NBER WORKING PAPER SERIES

TASKNET: BUILDING OCCUPATIONAL TASK INVENTORIES FROM WORKER INTERVIEWS AT SCALE

Jennifer Wang
Andreas Haupt
Isaac Robinson
Phil Lewis
Erik Brynjolfsson
Diyi Yang

Working Paper 35987
http://www.nber.org/papers/w35987

NATIONAL BUREAU OF ECONOMIC RESEARCH
1050 Massachusetts Avenue
Cambridge, MA 02138
November 2026

We would like to thank members of the Stanford Social and Language Technology Lab and the Stanford Digital Economy Lab for their helpful discussions and feedback. We are especially grateful to Tijana Zrnic, Bharat Chandar, David Autor, Luca Vendraminelli, Henri Beyer, Joachim Baumann, Vishakh Padmakumar, and Michael Ryan. This research is supported in part by the Laude Moonshot Seed Grant on "Collaborative Intelligence for the Future of Work", ONR grant N000142412532, Singapore Ministry of Digital Development and Information (to DY), and the Stanford Digital Economy Lab. JW is partially supported by NSF DGE-2146755. EB and AH received support from Schmidt Sciences. The views expressed herein are those of the authors and do not necessarily reflect the views of the National Bureau of Economic Research.

At least one co-author has disclosed additional relationships of potential relevance for this research. Further information is available online at http://www.nber.org/papers/w35987

NBER working papers are circulated for discussion and comment purposes. They have not been peer-reviewed or been subject to the review by the NBER Board of Directors that accompanies official NBER publications.

© 2026 by Jennifer Wang, Andreas Haupt, Isaac Robinson, Phil Lewis, Erik Brynjolfsson, and Diyi Yang. All rights reserved. Short sections of text, not to exceed two paragraphs, may be quoted without explicit permission provided that full credit, including © notice, is given to the source.

TaskNet: Building Occupational Task Inventories from Worker Interviews at Scale
Jennifer Wang, Andreas Haupt, Isaac Robinson, Phil Lewis, Erik Brynjolfsson, and Diyi Yang
NBER Working Paper No. 35987
November 2026
JEL No. J01, M50

ABSTRACT

Understanding how AI is reshaping work requires an accurate account of the tasks workers perform. Workers are often the first to see how their jobs are changing, as new responsibilities and tools often appear in everyday work before these changes reach formal descriptions. Interviews with workers can help capture these emerging changes in job tasks. Yet existing methods support standardized ratings across workers, while open-ended conversations with workers remain costly to collect and synthesize at scale. We introduce the Stanford Task Network (TaskNet), an occupational taxonomy grounded directly in workers' accounts of their everyday work. To construct TaskNet, we develop a mixed-initiative system that induces task inventories directly from workers. The system conducts interviews, standardizes reports into individual task lists, and aggregates these lists into occupational task inventories. Applied to six occupations spanning a range of education requirements and AI exposure, our system interviews 132 workers and produces 202 task statements from 2726 reports, all of which were accepted by O*NET Center analysts, compared to roughly 65%of workers' write-in task suggestions collected through the current O*NET data collection process, a major alternative source for emerging tasks. Every statement is validated by 318 incumbents in 7923 ratings of relevance and importance. Relative to the task data from the O*NET 31.0 database for the sampled occupations, TaskNet covers 75% of the 118 published O*NET tasks and adds 79 tasks with no O*NET counterpart and 58 that refine existing statements. It covers 89% of tasks reported by held-out workers, compared with 79% for O*NET, and a larger share of the activities described in employer job postings (72.5% vs. 70.2%). Downstream, TaskNet changes AI exposure estimates in both directions, raising them for Software Developers and Lawyers, two occupations where AI's effects are widely debated. Because such estimates guide workforce policy and training, more complete and up-to-date task lists can help direct these efforts toward the workers most likely to be affected. TaskNet provides a foundation for keeping occupational information current and grounded in workers' experiences as AI reshapes work.

Jennifer Wang
Stanford University
jennjwang@stanford.edu

Andreas Haupt
Stanford University
h4upt@stanford.edu

Isaac Robinson
Mercor
isaacrobinson@mercor.com

Phil Lewis
National Center for O*NET Development
phil.lewis@commerce.nc.gov

Erik Brynjolfsson
Stanford University
and NBER
erik.brynjolfsson@gmail.com

Diyi Yang
Stanford University
Computer Science
diyiy@stanford.edu

TaskNet: Building Occupational Task Inventories from Worker Interviews at Scale

Jennifer Wang[*,1] Andy Haupt[*,2] Isaac Robinson[3]
Phil Lewis[4] Erik Brynjolfsson[†,2,5] Diyi Yang[†,1]

[1]Stanford University, Department of Computer Science
[2]Stanford Institute for Human-Centered Artificial Intelligence
[3]Mercor Inc.
[4]National Center for O*NET Development
[5]National Bureau of Economic Research

[*]Equal contribution. [†]Equal supervision.

Abstract

Understanding how AI is reshaping work requires an accurate account of the tasks workers perform. Workers are often the first to see how their jobs are changing, as new responsibilities and tools often appear in everyday work before these changes reach formal descriptions. Interviews with workers can help capture these emerging changes in job tasks. Yet existing methods support standardized ratings across workers, while open-ended conversations with workers remain costly to collect and synthesize at scale. We introduce the Stanford Task Network (**TaskNet**), an occupational taxonomy grounded directly in workers' accounts of their everyday work. To construct TaskNet, we develop a mixed-initiative system that induces task inventories directly from workers. The system conducts interviews, standardizes reports into individual task lists, and aggregates these lists into occupational task inventories. Applied to six occupations spanning a range of education requirements and AI exposure, our system interviews 132 workers and produces 202 task statements from 2726 reports, all of which were accepted by O*NET Center analysts, compared to roughly 65 % of workers' write-in task suggestions collected through the current O*NET data collection process, a major alternative source for emerging tasks. Every statement is validated by 318 incumbents in 7923 ratings of relevance and importance. Relative to the task data from the O*NET 31.0 database for the sampled occupations, TaskNet covers 75% of the 118 published O*NET tasks and adds 79 tasks with no O*NET counterpart and 58 that refine existing statements. It covers 89 % of tasks reported by held-out workers, compared with 79 % for O*NET, and a larger share of the activities described in employer job postings (72.5% vs. 70.2%). Downstream, TaskNet changes AI exposure estimates in both directions, raising them for Software Developers and Lawyers, two occupations where AI's effects are widely debated. Because such estimates guide workforce policy and training, more complete and up-to-date task lists can help direct these efforts toward the workers most likely to be affected. TaskNet provides a foundation for keeping occupational information current and grounded in workers' experiences as AI reshapes work.

1 Introduction

Thirty years ago, the Occupational Information Network (O*NET) was created with an ambitious goal: to prepare workers and the institutions that train and employ them for a rapidly changing economy. Workers needed to understand how their skills translated to new opportunities, educators needed to know what capabilities to develop, and employers needed a common language for

describing the work they needed done. O*NET became the main source of occupational information in the U.S., providing systematic descriptions of more than 900 occupations, including the tasks workers perform and the knowledge and skills those tasks require (Peterson et al., 1999).

Today, O*NET's mission has taken on new importance. As AI systems expand their capabilities on economically relevant tasks, researchers and policymakers need occupational information that is both current and granular (Autor et al., 2024, Denain et al., 2026). Task inventories have long been used to study how technology changes work activities and, through them, labor demand (Acemoglu and Restrepo, 2019, Autor and Handel, 2013, Autor et al., 2003). Researchers now rely extensively on O*NET task statements to estimate AI exposure (Brynjolfsson et al., 2018, Eloundou et al., 2024, Felten et al., 2021, Putka et al., 2026, Webb, 2020), characterize AI use (Chatterji et al., 2025, Handa et al., 2025, Tomlinson et al., 2025), and study AI's effects on employment (Brynjolfsson et al., 2025a).

The quality of these analyses depends on how occupational tasks are collected. O*NET primarily gathers task information through structured surveys of workers and occupational experts. Incumbents receive a list of task statements, rate each for relevance, frequency, and importance, and may write in tasks the list omits (Dierdorff and Norton, 2011). Surveys provide standardized measurements of known tasks but are less suited to discovering work not already represented in the inventory. Interviews offer a complementary approach. Rather than asking workers to react to pre-written statements, interviews allow them to describe their work in their own words. Workers can explain how they perform their responsibilities in practice, identify activities that are new or changing, and elaborate on them through follow-up questions. This makes interviews especially valuable for surfacing work that existing task lists or employee-facing records may not capture.

Until recently, conducting and synthesizing interviews at scale was prohibitively costly. Language models now make both more feasible. AI interviewers can elicit detailed worker accounts through open-ended conversations and follow-ups. AI can also support concept discovery and aggregation to organize those accounts into a task taxonomy (Section 2). We bring these capabilities together in the Stanford Task Network (TaskNet) Interviewer, an adaptive interview system that induces occupational task inventories from workers' accounts of their everyday work. Turning these accounts into comprehensive, comparable task inventories raises two challenges. First, workers tend to describe their work through episodes and anecdotes rather than discrete tasks, so the interviewer must elicit task-level detail and cover the full range of responsibilities without leading the worker. The TaskNet Interviewer tracks coverage of a topic guide and infers a likely task inventory from the worker's role to probe tasks not yet mentioned. Second, interviews do not deliver the comparability that surveys get by design. Tasks reported by different workers must be combined into a single inventory, which means merging paraphrases, reconciling differences in granularity, and preserving infrequent but genuine tasks. Standard embedding-based clustering is poorly suited to this, because textual similarity does not track whether two reports describe the same work, nor at what level of detail they should count as one task. The TaskNet Interviewer aggregates tasks through an iterative procedure that adapts hierarchical task analysis and functional job analysis to language-model clustering. Each task statement remains traceable to the interview excerpts that support it.

We apply the TaskNet Interviewer to six occupations—Personal Care Aides, Software Developers, Elementary School Teachers, Except Special Education, Emergency Medicine Physicians, Dishwashers, and Lawyers—selected to span education and training requirements and AI exposure, with recently updated O*NET task data and clear occupational boundaries. From interviews with 132 workers (20–25 per occupation), we construct TaskNet, a taxonomy of 202 task statements synthesized from 2726 task reports. Trained O*NET Center analysts judge all 202 TaskNet statements valid under O*NET 's standards, making them eligible for review as *Emerging Tasks*. Another 318 incumbents (31–69

per occupation) provide 7923 ratings of task importance, time spent, verifiability, and variation in performance, enriching the inventory for analyses of task differences and AI exposure. Compared with O*NET, TaskNet identifies 79 new and 58 refined tasks while covering 75% of existing task statements (Section 4.3). It also covers more tasks reported by held-out workers (89.3% vs. 78.6%; higher in five of six occupations) and more activities in employer job postings (72.5% vs. 70.2%; higher in four of six; Section 5.2). These differences affect downstream AI exposure estimates (Putka et al., 2026). Under both the measures of Eloundou et al. (2024) and Tomei and Teeselink (2026), TaskNet raises exposure scores relative to O*NET for Lawyers, Software Developers, and Elementary School Teachers, but lowers them for Personal Care Aides. Much of this difference comes from newly identified coordination and oversight tasks. Although these tasks are highly exposed to AI, their reliance on human accountability and tacit knowledge suggests that such exposure may reflect complementarity as much as substitution. Finer-grained task descriptions also distinguish exposed from unexposed activities that O*NET combines into broader tasks. Thus, task inventories shape both the overall level of estimated AI exposure and its distribution within occupations.

