputation as one of the attributes in their classification approaches (Kokkodis et al. 2015, Abhinav et al. 2017). Career-path recommenders use worker-reputation scores to provide relevant skill recommendations (Kokkodis and Ipeirotis 2020). Systems that recommend tasks to workers use reputation scores to identify the most appropriate assignments (Hossain and Arefin 2019).

2.3.2. Recommender systems as worker-reputation frameworks: Despite these conceptual differences, recommender systems can adjust to provide worker-reputation scores. Traditional recommenders predict the rating a user would assign to an item (Ricci et al. 2011, Adamopoulos and Tuzhilin 2014). To apply these systems in the focal context, we need to map ratings, users, and items to their respective entities in a worker-reputation framework for online labor markets. Since workers' reputation is the desired outcome of the framework, workers map to a recommender system's users.

The mapping of items and ratings is more complicated. In traditional recommender systems, the items are static entities that do not evolve over time (e.g., movies, smartphones, songs). Besides, multiple users of traditional recommender systems buy, experience, and rate identical items (e.g., the same movie, song, smartphone). These multiple ratings per item are necessary for algorithms such as collaborative filtering to provide item recommendations that similar users to the focal user have liked in the past (Koren 2009, Adamopoulos and Tuzhilin 2013). In the focal context, the rated items are the completed tasks. In online labor markets, very rarely (if ever) two tasks are identical; even tasks that require the same skills might have different objectives. As a result, we do not observe multiple ratings for each task; instead, each task receives only a single rating. Hence, to transform recommender systems to a worker-reputation framework, we need to create static items for which multiple workers receive ratings.

A straightforward noisy transformation is to consider tasks that require the same skillsets as identical items. Then, the skillset-specific average feedback score that each worker receives can map to a recommended item's rating. Using these mappings, I can fill the user-item matrix, and implement matrix-completion algorithms (Adomavicius and Tuzhilin 2005, Koren 2009) that will provide worker-reputation scores for each available skillset (item). Even though this mapping is not perfect (e.g., each worker might perform multiple tasks that require the same skillsets over time and receive different ratings), it provides skillset-specific reputation scores and potentially addresses reputation attribution. What about reputation staticity?

The rich literature on recommender systems provides a plethora of sequence-aware (or session-based) approaches (Hsueh et al. 2008, Zang et al. 2010, Jannach et al. 2015, Quadrana et al. 2018) that could potentially address reputation staticity. These systems explicitly model sequences of past events and recommend items according to the user's short-term behavior (Quadrana et al. 2018). Because they focus on next-item recommendations, sequence-aware approaches use implicit user feedback (i.e., whether or not a user has bought a product) ignoring explicit feedback ratings that describe whether or not the user actually liked the bought product (Kula 2018, Devooght and

Table 2 Comparison of relevant literature of recommender systems

Type of recommender systems	Objective	Required modifications	Reputation inflation	Reputation attribution	Reputation staticity	Methodology
Collaborative filtering (Adomavicius and Tuzhilin 2005, Koren et al. 2009)	Recommend items that similar users to the targeted user have liked in the past	Skillset ↦ item Worker ↦ user Avg. feedback ↦ rating	✗	✠	✗	Singular value decomposition, Slope one
Content-based (Adomavicius and Tuzhilin 2005)	Recommend items based on commonalities between items that a user has rated highly in the past	Skillset ↦ item Worker ↦ user Avg. feedback ↦ rating	✗	✠	✗	Cosine similarity, Euclidean distance, Predictive modeling
Basket analysis (Agrawal et al. 1994)	Recommend items that users often buy together	Not applicable	✗	✗	✗	Apriori, Association rule mining
Sequence-aware (next-item) recommenders (Quadrana et al. 2018)	Recommend an item that a user should buy next based on the observed user history	Encoding of user actions (implicit feedback) to represent worker reputation (Section 5.4.1)	✗	✠	✠	Convolutional neural networks, Long short-term memory, Factorized personalized Markov chain
Recommenders in OLMs (Abhinav et al. 2017, Kokkodis et al. 2015, Hossain and Arefin 2019)	Recommend skills and tasks to workers, or job-applicants to employers	Predictive models adjusted to regression output	✗	✗	✗	Logistic regression, Bayesian networks, Support vector machines, Decision trees, Random forest
This research	Provide current and skillset-specific worker reputation	None	✓	✓	✓	Deep learning, Hidden Markov Models

Avg. feedback: the average feedback score that a worker receives on a given skillset (item). OLMs: Online labor markets.

Column "Required modifications" summarizes the necessary modifications and assumptions that recommenders need to make in order to provide skillset-specific worker-reputation scores. Columns "Reputation inflation," "Reputation attribution," and "Reputation staticity" identify whether the recommender system adaptation addresses these shortcomings of current reputation systems in online labor markets.

✠: Required encoding and item assumptions could potentially address reputation attribution and reputation staticity. Section 5.4 empirically shows that these assumptions do not resolve reputation attribution and reputation staticity sufficiently, and a s a result, they also do not resolve reputation inflation.

Bersini 2017, Quadrana et al. 2018). Yet, most reputation frameworks require explicit feedback ratings to work (Tadelis 2016, Luca 2017, Einav et al. 2016, Stackoverflow 2018, Amazon 2018, eBay 2018, Kokkodis and Lappas 2020, Kokkodis et al. 2020a). As a result, the application of next-item recommenders in the studied context will require significant encoding assumptions of user actions to generate next-item recommendations that provide skillset-specific worker-reputation scores. I discuss these encoding assumptions in detail in Section 5.4.1.

Table 2 compares relevant literature in recommender systems with the proposed approach and demonstrates the necessary modifications that relevant recommender systems need to make to provide worker-reputation scores. When applied in Section 5.4 and Figure 8, these modifications significantly hurt the performance of various recommenders, highlighting the need for a context-appropriate reputation system that addresses reputation inflation, reputation attribution, and reputation staticity.

3. A dynamic, multidimensional reputation framework

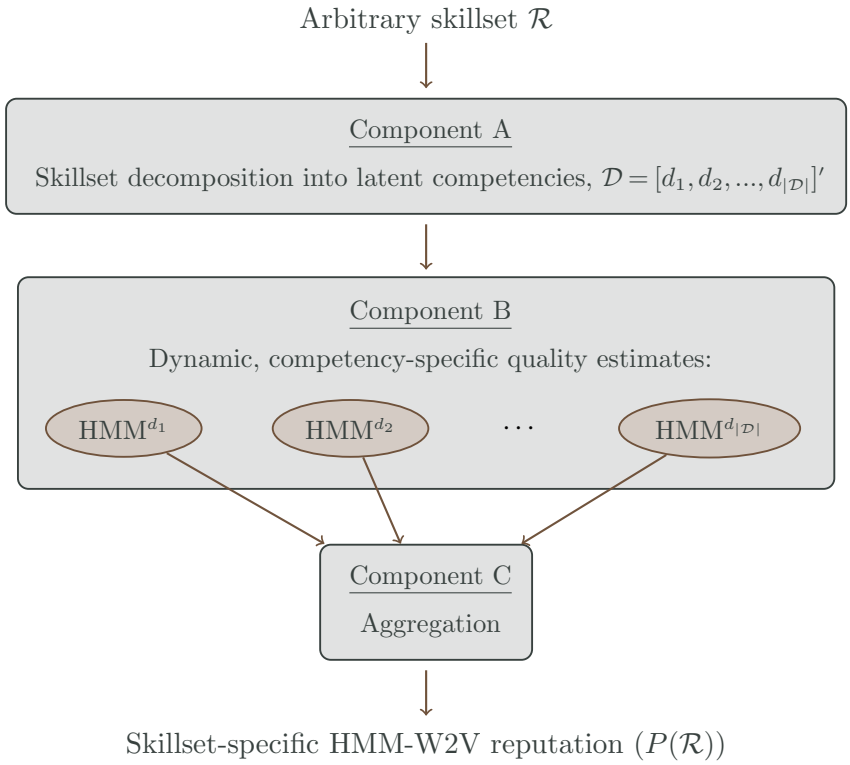
Section 2.2.3 summarized the three design principles that a reputation framework should follow in order to provide current skillset-specific worker quality scores. Based on these principles, I structure a reputation framework (`HMM-W2V-framework`) that consists of three components. Component A focuses on decomposing skills into competency dimensions that allow the framework to generalize and accommodate any arbitrary number of skills. Component B builds a dynamic model that combines multiple signals to estimate a worker's current, competency-specific quality. Component C aggregates competency-specific predictions to get a current estimate of the worker's quality on any set of skills. Figure 1 draws the interconnections of these three components, which I describe in detail next.

3.1. Component A: Skills decomposition

Workers can work on any arbitrary combination of skills, which creates a space of tens of thousands of available combinations of unique skillsets (Table 3). In theory, I could directly estimate the reputation of each worker on any observed skillset. This approach, however, has three drawbacks. First, independent of the size of the analyzed data, considering distinct skillset-specific observations will result in sparse training datasets. Second, such observations would ignore (by construction) correlations between various skillsets. Third, the entrance of new skills would require retraining of the framework.

To overcome these drawbacks, I use a distributed representation of words model (Word Embedding, W2V; see Mikolov et al. 2013) that projects individual skills into a set of competency dimensions. W2V embeds words from a vocabulary into a lower-dimensional space, in which semantically similar words appear close to each other, while semantically dissimilar words appear far away from each other (Mikolov et al. 2013). In the context of online work, a "skill" maps to a "word" and a "skillset" to a "document." Based on this representation, W2V projects contextually similar skills close to each other in a $|\mathcal{D}|$-dimensional space of competencies. (The actual number of competency dimensions $|\mathcal{D}|$ is a hyperparameter of the framework.)

Figure 1 **The `HMM-W2V-framework` provides current, skillset-specific worker-quality estimates**



The three design principles (Section 2.2.3) define the three-component structure of the `HMM-W2V-framework`. Component A maps any set of skills to competency dimensions. Component B provides dynamic models (hidden Markov models, HMM) that make current, competency-specific quality estimates. Component C aggregates these estimates to provide a skillset-specific reputation.

The `HMM-W2V-framework` maps any observed skillset $\mathcal{R}$ into a $|\mathcal{D}|$-dimensional vector space of competencies (Figure 1). To do so, it averages competency-specific scores of each skill in a given skillset. In particular, a skillset $\mathcal{R}$ maps to an aggregated W2V representation as follows:

$$\hat{w}^d_{\mathcal{R}} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \text{W2V}^d(r), \ \forall d \in \mathcal{D}\,, \tag{1}$$

where $\hat{w}^d_{\mathcal{R}}$ is the $d-$competency score of skillset $\mathcal{R}$, and $\text{W2V}^d(r)$ is the W2V score for skill r in dimension d. I normalize these weights to create a scale-invariant decomposition through a softmax transformation:

$$\mathbf{w}_{\mathcal{R}} = \text{softmax}\left([\hat{w}^1_{\mathcal{R}}, \hat{w}^2_{\mathcal{R}}, ..., \hat{w}^{|\mathcal{D}|}_{\mathcal{R}}]'\right) = [w^1_{\mathcal{R}}, w^2_{\mathcal{R}}, ..., w^{|\mathcal{D}|}_{\mathcal{R}}]'\,. \tag{2}$$

Alternatively, a distributed memory model (D2V; see Le and Mikolov 2014) or simpler clustering approaches could map skillsets into vectors of real numbers. Furthermore, in addition to the skillsets, W2V or D2V could also consider the job-description text. Appendix C.1 and Figure 11A discuss and empirically compare these alternative approaches.

3.2. Component B: Competency-specific dynamic quality assessment

At any given time, each worker has a quality level in each competency dimension $d \in \mathcal{D}$. This quality is latent (unobserved). However, for each completed task t with required skills $\mathcal{R}$, the market observes a feedback score (Y_t) that the worker receives. This feedback maps into competency-specific scores through the weighting vector $\boldsymbol{w}_{\mathcal{R}}$: $Y_t^d = w_{\mathcal{R}}^d Y_t$, $\forall d \in \mathcal{D}$. These scores ($Y_t^d$) form a sequence of proxies of the worker's quality in each competency dimension $d \in \mathcal{D}$.

In addition to latent, a worker's quality dynamically evolves. Specifically, a worker's quality can change in any competency dimension, either by gaining experience on the platform and better understanding the expectations of the employers or by learning new skills and continuously expanding current knowledge and abilities. The `HMM-W2V-framework` formulates this evolution through a hidden Markov model (HMM): Each worker operates from a latent, competency-specific state, which determines the worker's propensity to perform with score Y_t^d. Every time a worker completes a new task and receives a new feedback score, the framework observes new evidence about the worker's competency-specific qualities and stochastically transitions the worker to new latent states. The framework assumes a set of $\mathcal{S}^d$ latent states that describe K^d different levels of quality for each competency d, $\mathcal{S}^d = \{s_1, s_2, \ldots, s_{K^d}\}$.

HMM structure: Every new worker who joins the platform has an unknown quality across the competency dimensions $d \in \mathcal{D}$. As the worker completes new tasks on the platform, the market observes signals (i.e., through task outcomes) that correlate with the latent worker quality. To capture this behavior, the `HMM-W2V-framework` assumes an initial latent state s_1, where all new workers land. This state makes an average initial estimate of workers' competency-specific quality.[2] Once

[2] Alternatively, workers could land stochastically to any of the available states. This would add noise to the estimates, and as a result it could potentially hurt the performance of the framework.

the workers complete their first task and emit an observation (i.e., $Y_1^d,\ \forall d \in \mathcal{D}$), they stochastically transition to different states according to the parameters of the model.

To define an HMM for a given competency dimension d, I need (1) a vector of initial state probabilities $\boldsymbol{\pi}^d$, (2) a transition matrix T^d that stores the transition probabilities between states, and (3) an emission matrix E^d that describes the state-specific probability distributions for observations Y_t^d. Since every new worker lands in state s_1, the initial probability vector of each HMM is the following:

$$\boldsymbol{\pi}^d = [1, 0, 0, ..., 0]' \,. \tag{3}$$

A worker's history provides multiple observable signals that correlate with transitions to new quality states (e.g., total wages received, hiring rates, number of completed tasks). Such historical attributes define a vector $\mathbf{Z}_{t-1}$.[3] By weighing this vector with $w_{\mathcal{R}}^d$, the framework forms vectors $\boldsymbol{Z}_{t-1}^d = w_{\mathcal{R}}^d \mathbf{Z}_{t-1}$ that capture competency-specific histories. Each $\boldsymbol{Z}_{t-1}^d$ directly affects the transition probabilities to different states (i.e., matrix T^d). Formally, assume that a given worker completes task $t-1$ from state s_k in a given dimension d. Once the framework observes the outcome of task $t-1$, it estimates the transition probability of this worker to move to state state s_l as follows:

$$\lambda^{d,s_k s_l}_{\boldsymbol{\gamma}_{kl}^d \boldsymbol{Z}_{t-1}^d} = \Pr(S_t^d = s_l | S_{t-1}^d = s_k; \boldsymbol{\gamma}_{kl}^d, \boldsymbol{Z}_{t-1}^d) = g^d(\boldsymbol{\gamma}_{kl}^d \boldsymbol{Z}_{t-1}^d) \,. \tag{4}$$

In the previous Equation, $\boldsymbol{\gamma}_{kl}^d$ is the vector of coefficients of state s_k that define the weights of $\mathbf{Z}_{t-1}^d$ in estimating the transition probability to state s_l. Function g^d transforms the product $\boldsymbol{\gamma}_{kl}^d \boldsymbol{Z}_{t-1}^d$ into a probability. (The choice of function g^d is context- and data-specific. I discuss this in Section 5.1 and Appendix A.) The complete transition matrix for a given worker after completing task $t-1$ in dimension d is as follows:

[3] Vectors $\mathbf{Z}_{t-1}$ and $\mathbf{X}_t$ (that I use later) are worker-specific. As a result, the transition and emission probabilities are also worker-specific. For simplicity, I drop worker subscript i from the notation of this subsection. I use the subscript i in Section 3.3 and Appendix B to highlight that the skillset-specific reputation scores are also worker-specific.

$$T^d(\mathbf{\Gamma}^d, \boldsymbol{Z}^d_{t-1}) = \begin{bmatrix} \lambda^{d,s_1 s_1}_{\boldsymbol{\gamma}^d_{11}\boldsymbol{Z}^d_{t-1}} & \lambda^{d,s_1 s_2}_{\boldsymbol{\gamma}^d_{12}\boldsymbol{Z}^d_{t-1}} & \dots & \lambda^{d,s_1 s_{K^d}}_{\gamma_{1K^d}\boldsymbol{Z}_{t-1}} \\ \lambda^{d,s_2 s_1}_{\boldsymbol{\gamma}^d_{21}\boldsymbol{Z}^d_{t-1}} & \lambda^{d,s_2 s_2}_{\boldsymbol{\gamma}^d_{22}\boldsymbol{Z}^d_{t-1}} & \dots & \lambda^{d,s_2 s_{K^d}}_{\boldsymbol{\gamma}^d_{2K^d}\boldsymbol{Z}^d_{t-1}} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda^{d,s_K s_1}_{\boldsymbol{\gamma}^d_{K1}\boldsymbol{Z}^d_{t-1}} & \lambda^{d,s_{K^d} s_2}_{\boldsymbol{\gamma}^d_{K2}\boldsymbol{Z}^d_{t-1}} & \vdots & \lambda^{d,s_{K^d} s_{K^d}}_{\boldsymbol{\gamma}^d_{K^d K^d}\boldsymbol{Z}^d_{t-1}} \end{bmatrix}, \tag{5}$$

where $\mathbf{\Gamma}^d = [\boldsymbol{\gamma}^d_{11}, \boldsymbol{\gamma}^d_{12}, \dots, \boldsymbol{\gamma}^d_{K^d K^d}]'$.

Similarly, observed worker characteristics (e.g., hourly rate, average feedback score) are correlated with the observed emissions of the HMM (matrix E^d). These characteristics form a vector $\mathbf{X}_t$. The framework makes this vector competency-specific by weighting its elements with $w^d_{\mathcal{R}}$, i.e., $\mathbf{X}^d_t = w^d_{\mathcal{R}}\mathbf{X}_t$. Formally, the conditional probability of observing Y^d_t given the current state of the worker S^d_t is:

$$\Pr(Y^d_t | S^d_t = s_k; \boldsymbol{\theta}^d_k, \boldsymbol{X}^d_t) = f^d(\boldsymbol{\theta}^d_k \boldsymbol{X}^d_t)\,, \tag{6}$$

where $f^d(.)$ is a continuous probability distribution (e.g., Beta), and $\boldsymbol{\theta}^d_k$ is the parameter vector of the continuous distribution for state k and competency d. The complete parameter vector $\mathbf{\Theta}^d$ (for all states $s_k \in \mathcal{S}^d$) is as follows:

$$\mathbf{\Theta}^d = [\boldsymbol{\theta}^d_1, \boldsymbol{\theta}^d_2, ..., \boldsymbol{\theta}^d_{K^d}]'. \tag{7}$$

Appendix B presents the derivation of the likelihood and the subsequent process of estimating the parameters of the model. Appendix C discusses the choice of emission (g^d) and transition (f^d) functions, as well as the tuning of the total number of states K^d. Finally, Appendix C compares alternative approaches for modeling component B of the `HMM-W2V-framework`, including recurrent neural networks and gradient boosting.

3.3. Component C: Aggregation

This process happens independently for each competency dimension $d \in \mathcal{D}$. As a result, for a given worker i who has completed t tasks, each HMM estimates a current competency-specific quality p^d_{it} (i.e., a stochastic draw from the continuous emission distribution of Equation 6). To estimate the

quality of worker i for any given set of skills $\mathcal{R}$, the `HMM-W2V-framework` aggregates the available competency-specific estimates:

$$P_{it}(\mathcal{R}) = \sum_{d \in \mathcal{D}} p_{it}^{d} \,. \tag{8}$$

Summation of these estimates allows each competency to contribute to the skillset-specific reputation by its respective, skillset-specific weight (recall that Equation 2 softmaxes the weights of each dimension). Appendix C and Figure 11C compare alternative aggregating approaches.