Beyond these six occupations, TaskNet offers a repeatable approach to updating occupational task data from workers' accounts. Extending the method across occupations and repeating it over time could reveal emerging responsibilities and changes in established tasks, providing a worker-centered complement to existing occupational data. Our contributions are as follows:

1. We introduce the TaskNet Interviewer (Section 3.1), a mixed-initiative system that elicits tasks through semi-structured interviews with adaptive task probing, and verifies them through worker review. To our knowledge, we are the first to build occupational task inventories directly from workers' own accounts in AI-conducted interviews.
2. We present a task aggregation framework that derives an occupational task inventory from individual task lists (Section 3.2). The framework separates the definition of a task from the aggregation procedure, allowing task definitions to be revised while maintaining coverage and keeping tasks distinct.
3. We propose a framework for evaluating task inventories as representations of occupational work. Because there is no ground truth for task lists, we triangulate across stakeholders and independent sources to test whether the inventory captures the work workers do.
4. We release TaskNet, a worker-grounded taxonomy of 202 validated tasks across six occupations. Relative to O*NET 31.0, it adds 79 newly identified tasks and 58 refinements, achieving higher coverage and finer granularity.
5. We find that TaskNet produces different AI exposure estimates than O*NET, with the largest increases for Lawyers and Software Developers and a decrease for Personal Care Aides. These differences highlight how more comprehensive and detailed task inventories can sharpen our understanding of where and how AI may affect work.

2 Related Work

Task inventories Task lists have traditionally been built by job analysts through labor-intensive observation and analysis. Examples include the four editions of the Dictionary of Occupational Titles from 1939-1991 (Miller et al., 1980, Peterson et al., 1999, U.S. Department of Labor, Employment and Training Administration, 1991) and the U.S. Air Force task inventories (Christal, 1974, Christal and Weissmuller, 1988, Ramsey-Klee, 1984). These efforts grounded task lists in direct study of workers but were costly to sustain across many occupations. O*NET broadens the sources of task information,

drawing on surveys, write-in task statements, AI-assisted subject matter expert input, and other data sources (Advisory Panel for the Dictionary of Occupational Titles, 1993, Dierdorff and Norton, 2011, Klein et al., 2025, Lewis et al., 2025). Candidate tasks undergo review by occupational analysts before entering the inventory in a later survey cycle. TaskNet introduces AI-conducted interviews as a complementary source of task statements, producing statements that meet O*NET's standards and can enter its existing review process.

More recently, researchers have used language models to augment occupational data. Putka et al. (2023) predict the knowledge, skills, abilities, and other characteristics (KSAOs) associated with occupations, Lewis and Morris (2024) predict additional task information, and Lewis et al. (2025) generate candidate tasks for drone manufacturing. Klein et al. (2025) describe the use of language models to assist O*NET analysts in reviewing write-in task statements. In these applications, language models either generate occupational information from their own knowledge or help process statements that workers have already submitted. In TaskNet, workers provide the content of each task, while the model guides the conversation and structures what workers report.

Separately, researchers have applied natural language processing methods to extract occupational information from job postings (Howison et al., 2025), including skills (Atalay et al., 2020, Deming and Kahn, 2018). These approaches speed up the production of occupational data, but rely on model knowledge or employer-provided text rather than the workers who perform the tasks.

TaskNet brings these strands together. Like the computational approaches, it uses language models to make the production of occupational data faster and cheaper. Like O*NET, it takes a human-centered approach, grounding task lists in incumbents' own accounts.

LLM-Based Interviewing Recent advances in language models have enabled flexible systems for conducting semi-structured or adaptive interviews, ranging from single-LLM approaches guided by prompting (Geiecke and Jaravel, 2024, Park et al., 2026, Wuttke et al., 2025) to multi-agent systems (Anugraha et al., 2026, Liu and Yu, 2025, Talaei et al., 2025). Many applications of LLM-based interviews focus on qualitative accounts of respondents' attitudes, experiences, and perspectives (Cuevas et al., 2024, Wong et al., 2025, Wuttke et al., 2025), with more recent work extending to professionals' descriptions of their work (Anthropic, 2025). We build on this literature to elicit occupational tasks, drawing on mixed-initiative interaction (Horvitz, 1999, Ziems et al., 2025) and conversational interviewing (Schober and Conrad, 1997).

AI-Assisted Qualitative Analysis Prior work uses language models to organize and synthesize open-ended text, including developing qualitative codebooks (Dai et al., 2023), inducing interpretable concepts (Lam et al., 2024), and identifying latent topics (Pham et al., 2024). Related methods extract recurring patterns from large conversational datasets (Tamkin et al., 2024) and cluster semantically related observations (Viswanathan et al., 2024, Zhang et al., 2023).

3 System Design

The TaskNet Interviewer builds an occupational task inventory in three stages. *Discovery* interviews individual workers and produces and verifies a task list for each worker (Section 3.1). *Aggregation* consolidates these lists into a single candidate inventory for the occupation (Section 3.2). *Validation* brings in job analysts to review the inventory and workers to rate its tasks (Section 3.3).

We designed this system around three guiding principles. First, *workers are the authority on their own work*. Every task in the inventory originates in an activity reported by workers during discovery. Second, *task statements meet the standard that the O*NET Center applies in its own emerging tasks process*. This allows newly discovered tasks to enter the same adoption process as existing O*NET tasks and preserves compatibility with the broader Occupational Information Network. Third, *the inventory must support analyses that aggregate over tasks*. Such analyses require task statements to be consistent in granularity, collectively exhaustive of the work workers reported, and sufficiently distinct to prevent the same work from being counted twice.

3.1 Task Discovery

Task discovery aims to create a comprehensive task list for each worker. The challenge with creating exhaustive task lists is that a worker's first account of a job is rarely exhaustive (Morgeson and Campion, 1997). Workers are more likely to report recent or salient tasks while overlooking activities that are routine, infrequent, or taken for granted. Discovery therefore cannot simply record what workers volunteer. We therefore treat task discovery as an **active elicitation** problem. To maximize coverage of the worker's task set, we apply a mixed-initiative design in which the interviewer helps the worker recall and elaborate on their responsibilities, a model constructs a candidate task list from the conversation and proposes likely missing tasks, and the worker confirms or revises the result (see Figure 1). Building on SparkMe (Anugraha et al., 2026), our task induction system introduces three design innovations to a multi-agent framework for adaptive semi-structured interviews that tracks topic-guide coverage, incorporates emerging themes, and manages interview length.

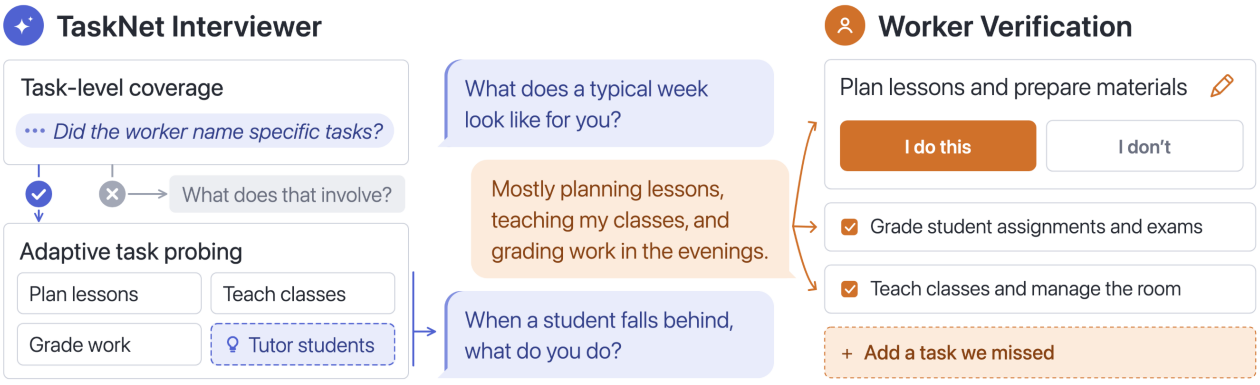


Figure 1: **The TaskNet Interviewer elicits occupational tasks through a mixed-initiative process:** The interviewer (purple) conducts a semi-structured interview covering the core aspects of the worker's role. After each answer, it checks whether the worker has named concrete tasks and follows up. It adaptively probes for tasks by comparing the tasks extracted so far with those likely for the role. The worker (orange) then verifies the resulting task list: they confirm, revise, or reject tasks the interviewer extracted and inferred, and add any tasks it missed.

First, we define topic coverage by tasks. Typically, in semi-structured interviews, a topic may be considered complete once the participant has addressed it. For task induction, this standard is insufficient: a participant can respond to every topic in broad terms (e.g., "I see patients") while yielding few concrete tasks. We therefore define coverage at the level of the task. A topic is complete only once the participant's response targets the granularity of an O*NET task statement. Each coverage criterion evaluates the response in terms of the tasks it specifies. When a participant names

a responsibility, tool, or stakeholder without describing the tasks it entails, the interviewer follows up to develop that description into concrete tasks.

Second, we search for uncovered tasks through adaptive task probing. An interviewer can adapt to themes that emerge in conversation, but it cannot pursue tasks that never surface in the first place. Adaptive task probing addresses this gap by using knowledge of the worker's role to identify plausible omissions. Throughout the interview, the system maintains a running list of reported tasks. Separately, it infers a candidate task set for the worker's described role. This inventory draws on the model's prior knowledge of the role, conditioned on the conversation. The interviewer then compares these two lists and turns the most relevant gaps into open-ended follow-up questions.

Finally, we present the induced task list for worker verification. Because recognizing a task is generally easier than recalling it (Mandler, 1980), job analysis has long relied on inventories that incumbents review rather than requiring them to generate complete lists from scratch (Christal, 1974). Our approach preserves that advantage while personalizing the inventory to each worker. The system consolidates the reported tasks at O*NET granularity and proposes additional tasks implied by the interview. Across the 132 interviews that contribute to our six task inventories, workers reviewed 2798 proposed tasks, of which they confirmed $94.2\,\%$ unchanged, revised $5.3\,\%$, deleted $0.5\,\%$, and added 163 of their own.

3.2 Task Aggregation

Interviews provide rich accounts of work, but workers use different vocabulary and levels of detail, making their reports difficult to compare. Similar wording can describe distinct tasks, such as preparing meals and feeding clients, while different wording can describe the same task, such as recording vitals in an EHR and writing blood pressure in a care log. Aggregation must therefore resolve both *task identity*—whether reports describe the same work—and *task granularity*—the level of detail at which that work should be represented. We introduce a framework that makes these boundary decisions explicit and produces standardized task inventories. O*NET's guidelines specify how to formulate task statements but leave task boundaries largely to analyst judgment. Our framework separates the definition of a task from the aggregation procedure, allowing researchers to inspect boundary decisions and adapt the definition to different measurement needs.

Our definition combines hierarchical task analysis (HTA) and functional job analysis (FJA). HTA organizes work into goals and the actions used to achieve them, providing a basis for identifying reports of the same task (Annett and Duncan, 1967, Stanton, 2006). Because goals nest, however, HTA alone does not establish a consistent level of granularity (Stanton, 2006). FJA helps establish that level by distinguishing workers' actions from their tools, materials, and settings (Fine and Cronshaw, 1999, Fine and Wiley, 1971). Together, these principles define *a task as the highest node in a goal hierarchy whose subtasks differ only in context*. Thus, recording vitals in different systems constitutes one task, while preparing meals and feeding clients remain distinct. Unlike similarity-based clustering, this criterion explicitly distinguishes differences in action from differences in context (Steck et al., 2024, Zhou et al., 2022).

We apply this definition through four iterative steps: *clustering, synthesis, overlap resolution, and coverage checking*. First, we cluster worker reports by task identity and granularity, retaining only clusters reported by at least two workers, following O*NET's emerging-tasks rule (Klein et al., 2025). Second, a language model synthesizes each cluster into a task statement using O*NET's guidelines (National Center for O*NET Development, 2009) and writing principles developed with feedback from O*NET Center staff. Each statement generalizes only as far as its source reports support and

is split if it combines distinct tasks. Third, an LLM judge checks pairs of statements for overlap: duplicate or nested statements are merged, and partially overlapping statements are reworded to separate their scopes. Finally, an LLM judge checks whether the revised statements still cover their source reports. Uncovered and unclustered reports are reassigned where possible; the remainder enter another round, continuing while new shared tasks emerge. We then manually compare each statement with its source reports and correct omissions or distortions. These steps preserve distinct, measurable tasks while recovering coverage that may be lost during synthesis and revision.

3.3 Task Validation

We use a multi-stakeholder validation procedure that pairs O*NET analyst reviews with worker ratings. Job analysts assess task-writing quality and occupational fit, while workers assess whether the tasks reflect their jobs. These two assessments evaluate adherence to O*NET's standards and fidelity to workers' accounts in line with our guiding principles.

Two trained job analysts at the O*NET Center jointly reviewed every statement using O*NET's task-review procedure (Lewis et al., 2025). They reached consensus on three questions: does the statement describe a task, does it meet O*NET's writing guidelines (National Center for O*NET Development, 2009), and is the work primarily performed by the target occupation? Their judgments distinguish valid occupational tasks from *adjacent* tasks—valid work that would not belong in the target occupation's task list. We revised flagged statements in consultation with the analysts.

Workers assessed each statement on O*NET's dimensions of *relevance*, *importance*, and *frequency* (National Center for O*NET Development, 2026b). We retain tasks rated relevant by at least a quarter of incumbents. Among retained tasks, those with relevance of at least 67 % and mean importance of at least 3 are classified as *core*, and the remainder are *supplemental*. Workers also rated five additional, established properties relevant to AI capabilities, including duration (Kwa and Cheng, 2025, Mertens et al., 2026), expertise requirements (Autor and Thompson, 2025), and verifiability (Polanyi, 1966) (Section E.1; Section K). Each worker rated a subset of statements to limit survey fatigue, with adaptive assignment balancing the number of ratings across tasks (Jeong et al., 2023).

4 Task Inventories for Six Occupations

To test the TaskNet Interviewer, we recruited workers from six occupations in the O*NET-SOC Taxonomy: Personal Care Aides (31-1122.00), Software Developers (15-1252.00), Elementary School Teachers, Except Special Education (25-2021.00), Emergency Medicine Physicians (29-1214.00), Dishwashers (35-9021.00), and Lawyers (23-1011.00)[1]. We selected these occupations based on three criteria: the recency of their O*NET task information (maximum of three years old with majority updated in 2025), their diversity in job zone[2] and AI exposure (see Section 7), and low ambiguity of their occupational titles. The last criterion avoids, for example, titles such as Accountants and Auditors (13-2011.00), for which respondents may conflate distinct specializations.

[1] We write Elementary School Teachers as shorthand, and drop the codes hereafter.

[2] An O*NET Job Zone is a group of occupations that share similar levels of required education, related work experience, and on-the-job training.

Table 1: **TaskNet organizes content from worker interviews into 202 task statements, covering 89.3% of reported activities.** Of the retained statements, 89 % are core tasks. The table summarizes inventory composition, source interviews (those used to construct the inventories), and coverage of their task reports by occupation. Supp. denotes supplemental tasks.

	TaskNet Inventory				Source Interviews		
Occupation	Tasks	Core	Supp.	Core Share (%)	Workers	Task Reports	Coverage (%)
Software Developers	44	40	4	91	21	468	87.4
Lawyers	32	28	4	88	22	529	83.6
Elementary School Teachers	39	37	2	95	24	562	95.6
Emergency Medicine Physicians	38	36	2	95	25	556	91.5
Personal Care Aides	36	30	6	83	20	344	94.2
Dishwashers	13	9	4	69	20	267	79.4
Total	202	180	22	89	132	2 726	89.3

4.1 Respondent Sample

We recruited workers through Mercor Inc., an AI human data company. Workers reside and work in the 50 U.S. states and the District of Columbia, excluding California. Individuals living and working in Puerto Rico and other U.S. territories are not included. Most participants came from Mercor's existing platform population, while Dishwashers were also recruited externally through job-posting sites. A language model first identified likely occupational matches from candidates' work histories and O*NET occupation information, and Mercor staff then reviewed each match. Candidates who passed this review completed an interest form reporting their occupation, years of experience, and location. We required at least two years of relevant experience for Personal Care Aides and Dishwashers, and at least four years for the other four occupations.

Recruitment prioritized occupational fit and experience over representativeness. The sample is not stratified to match the U.S. workforce in the selected occupations (see Section C for sample characteristics). Workers were paid a minimum of $80 per hour, or at a higher market rate for their occupation. Participation on the platform was voluntary. This research was approved under IRB protocol 84069.

Across six occupations, the TaskNet Interviewer conducted discovery interviews with 132 workers (Section 3.1) and collected task validations from 318 incumbents (Section 3.3), with modest demands on participants' time. Discovery interviews lasted a median of 29 minutes (IQR: 19–46 minutes)[3], with the shortest median durations among Dishwashers and Personal Care Aides (23 minutes) and the longest among Software Developers (43 minutes; Section C). Task validation took a comparable median of 28 minutes (IQR: 19–49 minutes), ranging from 16 minutes among Dishwashers to 40 minutes among Software Developers.

4.2 TaskNet Inventory

Across the six occupations, TaskNet contains 202 task statements derived from interviews with 132 workers (Table 1; full lists in Section F). Inventories range from 13 tasks for Dishwashers to 44 for Software Developers, with Elementary School Teachers (39), Emergency Medicine Physicians

[3]Durations include breaks, which inflate mean completion times, so we report medians.

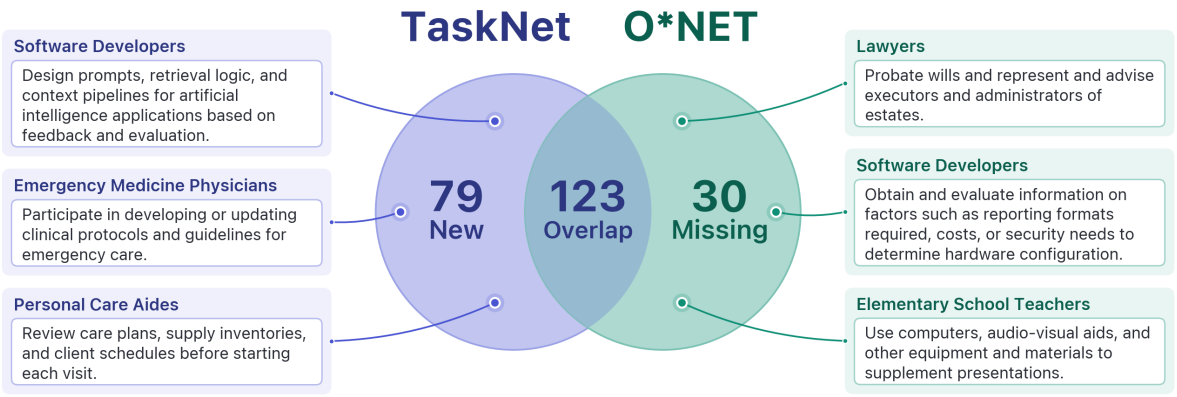


Figure 2: TaskNet recovers many existing O*NET tasks while expanding coverage to additional activities. Across six occupations, TaskNet contains 202 validated tasks and O*NET contains 118. TaskNet identifies 79 tasks with no O*NET counterpart (*new*), while 30 O*NET tasks have no TaskNet counterpart (*missing*). The overlap is reported at TaskNet's task granularity: 123 TaskNet tasks correspond to 88 O*NET tasks because a single O*NET task can be represented by multiple, more granular TaskNet tasks. Examples of tasks unique to each inventory are shown.

(38), Personal Care Aides (36), and Lawyers (32) in between. Of these tasks, 180 (89 %) are *core* (i.e., relevant to at least two-thirds of workers and rated at least 3 in mean importance), capturing work that is both widespread and important. The remaining 22 are *supplemental*, capturing less common or less central activities. Together, the statements cover 89.3 % of the 2726 tasks reported across the interviews, from 79.4 % for Dishwashers to 95.6 % for Elementary School Teachers.

4.3 Task Overlap and Distinctiveness

TaskNet overlaps substantially with O*NET tasks while also identifying tasks not already captured by the existing inventory (Figure 2). Of the 118 O*NET tasks, 88 (75 %) are covered by 123 tasks in TaskNet, as some existing tasks correspond to multiple tasks in TaskNet. The remaining 30 (25 %) are not covered, over a third of which are supplemental tasks. TaskNet not only covers existing tasks but also adds tasks absent from O*NET's inventory. Of the 202 tasks in TaskNet, 79 (39 %) lack an identified O*NET counterpart (*new*). The share of new tasks varies across occupations, ranging from 15 % for Dishwashers to 52 % for Software Developers.

Among the overlapping tasks, TaskNet often provides finer-grained descriptions of existing tasks (Figure 3). Of the 123 TaskNet tasks with an identified O*NET counterpart, 58 (47 %) refine existing task descriptions by capturing more specific activities (*refined*). The remaining 65 (53 %) overlap without providing additional granularity (*related*).

We use an LLM judge to map these relationships (Section G.1). The first authors independently labeled 40 randomly sampled statements, agreeing on counterpart presence at 83 % ($\kappa = 0.66$) and the three-way relation at 70 % ($\kappa = 0.53$). Where the authors agreed, the judge matched their labels 85 % and 71 % of the time, respectively (Section G.2).