4. Data description and model-free evidence

I build and evaluate a version of the proposed framework on a set of real transactions from a major online labor market, `LaborBazaar` (pseudonym). The focal data forms a snapshot of 662,423 task applications that led to 58,459 completed tasks by 13,510 workers. `LaborBazaar` supports diverse tasks from different categories, including software and web development, writing, sales, marketing, and data science. Table 3 summarizes the dataset. Overall, 547 unique skills create a total of 17,563 unique skillsets. Workers participate in this platform remotely from 141 different countries. I follow the actions of these workers for twelve consecutive months.

4.1. Model-free evidence

On `LaborBazaar`, employers rate workers after the completion of a task with a score $Y \in \{0, 1/9, 2/9, ..., 9/9\}$. The platform supports a state-of-the-art reputation system (Filippas et al. 2018): The employer and the worker each get two weeks to leave their feedback score. This is a double-blind process, where neither party learns its rating before leaving a rating for the other party. Figure 2A shows the resulting feedback distribution of the focal data. The mean and median of this distribution are both 0.86. Most of the workers appear to perform almost perfectly (reputation inflation).

At the same time, the heterogeneity in terms of skills and qualifications on the platform in combination with the fact that feedback scores are assigned uniformly (reputation attribution) adds noise to the inflated distribution. Figure 2B shows that skills accumulate different feedback scores: from `translation`, with median 1 and mean 0.91, to `twitter-marketing` with median 7/9 and

Table 3 Data overview

	Obs.	Mean	Median	StD	Min	Max
Skills per worker	662,423	9.4	9	5.8	1	61
Tasks per worker	58,459	6.3	5	3.6	1	59
Skills per task	58,459	3.2	3	2.5	1	38
Task compensation ($)	58,459	170	31	1,042	3	70,218
Task applications	662,423					
Completed tasks	58,459					
Unique skills ($\mathbb{R}$)	547					
Unique observed skillsets	17,563					

Workers work remotely from 141 countries. Data spans 12 months.

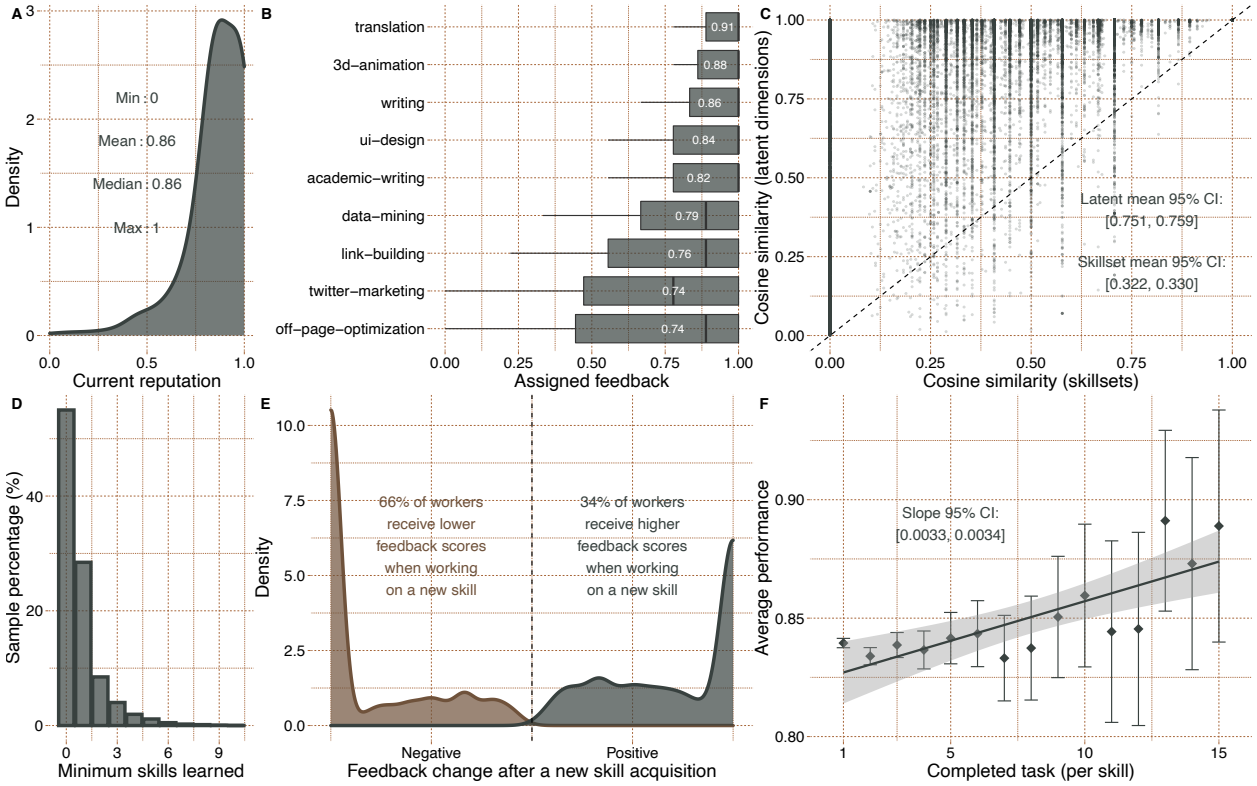
mean 0.74. Figure 2C further shows that workers work on heterogeneous tasks. Specifically, the $x-$axis shows the cosine similarity of consecutive tasks in terms of skillsets, and the $y-$axis shows the cosine similarity between consecutive tasks in terms of projected latent dimensions (Section 3.1). The mean cosine similarity of consecutive tasks is 0.33, suggesting that workers indeed work on heterogeneous tasks (reputation attribution). The same figure further shows how mapping skillsets through W2V captures contextual similarities between different skillsets, and as a result, yields higher cosine similarity between consecutive skillsets (average mean similarity of consecutive tasks in the latent space is 0.75).

Figure 2D shows that around 47% of the workers of this sample use in the marketplace at least one new skill. Recall that I follow the focal workers only for twelve months. As a result, the observed skillset evolution happens during these 12 months. Figure 2E shows shows that when workers acquire a new skill, they often (66%) initially receive lower feedback scores. However, as they gain experience, Figure 2F shows that their reputation increases. These three graphs highlight the dynamic nature of workers, which suggests that reputation systems should adjust for reputation staticity.

4.2. Emission and transition variables

The `HMM-W2V-framework` requires a set of emission and transition variables that describe vectors $\mathbf{Z}_{t-1}$ and $\mathbf{X}_t$. Recall that each latent state represents a different distribution of expected service

Figure 2 Model-free evidence of shortcomings of current reputation systems



The six figures describe the reputation system of `LaborBazaar`. Figure A shows that the accumulated feedback scores of workers are inflated. Figure B shows that different skills have different feedback score distributions. Figure C shows that workers work on heterogeneous tasks—skillset average cosine similarity between consecutive tasks is 0.33—hence highlighting reputation attribution. Figures D show that workers evolve by learning new skills. Such new skill acquisitions usually initially hurt worker reputation (Figure E). However, over time, workers gain experience and perform better (Figure F). As a result, Figures D, E and F combined highlight the problem of reputation staticity.

quality. Transitions between states are subject to the accumulated experience and feedback scores of each worker that form vector $\mathbf{Z}_{t-1}$. In particular, I allow five signals to affect transitions: (1) the current accumulated reputation of the worker, (2) the total money earned on the platform, (3) the total number of completed jobs, (4) the total number of hours worked, and (5) the worker's hiring rate.

Similarly, vector $\mathbf{X}_t$ that captures observed worker characteristics affects the emission probabilities. Such characteristics include the current reputation of the worker and the worker's current hourly rate. Table 4 shows the descriptive statistics of the variables that form vectors $\boldsymbol{X}_t, \boldsymbol{Z}_{t-1}$, as well as

Table 4 Descriptive statistics for the attributes in vectors $\boldsymbol{X}_t, \boldsymbol{Z}_{t-1}$, and the outcome variable $\boldsymbol{Y}_t$.

	Mean	Median	StD	Min	Max
Observed outcome (Y_t)	0.85	1	0.25	0	1
Transition vector $\boldsymbol{Z}_{t-1}$					
Current reputation	0.86	0.86	0.15	0	1
Total money earned ($)	4,289	459	11,735	0	152,526
Completed jobs	9.71	2	20.68	0	401
Work-hours	468	31	1,368	0	36,457
Hiring rate	0.07	0.05	0.07	0	1
Emission vector $\boldsymbol{X}_t$					
Current reputation	0.86	0.87	0.15	0	1
Hourly rate ($)	11.23	8.69	13.21	3	397

the outcome variable Y_t. I log-transform variables with long tails and standardize all non-binary variables for faster convergence. Finally, note that this illustrative list of variables that formulate HMM transitions and emissions is context-specific. Appendix E shows how alternative contexts require different variable choices.

5. Evaluation of the `HMM-W2V-framework`

The next paragraphs describe the modeling choices and hyperparameter tuning of the `HMM-W2V-framework` and compare its performance with various alternative advanced reputation systems and modified recommender systems. I split the data into ten folds that consist of different workers (i.e., each worker's complete history appears only in one of the ten folds). I use 10-fold cross-validation to evaluate and compare each alternative design approach.

5.1. Design choices and hyperparameter tuning

Alternative design options could model each component of the framework (Section 3). To identify the best design choices for the context of this study, I follow a grid-search approach. Specifically, I compare alternative modeling choices for components A, B, and C, and test the framework's performance under various numbers of dimensions $|\mathcal{D}|$. Furthermore, I evaluate various combinations of numbers of states K^d, choices of transition functions g^d, and choices of emission functions f^d. Appendix C discusses the details of this grid-search approach.

Based on this analysis, the combinations that performed best in the focal context are the following:

- ◇ Modeling component A: W2V performs better than D2V and Gaussian mixture models (Appendix C, Figure 11A). Furthermore, including the job-description text in W2V adds noise and does not improve the performance of component A (Appendix C, Figure 11A). Hence, for the focal dataset, I choose W2V.
- ◇ Modeling component B: Modeling dynamic transitions through a hidden Markov model performs significantly better ($p < 0.001$) than linear models, support vector machines, recurrent neural networks, and gradient boosting (Appendix C, Figure 11B). Hence I choose the HMM structure described in Section 3.2 for the focal dataset.
- ◇ Modeling component C: Aggregating dimension-specific feedback according to Equation 8 performs on par or better than alternative aggregation approaches (Appendix C, Figure 11C). Hence, I use Equation 8 for the rest of the analysis.
- ◇ Number of dimensions: $|\mathcal{D}| = 10$ performs better than alternative values (Appendix C, Figure 11D).
- ◇ HMM parameters: For the focal context I choose (Appendix C.5):
  - ○ $K = [3, 4, 4, 4, 4, 4, 3, 4, 3, 3]$.
  - ○ $f^d =$ Beta $\forall d \in \mathcal{D}$.
  - ○ $g^d =$ Multinomial logit $\forall d \in \mathcal{D}$.

Using these design choices and hyperparameter values, the `HMM-W2V-framework` estimates the reputation of each available worker on any arbitrary set of skills.

5.2. Alternative reputation systems

Alternative approaches could also generate reputation scores for each available worker. Recent advances in machine learning provide multiple approaches that could model sequential observations and capture dynamic behavior. Furthermore, applicable context-specific approaches (e.g., WorkerRank; see Daltayanni et al. 2015) could potentially address reputation inflation (Table 1). To benchmark the performance of the `HMM-W2V-framework` against such advanced alternative models, I implement and compare the following reputation systems:

- ◇ *Current reputation*: The accumulated feedback score of `LaborBazaar`. This score is a result of a human-based reputation system, where each worker gets rated upon completion of a task, and these ratings accumulate to form worker reputation.
- ◇ *Machine learning approaches*: Alternative hybrid reputation systems that combine (1) human input (ratings), (2) observed characteristics, and (3) alternative modeling choices. These alternative systems model the relationship

$$Y_t \sim G(\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t)\,, \tag{9}$$

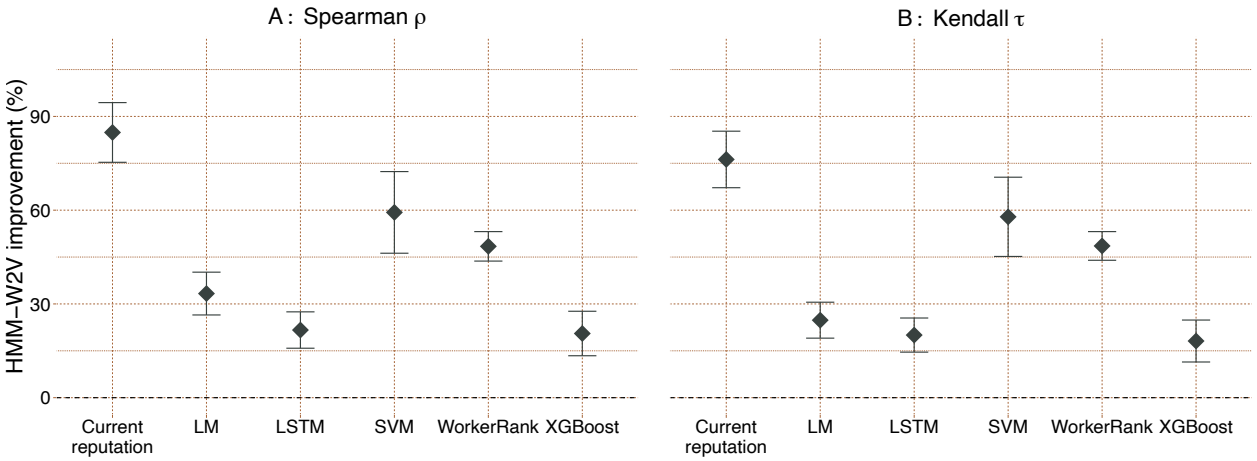
  where G represents the following:
  - ○ *Linear model*: G linearly regresses the dependent variable on $\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t$.
  - ○ *Recurrent neural networks (LSTM)*: G captures the relationships between vectors $\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t$, and Y_t through Long Short Term Memory networks (LSTM; see Hochreiter and Schmidhuber 1997).
  - ○ *Gradient boosting regression (XGBoost)*: G captures the relationships between vectors $\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t$, and Y_t through gradient boosting regression (Chen and Guestrin 2016).
  - ○ *Support vector regression (SVM-reg)*: G captures the relationships between vectors $\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t$, and Y_t through an SVM regression model (Smola and Schölkopf 2004).
- ◇ *WorkerRank*: An advanced reputation system for online labor markets that uses implicit feedback from employers (e.g., "hired," "invited") to rank workers through a link analysis approach (Daltayanni et al. 2015).

5.3. Results

The next paragraphs benchmark the performance of the `HMM-W2V-framework` against alternative reputation systems in terms of (1) ranking workers, (2) generating a representative reputation distribution, (3) estimating the service quality of the most dynamic "non-perfect" workers, and (4) ranking applicants within openings.

Ranking workers: The ultimate goal of any reputation system is to generate *rankings* of products or services (in this case, workers) according to their expected service quality. Hence, accurate

Figure 3 Ranking correlations of the `HMM-W2V-framework` and alternative reputation systems



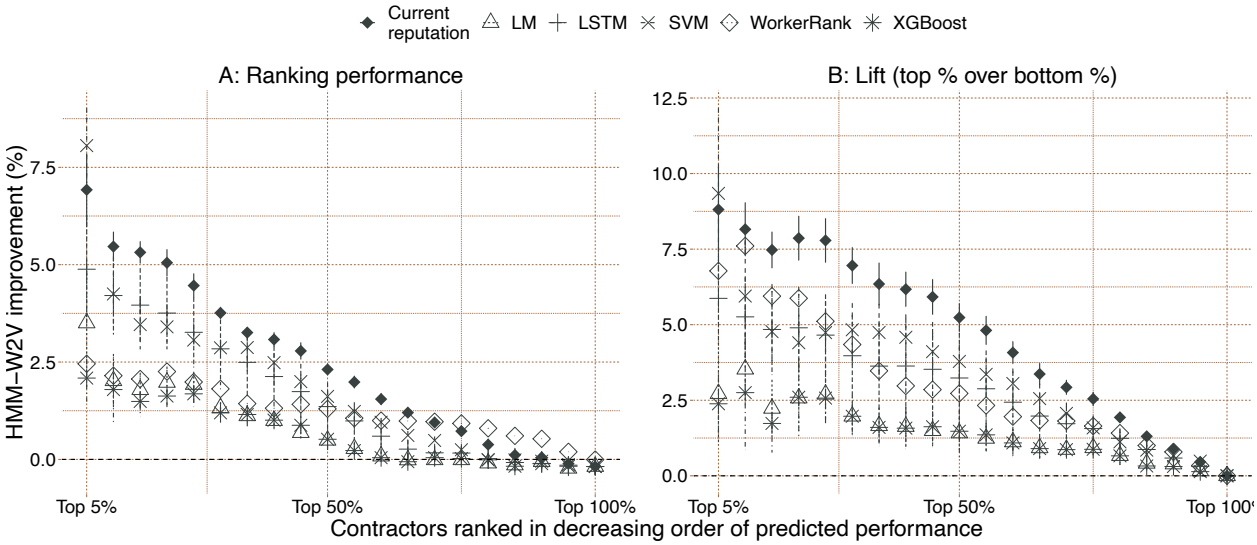
The $y-$axis shows the percentage improvement of the `HMM-W2V-framework` over the $x-$axis reputation systems, in terms of Spearman ρ and Kendall τ. The `HMM-W2V-framework` significantly ($p < 0.001$) outperforms all alternative reputation systems, with average 10-fold cross-validation improvements ranging between 20% and 85%. Error bars represent 95% confidence intervals.

assessment of quality should rank workers according to their likelihood of performing well on any given skillset $\mathcal{R}$. I capture the ranking performance of each reputation system through the following measures:

- ◇ Ranking correlations: Two ranking correlation coefficients (Kendall τ and Spearman ρ; see Kendall 1938, Spearman 1904) test the ordinal associations between quality estimates and observed outcomes.
- ◇ Ranking performance: Detailed analysis of the average performance of workers ranked in different cohorts tests whether workers ranked in the top tiers perform consistently better than workers ranked in the bottom tiers.
- ◇ Average lift: Lift analysis estimates how much better top-ranked workers perform than bottom-ranked workers.

Figure 3 compares the proposed approach with alternative reputation systems in terms of ranking correlations. The $y-$axis shows the 10-fold cross-validated percentage improvement of the `HMM-W2V-framework` over the $x-$axis reputation systems, in terms of Spearman ρ and Kendall

Figure 4 Ranking performance and lift of the `HMM-W2V-framework` and alternative reputation systems

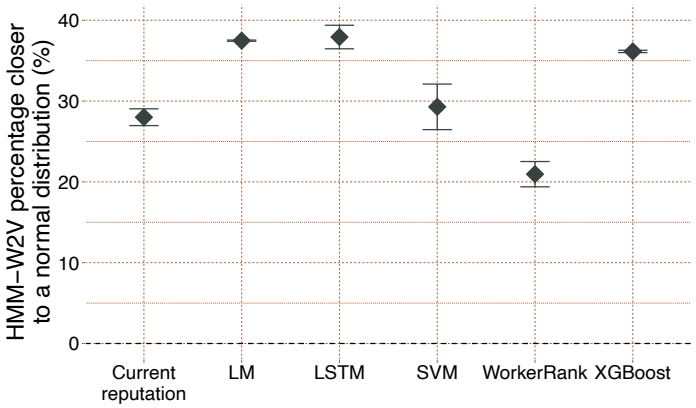


In both figures, the $x-$axis ranks workers according to their estimated reputation. The $y-$axis shows the 10-fold cross-validated percentage improvement of the `HMM-W2V-framework` over each alternative reputation system. The left figure shows the ranking performance: Top-ranked workers according to the HMM-W2V reputation clearly outperform ($p < 0.001$) top-ranked workers from all alternative reputation systems by up to 8%. The right figure shows the average lift of each ranked cohort. The `HMM-W2V-framework` yields up to 9% ($p < 0.001$) higher average lifts than the three baselines. Error bars represent 95% confidence intervals.