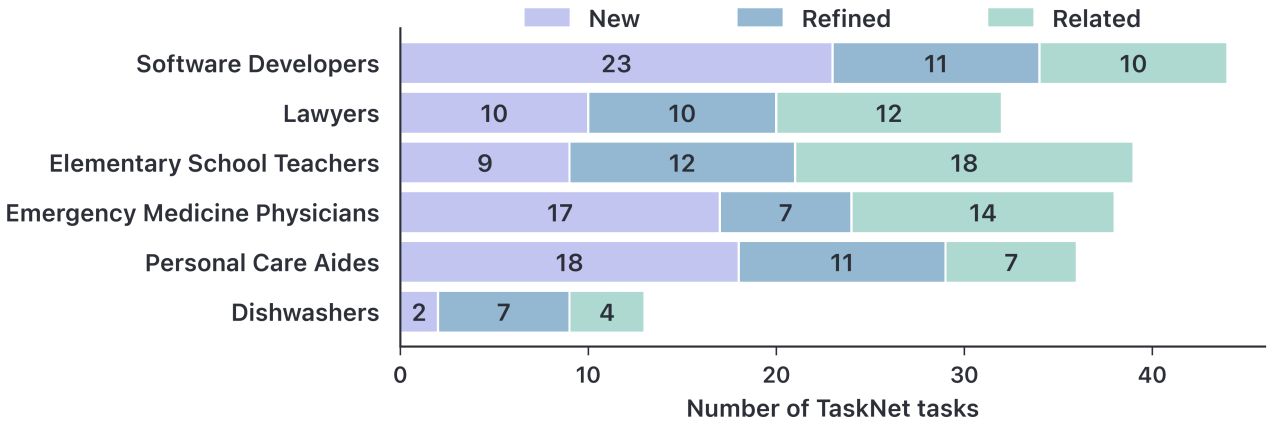


Figure 3: **Most TaskNet tasks add to or sharpen O*NET, and the mix varies by occupation.** Each TaskNet statement is assigned to *New* (79), *Refined* (a narrower subtask of an O*NET task; 58), or *Related* (overlapping with or broader than an O*NET task; 65). New tasks make up about half of the inventories for Software Developers (23 of 44) and Personal Care Aides (18 of 36), whereas most Dishwashers statements refine or relate to existing O*NET tasks (11 of 13).

5 Task Inventory Validation

Although task inventories underpin much of the research on the economic impact of AI, there has been no systematic way to assess them, since no definitive list establishes the scope of an occupation's tasks or their appropriate granularity. We propose such a framework, evaluating any task inventory at two levels: the quality of individual statements and the representativeness of the inventory as a whole. At the statement level, *expert review* assesses task-writing quality and occupational fit (Section 5.1). At the inventory level, we compare TaskNet with O*NET version 31.0 on how well they represent work beyond the construction sample from six occupations. An *internal evaluation* tests coverage and cluster coherence on held-out worker reports collected through the same interview process (Section 5.2). An *external evaluation* tests coverage of employers' descriptions of work in job postings (Section 5.2).

5.1 Task Statement Quality

Every statement meets O*NET 's standard for a task statement. Occupational analysts at the O*NET Centerreviewed 229 system-generated statements following Section 3.3 and judged every task statement to be a valid task. For comparison, 65.5 % of initial write-in task suggestions provided by job incumbents and occupational experts were accepted as task statements in O*NET's two most recent rounds of review (O*NET's Analysis Cycles 26 and 27) (National Center for O*NET Development, 2026a).

Although every statement was a valid task, analysts flagged some that departed from O*NET 's writing guidelines or described work primarily performed by another occupation. Their feedback led us to rewrite 21 % of statements (Section D). The first authors separately checked each statement against its underlying worker reports before review, revising 19 % for fidelity. Alongside revisions to wording, O*NET analysts classified 11 % of tasks as *adjacent*—valid tasks excluded from the

respective occupation's task list due to occupational fit. The remaining 204 statements formed the primary list of *occupational tasks*.

Workers recognize nearly every task as part of their work. Workers in each occupation rated the 204 occupational statements on O*NET's dimensions of relevance, importance, and frequency. We retain statements that at least a quarter of workers rate as relevant to their occupation (Section 3.3), which removes two statements and leaves 202. Of these statements, 180 (89 %) are core, against 82 % of O*NET's own tasks for these occupations (Table 1). The gap is largest for Software Developers (91 % vs. 65 %), Lawyers (88 % vs. 68 %), and Personal Care Aides (83 % vs. 73 %).

5.2 Inventory Representativeness

To assess whether our inventories generalize beyond the interviews used to construct them, we compare TaskNet and the sampled occupations from O*NET 31.0 on two independent sources: internally, on tasks reported by workers held out from inventory construction, and externally, on tasks extracted from job postings.

For internal evaluation, we interviewed 57 additional workers across the six occupations using the same recruitment and interview process. They reported 1184 tasks in total, none of which were used to build TaskNet. We assess both the breadth and organization of each inventory using *coverage* and *silhouette score*. Coverage is the share of held-out tasks an LLM judge finds described by at least one statement. Silhouette scores use task embeddings to assess whether tasks assigned to the same best-matching statement are more similar than those assigned elsewhere (Table 18).

For external evaluation, we use 2026 U.S. job postings from Revelio Labs, drawing on a 1 % random sample (Redivis, 2025) and supplementing it with 1000 randomly selected postings each for Personal Care Aides, Elementary School Teachers, and Emergency Medicine Physicians. We extract task sentences with the Job Ad Analysis Toolkit (JAAT) (Meisenbacher et al., 2025) and split them into single activities, yielding 12 474 activities from 2186 postings. An LLM then judges whether each inventory contains a statement describing each activity.

TaskNet covers more of the held-out tasks. Our inventories achieve higher coverage in five of the six occupations, averaging 89.3 % compared with O*NET's 78.6 % (Table 2). These differences are significant in all sampled occupations except Dishwashers, where both inventories cover 85.7 % of tasks. The gains are largest for Software Developers (17.0 points), Emergency Medicine Physicians (16.9), and Personal Care Aides (14.6), where O*NET leaves roughly a fifth to a third of the tasks workers report undescribed.

Table 2: **TaskNet coverage is higher in five of six occupations, averaging $89.3\,\%$ versus $78.6\,\%$ with occupations weighted equally.** Δ is TaskNet coverage minus O*NET's, in percentage points. Standard errors are the standard deviation of a paired bootstrap that resamples held-out workers within each occupation (10 000 replications). $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$.

Occupation	Workers	Tasks	TaskNet (%)	O*NET (%)	Δ (pp)		SE
Lawyers	22	451	84.0	75.6	+8.4	**	(2.4)
Software Developers	10	264	81.4	64.4	+17.0	***	(3.3)
Emergency Medicine Physicians	9	195	92.8	75.9	+16.9	***	(4.2)
Elementary School Teachers	6	143	97.9	90.9	+7.0	***	(1.5)
Personal Care Aides	5	82	93.9	79.3	+14.6	*	(5.5)
Dishwashers	5	49	85.7	85.7	+0.0		(8.5)
Pooled	57	1,184	89.3	78.6	+10.7	***	(2.0)

TaskNet draws sharper boundaries between tasks. Pooled across occupations, TaskNet 's silhouette score under language model cosine similarity is nearly double O*NET 's (0.052 vs. 0.027; $p < 0.001$), with significant gains in four of six occupations. These gains range from 0.020 for Lawyers to 0.063 for Teachers (Section H.1), suggesting that TaskNet's statements capture tighter groups of related tasks, with less overlap between them.

TaskNet also covers more activities in job postings. Across six occupations, TaskNet covers an average of $72.5\,\%$ of the activities in 2026 job postings, compared with $70.2\,\%$ for O*NET, a difference of 2.3 percentage points (s.e. 0.6, $p < 0.001$; Table 3). This difference varies by occupation. TaskNet's coverage is higher for Personal Care Aides (11.4 percentage points) and Emergency Medicine Physicians (10.1), but lower for Elementary School Teachers (-8.8) and Dishwashers (-7.4), each at $p < 0.001$. Because job postings advertise selected activities rather than the full range of work, this comparison tests generalization beyond the interview sample.

TaskNet's coverage advantage is driven by differences in task content. New TaskNet tasks cover 6.5 percentage points of activities that O*NET misses, whereas missing tasks in O*NET cover only 3.4 points. For Software Developers, new tasks such as reviewing code changes and mentoring junior developers account for 13.5 points of coverage. In contrast, for work covered by both inventories, O*NET's broader task descriptions better match generally worded job postings (7.4 versus 6.6 points). This is particularly evident for Dishwashers, where general requirements such as maintaining kitchen cleanliness match O*NET's broader tasks but not TaskNet's specific descriptions.

Table 3: **TaskNet covers more activities in job postings on average across occupations.** Coverage is the percentage of activities that a language model judges to be described by an inventory statement. Postings are deduplicated by merging reposts and include only unique templates containing at least one task activity. Δ is TaskNet minus O*NET, in percentage points. Standard errors are estimated by bootstrapping postings within occupations (4,000 replications). $^{**}p < .01$, $^{***}p < .001$.

Occupation	Postings	Activities	TaskNet (%)	O*NET (%)	Δ (pp)		SE
Software Developers	909	4,733	79.2	74.1	+5.0	***	(0.9)
Lawyers	279	1,837	69.3	65.8	+3.5	**	(1.2)
Elementary School Teachers	137	1,592	72.6	81.4	−8.8	***	(1.2)
Emergency Medicine Physicians	202	575	68.9	58.8	+10.1	***	(2.3)
Personal Care Aides	549	3,037	80.7	69.4	+11.4	***	(0.9)
Dishwashers	110	700	64.1	71.6	−7.4	***	(1.7)
Pooled	2,186	12,474	72.5	70.2	+2.3	***	(0.6)

6 New, Refined, and Missing Tasks

Task-based models describe technological change as the net effect of two forces: automation *displaces* labor from tasks it previously performed, while the creation of new tasks *reinstates* it (Acemoglu and Restrepo, 2019). The relation categories from Section 4.3 offer an initial view of both. New statements may signal emerging work, while missing tasks—O*NET tasks with no counterpart in TaskNet—may offer candidates for declining labor involvement or they may reflect methodological differences, such as sampling variation or O*NET's use of multiple sources of task information.

To characterize each category, the first authors read all of its statements and identified recurring themes. We defined each theme in a codebook with inclusion and exclusion rules and applied it by consensus (Section I). For comparison, we coded the 79 new statements together with the 65 related statements, and the 30 unconfirmed O*NET tasks together with the 88 covered ones, blind to category. Refined statements were coded against the O*NET task they refine. Because the themes were derived from the same statements they describe, the resulting comparisons show which features distinguish each category in our data rather than test hypotheses specified in advance. The themes are also not exhaustive, so some statements fall outside them, and others span several.

New tasks surface invisible work. Every occupation includes work that keeps it running but seldom appears in formal descriptions of the job (Suchman, 1995). Star and Strauss (1999) describe such work as invisible: backgrounded activity that enables occupations to carry out their core function. Over half of new statements (53 %) fall into three themes that echo documented forms of invisible work, compared with about a third of related statements (34 %). The most distinctive is *oversight and quality assurance* (24 % of new against 9 % of related): tasks that maintain standards and establish accountability for the quality and safety of the work, such as Lawyers approving staff-drafted reports and filings or Physicians countersigning residents' charts. The second is *coordination* (22 % against 18 %), in which workers hand off tasks, information, and responsibility between people, across shifts, and from one stage to the next, such as Physicians receiving patient information at shift changes or Developers passing completed features to quality assurance teams. The third is *teaching and professional development* (11 % against 8 %), tasks that sustain the expertise an occupation depends on, such as Physicians teaching residents or Personal Care Aides training staff in safe transfers.

Refined tasks clarify how work is structured and performed. Refined statements describe the core of each occupation (56 of 58 are core tasks). Refined statements add additional detail to their O*NET counterparts in two ways. 48% of our statements split an O*NET task into the steps or duties it bundles (e.g., Lawyers' "represent clients in court or before government agencies" becomes drafting motions and briefs, preparing exhibits and witnesses, and presenting arguments). In 59% of the tasks, it specifies how a task is performed, through particular methods, tools, or standards (e.g., Personal Care Aides' "Administer bedside or personal care, such as ambulation or personal hygiene assistance" becomes assisting clients with walking, transfers, and repositioning using mobility aids such as gait belts and ceiling lifts). The former reveals distinct duties within broad tasks, while the latter clarifies the conditions and requirements of performing them.

Missing tasks are O*NET's least prevalent tasks. On O*NET's own ratings, missing tasks are the ones its respondents report least often. 11 of the 30 missing tasks are supplemental, three times the share among covered tasks (37% against 11%). Our coding points the same way: 37% are specialized or dated, against 5% of covered tasks. Examples include Lawyers probating wills or serving as a trustee, guardian, or executor, and Dishwashers setting up banquet tables or packaging individual place settings. O*NET may better capture such activities due to its sample.

Technology creates new tasks and displaces old ones. $20\,\%$ of new statements depend on a recent tool or practice, against $3\,\%$ of shared ones. Technology enters the inventories in two ways. Sometimes it creates a task of its own, for example, Teachers reading and responding to email from colleagues or Developers building continuous integration and delivery pipelines. Technology also changes how existing tasks are done, for example, Teachers assigning computer-based activities to students. AI appears in the same two ways. For Software Developers, it creates new tasks such as designing prompts and evaluating model outputs. For Lawyers, it is a new object of existing review. Technology can also render older task descriptions obsolete. For instance, four of the 17 O*NET Software Developer tasks may predate current practice, such as "determine hardware configuration", a decision that cloud platforms now largely abstract away.[4]

7 Implications for AI Exposure Measurement

Nearly every AI exposure estimate relies on task inventories, making the choice of inventory consequential for measuring AI's potential labor market impacts. Conceptually, *AI exposure* measures the degree to which AI capabilities overlap with the activities involved in a task (Felten et al., 2021, Webb, 2020). Exposure estimates therefore depend on how tasks are defined and represented in an inventory. TaskNet matches O*NET's standards while covering more work in finer detail, making it a natural basis for revisiting exposure estimates.

We use two established measures of AI exposure that rate each task based on its written description and average across tasks within each occupation. Eloundou-β (Eloundou et al., 2024) captures whether access to a large language model would cut the time needed to complete a task by at least half, counting tasks that also require additional software at half weight (0–1 scale). RL feasibility (Tomei and Teeselink, 2026) uses a rubric to assess how feasible it is to train language models on a task through reinforcement learning (0–100 scale), based on dimensions including machine access, environment simulability, output gradeability, feedback frequency, and sequential decision

[4] All O*NET tasks for our occupations were last updated between 2023 and 2025. Because these recently updated statements contain some potentially outdated terminology, our count may be a lower bound on obsolescence across O*NET, especially where occupations were last updated considerably earlier.

depth (each rated 1–10; Section J.1). We then decompose the differences between inventories into contributions from newly identified, shared, and missing tasks.

Exposure estimates differ under TaskNet and O*NET in both directions. Lawyers show the largest differences (9.6 in RL feasibility and 0.16 in Eloundou-β), followed by Software Developers (9.1 and 0.07) and Elementary School Teachers (8.1 and 0.12). Emergency Medicine Physicians score slightly higher (2.0 and 0.03), Dishwashers are unchanged, while Personal Care Aides score lower (-6.5 and -0.11). Pooled across occupations, the net difference is 3.7 points in RL feasibility and 0.05 in Eloundou-β (Table 4).[5]

Table 4: **TaskNet raises predicted AI exposure for some occupations but lowers it for others.** Δ is TaskNet minus O*NET. The pooled row weights the six occupations equally.

	Eloundou-β (0–1)			RL feasibility (0–100)		
Occupation	TaskNet	O*NET	Δ	TaskNet	O*NET	Δ
Software Developers	0.81	0.74	+0.07	62.4	53.3	+9.1
Lawyers	0.55	0.39	+0.16	51.3	41.7	+9.6
Elementary School Teachers	0.38	0.26	+0.12	37.2	29.1	+8.1
Emergency Medicine Physicians	0.26	0.24	+0.03	36.9	34.9	+2.0
Personal Care Aides	0.17	0.27	−0.11	12.2	18.7	−6.5
Dishwashers	0.00	0.00	0.00	0.0	0.0	0.0
Pooled (occupation-weighted)	0.36	0.32	+0.05	33.3	29.6	+3.7

New tasks drive the pooled difference. We decompose the exposure difference into the addition of new tasks, changes to shared tasks, and omission of missing tasks (Table 23). New tasks raise RL feasibility in four of the six occupations, accounting for 2.55 of the 3.70 pooled difference, compared with 0.71 from shared tasks and 0.45 from missing tasks. For Eloundou-β, the 0.046 difference is more evenly distributed across the three sources. Among new tasks, the largest contributions come from coordination tasks (0.98), followed by oversight and quality assurance (0.85) and new technology (0.46; Section J.3). Coordination tasks are feasible because they do not require physical presence. Oversight tasks have more simulatable environments and verifiable outputs. And new technology tasks benefit from machine-accessible environments and frequent feedback.

Finer tasks reveal exposure within broader tasks. Changes to shared tasks are larger within occupations but run both ways: they raise RL feasibility for Lawyers (8.5), Software Developers (5.8), and Elementary School Teachers (6.1), but lower it for Emergency Medicine Physicians (-3.9) and Personal Care Aides (-12.3). These differences reflect how broad task descriptions can mask activities with different exposure levels. TaskNet refines 28 O*NET tasks into 58 statements, with the most exposed tasks scoring an average of 9.6 RL points and 0.18 on Eloundou-β above their O*NET counterparts. For example, TaskNet separates an Emergency Medicine Physician task on coordinating medical staff ($\beta = 0$; RL 16.7) into communicating patient status and orders ($\beta = 0.5$; RL 51.4) and supervising residents ($\beta = 0$; RL 0.0). This refinement locates AI exposure within a job, clarifying where AI capabilities may apply within occupations.

[5]Each occupation's exposure is a deterministic function of its task list rather than an estimate from a sample, so we report differences descriptively, without significance tests or confidence intervals. The differences describe the two inventories as constructed.

8 Discussion

Understanding how AI changes work requires an up-to-date account of what people do in their jobs. This paper makes five contributions toward such an account. First, we introduce the TaskNet Interviewer, a system that elicits tasks from incumbents through AI-conducted interviews with adaptive task probing and verifies them through worker review (Section 3.1). Second, we present an aggregation framework that makes explicit two choices that most inventories make implicitly—whether different reports describe the same task and how finely tasks should be defined (Section 3.2). It defines task identity through hierarchical task analysis and task granularity through functional job analysis (Fine and Cronshaw, 1999, Stanton, 2006), and applies these definitions in an iterative procedure that resolves overlaps and preserves coverage of the reported work. Third, since task lists have no ground truth, we propose an evaluation framework that triangulates across O*NET Center Analysts, incumbents, worker ratings, and job postings (Section 5). Fourth, we release validated inventories for six occupations: O*NET Center Analysts judged all statements to be valid task statements, and incumbents endorsed all but two of the 204 as relevant. Finally, we show that AI exposure estimates differ between TaskNet and O*NET, with higher estimates under TaskNet for Lawyers and Software Developers and lower estimates for Personal Care Aides (Section 7).

8.1 Explaining Differences in AI Exposure Estimates

Substituting TaskNet for O*NET changes exposure estimates, with new tasks accounting for much of the difference (Section 7). Three channels may explain this difference: task discovery methods, data collection periods, and worker samples. Our design cannot isolate each channel's respective contribution, so we discuss the evidence supporting each and its limitations.

Task Discovery Methods. O*NET's surveys ask incumbents to rate statements that already exist, and new tasks enter only through write-ins and Analyst review (Dierdorff and Norton, 2011, Klein et al., 2025). In contrast, TaskNet uses open-ended interviews that allow workers to describe their activities in their own words, with follow-ups to surface additional tasks (Section 3.1). Under this explanation, "new" tasks are not necessarily newly emerging or increasingly performed tasks, but existing tasks that are newly documented. Over half of the new statements concern such invisible tasks, which contribute to the difference in exposure scores. These are activities that keep an occupation running but seldom appear in formal descriptions of a job (Star and Strauss, 1999). TaskNet therefore captures AI-exposed work that existing inventories miss.

The high exposure of invisible tasks is notable because they typically draw on tacit knowledge and are difficult to fully automate. Work that is difficult to observe or articulate may still be exposed to AI, even if human judgment, accountability, and relationships may remain essential and prevent full automation. Hence, the extent to which AI replaces rather than supports human work depends not only on technical feasibility but also on how organizations choose to deploy it.

Task Collection Periods. While we selected occupations whose O*NET task information was updated recently (between 2023 and 2025), work may have evolved since, and O*NET's discovery process adds new tasks only in later survey cycles (Section 1). Some new statements do reflect genuinely recent work: 20 % depend on a recent tool or practice, against 3 % of related statements, and new technology is one theme behind the exposure difference. For these statements, "new" does mean newly performed. The high exposure of these tasks may partly reflect priorities of model

developers. AI and related digital tools create new activities, such as prompt design and model evaluation, that align closely with the capabilities and environments developers are optimizing for.

Worker Sampling. Finally, the inventories may differ because they draw on different worker samples. TaskNet recruits workers through an AI data platform based on occupational fit and experience (Section 4.1), whereas O*NET draws on multiple sampling frames, including business establishments, professional associations, and specialized occupational listings. These approaches may reach workers with different roles and experiences, potentially affecting which tasks are reported. Notably, O*NET tasks missing from TaskNet are more often specialized or dated than covered tasks (37 % versus 5 %). The omission of these tasks contributes to differences in AI exposure estimates.

8.2 Limitations and Future Work

Our approach has several limitations.

First, constructing task inventories requires judgments about task specificity and occupational fit. Although our task definition draws on existing theory (Fine and Cronshaw, 1999, Stanton, 2006), it cannot eliminate subjectivity in determining task boundaries. Similarly, judgments about occupational fit may exclude relevant tasks: 81 % of the statements analysts judged adjacent still met our task criterion, and more than half were core (Section D.1). Such tasks may fall outside prevailing occupational definitions even as they become more important in practice. These choices affect downstream measures, as specificity determines how work is represented in task averages, while occupational fit determines which activities are counted.

Second, although our inventories represent our sample well, roughly twenty workers per occupation recruited through a single platform may not capture the full range of occupational tasks. Expanding beyond six occupations would enable broader validation and help establish whether task-based exposure predicts short-term labor market outcomes (Brynjolfsson et al., 2025a, Lichtinger and Hosseini Maasoum, 2025) or long-term employment projections (Massenkoff and McCrory, 2026).