τ. These ranking correlations capture how aligned each reputation system's ranking is with the observed performance of the workers. Across both metrics, the proposed approach significantly outperforms both current and alternative reputation systems: The `HMM-W2V-framework` yields, on average, $\sim$85% better rankings than the current reputation scores. At the same time, it significantly ($p < 0.001$) yields better rankings than all alternative reputation systems (average improvement between 20% and 60%). Hence, the ordering of workers according to their predicted HMM-W2V reputation is significantly more accurate than their orderings according to alternative reputation systems.

Figures 4A and B show the ranking performance and lift improvement of the proposed approach over alternative reputation systems. The $x-$axis ranks workers according to their predicted reputation scores by each alternative approach. The $y-$axis shows the HMM-W2V reputation improvement

Figure 5 Reputation distributions of the `HMM-W2V-framework` and alternative reputation systems

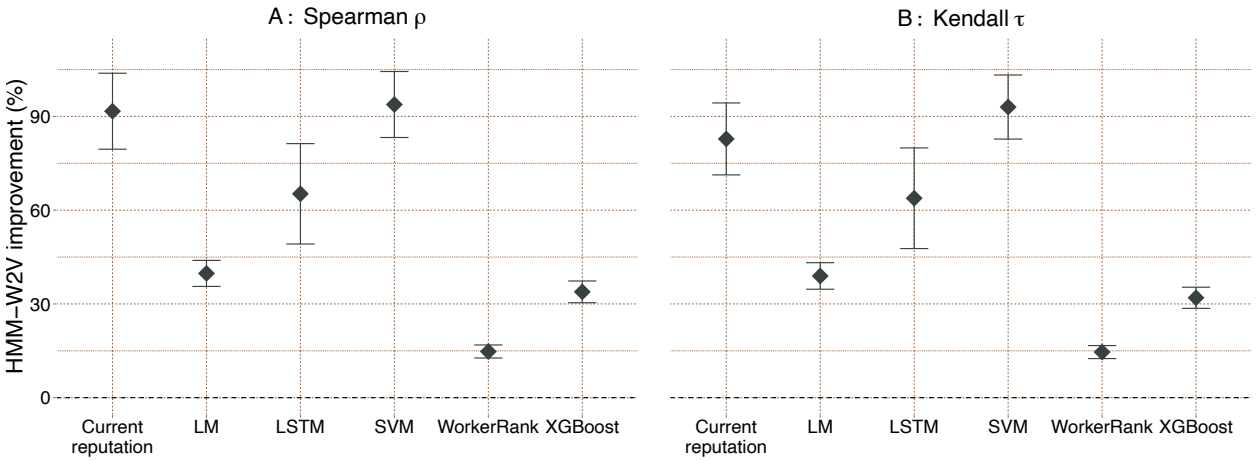


The $y-$axis shows the 10-fold cross-validated improvement of the `HMM-W2V-framework` in terms of the total variation distance compared to a normal distribution. Compared with alternative reputation systems, HMM-W2V reputation is up to 37% closer to a normal distribution. Error bars represent 95% confidence intervals.

over each alternative approach (A) in terms of the observed performance of the top-ranked workers, and (B) in terms of lift (i.e., how much better top-ranked workers perform than bottom-ranked workers). In Figure A, the `HMM-W2V-framework` significantly outperforms (up to 8%, $p < 0.001$) all alternative reputation systems: According to HMM-W2V reputation, higher-ranked workers yield significantly higher average performance. As the ranking moves from top to bottom, the improvement of the HMM-W2V reputation converges to zero as the top-ranked cohort includes a larger portion of the available contractors (i.e., the top-ranked sample's average performance converges to the population's average performance). Note that there is not a single point on the $x-$axis where an alternative reputation system outperforms the `HMM-W2V-framework`. Similarly, in Figure 4B, the proposed framework yields a higher average lift (up to 9%, $p < 0.001$) than all alternative approaches. This means that the rates of the HMM-W2V reputation top-ranked worker performance over the bottom-ranked worker performance are significantly ($p < 0.001$) higher than the respective rates of all alternative reputation systems.

Reputation distribution: Figure 5 compares the resulting reputation distributions of the `HMM-W2V-framework` and all alternative reputation systems. The focus is on estimating how close

Figure 6 Evaluation on "non-perfect" workers

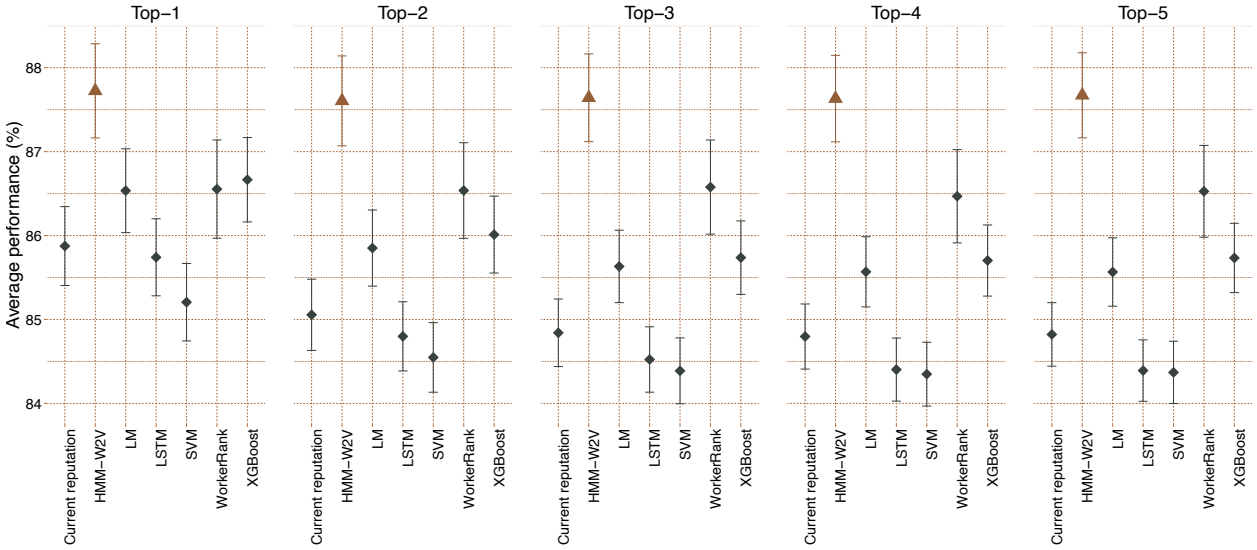


The y-axis indicates the improvement of the `HMM-W2V-framework` over the $x-$axis reputation systems in terms of Spearman ρ and Kendall τ. The `HMM-W2V-framework` outperforms all alternative reputation systems ($p < 0.001$) in identifying "non-perfect" workers. Error bars represent 95% confidence intervals.

each resulting distribution is to the normal distribution. A normal distribution is more likely to represent the skillset-specific abilities of workers (Schmidt and Hunter 1983) and facilitate worker differentiation. The total variation distance (Huber 2011, Kohl 2019) between any two distributions captures how close these distributions are. Hence, I use this distance to measure the closeness of each alternative reputation distribution to the normal distribution. Figure 5 shows that the proposed approach yields reputation distributions that are up to 37% closer to a normal distribution ($p < 0.001$) compared with the resulting distributions of alternative reputation systems.

Performance evaluation on "non-perfect" workers: Many workers in these markets always appear to perform well, in part due to reputation inflation (Table 4). As a result, models can have high accuracy by always correctly predicting these "perfect" workers. The real challenge is for algorithms to accurately predict the performance of workers who occasionally underperform (i.e., minority class prediction performance; see Longadge and Dongre 2013). This is important for the market, as early identification of potential underperforming workers could prevent employers from having a disappointing experience (Section 6).

Figure 7 Comparison of alternative reputation systems on within-opening rankings



The y-axis indicates the average performance of the Top$-n$ hired workers according to each reputation system. The `HMM-W2V-framework` outperforms ($p < 0.05$) all reputation systems but WorkerRank and XGBoost across all n. It further outperforms WorkerRank at $p < 0.05$ for $n \in \{4, 5\}$, and at $p < 0.1$ for $n \in \{1, 2, 3\}$, and XGBoost at $p < 0.1$ for $n = 1$, and at $p < 0.05$ for $n \in \{2, 3, 4, 5\}$. Error bars represent 95% confidence intervals.

To test the performance of each alternative reputation system on "non-perfect" workers, I estimate ranking correlations (Spearman ρ, Kendall τ) for the subset of workers who receive at least one imperfect (< 1) feedback score. Figure 6 shows the results. The focal framework significantly ($p < 0.001$) outperforms all alternative reputation systems in the populations that are harder to predict and are often costly for the marketplace (Section 6).

Performance evaluation on within-opening rankings: The previous paragraphs demonstrate the superiority of the proposed approach compared with alternative reputation systems in terms of differentiating workers (ranking correlations, reputation distribution). Each reputation system, however, could generate within-opening rankings of *applicants* (i.e., workers who have applied for the job). Such within-choice-set rankings often affect buyer decisions (Kokkodis et al. 2015, Ghose et al. 2012, 2014). To compare the set of alternative reputation systems in terms of within-opening rankings, I do the following:

- For each reputation system, rank applicants within openings according to their reputation score.
- For each $n \in \{1,2,3,4,5\}$, observe the average performance of workers who got hired while ranking at the Top$-n$ according to each reputation system.
- In the end, compare the average performance of each reputation system at Top$-n$.

Figure 7 shows the results. The $y-$axis indicates the average performance of workers that each algorithm ranks at Top$-n$. The $x-$axis lists all reputation systems. The `HMM-W2V-framework` outperforms ($p < 0.05$) all reputation systems but WorkerRank and XGBoost across all n. It further outperforms WorkerRank at $p < 0.05$ for $n \in \{4,5\}$, and at $p < 0.1$ for $n \in \{1,2,3\}$, and it outperforms XGBoost at $p < 0.1$ for $n = 1$ and at $p < 0.05$ for $n \in \{2,3,4,5\}$. These results suggest that the `HMM-W2V-framework` can offer better within-opening rankings of applicants than alternative reputation systems.

5.4. Empirical evaluation of recommender systems

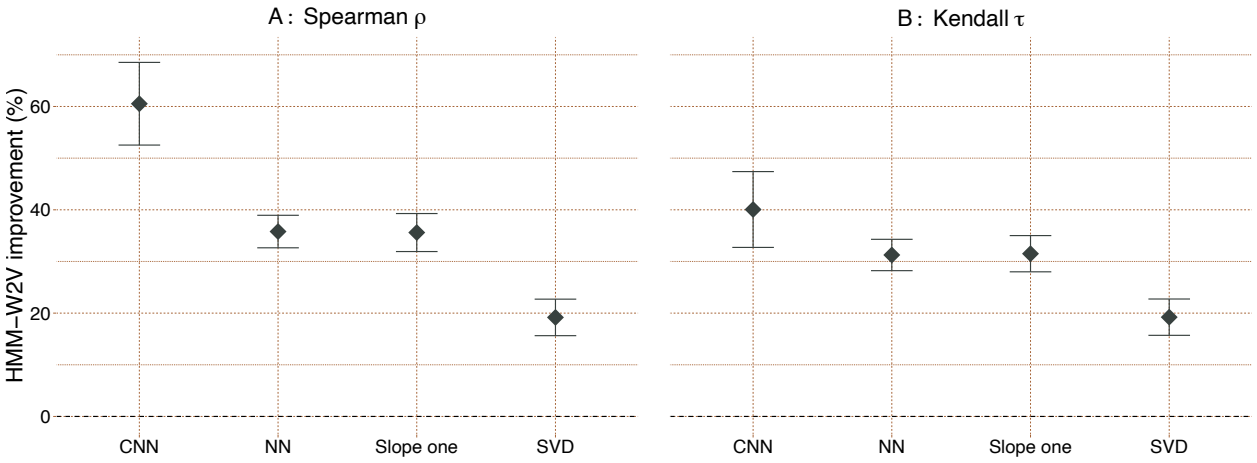
Section 2.3 summarized the conceptual differences between reputation and recommender systems. Table 2 further identified the necessary modifications and assumptions that recommender systems need to make in order to provide worker-reputation scores. Section 5.4.1 empirically tests such adaptations of recommender systems and shows that they significantly underperform compared with the `HMM-W2V-framework`. Finally, Section 5.4.2 illustrates how reputation and recommender systems can work together to enhance the transaction efficacy of a marketplace.

5.4.1. Adaptations of recommender systems as reputation frameworks: Section 2.3.2 and Table 2 identified that in order to adjust recommender systems to provide worker-reputation scores, I need to make the following assumptions:

- Skillset $\mapsto$ item.
- Worker $\mapsto$ user.
- Average worker's skillset-specific feedback score $\mapsto$ rating.

Based on these, I transform and implement the following popular recommender systems:

Figure 8 **Th `HMM-W2V-framework` outperforms adaptations of recommender systems**



The $y-$axis shows the percentage improvement of the `HMM-W2V-framework` over the $x-$axis recommender systems, in terms of Spearman ρ and Kendall τ. The `HMM-W2V-framework` significantly ($p < 0.001$) outperforms all alternative recommender systems, with average 10-fold cross-validation improvements ranging between 20% and 60%. Error bars represent 95% confidence intervals.

- *Collaborative filtering:* Collaborative filtering recommenders are among the most powerful and widespread industry approaches (Adomavicius and Tuzhilin 2005, Meyer 2012, Su and Khoshgoftaar 2009, Adamopoulos and Tuzhilin 2015). For the focal context, I customize three popular powerful collaborative filtering frameworks: k-nearest neighbors (NN, Kantor et al. 2011), singular value decomposition (SVD, Kantor et al. 2011), and slope one (Lemire and Maclachlan 2005).
- *Neural network sequence recommender systems:* The rise and popularity of neural networks have motivated approaches that use such networks to build deep recommender systems. Given the nature of the focal context and the fact that workers evolve dynamically, a sequential recommendation model could capture latent and dynamic relationships by treating recommendations as a sequential prediction problem (Kung-Hsiang 2018). Convolutional neural networks (CNN) often model such sequential recommender systems that provide next-item recommendations (Kula 2018, Quadrana et al. 2018). As mentioned in Section 2.3.2, because these approaches focus on implicit feedback (i.e., whether or not a user chooses an item next),

I need to encode the observed sequences of worker skillset-specific ratings into user actions. To do so, I combine a task's required skillset along with its respective observed performance. For instance, if a worker completes a sequence of tasks that require {`java`} and {`python`} and receives feedback scores 8/9, and 1, then the sequence will be [{`java`}-8/9, {`python`}-1]. These transformations generate sequences of observations that a deep recommender system can use to predict the next skillset-score combination of each worker. I use these next-skillset-score predictions to infer a worker's skillset-specific reputation.

Through a grid search approach, I tune these models, and compare their performance with the `HMM-W2V-framework`. Figure 8 shows how the proposed approach significantly ($p < 0.001$) outperforms all recommender systems in terms of ranking correlations. This underperformance shows that the necessary assumptions that transform recommender systems into worker-reputation frameworks likely hurt their performance. As a result, recommender systems do not sufficiently address reputation staticity, reputation attribution, and reputation inflation in online labor markets.

5.4.2. Collaboration of reputation and recommender systems: Reputation and recommender systems can work together to increase transaction efficacy. Specifically, worker-reputation scores can enhance the performance of recommender systems in online labor markets (Section 2.3.1). To demonstrate, I build existing job-applicant recommenders (Kokkodis et al. 2015, Abhinav et al. 2017), and I test their performance with and without using HMM-W2V reputation. Specifically, I build models that estimate the hiring probability of each job applicant (Kokkodis et al. 2015, Abhinav et al. 2017):

$$\Pr(\text{Hire}|\mathbf{W}) = h(\mathbf{W})\,, \tag{10}$$

where $h \in \{$Logistic regression, Bayesian networks, Random forest, Gradient boosting, Neural networks$\}$. The vector of observed job-applicant characteristics $\mathbf{W}$ takes the following forms:

$$\mathbf{W} = \begin{cases} \mathbf{W}_{cr} := \text{Current reputation} \\ \mathbf{W}_{hmm} := \text{HMM-W2V reputation} \\ \mathbf{W}_{p} := \text{Predictive features in Kokkodis et al. (2015)} \\ \mathbf{W}_{hmm \wedge p} := \mathbf{W}_{p} \wedge \text{HMM-W2V reputation} \end{cases} . \tag{11}$$

Appendix G presents the list of predictive features $\mathbf{W}_p$.

I evaluate the performance of each recommender by estimating their area under the curve (AUC). To highlight the benefits of using HMM-W2V reputation, I estimate the following improvements:

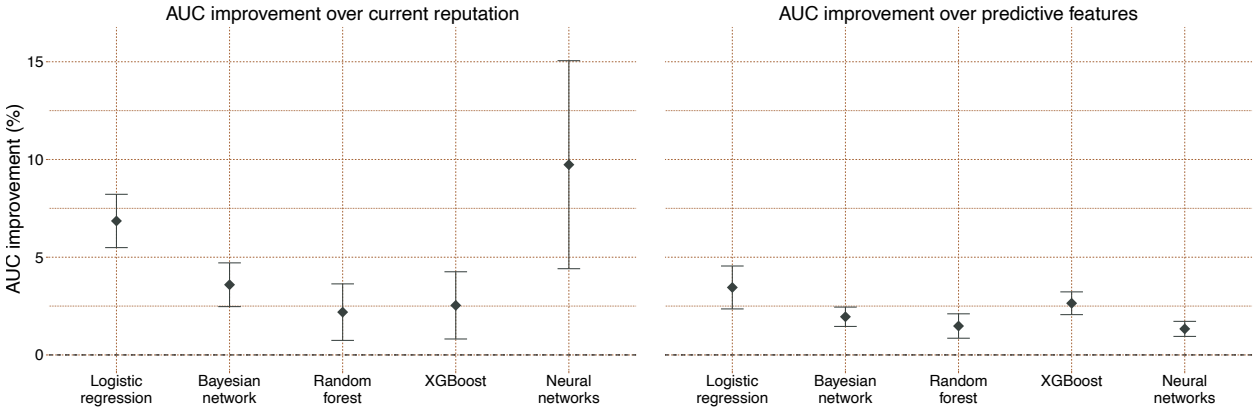
$$\text{Improvement over current reputation} = \frac{AUC(\mathbf{W}_{hmm}) - AUC(\mathbf{W}_{cr})}{AUC(\mathbf{W}_{cr})} * 100,$$
$$\text{Improvement over predictive features} = \frac{AUC(\mathbf{W}_{hmm \wedge p}) - AUC(\mathbf{W}_{p})}{AUC(\mathbf{W}_{p})} * 100\,.$$

Figure 9 shows the 10-fold cross-validated AUC improvements. First, compared with the current reputation, HMM-W2V reputation provides significantly ($p < 0.001$) better recommendations across all classifiers that yield an AUC improvement between 2.4% and 10%. Second, once I include the HMM-W2V reputation in the set of predictive features presented in Appendix G, the performance of the recommenders increases between 1.3% ($p < 0.001$) for neural networks and 3.5% ($p < 0.001$) for logistic regression. Such AUC improvements could generate better matches, happier employers, better collaborations, and a subsequent increase in market revenue (Kokkodis et al. 2015).[4]

To conclude, the proposed reputation framework outperforms adaptations of recommender systems, hence further highlighting the benefits of using the three-component architecture that addresses reputation attribution, reputation staticity, and reputation inflation. Furthermore, Figure 9 illustrates that reputation and recommender systems are not at odds, but instead, they provide better user experience when they work together.

[4] The high variance of the neural networks performance when modeling only reputation scores (i.e., only one feature) suggests that the networks possibly overfit the training sets and often fail in the test sets. Once additional features enter the model, trained neural networks make robust predictions that perform better than alternative approaches.

Figure 9 HMM-W2V reputation enhances the performance of job-applicant recommender systems



Using HMM-W2V reputation in recommender systems that rank job applicants according to their likelihood of getting hired (Kokkodis et al. 2015, Abhinav et al. 2017) results in better ($p < 0.001$) recommendations. The y-axis shows the AUC improvement when each recommender system on the $x-$axis includes HMM-W2V reputation as an attribute. This example illustrates how reputation and recommender systems can work together to improve transaction efficiency in a marketplace. The figure shows 10-fold cross-validation average improvements. Error bars represent 95% confidence intervals.

5.5. Additional evaluations and generalizability

Besides this analysis, I further evaluate the predictive and explanatory performance of the `HMM-W2V-framework` and its generalizability.

Predictive and explanatory performance: Appendix D provides performance evaluations and comparisons across different metrics. Specifically, Figure 13 shows that the `HMM-W2V-framework` significantly ($p < 0.001$) outperforms all alternative reputation systems in terms of predictive mean absolute error (MAE) and root mean squared error (RMSE). Even further, Table 5 shows that, compared with alternative reputation systems, HMM-W2V reputation better explains the variance of the observed performance in terms of R^2 in a linear regression specification.

Generalizability: One advantage of the focal approach is that it can generalize to other contexts that experience reputation attribution, reputation staticity, and reputation inflation. One such example is online reputation platforms (e.g., TripAdvisor, Yelp). These platforms deliver positively skewed reputation scores (Luca and Zervas 2016). At the same time, because venues evolve (e.g.,

change their menus, go through renovations, hire different personnel), their reputation is also dynamic. Finally, these restaurants receive evaluations for multiple dimensions, and as a result, their reputation systems likely experience reputation attribution as well. Appendix E implements the `HMM-W2V-framework` and the alternative reputation and recommender systems on 77,044 restaurant reviews from a major restaurant review platform: The `HMM-W2V-framework` provides significantly ($p < 0.001$) better restaurant reputation scores compared with alternative reputation systems (Figure 14) and adaptations of popular recommender systems (Figure 15). These results empirically show the generalizability of the `HMM-W2V-framework`.

One concern of the proposed approach is that platforms are already developing multidimensional reputation systems, and as a result, the `HMM-W2V-framework` might be only recovering noisy information from such human-rated dimensions. The focal restaurant review platform creates a perfect environment to test this, as it allows customers to rate venues across four dimensions: "Food," "Atmosphere," "Value," and "Service." Appendix E and Figure 16 compare the `HMM-W2V-framework` with alternative approaches that use as input these human-rated reputation dimensions. The `HMM-W2V-framework` significantly ($p < 0.001$) outperforms these alternatives, suggesting that the latent dimensions captured by the `HMM-W2V-framework` contain different information than the observed multidimensional reputation scores.

Overall, the empirical analysis in this section shows the advantages of the HMM-W2V reputation over a set of advanced alternative reputation and recommender systems in terms of ranking workers, identifying "non-perfect" workers, generating better within-opening rankings, and generating a more representative, normal-like reputation distribution.

6. Discussion

Current reputation systems in online labor markets experience three shortcomings: (1) reputation attribution (attribution of reputation to specific skills is infeasible), (2) reputation staticity (the assumption that worker quality is static and does not evolve), and (3) reputation inflation (feedback scores are positively skewed). To address these shortcomings, this work presents the

`HMM-W2V-framework`, which combines human input (assigned feedback scores) with machine learning (Word Embedding, Hidden Markov Models) to provide accurate quality estimates for any online worker on any given set of skills. The proposed framework includes three components. The first component maps skills into a latent space of finite competency dimensions and addresses reputation attribution. The second builds dynamic competency-specific quality assessment models and addresses reputation staticity. The final component aggregates these competency-specific assessments to generate a reputation score for any given set of skills. Because these reputation scores are skillset-specific and evolve over time, they generate a representative, closer-to-normal reputation distribution, hence addressing reputation inflation. Application of the proposed framework to two different datasets illustrates that, compared with alternative reputation systems, HMM-W2V reputation performs better in terms of (1) ranking workers according to their likelihood of performing well, (2) identifying "non-perfect" workers who are more likely to underperform and are harder to predict, (3) improving the ranking of within-opening choices, and (4) creating closer-to-normal reputation distributions that facilitate worker differentiation.

6.1. Research contributions

Given the projected growth of the number of online workers in the coming years (Agile-1 2016, Sundararajan 2016) and their dynamic nature (Oliver 2015, Kokkodis and Ipeirotis 2016), accurate quality assessment could be a defining factor of the ultimate reach of online work. This paper is the first to outline shortcomings of current reputation systems (i.e., reputation staticity and reputation attribution) and to explain why such systems underperform in this context. By identifying these shortcomings, this work provides a solution that generalizes to arbitrary sets of skills. Because it makes skillset-specific estimates that dynamically evolve, this work provides accurate skillset-specific reputation.

From a design perspective, this work extends the rich literature (Table 1) of reputation systems by combining human input with machine intelligence to enhance employer judgment in choosing appropriate online workers. Compared with previous human-based, machine-based, and hybrid

reputation systems, the proposed approach has unique dynamic attributes that fit the particular context of online work. Given that algorithmic collaboration between humans and machines is expected to grow (Jain et al. 2018), the proposed hybrid approach could be a baseline for future intelligence augmentation systems that could potentially use human input beyond the current uniform worker assessment (e.g., by having employers rating workers only in dimensions that employers have appropriate expertise in, or by using expert, third-party human raters).

6.2. Methodological contributions and generalizability

Methodologically, this paper provides a detailed guideline for markets that are interested in developing dynamic reputation systems by addressing methodological challenges that include the conceptualization, modeling, and estimation of a worker's reputation. Specifically:

- ◇ Skillset decomposition: This paper is the first to conceptualize skillsets as documents and propose the application of a text-analysis algorithm in a completely different context. As discussed in Sections 2 and 3 and illustrated in Appendix C and Figure 11, this decomposition is necessary, as it allows the proposed reputation framework to generalize on any number of available skills.
- ◇ HMM architecture: Section 3.2 describes how practitioners can conceptualize and formulate a suitable structure for an HMM that allows a series of observed signals to shape the transition and emission probabilities of workers of various qualities.
- ◇ Parameter estimation: Appendix B guides practitioners through the derivation of the global likelihood of the model and the estimation process of all the parameters.
- ◇ Design choices and evaluation: Appendix C presents the process of evaluating different components of the `HMM-W2V-framework` and selecting an appropriate configuration of the HMM. Section 5 and Appendices C and D guide practitioners on how to evaluate and compare alternative design choices and reputation systems.

These methodological contributions generalize beyond the focal context of online work. The `HMM-W2V-framework` can be adjusted and implemented in any online platform that experiences

reputation attribution, reputation staticity, and reputation inflation. For instance, Appendix E shows that, under the assumption that hotels and restaurants dynamically change, reputation platforms such as Yelp and TripAdvisor could use a similar framework to develop a more dynamic, up-to-date reputation system. Sharing economy platforms such as Uber and Lyft can borrow ideas from the proposed approach and develop similar reputation systems internally. Such internal systems could estimate the dynamic service quality of each driver and even identify heterogeneity in reputation across various types of trips (e.g., airport and train station trips, long trips out of town, trips in rush hour, Saturday night trips). Finally, online platforms that track workers' career paths, such as LinkedIn, can adapt the proposed approach and estimate the evolving expertise of their users across multiple skillsets. Recommender systems could then use such estimates to suggest new skills or promote workers to potentially relevant jobs.

6.3. Implications for platforms, workers, employers, and the future of work

Online labor platforms stand to benefit through implementing the proposed approach, as accurate reputation scores (1) help workers to differentiate, (2) guide employers to make informed and fast (reduced search cost; see Bakos 1997) decisions, and (3) enable the market to improve its recommendation algorithms. When workers can be accurately differentiated, the quality of the supply side of the market naturally increases. High-quality relevant workers are more likely to keep participating as they are in high demand. Lower-quality workers could be motivated to invest in different skills that are uncorrelated with their current skillset and respective reputation. At the same time, and as I showed in Section 2.3 and Figure 9, markets can improve the performance of their recommendation algorithms by using HMM-W2V reputation as an additional attribute. Better recommendations imply higher income and higher transaction efficacy (Kokkodis et al. 2015).

The performance of the proposed approach in terms of identifying "non-perfect" workers is of particular importance to market managers. Accurately predicting underperformance allows informing employers preemptively. Such interventions could potentially reduce the number of adverse outcomes. Employers who make better-informed and faster decisions that lead to better

outcomes are more likely to be happy and keep participating in the marketplace, thereby generating a continuous stream of revenue for the platform (Tripp and Grégoire 2011).

Through the proposed reputation framework, platforms can better understand the supply distributions across latent competencies and any arbitrary combination of skills. Based on such information, market managers can intervene where they deem appropriate (e.g., through targeted advertising on competencies that workers tend to underperform). Appendix H empirically analyzes the focal dimensions and explains how market managers can track worker performance and employer demand across competencies and devise appropriate interventions.

The proposed reputation framework combines human input (feedback scores) with machine learning (AI) to augment intelligence in decision making. Over time, through continuous training, the AI component of the framework will improve its performance. As the framework becomes better and more accurate, its effect on humans (both workers and employers) who interact with it will also intensify. Specifically, the AI framework will be facilitating increased differentiation of intelligent crowd and crowd abilities. Due to better differentiation, employers who interact with the system will be able to make intelligent decisions that likely lead to successful outcomes. Such outcomes could encourage participation in these markets and attract new employers. Similarly, workers that the system evaluates to have high relevant expertise will be able to find tasks to complete online seamlessly. On the other hand, the framework might marginalize workers that it does not evaluate as experts. For such workers, career development systems can provide recommendations for new skills to learn (Kokkodis and Ipeirotis 2020). At the same time, as the IA framework's performance improves, it will potentially allow new workers (without prior history on the platform) to complete tasks for which the system deems workers with history on the platform as inadequate. As a result, a better IA system will likely help workers with high-in-demand skills and abilities to succeed, and it will potentially drive workers who exercise low-demand skills to consider reskilling.

These effects of augmented intelligence could extend to offline work, as the proposed framework could generalize (for instance, through LinkedIn) to offline worker reputation. As automation keeps

evolving, the nature of many jobs will change, while other jobs will become obsolete (Brynjolfsson et al. 2018). This transition will require many workers to learn new skills: By some estimates, 120 million workers worldwide will need to be retrained as a result of automation in the next three years (Institute of Business Value 2019). Given this dynamic evolution of skills and workers, a reputation framework such as the one proposed in this work could successfully facilitate supply redistribution while intelligently differentiating relevant hireable workers.

6.4. Additional discussion of the proposed framework

To provide dynamic recommendations the proposed approach assumes that the competency-specific hidden quality states are discrete. In particular, the HMMs allow workers to transition across these discrete states as they complete new tasks. Once an HMM estimates the state of a worker, it predicts quality estimates that are continuous through Equation 6 and subsequent Equation 8. Hence, although the `HMM-W2V-framework` assumes discrete hidden states, it provides scores that are continuous and capture the complete spectrum of worker competency-specific qualities. Future work could also explore state-space models (e.g., Linear Dynamical Systems) that allow hidden states to be continuous (Murphy 2012). Given that the `HMM-W2V-framework` already provides continuous worker scores, the additional complexity of these models does not guarantee better performance. Even further, latent-space models might not provide clear market insights to managers: Because workers will not reside in discrete latent states, competency-specific market analysis (such as the one in Appendix H) will require threshold tuning.

Through the examination of different transition functions (Appendix A), I concluded that for the focal dataset, multinomial transitions yield better results. By definition, multinomial transitions allow workers to downgrade to lower-quality states. This might appear irrational at first: why would a worker's quality decrease over time? The way that an HMM works might provide some rationale. Because each HMM focuses on accurately capturing a worker's latent quality, more worker observations will generate less-noisy estimates of the worker's quality. For instance, consider a worker who receives high feedback scores in early tasks and low feedback scores in later tasks.

The HMM will (likely) initially assign this worker to a high-quality state. As the HMM evaluates new evidence, it will (likely) adjust the worker to a lower-quality state. As a result, allowing transitions to lower-quality states is vital for the `HMM-W2V-framework` in order to correct and adjust worker-quality estimates as new information arrives. Appendix I discusses this process in greater detail, and Figure 18 empirically shows the benefits of facilitating unconstrained HMM transitions.

An additional concern for the proposed approach is that platforms are already developing multidimensional reputation systems, and as a result, the `HMM-W2V-framework` might be only recovering noisy information from such human-rated dimensions. To investigate, Appendix H presents the skills that define each competency in the focal dataset. The complexity of these competency-specific skillsets demonstrate the mental load that humans would need to follow to decompose evaluations across different dimensions. Furthermore, in a human-rating scenario, the mapping of each skill to a non-primary dimension (that the proposed framework does automatically) would be practically infeasible. For instance, if a worker completes a data-mining job, a human rater would evaluate the worker's performance in that skill, but not in every other skill available on the market (e.g., video-production). The `HMM-W2V-framework` does this mapping automatically, as it implicitly estimates correlations between skills through the W2V decomposition (Section 3.1). Appendix H discusses this in detail, and Appendix E.3 empirically shows that the proposed mapping captures different information than the already implemented human-rated dimensions.

Finally, the fact that, compared with D2V, W2V was a more appropriate decomposition approach in the worker-reputation context (Appendix C.1) but less appropriate in the restaurant-reputation context (Appendix E) raises a question of when should markets prefer W2V over D2V. Conceptually, W2V is likely to perform better when the pool of terms is predefined, well-structured, and where every term includes crucial information. In these scenarios, where each individual term (e.g., a skill) contains crucial information, summing up mappings through Equation 1 should generate more informative representations than D2V mappings that will unavoidably include noise from rare combinations of terms (e.g., rare combinations of skills). On the other hand, D2V should perform

better when the analyzed text is unstructured (e.g., review text). In those cases, summing up weights of seemingly random terms (e.g., found in reviews) through Equation 1 will likely generate noisy representations.

6.5. Conclusion

Conclusively, this work presents an intelligence augmentation framework that addresses reputation attribution, reputation staticity, and reputation inflation. Application of this framework in two different contexts (online labor markets and online reputation platforms) shows that it can track evolving entities across multiple dimensions and provide accurate service quality estimates. As a result, its deployment on different types of online platforms could have significant implications for workers, employers, businesses, and the future of work.

References

Abhinav, Kumar, Alpana Dubey, Sakshi Jain, Gurdeep Virdi, Alex Kass, Manish Mehta. 2017. Crowdadvisor: A framework for freelancer assessment in online marketplace. *International Conference on Software Engineering*. 93–102.

Adamopoulos, Panagiotis. 2013. Beyond rating prediction accuracy: on new perspectives in recommender systems. *Conference on Recommender systems*. 459–462.

Adamopoulos, Panagiotis, Alexander Tuzhilin. 2013. Recommendation opportunities: improving item prediction using weighted percentile methods in collaborative filtering systems. *Proceedings of the 7th ACM conference on Recommender systems*. 351–354.

Adamopoulos, Panagiotis, Alexander Tuzhilin. 2014. On over-specialization and concentration bias of recommendations: Probabilistic neighborhood selection in collaborative filtering systems. *Conference on Recommender Systems*. 153–160.

Adamopoulos2015 Adamopoulos, Panagiotis, Alexander Tuzhilin. 2015. The business value of recommendations: A privacy-preserving econometric analysis. *International Conference on Information Systems*.

Adomavicius, Gediminas, Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Transactions on Knowledge and Data Engineering* **17** 734–749.

Agile 1. 2016. Gig economy. http://www.hrotoday.com/wp-content/uploads/2016/07/Whitepaper_Agile2016-single.pdf. [Online; accessed 02-December-2019].

Agrawal, Ajay, Nicola Lacetera, Elizabeth Lyons. 2013. Does information help or hinder job applicants from less developed countries in online markets? Tech. rep., National Bureau of Economic Research.

Agrawal, Rakesh, Ramakrishnan Srikant, et al. 1994. Fast algorithms for mining association rules. *Conference on Very Large Data Bases*, vol. 1215. 487–499.

Akerlof, George A. 1978. The market for lemons: Quality uncertainty and the market mechanism. *Uncertainty in Economics*. Elsevier, 235–251.

Allahbakhsh, Mohammad, Aleksandar Ignjatovic, Boualem Benatallah, Elisa Bertino, Norman Foo, et al. 2012. Reputation management in crowdsourcing systems. *International Conference on Collaborative Computing*. IEEE, 664–671.

Amazon. 2018. Customer reviews. https://www.amazon.com/gp/help/customer/display.html?nodeId=202094910. [Online; accessed 02-December-2019].

Amazon. 2020. Search by "avg. customer review". https://www.amazon.com/gp/help/customer/display.html?nodeId=201889520 [Online; accessed 02-May-2020].

Autor, David H. 2001. Wiring the labor market. *Journal of Economic Perspectives* **15** 25–40.

Autor, David H., Lawrence F. Katz, Alan B. Krueger. 1998. Computing inequality: Have computers changed the labor market? *The Quarterly Journal of Economics* **113** 1169–1213.

Ba2002 Ba, Sulin, Paul A. Pavlou. 2002. Evidence of the effect of trust building technology in electronic markets: Price premiums and buyer behavior. *MIS Quarterly* 243–268.

Baba, Yukino, Kei Kinoshita, Hisashi Kashima. 2016. Participation recommendation system for crowdsourcing contests. *Expert Systems with Applications* **58** 174–183.

Bakos, Yannis. 1997. Reducing buyer search costs: Implications for electronic marketplaces. *Management Science* **43** 1676–1692.

Balog, Krisztian, Maarten De Rijke. 2007. Determining expert profiles (with an application to expert finding). *International Joint Conference on Artificial intelligence*. Morgan Kaufmann Publishers Inc., 2657–2662.

Banker, Rajiv D., Iny Hwang. 2008. Importance of measures of past performance: Empirical evidence on quality of e-service providers. *Contemporary Accounting Research* **25** 307–337.

Billsus, Daniel, Michael J. Pazzani. 1998. Learning collaborative information filters. *International Conference on Machine Learning*, vol. 98. 46–54.

Bouguessa, Mohamed, Benoît Dumoulin, Shengrui Wang. 2008. Identifying authoritative actors in question-answering forums: The case of Yahoo! answers. *International Conference on Knowledge Discovery and Data Mining*. ACM, 866–874.

Breese, S. John, David Heckerman, Carl Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. *Conference on Uncertainty in Artificial Intelligence*. 43–52.

Brynjolfsson, Erik, Tom Mitchell, Daniel Rock. 2018. What can machines learn, and what does it mean for occupations and the economy? *AEA Papers and Proceedings*, vol. 108. 43–47.

Chen, Tianqi, Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. *International Conference on Knowledge Discovery & Data Mining*. ACM, 785–794.

Chevalier, Judith A, Dina Mayzlin. 2006. The effect of word of mouth on sales: Online book reviews. *Journal of Marketing Research* **43** 345–354.

Christoforaki, Maria, Panagiotis G. Ipeirotis. 2015. A system for scalable and reliable technical-skill testing in online labor markets. *Computer Networks* **90** 110–120.

Cui, Geng, Hon Kwong Lui, Xiaoning Guo. 2012. The effect of online consumer reviews on new product sales. *International Journal of Electronic Commerce* **17** 39–58.

Curtis, Nathan, Rei Safavi Naini, Willy Susilo. 2004. X2rep: Enhanced trust semantics for the xrep protocol. *International Conference on Applied Cryptography and Network Security*. Springer, 205–219.

Daltayanni, Maria, Luca de Alfaro, Panagiotis Papadimitriou. 2015. Workerrank: Using employer implicit judgements to infer worker reputation. *International Conference on Web Search and Data Mining*. ACM, 263–272.

Damiani, Ernesto, De Capitani di Vimercati, Stefano Paraboschi, Pierangela Samarati, Fabio Violante. 2002. A reputation-based approach for choosing reliable resources in peer-to-peer networks. *Conference on Computer and Communications Security*. ACM, 207–216.

Delgado, Joaquin, Naohiro Ishii. 1999. Memory-based weighted-majority prediction. *Special Interest Group on Information Retrieval Workshop on Recommender Systems*.