Third, our task inventories capture individual activities but omit workers' preferences and broader attributes, including knowledge, skills, and abilities. By representing tasks as unordered lists, they also overlook how activities interact within workflows, even though these relationships shape how automation affects production (Demirer et al., 2026, Gilbreth and Gilbreth, 1921, Hammer, 1990, Malone and Crowston, 1994). Future work could use AI-guided interviews to capture both the human and organizational context surrounding tasks.

Despite these limitations, our study offers a proof of concept for using worker interviews to document tasks at scale and modest expense. Scaling our method across occupations could support more current and detailed representations of work, as a stronger empirical foundation for anticipating changes in skill demand and guiding economic policy. AI may be driving an unprecedented transformation of work. Our information infrastructure for understanding this transformation must advance alongside it.

9 Acknowledgments

We would like to thank members of the Stanford Social and Language Technology Lab and the Stanford Digital Economy Lab for their helpful discussions and feedback. We are especially grateful

to Tijana Zrnic, Bharat Chandar, David Autor, Luca Vendraminelli, Henri Beyer, Joachim Baumann, Vishakh Padmakumar, and Michael Ryan. This research is supported in part by the Laude Moonshot Seed Grant on "Collaborative Intelligence for the Future of Work"[6], ONR grant N000142412532, Singapore Ministry of Digital Development and Information (to DY), and the Stanford Digital Economy Lab. JW is partially supported by NSF DGE-2146755. EB and AH received support from Schmidt Sciences.

10 Author Contributions

JW led the design and implementation of the TaskNet Interviewer system, including the AI interviewer and the worker verification interface. JW developed the task aggregation pipeline, with design input from AH. JW and AH jointly conducted the task inventory validation and exposure score analysis. JW and AH collaborated on writing the paper. IR coordinated the data partnership with Mercor, including worker recruitment and research operations. PL advised on validating the task inventory against O*NET's standards and contributed to refining the paper's framing. EB and DY supervised the project, shaped its research direction and framing, as well as advised on study design and interpretation.

References

Daron Acemoglu and Pascual Restrepo. Automation and new tasks: How technology displaces and reinstates labor. *Journal of Economic Perspectives*, 33(2):3–30, 2019. doi: 10.1257/jep.33.2.3.

Advisory Panel for the Dictionary of Occupational Titles. The new DOT: A database of occupational titles for the twenty-first century. Final report, U.S. Department of Labor, Employment and Training Administration, United States Employment Service, Washington, DC, May 1993. URL https://www.onetcenter.org/dl_files/omb2002/AppendixC.pdf. Reproduced as Appendix C of the O*NET Data Collection Program OMB clearance package (2002).

Herman Aguinis and Ernest O'Boyle, Jr. Star performers in twenty-first century organizations. *Personnel Psychology*, 67(2):313–350, 2014. doi: 10.1111/peps.12054.

John Annett and Keith D. Duncan. Task analysis and training design. *Occupational Psychology*, 41: 211–221, 1967.

Anthropic. Introducing Anthropic Interviewer: What 1,250 professionals told us about working with AI. Anthropic research report, 4 December 2025, 2025. URL https://www.anthropic.com/research/anthropic-interviewer.

David Anugraha, Vishakh Padmakumar, and Diyi Yang. SparkMe: Adaptive semi-structured interviewing for qualitative insight discovery, 2026. URL https://arxiv.org/abs/2602.21136.

Ruth Appel, Peter McCrory, Alex Tamkin, Miles McCain, Tyler Neylon, and Michael Stern. Anthropic economic index report: Uneven geographic and enterprise AI adoption. Anthropic, 2025. URL https://www.anthropic.com/research/anthropic-economic-index-september-2025-report.

[6] https://www.laude.org/moonshots/one/collaborative-intelligence-future-of-work

Ruth Appel, Maxim Massenkoff, Peter McCrory, Miles McCain, Ryan Heller, Tyler Neylon, and Alex Tamkin. Anthropic Economic Index report: Economic primitives. Anthropic research report, January 2026. URL https://www.anthropic.com/research/anthropic-economic-index-january-2026-report.

Enghin Atalay, Phai Phongthiengtham, Sebastian Sotelo, and Daniel Tannenbaum. The evolution of work in the United States. *American Economic Journal: Applied Economics*, 12(2):1–34, 2020. doi: 10.1257/app.20190070.

David Autor and Neil Thompson. Expertise. *Journal of the European Economic Association*, 23(4): 1203–1271, 2025. doi: 10.1093/jeea/jvaf023. Also NBER Working Paper 33941.

David Autor, Caroline Chin, Anna Salomons, and Bryan Seegmiller. New frontiers: The origins and content of new work, 1940–2018. *The Quarterly Journal of Economics*, 139(3):1399–1465, 2024. doi: 10.1093/qje/qjae008.

David H. Autor. Polanyi's paradox and the shape of employment growth. Technical Report Working Paper 20485, National Bureau of Economic Research, 2014.

David H. Autor. Why are there still so many jobs? the history and future of workplace automation. *Journal of Economic Perspectives*, 29(3):3–30, 2015. doi: 10.1257/jep.29.3.3.

David H. Autor and Michael J. Handel. Putting tasks to the test: Human capital, job tasks, and wages. *Journal of Labor Economics*, 31(S1):S59–S96, 2013. doi: 10.1086/669332.

David H. Autor, Frank Levy, and Richard J. Murnane. The skill content of recent technological change: An empirical exploration. *The Quarterly Journal of Economics*, 118(4):1279–1333, 2003. doi: 10.1162/003355303322552801.

Erik Brynjolfsson, Tom Mitchell, and Daniel Rock. What can machines learn, and what does it mean for occupations and the economy? *AEA Papers and Proceedings*, 108:43–47, 2018. doi: 10.1257/pandp.20181019.

Erik Brynjolfsson, Bharat Chandar, and Ruyu Chen. Canaries in the coal mine? six facts about the recent employment effects of artificial intelligence. Stanford Digital Economy Lab working paper, 2025a.

Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. Generative AI at work. *The Quarterly Journal of Economics*, 140(2):889–942, 2025b. doi: 10.1093/qje/qjae044.

Roger Buehler, Dale Griffin, and Michael Ross. Exploring the "planning fallacy": Why people underestimate their task completion times. *Journal of Personality and Social Psychology*, 67(3): 366–381, 1994. doi: 10.1037/0022-3514.67.3.366.

Aaron Chatterji, Thomas Cunningham, David J. Deming, Zoë Hitzig, Christopher Ong, Carl Yan Shan, and Kevin Wadman. How people use ChatGPT. NBER Working Paper 34255, National Bureau of Economic Research, 2025.

Raymond E. Christal. The United States Air Force occupational research project. Technical Report AFHRL-TR-73-75, Air Force Human Resources Laboratory, Occupational Research Division, Lackland AFB, TX, 1974.

Raymond E. Christal and Johnny J. Weissmuller. Job-task inventory analysis. In Sidney Gael, editor, *The Job Analysis Handbook for Business, Industry, and Government*, volume 2, pages 1036–1050. Wiley, New York, 1988.

Alejandro Cuevas, Jennifer V. Scurrell, Eva M. Brown, Jason Entenmann, and Madeleine I. G. Daepp. Collecting qualitative data at scale with large language models: A case study, 2024. URL https://arxiv.org/abs/2309.10187.

Shih-Chieh Dai, Aiping Xiong, and Lun-Wei Ku. LLM-in-the-loop: Leveraging large language model for thematic analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9993–10001, 2023. doi: 10.18653/v1/2023.findings-emnlp.669.

Fabrizio Dell'Acqua, Edward McFowland III, Ethan Mollick, Hila Lifshitz, Katherine C. Kellogg, Saran Rajendran, Lisa Krayer, François Candelon, and Karim R. Lakhani. Navigating the jagged technological frontier: Field experimental evidence of the effects of artificial intelligence on knowledge worker productivity and quality. *Organization Science*, 37(2):403–423, 2026. doi: 10.1287/orsc.2025.21838.

David J. Deming and Lisa B. Kahn. Skill requirements across firms and labor markets: Evidence from job postings for professionals. *Journal of Labor Economics*, 36(S1):S337–S369, 2018. doi: 10.1086/694106.

Mert Demirer, John J. Horton, Nicole Immorlica, Brendan Lucier, and Peyman Shahidi. Chaining tasks, redefining work: A theory of AI automation. NBER Working Paper 34859, National Bureau of Economic Research, February 2026.

Jean-Stanislas Denain, Joe Kwon, and Anson Ho. Toward an O*NET for AI R&D, 2026. URL https://epoch.ai/gradient-updates/toward-an-onet-for-ai-rnd. Accessed: 2026-09-26.

Erich C. Dierdorff and Jennifer J. Norton. Summary of procedures for O*NET task updating and new task generation. Technical report, National Center for O*NET Development, 2011. URL https://www.onetcenter.org/reports/TaskUpdating.html.

Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. GPTs are GPTs: Labor market impact potential of LLMs. *Science*, 384(6702):1306–1308, 2024. doi: 10.1126/science.adj0998.

Edward Felten, Manav Raj, and Robert Seamans. Occupational, industry, and geographic exposure to artificial intelligence: A novel dataset and its potential uses. *Strategic Management Journal*, 42 (12):2195–2217, 2021. doi: 10.1002/smj.3286.

Sidney A. Fine and Steven F. Cronshaw. *Functional Job Analysis: A Foundation for Human Resources Management*. Psychology Press, 1st edition, 1999. doi: 10.4324/9781410602497.

Sidney A. Fine and Wretha W. Wiley. An introduction to functional job analysis: A scaling of selected tasks from the social welfare field. Monograph Methods for Manpower Analysis No. 4, W. E. Upjohn Institute for Employment Research, Kalamazoo, MI, 1971.

Luis Garicano. Hierarchies and the organization of knowledge in production. *Journal of Political Economy*, 108(5):874–904, 2000. doi: 10.1086/317671.

Friedrich Geiecke and Xavier Jaravel. Conversations at scale: Robust AI-led interviews. SSRN Working Paper 4974382; CEPR Discussion Paper 19705, 2024. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4974382.

Frank B. Gilbreth and Lillian M. Gilbreth. Process charts: First steps in finding the one best way to do work. *Transactions of the American Society of Mechanical Engineers*, 43:1029–1043, 1921. doi: 10.1115/1.4058133. Paper No. 1818, presented at the ASME Annual Meeting, December 1921.

Michael Hammer. Reengineering work: Don't automate, obliterate. *Harvard Business Review*, 68(4): 104–112, 1990.

Kunal Handa, Alex Tamkin, Miles McCain, Saffron Huang, Esin Durmus, Sarah Heck, Jared Mueller, Jerry Hong, Stuart Ritchie, Tim Belonax, Kevin K. Troy, Dario Amodei, Jared Kaplan, Jack Clark, and Deep Ganguli. Which economic tasks are performed with AI? evidence from millions of Claude conversations, 2025. URL https://arxiv.org/abs/2503.04761.

Eric Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*, pages 159–166, 1999. doi: 10.1145/302979.303030.

Mark Howison, William O. Ensor, Suraj Maharjan, Rahil Parikh, Srinivasan H. Sengamedu, Paul Daniels, Amber Gaither, Carrie Yeats, Chandan K. Reddy, and Justine S. Hastings. Extracting structured labor market information from job postings with generative AI. *Digital Government: Research and Practice*, 6(1):1–12, 2025. doi: 10.1145/3674847.