Dellarocas, Chrysanthos. 2003. The digitization of word of mouth: Promise and challenges of online feedback mechanisms. *Management Science* **49** 1407–1424.

Dellarocas, Chrysanthos. 2006. Reputation mechanisms. *Handbook on Economics and Information Systems* 629–660.

Devooght, Robin, Hugues Bersini. 2017. Collaborative filtering based on sequences. https://github.com/rdevooght/sequence-based-recommendations. [Online; accessed: 02-December-2019].

Duan2008 Duan, Wenjing, Bin Gu, Andrew B Whinston. 2008. Do online reviews matter? An empirical investigation of panel data. *Decision Support Systems* **45** 1007–1016.

eBay. 2018. Feedback forum. https://pages.ebay.com/services/forum/feedback.html. [Online; accessed 02-December-2019].

Einav, Liran, Chiara Farronato, Jonathan Levin. 2016. Peer-to-peer markets. *Annual Review of Economics* **8** 615–635.

Filippas, Apostolos, John J. Horton, Joseph Golden. 2018. Reputation inflation. *Conference on Economics and Computation*. ACM, 483–484.

Freelancers union, Upwork. 2017. Freelancing in america. https://s3-us-west-1.amazonaws.com/adquiro-content-prod/documents/Infographic-UP-URL-2040x1180.pdf. [Online; accessed: 02-December-2019].

Gandini, Alessandro, Ivana Pais, Davide Beraldo. 2016. Reputation and trust on online labour markets: The reputation economy of elance. *Work Organisation, Labour and Globalisation* **10** 27–43.

Ghose, Anindya, Panagiotis G. Ipeirotis. 2011. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *Transactions on Knowledge and Data Engineering* **23** 1498–1512.

Ghose, Anindya, Panagiotis G. Ipeirotis, Beibei Li. 2012. Designing ranking systems for hotels on travel search engines by mining user-generated and crowdsourced content. *Marketing Science* **31** 493–520.

Ghose, Anindya, Panagiotis G. Ipeirotis, Beibei Li. 2014. Examining the impact of ranking on consumer behavior and search engine revenue. *Management Science* **60** 1632–1654.

Goswami, Anjan, Fares Hedayati, Prasant Mohapatra. 2014. Recommendation systems for markets with two-sided preferences. *International Conference on Machine Learning and Applications*. 282–287.

Graham, Mark, Isis Hjorth, Vili Lehdonvirta. 2017. Digital labour and development: Impacts of global digital labour platforms and the gig economy on worker livelihoods. *Transfer: European Review of Labour and Research* **23** 135–162.

Hambleton, Ronald K, Hariharan Swaminathan, H Jane Rogers. 1991. *Fundamentals of item response theory*. Sage.

Hendrikx, Ferry, Kris Bubendorfer. 2013. Malleable access rights to establish and enable scientific collaboration. *International Conference on eScience*. IEEE, 334–341.

Hendrikx, Ferry, Kris Bubendorfer, Ryan Chard. 2015. Reputation systems: A survey and taxonomy. *Journal of Parallel and Distributed Computing* **75** 184–197.

Ho2017b Ho, Yi-Chun, Junjie Wu, Yong Tan. 2017. Disconfirmation effect on online rating behavior: A structural model. *Information Systems Research* **28** 626–642.

Hochreiter, Sepp, Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* **9** 1735–1780.

Horton, John J. 2017. The effects of algorithmic labor market recommendations: Evidence from a field experiment. *Journal of Labor Economics* **35** 345–385.

Hossain, Md Sabir, Mohammad Shamsul Arefin. 2019. Development of an intelligent job recommender system for freelancers using clients feedback classification and association rule mining techniques. *Journal of Software* **14** 312–339.

Hsueh, Sue-Chen, Ming-Yen Lin, Chien-Liang Chen. 2008. Mining negative sequential patterns for e-commerce recommendations. *Asia-Pacific Services Computing Conference*. 1213–1218.

Hu2017 Hu, Nan, Paul A. Pavlou, Jie Zhang. 2017. On self-selection biases in online product reviews. *MIS Quarterly* **41** 449–471.

Hu, Nan, Jie Zhang, Paul A. Pavlou. 2009. Overcoming the j-shaped distribution of product reviews. *Communications of the ACM* **52** 144–147.

Huber, Peter J. 2011. *Robust statistics*. Springer.

Institute of Business Value. 2019. The enterprise guide to closing the skills gap. https://www.ibm.com/downloads/cas/EPYMNBJA. [Online; accessed 02-December-2019].

Ipeirotis, Panagiotis. 2013. Badges and the lake wobegon effect. https://www.behind-the-enemy-lines.com/2013/10/badges-and-lake-wobegon-effect.html. [Online; accessed 30-December-2018].

Jagabathula, Srikanth, Lakshminarayanan Subramanian, Ashwin Venkataraman. 2014. Reputation-based worker filtering in crowdsourcing. *Advances in Neural Information Processing Systems*. 2492–2500.

Jain2018 Jain, Hemant, Balaji Padmanabhan, Paul A. Pavlou, Raghu T. Santanam. 2018. Call for papers–special issue of information systems research–humans, algorithms, and augmented intelligence: The future of work, organizations, and society. *Information Systems Research* **29** 250–251.

Jannach, Dietmar, Lukas Lerche, Michael Jugovac. 2015. Adaptation and evaluation of recommendations for short-term shopping goals. *Conference on Recommender Systems*. 211–218.

Jerath, Kinshuk, Peter S. Fader, G.S. Bruce Hardie. 2011. New perspectives on customer "death" using a generalization of the pareto/nbd model. *Marketing Science* **30** 866–880.

Jøsang, Audun, Jennifer Golbeck. 2009. Challenges for robust trust and reputation systems. *Proceedings of the 5th International Workshop on Security and Trust Management*. 52.

Jøsang, Audun, Guibing Guo, Maria Silvia Pini, Francesco Santini, Yue Xu. 2013. Combining recommender and reputation systems to produce better online advice. *International Conference on Modeling Decisions for Artificial Intelligence*. Springer, 126–138.

Joesang2007 Jøsang, Audun, Roslan Ismail, Colin Boyd. 2007. A survey of trust and reputation systems for online service provision. *Decision Support Systems* **43** 618–644.

Jurczyk, Pawel, Eugene Agichtein. 2007. Discovering authorities in question answer communities by using link analysis. *International Conference on Information and Knowledge Management*. ACM, 919–922.

Kamvar, Sepandar D, Mario T Schlosser, Hector Garcia-Molina. 2003. The eigentrust algorithm for reputation management in p2p networks. *International Conference on World Wide Web*. ACM, 640–651.

Kanat2018 Kanat, Irfan, Yili Hong, Santanam Raghu T. 2018. Surviving in global online labor markets for it services: A geo-economic analysis. *Information Systems Research* **29** 893–909.

Kantor, B. Paul, Lior Rokach, Francesco Ricci, Bracha Shapira. 2011. *Recommender systems handbook*. Springer.

Kendall, Maurice G. 1938. A new measure of rank correlation. *Biometrika* **30** 81–93.

Kohl, Matthias. 2019. Totalvardist: Generic function for the computation of the total variation distance of two distributions. https://www.rdocumentation.org/packages/distrEx/versions/2.5/topics/TotalVarDist. [Online; accessed 02-December-2019].

Kokkodis, Marios. 2018. Dynamic recommendations for sequential hiring decisions in online labor markets. *International Conference on Knowledge Discovery & Data Mining*. ACM, 453–461.

Kokkodis, Marios. 2020. Diversify or specialize? Demand reputation trade-offs and career paths in online labor markets. Working paper.

Kokkodis2014 Kokkodis, Marios, Panagiotis G. Ipeirotis. 2014. The utility of skills in online labor markets. *International Conference on Information Systems*.

Kokkodis, Marios, Panagiotis G. Ipeirotis. 2016. Reputation transferability in online labor markets. *Management Science* **62** 1687–1706.

Kokkodis, Marios, Panagiotis G. Ipeirotis. 2020. Demand-aware career path recommendations: A reinforcement learning approach. *Management Science* (forthcoming).

Kokkodis2020 Kokkodis, Marios, Theodoros Lappas. 2020. Your hometown matters: Popularity-difference bias in online reputation platforms. *Information Systems Research* **31** 412–430.

Kokkodis, Marios, Theodoros Lappas, Gerald Kane. 2020a. Direct and indirect effects of introducing purchase verification in e-commerce platforms. Working paper.

Kokkodis2020a Kokkodis, Marios, Theodoros Lappas, Sam Ransbotham. 2020b. From lurkers to workers: Predicting voluntary contribution and community welfare. *Information Systems Research* **31** 607–626.

Kokkodis, Marios, Panagiotis Papadimitriou, Panagiotis G. Ipeirotis. 2015. Hiring behavior models for online labor markets. *International Conference on Web Search and Data Mining*. 223–232.

Kokkodis, Marios, Sam Ransbotham. 2020. Asymmetric reputation spillover from agencies on digital platforms. (Working paper).

Koren, Yehuda. 2009. Collaborative filtering with temporal dynamics. *International Conference on Knowledge Discovery and Data Mining*. 447–456.

Koren, Yehuda, Robert Bell, Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* **42** 30–37.

Kuhn, Peter, Mikal Skuterud. 2004. Internet job search and unemployment durations. *American Economic Review* **94** 218–232.

Kula, Maciej. 2018. Deep recommender models using pytorch. https://github.com/maciejkula/spotlight. [Online; accessed: 02-December-2019].

Kung Hsiang, Huang. 2018. Introduction to recommender systems (neural network approach). https://towardsdatascience.com/introduction-to-recommender-system-part-2-adoption-of-neural-network-831972c4cbf7. [Online; accessed: 02-December-2019].

Le, Quoc, Tomas Mikolov. 2014. Distributed representations of sentences and documents. *International Conference on Machine Learning*. 1188–1196.

Lee2014b Lee, Gunwoong, Santanam Raghu T. 2014. Determinants of mobile apps' success: Evidence from the appstore market. *Journal of Management Information Systems* **31** 133–170.

Lemire, Daniel, Anna Maclachlan. 2005. Slope one predictors for online rating-based collaborative filtering. *International Conference on Data Mining*. SIAM, 471–475.

Lin, Mingfeng, Yong Liu, Siva Viswanathan. 2016. Effectiveness of reputation in contracting for customized production: Evidence from online labor markets. *Management Science* **64** 345–359.

Longadge, Rushi, Snehalata Dongre. 2013. Class imbalance problem in data mining review. *arXiv preprint* 1305.1707.

Lu2013 Lu, Xianghua, Sulin Ba, Lihua Huang, Yue Feng. 2013. Promotional marketing or word-of-mouth? Evidence from online restaurant reviews. *Information Systems Research* **24** 596–612.

Luca, Michael. 2016. Reviews, reputation, and revenue: The case of yelp.com. (Working Paper).

Luca, Michael. 2017. Designing online marketplaces: Trust and reputation mechanisms. *Innovation Policy and the Economy* **17** 77–93.

Luca, Michael, Georgios Zervas. 2016. Fake it till you make it: Reputation, competition, and Yelp review fraud. *Management Science* **62** 3412–3427.

Meyer, Frank. 2012. Recommender systems in industrial contexts. *arXiv preprint* 1203.4487.

Mikolov, Tomas, Kai Chen, Greg Corrado, Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint* 1301.3781.

Moe, Wendy W, David A Schweidel. 2012. Online product opinions: Incidence, evaluation, and evolution. *Marketing Science* **31** 372–386.

Moreno2014 Moreno, Antonio, Christian Terwiesch. 2014. Doing business with strangers: Reputation in online service marketplaces. *Information Systems Research* **25** 865–886.

Murphy, Kevin P. 2012. *Machine learning: A probabilistic perspective*. The MIT Press.

Nica, Elvira, Ana Mădălina Potcovaru, Cătălina Oana Mircică, et al. 2017. A question of trust: Cognitive capitalism, digital reputation economy, and online labor markets. *Economics, Management and Financial Markets* **12** 64.

Oliver, Beverley. 2015. Redefining graduate employability and work-integrated learning: Proposals for effective higher education in disrupted economies. *Journal of Teaching and Learning for Graduate Employability* **6** 56.

Pallais, Amanda. 2014. Inefficient hiring in entry-level labor markets. *American Economic Review* **104** 3565–3599.

Paolacci, Gabriele, Jesse Chandler, Panagiotis G. Ipeirotis. 2010. Running experiments on amazon mechanical turk. *Judgment and Decision making* **5** 411–419.

Patel, Bharat, Varun Kakuste, Magdalini Eirinaki. 2017. CaPaR: A career path recommendation framework. *International Conference on Big Data Computing Service and Applications*. 23–30.

Pavlou2004 Pavlou, Paul A., David Gefen. 2004. Building effective online marketplaces with institution-based trust. *Information Systems Research* **15** 37–59.

Pelechrinis, Konstantinos, Vladimir Zadorozhny, Velin Kounev, Vladimir Oleshchuk, Mohd Anwar, Yiling Lin. 2015. Automatic evaluation of information provider reliability and expertise. *World Wide Web* **18** 33–72.

Proserpio, Davide, Georgios Zervas. 2017. Online reputation management: Estimating the impact of management responses on consumer reviews. *Marketing Science* **36** 645–665.

Quadrana, Massimo, Paolo Cremonesi, Dietmar Jannach. 2018. Sequence-aware recommender systems. *ACM Computing Surveys* **51** 66:1–66:36.

Rahman2018 Rahman, Hatim. 2018a. Don't worship the stars: Ratings inflation in online labor markets. *International Conference on Information Systems*.

Rahman, Hatim A. 2018b. Reputational ploys: Reputation and ratings in online markets. *Academy of Management Proceedings*.

Rendle, Steffen, Christoph Freudenthaler, Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. *International Conference on World Wide Web*. 811–820.

Ricci, Francesco, Lior Rokach, Bracha Shapira. 2011. Introduction to recommender systems handbook. *Recommender systems handbook*. Springer, 1–35.

Sabater, Jordi, Carles Sierra. 2001a. REGRET: Reputation in gregarious societies. *International Conference on Autonomous Agents*. ACM, 194–195.

Sabater, Jordi, Carles Sierra. 2001b. Social regret, a reputation model based on social relations. *ACM SIGecom Exchanges* **3** 44–56.

Schmidt, Frank L, John E Hunter. 1983. Individual differences in productivity: An empirical test of estimates derived from studies of selection procedure utility. *Journal of Applied Psychology* **68** 407.

Smola, Alex J, Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and Computing* **14** 199–222.

Spearman, Charles. 1904. The proof and measurement of association between two things. *The American Journal of Psychology* **15** 72–101.

Stackoverflow. 2018. What is reputation? How do i earn (and lose) it? https://stackoverflow.com/help/whats-reputation. [Online; accessed 02-December-2019].

Stevenson, Betsey. 2009. The internet and job search. *Studies of Labor Market Intermediation*. University of Chicago Press, 67–86.

Su, Xiaoyuan, Taghi M. Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence* **2009** 19.

Sundararajan, Arun. 2016. *The sharing economy: The end of employment and the rise of crowd-based capitalism*. Mit Press.

Tadelis, Steven. 2016. Reputation and feedback systems in online platform markets. *Annual Review of Economics* **8** 321–340.

Tian, Chunqi, Baijian Yang. 2011. R2Trust, a reputation and risk based trust management framework for large scale, fully decentralized overlay networks. *Future Generation Computer Systems* **27** 1135–1141.

Tripp, M. Thomas, Yany Grégoire. 2011. When unhappy customers strike back on the internet. *MIT Sloan Management Review* **52** 37–44.

Turkopticon. 2018. Turkopticon. https://turkopticon.ucsd.edu/. [Online; accessed 02-December-2019].

Upwork. 2014. Online work report. https://web.archive.org/web/20180228011632/http://elance-odesk.com:80/online-work-report-global. [Online; accessed: 28-April-2019].

Upwork. 2018. Add certifications. https://support.upwork.com/hc/en-us/articles/215650138-Add-Certifications. [Online; accessed 02-December-2019].

Wood Doughty, Alex. 2018. The role of reputation systems in an online labor market. (Working paper).

Xie, Hong, John CS Lui, Don Towsley. 2015. Incentive and reputation mechanisms for online crowdsourcing systems. *International Symposium on Quality of Service*. IEEE, 207–212.

Xiong, Li, Ling Liu. 2004. Peertrust: Supporting reputation based trust for peer-to-peer electronic communities. *Transactions on Knowledge and Data Engineering* **16** 843–857.

Ye, Qiang, Rob Law, Bin Gu. 2009. The impact of online user reviews on hotel room sales. *International Journal of Hospitality Management* **28** 180–182.

Yoganarasimhan, Hema. 2013. The value of reputation in an online freelance marketplace. *Marketing Science* **32** 860–891.

Zang, Hao, Yue Xu, Yuefeng Li. 2010. Non redundant sequential association rule mining and application in recommender systems. *International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 3. 292–295.

Zervas, Georgios, Davide Proserpio, John Byers. 2015. A first look at online reputation on airbnb, where every stay is above average. (Working paper).

Zhang, Jun, Mark S Ackerman, Lada Adamic. 2007. Expertise networks in online communities: structure and algorithms. *International Conference on World Wide Web*. ACM, 221–230.

A. Transition functions

To model transitions the framework considers the following three functions:

$$g^d \in \{\text{ Multinomial Logit, Ordered logit, Constrained ordered logit }\}\ \forall d \in \mathcal{D}\,. \tag{12}$$

Application of the multinomial logit is straightforward. The transition probability from state s_k to s_l is given by the following:

$$Pr(s_l|s_k,\mathbf{\Gamma},\mathbf{Z}_{t-1}) = \frac{\exp(\boldsymbol{\gamma}_{kl}\mathbf{Z}_{t-1})}{\sum_{m\in K}\exp(\boldsymbol{\gamma}_{km}\mathbf{Z}_{t-1})} = \text{softmax}(\boldsymbol{\gamma}_{kl}\mathbf{Z}_{t-1})\,, \tag{13}$$

where I dropped the superscript d for simplicity. This model does not assume any order, so for $k>l$, state s_k might model higher or lower quality from state s_l.

To the contrary, the ordered logit formulation assumes that states are ordered in terms of increasing quality, such that state s_k has a larger constant term than s_l when $k>l$ (Ghose and Todri 2016, Ghose et al. 2017, Todri et al. 2020, Zucchini et al. 2017). The transition probability from state s_k to a state s_l is as follows:

$$Pr(s_l|s_k,\mathbf{\Gamma},\mathbf{Z}_{t-1}) = \begin{cases} \Pr(\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon \le \alpha^1) = \Lambda(\alpha^1-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}), & \text{if } l=1 \\ \Pr(\alpha^{l-1} < \boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon \le \alpha^l) = \Lambda(\alpha^l-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}) - \Lambda(\alpha^{l-1}-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}), & \text{if } 1<l<K \\ \Pr(\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon > \alpha^{K-1}) = 1-\Lambda(\alpha^{K-1}-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}) & \text{if } l=K \end{cases}. \tag{14}$$

In the previous equation, ε is the unobserved error term that I assume to follow a logistic distribution (i.e., $\varepsilon_i|\boldsymbol{Z}_{t-1} \sim Logistic(0,1)$), Λ is the logit function, and $\boldsymbol{\alpha} = [\alpha^1,\alpha^2,...,\alpha^{K-1}]'$ are state-specific thresholds (Wooldridge 2010).

The constrained ordered logit adjusts Equation 14 to not allow transitions to lower-quality states. Specifically:

$$Pr(s_l|s_k,\mathbf{\Gamma},\mathbf{Z}_{t-1}) = \begin{cases} 0, & \text{if } l<k \\ \Pr(\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon \le \alpha^1) = \Lambda(\alpha^1-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}), & \text{if } l=k=1 \\ \Pr(\alpha^{l-1} < \boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon \le \alpha^l) = \Lambda(\alpha^l-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}) - \Lambda(\alpha^{l-1}-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}), & \text{if } 1<k<l<K \\ \Pr(\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}+\varepsilon > \alpha^{K-1}) = 1-\Lambda(\alpha^{K-1}-\boldsymbol{\gamma}_k\boldsymbol{Z}_{t-1}) & \text{if } l=K \end{cases}. \tag{15}$$

B. HMM identification

Given the structure of the HMM I focus on estimating the parameter vectors $\boldsymbol{\Theta}^d, \boldsymbol{\Gamma}^d$. To do so, I maximize the conditional probability of the set of observations given the HMM. (For simplicity, in the following analysis I drop the superscript d. However, keep in mind that this estimation process happens independently for each dimension $d \in \mathcal{D}$.)

Let us assume that I have the following sequence of M observations for a given worker i:

$$\boldsymbol{Y}_i = Y_{i1}, Y_{i2}, \ldots, Y_{iM} \,. \tag{16}$$

These observations correspond to a sequence of input vectors:

$$\boldsymbol{X}_{1:M} = \boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_M \,. \tag{17}$$

Furthermore, let us assume that $\boldsymbol{Y}_i$ is the result of a sequence of latent states, $\boldsymbol{S}_i$:

$$\boldsymbol{S}_i = S_{i1}, S_{i2}, \ldots, S_{iM} \,, \tag{18}$$

where $S_{im} \in \mathcal{S}$. This sequence of states is affected by the sequence of historic vectors $\boldsymbol{Z}_{1:M-1}$:

$$\boldsymbol{Z}_{1:M-1} = \boldsymbol{Z}_1, \boldsymbol{Z}_2, \ldots, \boldsymbol{Z}_{M-1} \,. \tag{19}$$

Figure 10 shows these sequences along with their interactions. Based on the structure of the graph, I get the conditional likelihood of observing sequence $\boldsymbol{Y}_i$:

$$\Pr(\boldsymbol{Y}_i | \boldsymbol{S}_i; \boldsymbol{\Theta}, \boldsymbol{X}_{1:M}) = \prod_{t=1}^{M} \Pr(Y_{it} | S_{it}; \boldsymbol{\Theta}, \boldsymbol{X}_t) \,, \tag{20}$$

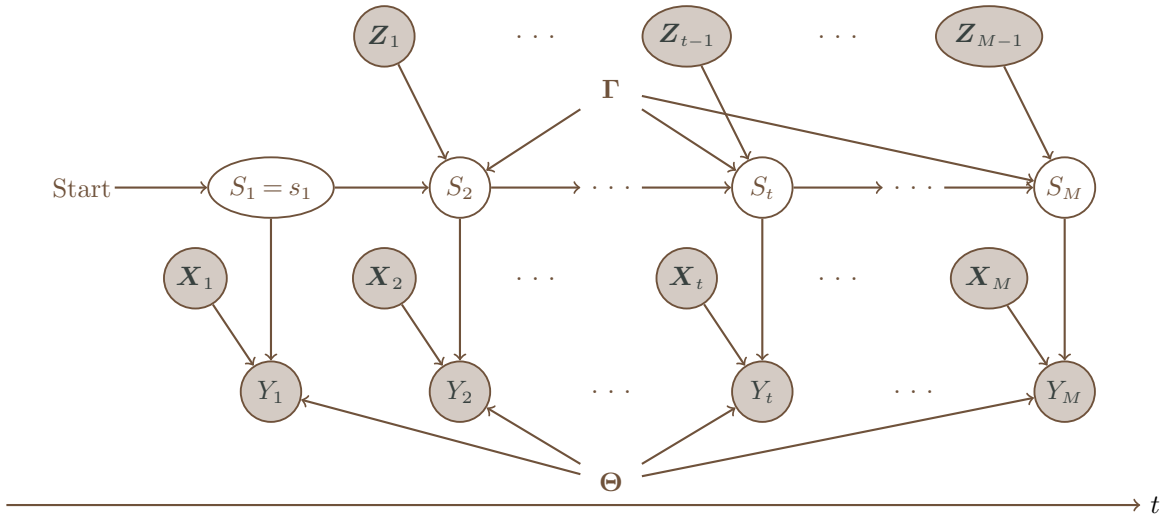
where Equation 6 estimates the right hand side. From Figure 10, the conditional probability of observing the sequence $\boldsymbol{S}_i$ is:

$$\Pr(\boldsymbol{S}_i | \boldsymbol{\Gamma}, \boldsymbol{Z}_{1:M-1}) = \boldsymbol{\pi}(S_1) \prod_{t=2}^{M} \Pr(S_{it} | S_{it-1}; \boldsymbol{\Gamma}, \boldsymbol{Z}_{t-1}) \,, \tag{21}$$

where $\boldsymbol{\pi}(S_1)$ is the the prior probability of being at state S_1. Equation 4 estimates these transition probabilities. Since the structure of the HMM imposes that every new worker lands in state s_1 ($\boldsymbol{\pi}(S_1 = s_1) = 1$), the previous equation becomes:

$$\Pr(\boldsymbol{S}_i | \boldsymbol{\Gamma}, \boldsymbol{Z}_{1:M-1}) = \prod_{t=2}^{M} \Pr(S_{it} | S_{it-1}; \boldsymbol{\Gamma}, \boldsymbol{Z}_{t-1}) \,. \tag{22}$$

Figure 10 Temporal evolution of the HMM



The structure of the latent state sequence $\boldsymbol{S}_i$, the observed sequence of outcomes $\boldsymbol{Y}_i$, the parameter vectors $\boldsymbol{\Theta}, \boldsymbol{\Gamma}$, and the sequences of input vectors $\boldsymbol{X}_{1:M}, \boldsymbol{Z}_{1:M-1}$ for a given worker i. For better readability I have dropped the worker subscript i and the competency superscript d. As with traditional probabilistic graphical models, latent states are in clear ellipses, and observed features in shaded ones (Koller and Friedman 2009).

Based on this analysis and the graph in Figure 10, the likelihood of this sequence of observations for worker i is as follows:

$$
\begin{aligned}
l(\boldsymbol{Y}_i; \boldsymbol{\Theta}, \boldsymbol{\Gamma}) &= \Pr(\boldsymbol{Y}_i | \boldsymbol{\Theta}, \boldsymbol{\Gamma}, \boldsymbol{X}_{1:M}, \boldsymbol{Z}_{1:M-1}) \\
&= \sum_{\forall \boldsymbol{S}_i} \Pr(\boldsymbol{Y}_i, \boldsymbol{S}_i | \boldsymbol{\Theta}, \boldsymbol{\Gamma}, \boldsymbol{X}_{1:M}, \boldsymbol{Z}_{1:M-1}) \\
&\overset{\text{Figure 10}}{=} \sum_{\forall \boldsymbol{S}_i} \Pr(\boldsymbol{Y}_i | \boldsymbol{S}_i; \boldsymbol{\Theta}, \boldsymbol{X}_{1:M}) \Pr(\boldsymbol{S}_i | \boldsymbol{\Gamma}, \boldsymbol{Z}_{1:M-1}) \\
&= Pr(Y_{i1} | S_{i1}; \boldsymbol{\Theta}, \boldsymbol{X}_1) \\
&\times \sum_{\forall \boldsymbol{S}_i} \prod_{t=2}^{M} \Pr(Y_{it} | S_{it}; \boldsymbol{\Theta}, \boldsymbol{X}_t) \\
&\times \Pr(S_{it} | S_{it-1}; \boldsymbol{\Gamma}, \boldsymbol{Z}_{t-1}),
\end{aligned} \tag{23}
$$

where I used the structure of the HMM to decompose the joint probability of $\Pr(\boldsymbol{Y}_i, \boldsymbol{S}_i | \boldsymbol{\Theta}, \boldsymbol{\Gamma}, \boldsymbol{X}_{1:M}, \boldsymbol{Z}_{1:M-1})$. Then, the complete likelihood for a dataset with N workers is:

$$
L(\boldsymbol{\Theta}, \boldsymbol{\Gamma}) = \prod_{i=1}^{N} l(\boldsymbol{Y_i}; \boldsymbol{\Theta}, \boldsymbol{\Gamma}) \,. \tag{24}
$$

Maximization of this likelihood estimates the parameters $\mathbf{\Theta}, \mathbf{\Gamma}$. I do this numerically through the limited memory Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm (Byrd et al. 1995). (In practice I minimize the negative log-likelihood.)

For more conservative reputation predictions, an option is to maximize this likelihood and then smooth the framework's estimate by adding the observed accumulated reputation:

$$P_{it}(\mathcal{R}) = \frac{1}{2}(P_{it}(\mathcal{R}) + \text{Current reputation}_{t-1}) \tag{25}$$

Finally, estimation of Equation 24 is highly parallelizable, as each individual likelihood l can run independently at every iteration.

C. Comparison of alternative design choices

The three components of the framework require hyper-parameter tuning. At the same time, the justification of each design choice presented in Section 3 requires comparison with alternative approaches. In this Appendix, I discuss such alternative modeling choices for each component of the proposed framework along with the process I follow to tune all the parameters of the focal approach. For the rest of the analysis, I split the data into ten folds that consist of different workers (i.e., each worker's complete history appears only in one of the ten folds), and I use 10-fold cross-validation to estimate the performance of each alternative design approach.

C.1. Alternative modeling choices for component A

Component A of the `HMM-W2V-framework` is required in order to map skillsets into a vector space of real numbers. To achieve this, in Section 3.1, I used a W2V approach. Alternative approaches can also achieve this mapping. For instance, document embedding (D2V, Le and Mikolov 2014) can directly map each skillset into numeric vectors. Even further, simpler clustering approaches can also perform this task. For instance, a Gaussian mixture model (GMM; see Murphy 2012) can provide membership weights for each skill to any number of predefined clusters. Finally, an alternative approach could also consider the raw text from job descriptions. To test this, I implement the

proposed W2V approach (Section 3.1) on the job-description text that includes the task's required skills (W2V-D). [5]

To compare these four alternative modeling choices for component A, I follow a grid search approach. Specifically, I estimate the 10-fold cross-validated ranking correlations (Spearman ρ) of each approach for the following parameter combinations:

$$\text{Component A grid-search: } \left\{ \overbrace{\{\text{W2V, D2V, GMM, W2V-D}\}}^{\texttt{Component A}} \times \overbrace{\{5, 10, 15\}}^{|\mathcal{D}|} \right\}. \tag{26}$$

Figure 11A shows the results. The $y-$axis shows improvement over the proposed W2V approach described in Section 3.1. The results show that W2V performs significantly ($p < 0.05$) better than the alternative approaches for the focal dataset. As a result, for the main analysis of this work, I use W2V. (For other contexts, one of the alternative approaches could be more appropriate. For instance, for the restaurant review dataset described in Appendix E, D2V worked better.)

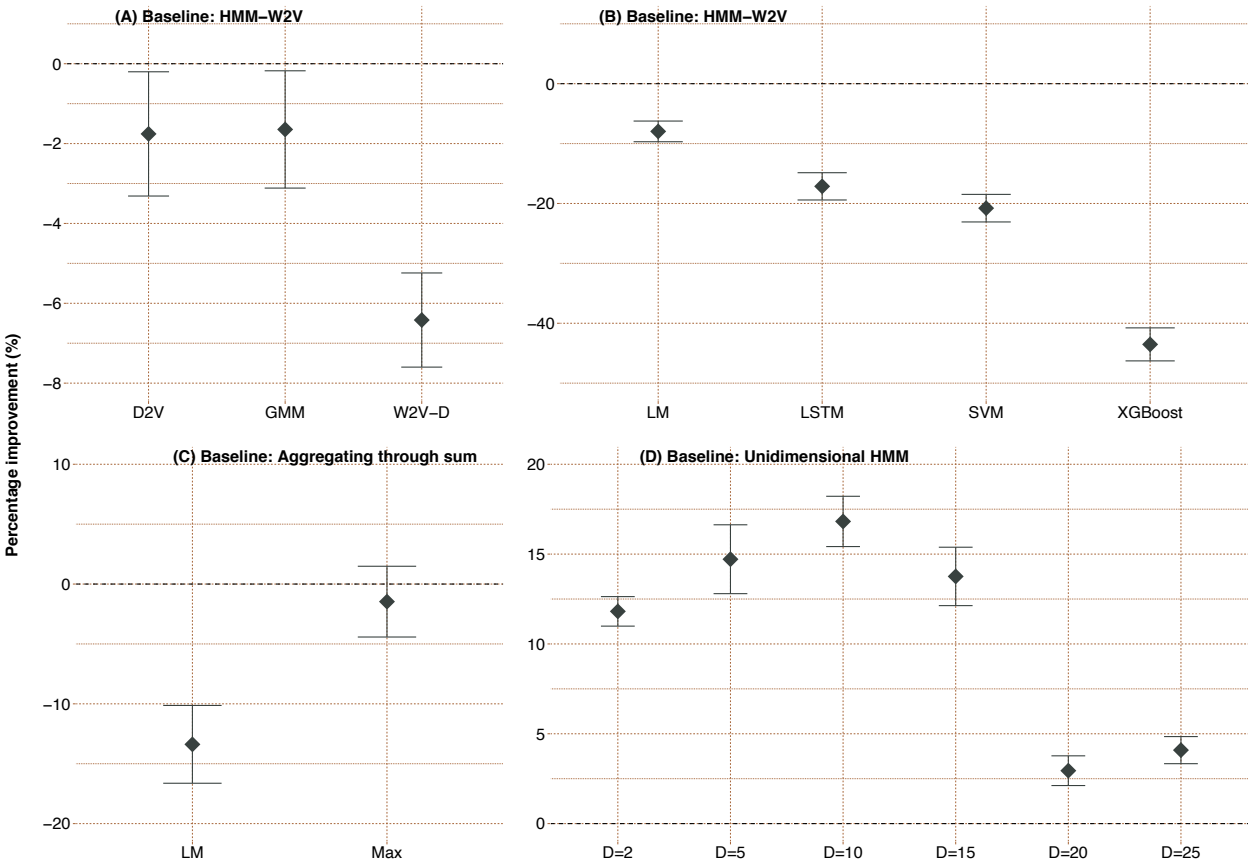
Figure 12 shows how W2V maps a randomly selected subset of skills into a reduced (through stochastic neighbor embedding, Hinton and Roweis 2003) two-dimensional space. The plot reveals hidden contextual similarities between skills. For instance, it shows that employers who request `C++` usually also request `SQLite`, while employers who request `Python` usually also request `MongoDB`. The plot hence suggests that `Python` is contextually closer to `MongoDB` than `SQLite`.

C.2. Alternative modeling choices for component B

For implementing component B of the `HMM-W2V-framework` that addresses reputation staticity, the main analysis uses an HMM. Other modeling approaches can also capture such a dynamic evolution of sequential observations. For instance, recurrent neural networks can capture such evaluations through Long Short Term Memory networks (Hochreiter and Schmidhuber 1997). In addition, I benchmark such dynamic approaches with simple (linear regression models—LM) and powerful regression approaches (SVM-regression and gradient boosting—XGBoost).

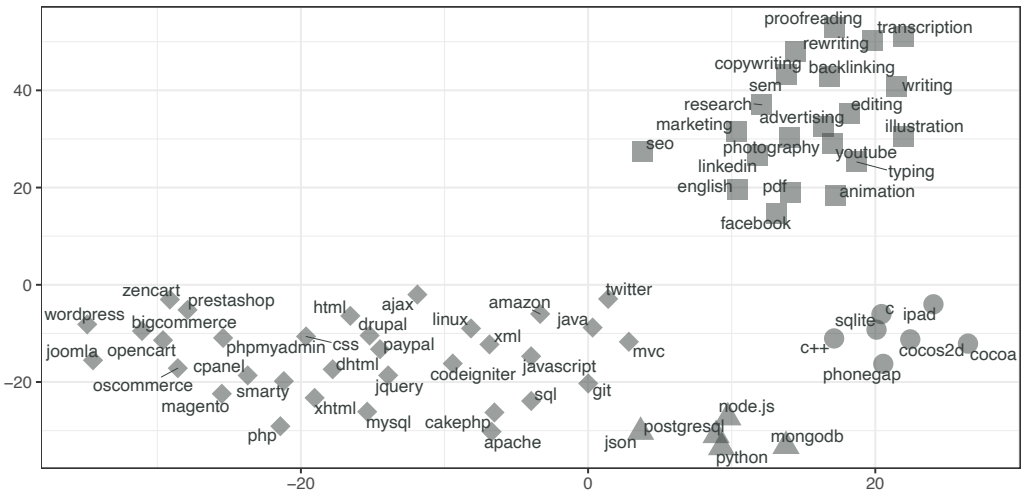
[5] I train these approaches on a separate dataset of 40,000 job-opening skillsets. I then use the pre-trained W2V, D2V and GMM models to decompose the skillsets of the focal dataset.

Figure 11 Design choices and tuning of the `HMM-W2V-framework`



The four figures compare alliterative modeling choices for each component of the `HMM-W2V-framework`. The $y-$axis shows a 10-fold cross-validated percentage improvement. W2V: Word2Vector. D2V: Doc2Vector. GMM: Gaussian mixture model. W2V-D: Word2Vector on job description text. Figure A shows that using W2V for component A of the `HMM-W2V-framework` outperforms ($p < 0.05$) D2V, GMM and W2V-D. Figure B shows that compared with alternative modeling approaches, using an HMM for component B of the `HMM-W2V-framework` better captures the dynamic nature of workers ($p < 0.001$). Figure C shows that a linear aggregation (for component C of the `HMM-W2V-framework`) performs worse than summing all dimensions, which performs on par with projecting the reputation of the dimension with the maximum weight. Figure D shows that $|\mathcal{D}| = 10$ dimensions better describe the focal dataset. In addition, it shows that configurations that include any number of dimensions (i.e., including component A of the `HMM-W2V-framework`) outperform a unidimensional HMM. Error bars represent 95% confidence intervals.

Figure 12 Skill decomposition through W2V



Visualization of mapped skills through W2V in a reduced, two-dimensional space through stochastic neighbor embedding (Hinton and Roweis 2003). Contextually similar skills appear close to each other (e.g., `C++, C, SQLite`), while contextually dissimilar skills appear far away from each other (e.g., `MongoDB` vs. `proofreading`).

To evaluate the performance of each alternative modeling of component B, I follow a grid search approach, and I estimate the 10-fold cross-validated ranking correlations (Spearman ρ) of each approach for the following parameter combinations:

$$\text{Component B grid-search: } \left\{ \overbrace{\{\text{HMM, LM, LSTM, SVM, XGBoost}\}}^{\texttt{Component B}} \times \overbrace{\{5, 10, 15\}}^{|\mathcal{D}|} \right\}. \quad (27)$$

Figure 11B shows the results. The $y-$ axis shows the percentage improvement of the HMM approach over the alternative approaches of modeling component B: It becomes clear that modeling dynamic transitions through a hidden Markov model performs significantly ($p < 0.001$) better than all alternative approaches for the focal dataset. One reason that could explain the underperformance of the LSTM approach, in particular, is the fact that the analyzed sequences are short (median completed tasks per worker is 5, Table 3). Perhaps, in larger datasets with longer sequences, LSTM could have been a more appropriate choice.

C.3. Alternative modeling choices for component C

Section 3.3 aggregates dimension-specific scores through a simple summation equation. Alternatively, an aggregation function could linearly regress each dimension (LM). Such a function could identify

dimensions that consistently provide erroneous estimates and underweight them. Another aggregation function could project the reputation score of the dimension with the maximum weight (Max). In theory, such an approach would rely on the dimension that is more relevant to the skillset at hand to make a reputation estimate.

To evaluate the performance of each alternative modeling choice of component C I follow a similar grid search approach to before, and estimate the 10-fold cross-validated ranking correlations (Spearman ρ) of each approach for the following parameter combinations:

$$\text{Component C grid-search: } \left\{ \overbrace{\{\text{Sum, LM, Max}\}}^{\texttt{Component C}} \times \overbrace{\{5, 10, 15\}}^{|\mathcal{D}|} \right\}. \tag{28}$$

Figure 11C shows the results. The $y-$axis shows the improvement of each approach compared with summing each dimension according to Equation 8. Linear modeling performs significantly ($p < 0.001$) worse compared with summing all dimensions. On the other hand, projecting the dimension with the maximum weight performs on average insignificantly ($p > 0.05$) worse than Equation 8. For the focal analysis, I choose Equation 8.