Mark A. Huselid, Richard W. Beatty, and Brian E. Becker. "A players" or "A positions"? The strategic logic of workforce management. *Harvard Business Review*, 83(12):110–117, 2005.

Dahyeon Jeong, Shilpa Aggarwal, Jonathan Robinson, Naresh Kumar, Alan Spearot, and David Sungho Park. Exhaustive or exhausting? Evidence on respondent fatigue in long surveys. *Journal of Development Economics*, 161:102992, 2023. doi: 10.1016/j.jdeveco.2022.102992.

Katherine Klein, Mitchell Pontikes, Jeffrey A. Dahlke, Dan J. Putka, Brittany F. Crawford, Matthew C. Reeder, and Phil Lewis. Identification of emerging tasks in the O*NET system: A revised approach. Technical Report 2025 No. 021, Human Resources Research Organization (HumRRO), prepared for the National Center for O*NET Development, February 2025. URL https://www.onetcenter.org/dl_files/EmergingTasks_RevisedApproach.pdf.

Thomas Kwa and Vincent Cheng. How does time horizon vary across domains? METR blog, July 2025. URL https://metr.org/blog/2025-07-14-how-does-time-horizon-vary-across-domains/. Code: https://github.com/METR/cross-domain-horizon.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney Von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Harold Miles, Seraphina Nix, Tao Lin, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel M. Ziegler, Elizabeth Barnes, and Lawrence Chan. Measuring AI ability to complete long software tasks. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, pages 102382–102435, 2025. doi: 10.52202/085713-3086.

Michelle S. Lam, Janice Teoh, James A. Landay, Jeffrey Heer, and Michael S. Bernstein. Concept induction: Analyzing unstructured text with high-level concepts using LLooM. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, page Article 933, 2024. doi: 10.1145/3613904.3642830.

Frank Levy and Richard J. Murnane. *The New Division of Labor: How Computers Are Creating the Next Job Market*. Princeton University Press, Princeton, NJ, 2004. doi: 10.1515/9781400845927.

Phil Lewis and Jeremiah Morris. Supplementing the O*NET sources of additional information: A preliminary exploration of the use of ChatGPT. Technical report, National Center for O*NET Development, 2024. URL https://www.onetcenter.org/reports/SAI_GPT.html.

Phil Lewis, Christina Gregory, and Jeremiah Morris. Adding drone-specific tasks to the O*NET database: Initial identification of emerging tasks using ChatGPT. Technical report, National Center for O*NET Development, May 2025. URL https://www.onetcenter.org/reports/Drone_Tasks.html.

Guy Lichtinger and Seyed Mahdi Hosseini Maasoum. Generative AI as seniority-biased technological change: Evidence from U.S. résumé and job posting data. SSRN Working Paper 5425555, 2025.

Fengming Liu and Shubin Yu. MimiTalk: Revolutionizing qualitative research with dual-agent AI, 2025. URL https://arxiv.org/abs/2511.03731.

Thomas W. Malone and Kevin Crowston. The interdisciplinary study of coordination. *ACM Computing Surveys*, 26(1):87–119, 1994. doi: 10.1145/174666.174668.

George Mandler. Recognizing: The judgment of previous occurrence. *Psychological Review*, 87(3): 252–271, 1980. doi: 10.1037/0033-295X.87.3.252.

Maxim Massenkoff and Peter McCrory. Labor market impacts of AI: A new measure and early evidence. Anthropic working paper, March 2026. URL https://www.anthropic.com/research/labor-market-impacts.

Stephen Meisenbacher, Svetlozar Nestorov, and Peter Norlander. Extracting O*NET features from the NLx corpus to build public use aggregate labor market data, 2025. URL https://arxiv.org/abs/2510.01470.

Matthias Mertens, Adam Kuzee, Brittany S. Harris, Harry Lyu, Wensu Li, Jonathan Rosenfeld, Meiri Anto, Martin Fleming, and Neil Thompson. Crashing waves vs. rising tides: Findings on AI automation from thousands of worker evaluations of labor market tasks, 2026. URL https://arxiv.org/abs/2604.01363.

Ann R. Miller, Donald J. Treiman, Pamela S. Cain, and Patricia A. Roos, editors. *Work, Jobs, and Occupations: A Critical Review of the Dictionary of Occupational Titles*. National Academy Press, Washington, DC, 1980. doi: 10.17226/92.

Brian E. Moore. Occupational analysis for human resource development: A review of utility of the task inventory. Technical Report Research Report No. 25 (DTIC ADA026707), Office of Civilian Manpower Management, Department of the Navy, Washington, DC, 1976.

Frederick P. Morgeson and Michael A. Campion. Social and cognitive sources of potential inaccuracy in job analysis. *Journal of Applied Psychology*, 82(5):627–655, 1997. doi: 10.1037/0021-9010.82.5.627.

National Center for O*NET Development. Appendix B: Task writing guidelines. Technical report, National Center for O*NET Development, 2009. URL https://www.onetcenter.org/dl_files/GreenTask_AppB.pdf. Appendix to *Greening of the World of Work: Implications for O*NET-SOC and New and Emerging Occupations*.

National Center for O*NET Development. Personal communication, August 2026a. O*NET task analysis data for Cycles 26 and 27.

National Center for O*NET Development. O*NET resource center. https://www.onetcenter.org/, 2026b. Accessed October 22, 2026.

National Center for O*NET Development. O*NET 31.0 data dictionary, 2026c. URL https://www.onetcenter.org/dictionary/31.0/excel/task_statements.html. Section "Task Statements".

Shakked Noy and Whitney Zhang. Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654):187–192, 2023. doi: 10.1126/science.adh2586.

Joon Sung Park, Carolyn Q. Zou, Jonne Kamphorst, Niles Egan, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Percy Liang, Robb Willer, and Michael S. Bernstein. LLM agents grounded in self-reports enable general-purpose simulation of individuals, 2026. URL https://arxiv.org/abs/2411.10109.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, Natalie S. Kim, Patrick Chao, Samuel Miserendino, Gildas Chabot, David Li, Michael Sharman, Alexandra Barr, Amelia Glaese, and Jerry Tworek. GDPval: Evaluating AI model performance on real-world economically valuable tasks, 2025. URL https://arxiv.org/abs/2510.04374.

Sida Peng, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer. The impact of AI on developer productivity: Evidence from GitHub Copilot, 2023. URL https://arxiv.org/abs/2302.06590.

Norman G. Peterson, Michael D. Mumford, Walter C. Borman, P. Richard Jeanneret, and Edwin A. Fleishman, editors. *An Occupational Information System for the 21st Century: The Development of O*NET*. American Psychological Association, Washington, DC, 1999. doi: 10.1037/10313-000.

NORMAN G. PETERSON, MICHAEL D. MUMFORD, WALTER C. BORMAN, P. RICHARD JEANNERET, EDWIN A. FLEISHMAN, KERRY Y. LEVIN, MICHAEL A. CAMPION, MELINDA S. MAYFIELD, FREDERICK P. MORGESON, KENNETH PEARLMAN, MARILYN K. GOWING, ANITA R. LANCASTER, MARILYN B. SILVER, and DONNA M. DYE. Understanding work using the occupational information network (o*net): Implications for practice and research. *Personnel Psychology*, 54(2):451–492, 2001. doi: https://doi.org/10.1111/j.1744-6570.2001.tb00100.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1744-6570.2001.tb00100.x.

Chau Minh Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. TopicGPT: A prompt-based topic modeling framework. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2956–2984, 2024. doi: 10.18653/v1/2024.naacl-long.164.

Michael Polanyi. *The Tacit Dimension*. Doubleday, Garden City, NY, 1966.

Dan J. Putka, Frederick L. Oswald, Richard N. Landers, Adam S. Beatty, Rodney A. McCloy, and Martin C. Yu. Evaluating a natural language processing approach to estimating KSA and interest job analysis ratings. *Journal of Business and Psychology*, 38(2):385–410, 2023. doi: 10.1007/s10869-022-09824-0.

Dan J. Putka, Nathaniel M. Voss, and Phil Lewis. Indexing the impact of AI within the O*NET system: A review of methods and development of recommendations. Technical report, National Center for O*NET Development, June 2026. URL https://www.onetcenter.org/reports/AI_Impact_Review.html.

Diane M. Ramsey-Klee. Handbook for construction of task inventories for Navy enlisted ratings. Technical Report DTIC ADA150207, Navy Occupational Development and Analysis Center (NODAC), Washington, DC, 1984. Documents NOTAP task-inventory construction procedures.

Redivis. Redivis, 2025. URL https://redivis.com.

Michael M. Roy, Nicholas J. S. Christenfeld, and Craig R. M. McKenzie. Underestimating the duration of future events: Memory incorrectly used or memory bias? *Psychological Bulletin*, 131(5): 738–756, 2005. doi: 10.1037/0033-2909.131.5.738.

Michael F. Schober and Frederick G. Conrad. Does conversational interviewing reduce survey measurement error? *Public Opinion Quarterly*, 61(4):576–602, 1997. doi: 10.1086/297818.

Seamus Somerstep, Aritra Guha, Divesh Srivastava, and Yuekai Sun. The open source economic index of AI adoption and capability, 2026. URL https://arxiv.org/abs/2606.26118.

Neville A. Stanton. Hierarchical task analysis: Developments, applications, and extensions. *Applied Ergonomics*, 37(1):55–79, 2006. doi: 10.1016/j.apergo.2005.06.003.

Susan Leigh Star and Anselm Strauss. Layers of silence, arenas of voice: The ecology of visible and invisible work. *Computer Supported Cooperative Work*, 8(1–2):9–30, February 1999. doi: 10.1023/A:1008651105359.

Harald Steck, Chaitanya Ekanadham, and Nathan Kallus. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24)*, pages 887–890. ACM, May 2024. doi: 10.1145/3589335.3651526.

Lucy Suchman. Making work visible. *Communications of the ACM*, 38(9):56–64, 1995. doi: 10.1145/223248.223263.

Shayan Talaei, Meijin Li, Kanu Grover, James Kent Hippler, Diyi Yang, and Amin Saberi. StorySage: Conversational autobiography writing powered by a Multi-Agent framework, 2025. URL https://arxiv.org/abs/2506.14159.

Alex Tamkin, Miles McCain, Kunal Handa, Esin Durmus, Liane Lovitt, Ankur Rathi, Saffron Huang, Alfred Mountfield, Jerry Hong, Stuart Ritchie, Michael Stern, Brian Clarke, Landon Goldberg, Theodore R. Sumers, Jared Mueller, William McEachen, Wes Mitchell, Shan Carter, Jack Clark, Jared Kaplan, and Deep Ganguli. Clio: Privacy-preserving insights into real-world AI use, 2024. URL https://arxiv.org/abs/2412.13678.

Philip Moreira Tomei and Bouke Klein Teeselink. What jobs can AI learn? measuring exposure by reinforcement learning, 2026. URL https://arxiv.org/abs/2605.02598.

Kiran Tomlinson, Sonia Jaffe, Will Wang, Scott Counts, and Siddharth Suri. Working with AI: Measuring the applicability of generative AI to occupations, 2025. URL https://arxiv.org/abs/2507.07935.