C.4. Choosing number of competency dimensions

The proposed framework requires as input the number of competency dimensions $|\mathcal{D}|$. To identify the best parameter $|\mathcal{D}|$, I use the best design choices for each component described above, and I estimate the 10-fold cross-validated ranking correlations for the following:

$$|\mathcal{D}| \text{ search: } \{1,2,5,10,15,20,25\}. \tag{29}$$

Figure 11D shows the results. The $y-$axis shows the improvement over a unidimensional HMM approach with $|\mathcal{D}| = 1$. For the uni-dimensional model, I remove component A, implicitly ignoring reputation attribution. For all $|\mathcal{D}| > 1$ the ranking correlation (Spearman ρ) is significantly ($p < 0.001$) higher than the unidimensional model. This suggests that component A itself provides a significant contribution to the performance of the `HMM-W2V-framework`. Furthermore, $|\mathcal{D}| = 10$ significantly outperforms ($p < 0.05$) $|\mathcal{D}| \in \{2, 15, 20, 25\}$. The performance of $|\mathcal{D}| = 5$ is, on average lower than $|\mathcal{D}| = 10$, but not statistically significant ($p > 0.05$). As a result, for the main analysis I use $|\mathcal{D}| = 10$.

Summary of comparisons: Overall, and based on this analysis, it becomes clear that each component of the framework contributes to the observed performance. Specifically, compared with alternative design choices, (1) using W2V for component A increases the framework's performance by up to 6%, (2) using an HMM in component B increases the framework's performance by up to 43% (Figure 11B), and (3) using Equation 8 to aggregate dimension-specific reputation estimates increases the performance of the framework by up to 13% (Figure 11C). Finally, including component A and using $|\mathcal{D}| = 10$ increases the performance of the framework by up to$\sim 17\%$ (Figure 11D).

C.5. HMM tuning

Each HMM considers various options for the transition function g^d, emission function f^d, and number of states K^d. As I mentioned earlier in Appendix A, the following continuous probability distributions can model function g^d:

$$g^d \in \{ \text{ Multinomial Logit, Ordered logit, Constrained ordered logit } \} \, \forall d \in \mathcal{D} \, . \tag{30}$$

Similarly, for the emission function f^d, the HMM considers the following set of probability distributions:

$$f^d \in \{ \text{ Beta, Truncated normal, Truncated exponential} \} \, \forall d \in \mathcal{D} \, . \tag{31}$$

Since emissions are bounded in [0,1], I truncate the normal and the exponential distribution. (The support of the Beta distribution is by default $\in [0, 1]$). Finally, I consider the following number of states:

$$K^d \in \{2, 3, ..., 6\}, \forall d \in \mathcal{D} \, . \tag{32}$$

To choose the most appropriate combination for the focal dataset, I estimate the configurations that yield the lowest 10-fold cross-validated Bayesian information criterion (BIC) scores (Schwarz 1978, Murphy 2012, Koller and Friedman 2009, Bishop 2006). Because the optimization process depends on the initialization of $\mathbf{\Theta}^d, \mathbf{\Gamma}^d$, it is prone to stuck in local maxima. To increase the

likelihood of reaching a potential global maximum, I conduct a search of 100 random initializations for each combination. Specifically, I perform the following grid-search:

$$K^d, f^d, g^d \text{ grid-search: } \Bigg\{ \overbrace{\{2,3,4,5,6\}}^{K^d} \times \overbrace{\{\text{Multinomial, Ordered logit, Constrained ordered logit}\}}^{g^d} \times \overbrace{\{\text{Beta, Truncated normal, Truncated exponential}\}}^{f^d} \times 100 \Bigg\}. \tag{33}$$

Based on the results of these searches, I pick:

- ◇ Number of states for each dimension, $K = [3,4,4,4,4,4,3,4,3,3]$.
- ◇ $f^d =$ Beta $\forall d \in \mathcal{D}$.
- ◇ $g^d =$ Multinomial logit $\forall d \in \mathcal{D}$.

D. Additional comparisons

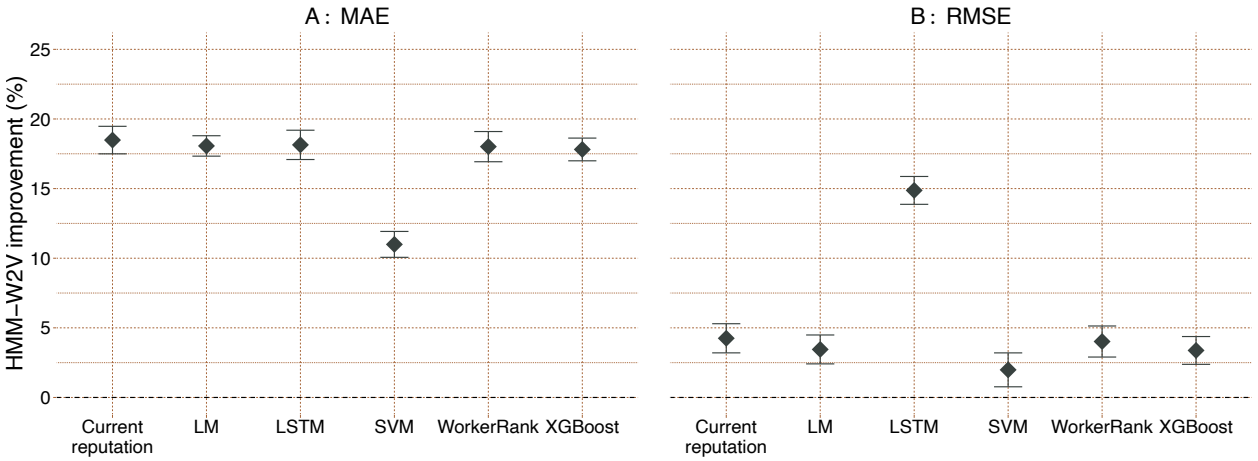
Section 5 illustrates the superiority of the `HMM-W2V-framework` in terms of ranking workers according to their likelihood of performing well, presenting accurate reputation distributions, and identifying "non-perfect" workers. Indeed, these evaluation metrics are the most relevant to benchmark a reputation system. Nevertheless, in this section I examine how the resulting HMM-W2V reputation compares to alternative reputation systems in terms of predictive performance and explanatory power.

Predictive performance: To test the predictive performance of the proposed approach, I estimate the 10-fold cross-validated mean absolute error (MAE) and the root mean squared error (RMSE) of each of the alternative reputation systems through the following linear model:

$$Y_t \sim \beta_0 + \beta Rep_t^j + \varepsilon, \tag{34}$$

where $Rep^j \in$ {HMM-W2V, Current feedback, LM, LSTM, SVM, WorkerRank, XGBoost}. Figures 13A and B show the improvement of `HMM-W2V-framework` in terms of MAE and RMSE respectively. HMM-W2V reputation is significantly ($p < 0.01$) more informative in predicting outcomes than alternative reputation approaches.

Figure 13 **Predictive performance of alternative reputation systems**



The $y-$axis shows the percentage improvement of the `HMM-W2V-framework` over the $x-$axis reputation systems in terms of MAE and RMSE. Error bars represent 95% confidence intervals.

Regression analysis: Regression analysis of multiple specifications can reveal the linear relationship of each reputation system with the observed outcomes. Table 5 shows the results of the regression analysis (Equation 34). The comparison shows that HMM-W2V reputation has higher explanatory power than the two baselines, as it yields greater R^2 than all alternative reputation systems.

Table 5 Explanatory power of alternative reputation systems

	DV: Observed performance						
	(A1)	(A2)	(A3)	(A4)	(A5)	(A6)	(A7)
Current reputation	0.025***						
	(0.00)						
LSTM		0.027***					
		(0.00)					
LM			0.039***				
			(0.00)				
SVM				0.009***			
				(0.00)			
WorkerRank					0.030***		
					(0.00)		
XGBoost						0.042***	
						(0.00)	
HMM-W2V							0.044***
							(0.00)
R^2	0.010	0.012	0.026	0.001	0.015	0.029	0.032

$^{***}p-$value < 0.001.

E. Generalizability: Application in online reputation platforms

The proposed approach, in theory, generalizes to other contexts that experience reputation inflation, reputation staticity, and reputation attribution. One such context is online reputation platforms (e.g., Yelp, TripAdvisor). These platforms experience reputation inflation (Luca 2016, Hu et al. 2017). At the same time, like contractors, venues (restaurants, hotels) in these platforms evolve. For instance, one venue might do a renovation, change menu offerings, hire a new chef, or change management. Hence, reputation staticity might also be present. Finally, because venues (like workers) are multidimensional entities, unidimensional ratings suggest the existence of reputation attribution.

It is important to highlight that some unique characteristics of online labor markets do not transfer to this alternative context. Specifically, in online labor markets, we are interested in estimating a skillset-specific reputation (Figure 1). The framework achieves this by observing skillset-specific evaluations for each worker. On the contrary, in online reputation platforms, there is no observed equivalent of skillsets. Reviewers rate their overall experience, and sometimes, they explain what they like and what they did not like in the review text. The proposed framework decomposes this text to latent dimensions, but the outcome of the framework is not "skillset-equivalent"-specific anymore. Nevertheless, the framework can be adjusted to this alternative context to provide current restaurant reputation scores. Table 6 summarizes the differences in objectives and data availability between the main and this alternative context.

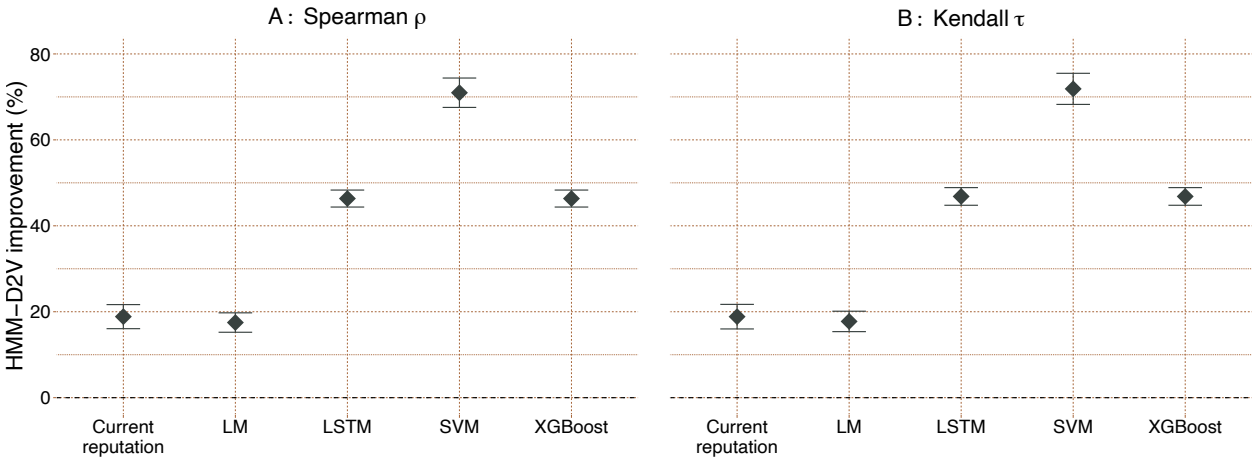
Table 6 Differences between the worker-reputation and the restaurant-reputation contexts

Context	Framework's objective	How it works
Online labor markets	Provide *current, skillset-specific* worker reputation	Decomposes observed skillset evaluations to latent dimensions. Builds dimension-specific HMMs.
Online reputation platforms	Provide *current* restaurant reputation	Decomposes review text into latent dimensions. Builds dimension-specific HMMs.

E.1. Comparison with alternative reputation systems

To test the proposed approach and alternative reputation systems in this different context, I analyze a set of 77,044 online reviews from a major restaurant review platform. As mentioned earlier in

Figure 14 **Ranking performance of alternative reputation systems on restaurant reputation**



The $y-$axis shows the percentage improvement of the `HMM-D2V` framework over the $x-$axis reputation system, in terms of Spearman ρ and Kendall τ. The `HMM-D2V` framework significantly ($p < 0.001$) outperforms all alternative reputation systems, with average 10-fold cross-validated improvements ranging between 20% and 70%. Error bars represent 95% confidence intervals.

Appendix C.1, D2V on review text was a better solution for implementing component A.[6] Hence, I use D2V, and allow latent restaurant dimensions of a venue to evolve as venues receive new ratings. Furthermore, I form vector $\mathbf{Z}_{t-1}$ by including the current reputation of each restaurant, the total number of reviews, and the days since its first review. Similarly, vector $\mathbf{X}_t$ includes the restaurant's current reputation.

Figure 14 shows the results. The $y-$axis shows the improvement of the proposed approach over alternative reputation systems, in terms of 10-fold cross-validated ranking correlations. The HMM-D2V framework significantly ($p < 0.001$) outperforms all alternative reputation systems, providing more accurate and current rankings of the listed venues.

[6] D2V likely works better in this context because review text is unstructured and unique. Conceptually, summing up weights of seemingly random words found in reviews through Equation 1 will generate noisy representations. On the contrary, summing up mappings of structured skill terms through W2V in the online labor market context should generate more informative representations, as each individual single skill contains crucial information.

E.2. Comparison with recommender systems

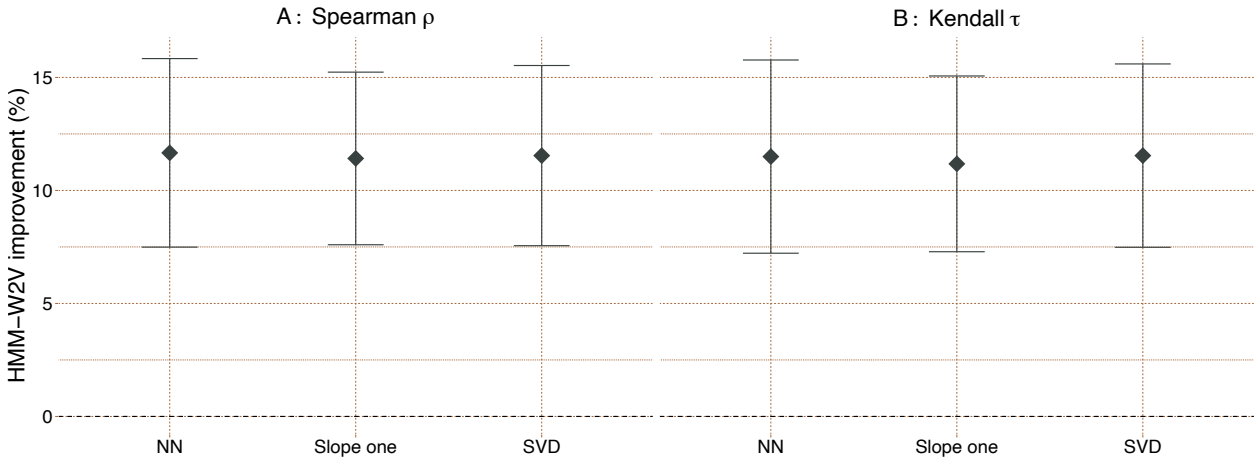
The restaurant-reputation context is closer to recommender systems than the focal worker-reputation context. Specifically, in this context, a mapping for recommender systems that could provide restaurant reputation is:

- Reviewer $\mapsto$ user.
- Restaurant rating $\mapsto$ rating.
- Restaurant $\mapsto$ item.

Compared with the main context, because in this context there is no objective restaurant equivalent of "skillset-specific" reputation (Table 6), restaurants map directly to items. In addition, reviewers rarely rate the same restaurant twice; hence, compared with online labor markets where workers get rated for the same skills repeatedly and I had to average their skillset-specific ratings (Table 2), restaurant ratings map directly to the recommender systems ratings. Finally, in online reputation platforms, it is reasonable to assume that reviewers are the recommender system's users, as every user's objective is to dine at a restaurant; instead, in online labor markets, it is not as reasonable to recommend workers to employers without considering the type of jobs and required skills that employers are looking for.

Given this more straightforward fit of recommender systems, I expect them to perform better compared with their performance in predicting worker-reputation scores (Section 5.4). Figure 15 shows the results. Indeed, explicit-feedback recommender systems perform better in this context than in online labor markets (Figure 8). Yet, and similar to online labor markets, the `HMM-W2V-framework` significantly ($p < 0.001$) outperforms these approaches, as it better captures restaurant evolution through the multi-dimensional HMM modeling (standard collaborative filtering approaches do not model item (restaurant) evolution). Finally, note that the proposed framework outperforms next-item recommenders (CNN) on average by 400% ($p < 0.001$), as the necessary encoding (Section 5.4.1) introduces significant noise in the prediction of restaurant reputation. I omit this comparison from Figure 8 for better presentation clarity.

Figure 15 Comparison with recommender systems



The `HMM-W2V-framework` significantly outperforms adaptations of recommender systems in this alternative context. As a result, it provides more accurate restaurant-reputation scores. The improvement of the `HMM-W2V-framework` over the deep learning recommender framework (CNN) is over 400%. I omit this point for presentation clarity. Error bars represent 95% confidence intervals.

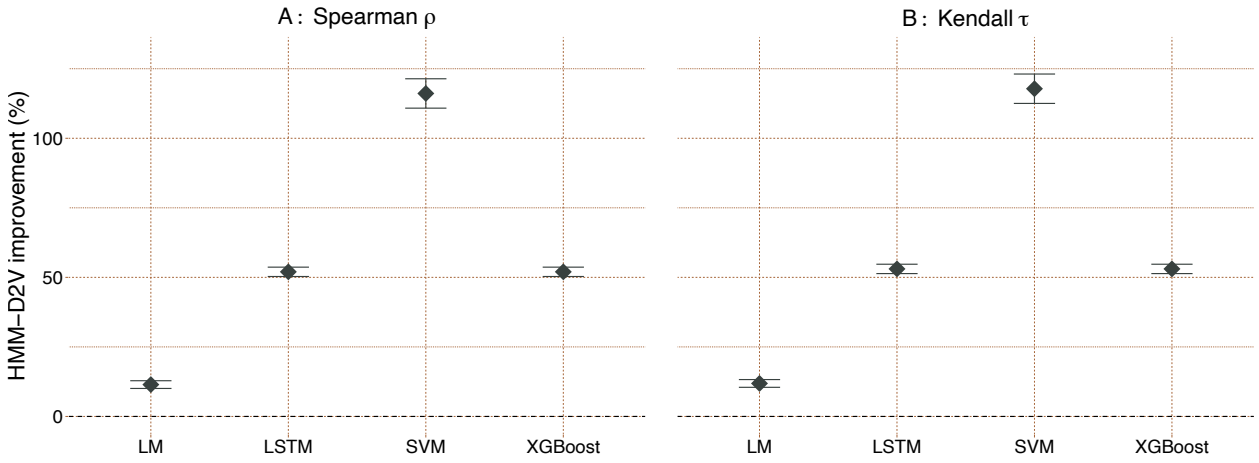
E.3. Comparison with human-rated dimensions

One concern of the proposed approach is that platforms are already developing multidimensional reputation systems, and as a result, the `HMM-W2V-framework` might be only recovering noisy information from such humanly rated dimensions. The focal restaurant review platform is an ideal context to test this, as it allows customers to rate venues across four dimensions: "Food," "Atmosphere," "Value," and "Service." Figure 16 compares the `HMM-W2V-framework` with alternative approaches that use these human-inputted multidimensional reputation scores as input. Specifically, these approaches estimate the following specification:

$$Y_t \sim G(\boldsymbol{Z}_{t-1}, \boldsymbol{X}_t, \bar{\text{Food}}_{t-1}, \bar{\text{Atmosphere}}_{t-1}, \bar{\text{Value}}_{t-1}, \bar{\text{Service}}_{t-1}) \,, \tag{35}$$

where $G \in \{\text{LM, LSTM, SVM, XGBoost}\}$ and $\bar{M}_{it-1}$ is the average reputation score of dimension M up to time $t-1$, $M \in \{$'Food", "Atmosphere," "Value," "Service"$\}$. The proposed approach significantly ($p < 0.001$) outperforms these alternative specifications. This suggests that the latent dimensions captured by the focal framework contain different information than the observed multidimensional feedback scores.

Figure 16 **Comparison with human-inputted multidimensional reputation**



The $y-$axis shows the percentage improvement of the `HMM-D2V` framework over the $x-$axis reputation system in terms of Spearman ρ and Kendall τ. Each one of the alternative reputation systems uses the four additional humanly-inputted reputation dimensions (Equation 35). The `HMM-D2V` framework significantly ($p < 0.001$) outperforms all alternative reputation systems, with average 10-fold cross-validated improvements ranging between 15% and 120%. Error bars represent 95% confidence intervals.

F. Parameter tuning

The alternative reputation systems discussed in Section 5 require hyperparameter tuning. This Appendix discusses the grid search approach I follow to tune the parameters of Gradient boosting and the dynamic LSTM network.

For Gradient boosting I use the Python package `xgboost`. I tune four hyperparameters: the number of trees to fit ("`n_estimators`"), the maximum tree depth ("`max_depth`"), the boosting learning rate ("`learning_rate`") and the subsample ratio of the training instance ("`subsample`"). I estimate the 10-fold cross-validated ranking correlation scores for the following combinations:

$$\text{XGBoost combinations:} \left\{ \overbrace{\{50, 100, 500\}}^{\texttt{n\_estimators}} \times \overbrace{\{3, 5, 10\}}^{\texttt{max\_depth}} \times \overbrace{\{0.8, 0.9, 1\}}^{\texttt{subsample}} \times \overbrace{\{0.001, 0.005, 0.01\}}^{\texttt{learning\_rate}} \right\}. \quad (36)$$

For building LSTM networks, I use the Python packages `keras.models.Sequential` and `keras.layers.LSTM`. I use adaptive learning rate optimization (Kingma and Ba 2014) to minimize the Mean Absolute Error (MAE) of the model. I tune three hyperparameters: the dimensionality of the output space ("`units`"), the number of "`epochs`" to train the model, and the number of samples

per gradient update "`batch_size`." I estimate the 10-fold cross-validated ranking correlation scores for the following combinations:

$$\text{LSTM combinations:} \left\{ \overbrace{\{20, 30, 40\}}^{\texttt{units}} \times \overbrace{\{10, 20, 30\}}^{\texttt{epochs}} \times \overbrace{\{32, 64, 128\}}^{\texttt{batch\_size}} \right\}. \tag{37}$$

Based on the resulting 10-fold cross-validated scores, I choose the following configurations for the main analysis:

- XGBoost: `learning_rate`: 0.01, `max_depth`: 2 , `n_estimators`: 100, `subsample`: 1.
- LSTM: `units`: 30, `epochs`: 20 , `batch_size`: 64.

G. Predictive features for job-applicant recommendations

To form vector $\mathbf{W}_p$ and build the job-applicant recommender systems described in Section 5.4.2, I use the same set of predictive variables as in Kokkodis et al. (2015). These are:

1. Years of experience: the self-reported job-applicant's experience (numeric).
2. Education: the self-reported education level of the job applicant (categorical).
3. Work-hours: the number of hours that the job-applicant has worked on the platform (numeric).
4. Rehire: whether or not the job-applicant has worked with the employer (numeric).
5. Current reputation: the accumulated reputation of the job applicant (numeric).
6. Certifications: the number of certification tests of the job applicant (numeric)
7. Bid: the bid price of the job applicant (numeric).
8. Completed jobs: the total number of the job-applicant's completed jobs (numeric).
9. Applicant-employer countries' PMI: the pairwise mutual information between the job-applicant's country and the employer's country (numeric; see Equation 38).
10. Certifications inner product: the inner product between the job-applicant certifications and the required skills by the opening (numeric).
11. Skills inner product: the inner product between the skillset of the job-applicant and the required skills by the opening (numeric).

Most of these variables are self-explained. The pairwise mutual information (PMI) of the job-applicant and employer countries is:

$$\text{Applicant-employer countries' PMI}(C_a, C_e) = \log \frac{\Pr(C_a, C_e)}{\Pr(C_a)\Pr(C_e)}\,, \tag{38}$$

where C_a is the country of the job-applicant, and C_e is the country of the employer.

H. Data-driven managerial insights

Section 3 describes the mapping process of skillsets to latent competency dimensions (W2V). The actual competencies' representations remain hidden, as the framework's primary goal is to provide worker-reputation scores. However, the decomposition of skillsets could provide interesting managerial insights. To illustrate, in this appendix, I examine each competency and try to extract market information that could guide managerial interventions.

H.1. Competency-specific skillsets

Which are the most representative skillsets in each competency? Recall that all skillsets decompose to all competencies (Equation 1). The skillsets that decompose to high weights within each competency are the ones that are more representative and allow workers to transition to higher competency-specific quality states. This is modeled explicitly in in the definition of the HMM observations (i.e., $Y_t^d = w_{\mathcal{R}}^d Y_t$). Hence, I can identify the skillset-specific workers' qualities that each competency captures by extracting the skillsets with the largest weights. Specifically, for each of the ten competencies I consider, the top five weighted skillsets include the following skills:

- Competency-0: {video-production, final-cut-pro, voice-over, video-postediting, video-editing, youtube-marketing}
- Competency-1: {brochure-design, print-design, print-layout-design, illustration}
- Competency-2: {manual-testing, software-testing, software-qa-testing, functional-testing}
- Competency-3: {google-adwords, google-analytics, google-adsense, ppc-advertising, sem}
- Competency-4: {android-app-development, 3d-rendering, unity-3d, rest, node.js, objective-c, angularjs}

- Competency-5: {data-scraping, google-spreadsheet, format-and-layout, data-entry, web-scraping, pdf-conversion, microsoft-excel}
- Competency-6: {portuguese, chinese, german, dutch, japanese, french, spanish, russian, italian}
- Competency-7: {business-analysis, internet-research, data-entry, market-research, business-plans}
- Competency-8: {translation-english-vietnamese, translation-english-italian, translation-english-portuguese, translation-english-korean, translation-english-malay }
- Competency-9: {medical-writing, writing, creative-writing, writing-slang-style, recipe-writing, article-writing, content-writing}

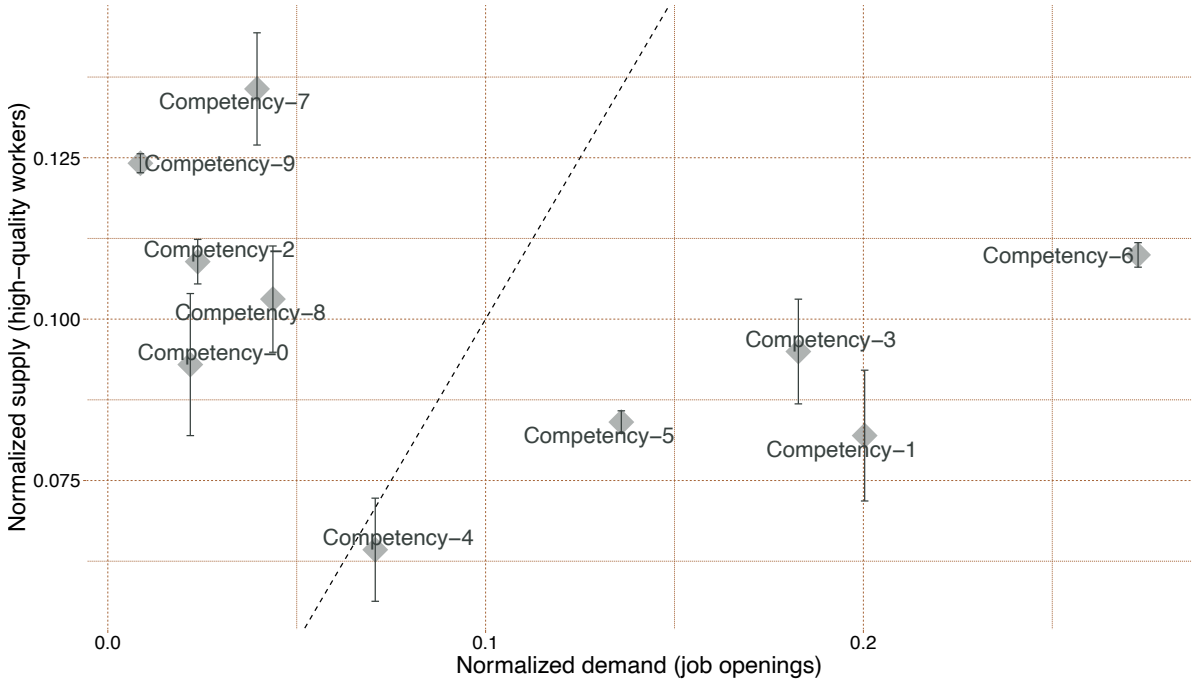
Through Equation 1, each skillset maps to all competencies. As a result, the framework implicitly identifies relationships (correlations) between the available competencies. For instance, when a worker receives a rating that primarily evaluates the worker's performance on a given skillset (e.g., video-production and final-cut-pro, competency-0), the rating maps to all competencies, allowing the framework to estimate an effect of this rating in seemingly uncorrelated competencies (e.g., competency-8). This unique characteristic of the proposed skillset-mapping approach demonstrates that it goes beyond any future human-rated dimensions, such as the ones discussed in Appendix E.3.

H.2. Competency-specific demand and supply

Once I know the skillset representation for each competency, I can identify competencies for which demand is higher than the supply of good workers. In particular, I first identify (through simulations) how many workers exist in the highest-quality state within each competency. This is a proxy of the supply of capable competency-specific workers. Then, I estimate the number of available job openings within each competency. This is a proxy of the competency-specific demand.

Figure 17 shows the scatterplot of the normalized competency-specific demand with the normalized competency-specific supply. The dashed line is the diagonal (slope = 45 degrees). Points below the diagonal represent high demand and low supply of good-quality workers. Points above the diagonal represent high supply of good workers and lower demand. Based on this analysis, the most

Figure 17 **Competency-specific demand and supply of high-quality workers**



The $y-$axis shows the normalized supply of high-quality workers (i.e., workers in the highest-quality state in each competency). The $x-$axis shows the normalized competency-specific demand (i.e., job openings). The dashed line is the diagonal (slope = 45 degrees). Error bars represent 95% confidence intervals of simulated worker paths.

in-demand competency that lacks high-quality workers is competency-6, which includes language-related skillsets. Platform managers can look into the origins of this demand-supply discrepancy and try to attract (new) workers with foreign language skills. On the other hand, competency-7 seems to provide an oversupply of high-quality workers with expertise in business analysis, and internet research. Hence, managers do not need to target workers in this competency at this moment.

To summarize, the `HMM-W2V-framework` allows managers to get deeper market insights through a formal latent-competency and hidden-state analysis, without the need to manually combine skillsets, identify thresholds of high-quality workers, and make assumptions of the current quality of each worker.

I. Discussion of worker competency-specific transitions

Figure 18A shows the cross-competency 10-fold cross-validated improvement of each alternative transition function over the constrained ordered logit function in terms of ranking correlations

(Spearman ρ). As mentioned in Appendix C.5 and is evident in Figure 18A, multinomial transitions perform significantly ($p < 0.001$) better than both ordered logit and constrained ordered logit transitions.

Of particular interest is the comparison between the constrained and unconstrained ordered logit functions. Figure 18A shows that an ordered logit without constrains performs slightly ($p < 0.05$) better. In other words, empirical evidence suggests that restricting transitions to higher-quality states hurts the performance of the model. This is in line with the expected behavior of the `HMM-W2V-framework`, as it continuously uses new evidence to update its quality estimates for each worker (Section 6.4).

To examine the frequency with which workers transition to lower-quality states, I simulate worker paths on the learned competency-specific HMMs and count the times that workers transition to a lower-quality state. Figure 18B shows that, on average, one out of five observed transitions is to a lower-quality state. This observation further highlights the need for the proposed framework to allow lower-state transitions in order to dynamically adjust and update its quality estimations for each worker.

Figure 18 **Constraining transitions to higher-quality states hurts the performance of the `HMM-W2V-framework`**

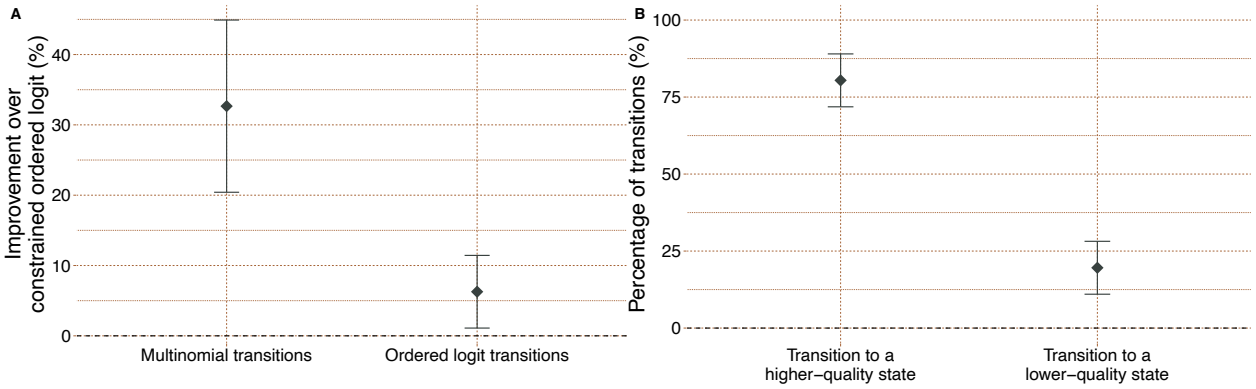


Figure A shows that multinomial and ordered logit transitions yield significantly ($p < 0.05$) better results than constrained ordered logit (Appendix A). Figure B shows that one out of five transitions across the competency-specific HMMs is to a lower-quality state. This highlights the need for the HMMs to adjust and correct their worker-quality estimates in the presence of new evidence. Error bars represent 95% confidence intervals.

J. Markov assumption

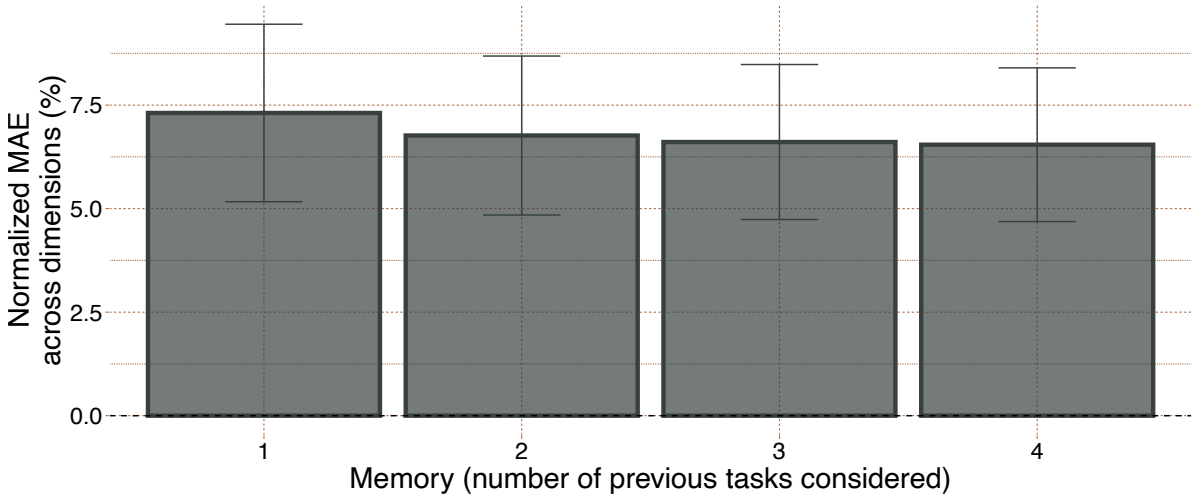
The HMM approach assumes that the next state of each worker depends on the current state of the worker and the set of observed characteristics $\boldsymbol{Z}_{it-1}, \boldsymbol{X}_t$. The presented results in Sections 5, 2.3 and Appendix D show that this assumption does not significantly hurt the performance of the framework.

There are multiple reasons that explain this. First, even though the Markovian assumption suggests that the next state depends on the current state, reaching a current state accumulates effects from the complete sequence of observations up to that point (Murphy 2012, Zucchini et al. 2017, Sahoo et al. 2012). In other words, the current state essentially encapsulates the likelihood of observing the sequence of states up to that point. Second, the fact that I consider up-to-date observed characteristics to affect both emissions and transitions controls for accumulated information up to that point for each worker (e.g., total number of jobs, accumulated feedback scores). Finally, given that each dimension-specific HMM captures dimension-specific reputation, it is reasonable to assume that the last update of a worker's performance in each dimension is likely the most current representation of the worker's dimension-specific quality.

To test this last argument, I estimate the predictive performance of varying memory windows for each dimensions $d \in \mathcal{D}$. Figure 19 shows the results. The $y-$axis shows the normalized MAE across all dimensions. Increasing the memory window to higher orders (2,3,4) appears to not have a significant effect on the predictive performance ($p > 0.05$).

Finally, even in scenarios where the Markovian assumption does not seem to fit the data, application of the Viterbi (Forney 1973) algorithm would likely solve this issue, as the Viterbi algorithm can estimate the current state of each user by maximizing the likelihood of the sequence of observations up to that point.

Figure 19 **Predictive performance of alternative memory windows**



There is no significant ($p > 0.05$) predictive improvement when considering more than one completed tasks. Error bars represent 95% confidence intervals.

References

Bishop, M. Christopher. 2006. *Pattern Recognition and Machine Learning*. Springer.

Byrd, H. Richard, Peihuang Lu, Jorge Nocedal, Ciyou Zhu. 1995. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing* **16** 1190–1208.

Forney, G. David. 1973. The viterbi algorithm. *IEEE* **61** 268–278.

Ghose, Anindya, Param Vir Singh, Vilma Todri. 2017. Got annoyed? examining the advertising effectiveness and annoyance dynamics. *International Conference on Information Systems*.

Ghose, Anindya, Vilma Todri. 2016. Towards a digital attribution model: Measuring the impact of display advertising on online consumer behavior. *MIS Quarterly* **40** 889–910.

Hinton, Geoffrey E., Sam T. Roweis. 2003. Stochastic neighbor embedding. *Advances in Neural Information Processing Systems*. 857–864.

Hochreiter, Sepp, Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* **9** 1735–1780.

Hu, Nan, Paul A. Pavlou, Jie Zhang. 2017. On self-selection biases in online product reviews. *MIS Quarterly* **41** 449–471.

Kingma, Diederik P, Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint* 1412.6980.

Kokkodis, Marios, Panagiotis Papadimitriou, Panagiotis G. Ipeirotis. 2015. Hiring behavior models for online labor markets. *International Conference on Web Search and Data Mining*. 223–232.

Koller, Daphne, Nir Friedman. 2009. *Probabilistic graphical models: Principles and techniques*. MIT press.

Le, Quoc, Tomas Mikolov. 2014. Distributed representations of sentences and documents. *International Conference on Machine Learning*. 1188–1196.

Luca, Michael. 2016. Reviews, reputation, and revenue: The case of yelp.com. (Working Paper).

Murphy, Kevin P. 2012. *Machine learning: A probabilistic perspective*. The MIT Press.

Sahoo, Nachiketa, Param Vir Singh, Tridas Mukhopadhyay. 2012. A hidden Markov model for collaborative filtering. *MIS Quarterly* **36** 1329–1356.

Schwarz, G. 1978. Estimating the dimension of a model. *The Annals of Statistics* **6** 461–464.

Todri, Vilma, Anindya Ghose, Param Vir Singh. 2020. Trade-offs in online advertising: Advertising effectiveness and annoyance dynamics across the purchase funnel. *Information Systems Research* **31** 102–125.

Wooldridge, M. Jeffrey. 2010. *Econometric analysis of cross section and panel data*. MIT Press.

Zucchini, Walter, Iain L. MacDonald, Roland Langrock. 2017. *Hidden Markov models for time series: An introduction using R*. CRC press.