U.S. Department of Labor, Employment and Training Administration. *Dictionary of Occupational Titles*. U.S. Government Printing Office, Washington, DC, 4th, revised edition, 1991. Appendix C: Specific Vocational Preparation (SVP) levels 1–9.

Vijay Viswanathan, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. Large language models enable few-shot clustering. *Transactions of the Association for Computational Linguistics*, 12:321–333, 2024. doi: 10.1162/tacl_a_00648.

Michael Webb. The impact of artificial intelligence on the labor market. SSRN Working Paper 3482150, 2020.

Mark A. Wilson and Robert J. Harvey. The value of relative-time-spent ratings in task-oriented job analysis. *Journal of Business and Psychology*, 4(4):453–461, 1990. doi: 10.1007/BF01013608.

Liang Ze Wong, Siti Amelia Juraimi, Yin Zhien Tan, Siyuan Brandon Loh, Mary F.-F. Chong, Prasanta Bhattacharya, and Aimee E. Pink. The AI interviewer: Exploring the use of conversational AI-enabled chatbots in qualitative data collection. SSRN Working Paper, March 2025. URL https://ssrn.com/abstract=5194078.

Alexander Wuttke, Matthias Aßenmacher, Christopher Klamm, Max M. Lang, Quirin Würschinger, and Frauke Kreuter. AI conversational interviewing: Transforming surveys with LLMs as adaptive interviewers. In *Proceedings of the 9th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2025)*, pages 179–204, 2025. doi: 10.18653/v1/2025.latechclfl-1.17.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. ClusterLLM: Large language models as a guide for text clustering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13903–13920, 2023. doi: 10.18653/v1/2023.emnlp-main.858.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. WildChat: 1M ChatGPT interaction logs in the wild. In *The Twelfth International Conference on Learning Representations (ICLR 2024)*, 2024.

Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester. A comprehensive survey on deep clustering: Taxonomy, challenges, and future directions, 2022. URL https://arxiv.org/abs/2206.07579.

Caleb Ziems, William Held, Jane Yu, Amir Goldberg, David Grusky, and Diyi Yang. Culture cartography: Mapping the landscape of cultural knowledge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP 2025)*, pages 1739–1757, 2025. doi: 10.18653/v1/2025.emnlp-main.91.

A Details on the TaskNet Interviewer

This appendix documents the details of discovery and aggregation (Section 3) and the prompts of the language-model calls they make: the interview application, the processing of verified per-worker lists, and the aggregation into an occupation-level inventory. Prompts not printed here are in the code release (see "Additional Prompts" after Prompt 17). Section G.2 reports the human evaluation of the judges marked with ⋆.

We show an activity diagram for the discovery phase in Figure 4. It follows one worker from intake to the processed task list. Rounded boxes are actions, blue boxes are actions of the worker, diamonds are decisions and merges, bars are forks and joins, and the filled and ringed circles are the initial and final nodes. Edges carry only the outcomes of decisions. A prompt is named by its key in typewriter type (for example `new-tasks`), which links to the prompt text below: in a decision when the reply only routes the flow, in an action when the call produces or changes content. Actions marked "no LLM" are deterministic. The diagram shows every language-model call and the decisions that route between calls. Fallbacks that would crowd it are stated in the paragraph that follows. ⋆ marks prompts whose output is subject to human evaluation (Section G.2).

A.1 Task Discovery

An interview opens with two free-text questions: the worker's job title and main responsibilities. From these, a language model ranks candidate O*NET-SOC occupations, adding the O*NET related occupations of the top match (P1). The worker selects an occupation from this list or searches the full taxonomy, and this self-declared code determines the occupation under which their tasks are aggregated. The topic guide covers eight topics—role and field, tenure, main responsibilities, weekly tasks, periodic tasks, work outputs, collaborators and stakeholders, and time use—each with explicit coverage criteria (P2). On each turn, one call extracts newly reported tasks (P3) and another marks topics complete, when their criteria are met or the worker declines, and asks about the most valuable remaining topic (P4).

A.1.1 Prompts for Task Discovery

Three further calls manage the conversation without producing task content, and their prompts are omitted: `screener-probe` asks for more detail when the intake description of responsibilities is shorter than 140 characters, `explain-question` rephrases a question the worker asks to have clarified, and `rewrite-varied` rewords a question whose opening repeats an earlier one.

P1 is the system message of a call to GPT-5.4 at the model's default sampling with a JSON-object reply. The instruction is followed by the complete O*NET-SOC 2019 list (omitted below), byte-identical across calls so that it is served from the prompt cache; the user message carries the intake answers and, when the worker asks for different options, the rejected codes and the worker's free-text hint. The application asks for five candidates, re-resolves every returned code against the canonical taxonomy, and widens the shortlist with O*NET related occupations.

Prompt 1: Occupation Self-ID Match

```
{# system (static, cacheable prefix) #}
You are an expert occupation coder. From the FULL O*NET-SOC 2019 list below, pick the TOP {{
top_n }} occupations the worker would most identify with, best first, by matching their PRIMARY
```

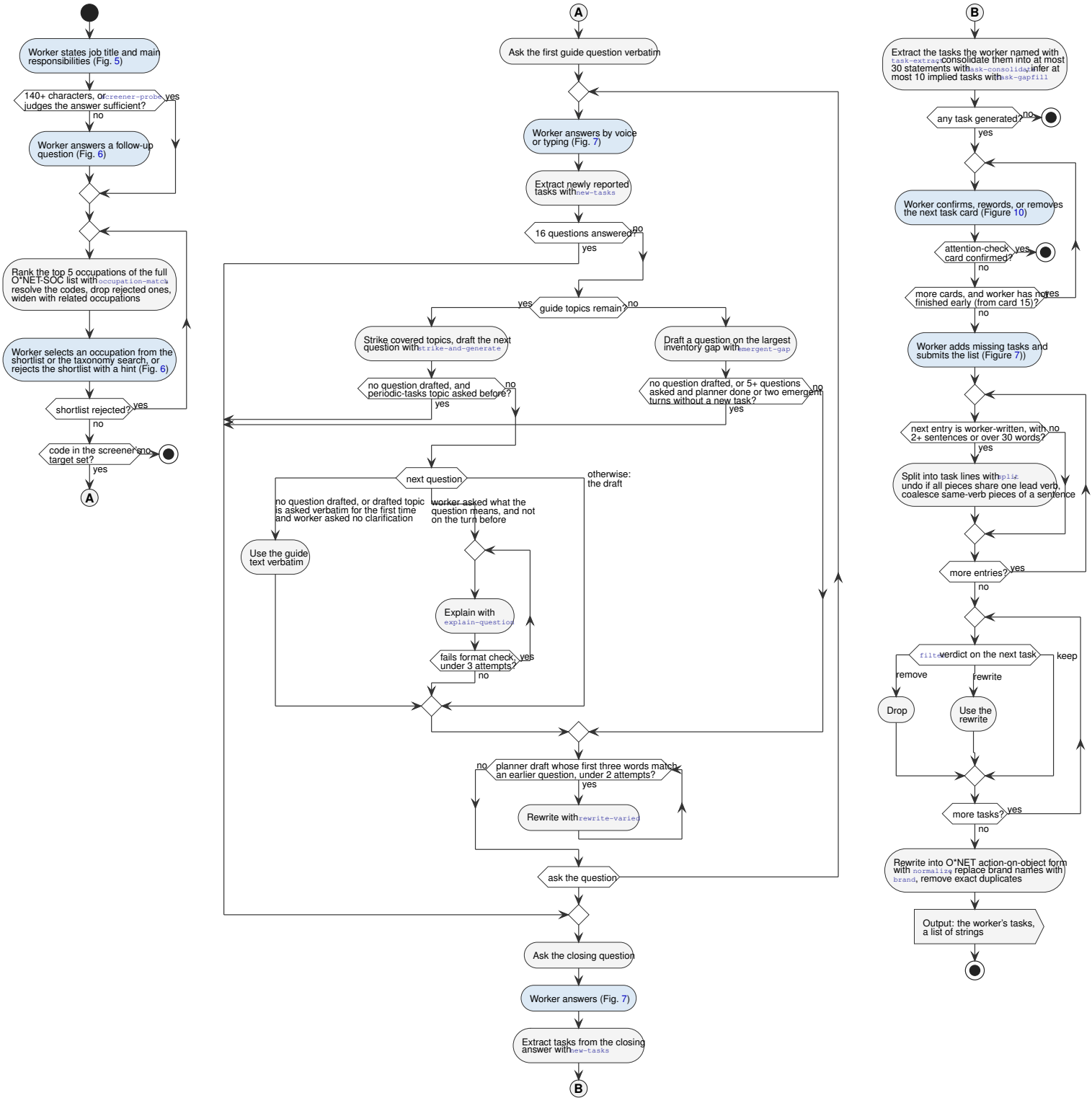


Figure 4: From intake to the processed task list. The activity takes no input: it runs against one worker, whose actions are the blue nodes (with the interface figure of Section B that shows them), and it outputs that worker's tasks as a list of strings. Left: intake and occupation selection (screener mode). Center: the adaptive interview. Right: task generation, the worker's review, and processing. A prompt is set in `typewriter` and links to its text: in a decision when its reply only routes the flow, in an action when it produces or changes content.

```
activities to the occupations. Use codes and titles EXACTLY as listed. Return ONLY JSON:
{"ranked":[{"code":"##-####.##","title":"...","why":"<=12 words why it fits"}, ... {{ top_n }}
items]}

O*NET-SOC occupations (code title):
{# the complete O*NET-SOC 2019 list from app/data/onet-soc.json, about 11,600 tokens,
byte-identical across calls so that it is served from the prompt cache #}
{% for o in onet_soc %}{{ o.code }} {{ o.title }}
{% endfor %}


{# user #}
Worker interview transcript:
{{ transcript }}  {# the intake answers as Q:/A: pairs #}
{% if rejected_codes %}
The participant has already rejected these -- do NOT return them:
{{ rejected_codes | join(', ') }}
{% endif %}{% if hint %}
The participant says none of the previous options fit, because: "{{ hint }}". Weight this
strongly.
{% endif %}
```

P2 is not a prompt but the data the planner reads: the topics not yet covered are listed with their criteria in the user message of P4, and the first topic's question opens the interview verbatim. We show the guide condensed: the general criterion summarizes the rules the per-topic criteria share, and the full criteria are in the released code.

Prompt 2: Interview Guide

```
General coverage criterion:
- The guide's goal is concrete tasks: what the worker actually does, not a label for an
  area, tool, or party. In the typical week and stakeholders topics (4 and 7), a bare label
  gets a follow-up asking what the worker has to do for it. Other topics are covered by a
  single answer, since those two topics and the gap questions after the guide collect task
  detail.
- Otherwise, accept brief answers and move on: follow up only on empty or evasive answers.
  Ask at most one follow-up per topic (up to four for the typical week), and never repeat a
  question in reworded form.

### Topic 1: jobTitle
Question: To start, what's your job and what field is it in?
Covered when: a job title or role and its field are named. The job they name is the subject of
the whole interview.

### Topic 2: tenure (asked verbatim)
Question: And how long have you been doing this job?
Covered when: a rough duration is given, here or earlier ('a few years', 'new grad').

### Topic 3: responsibilities
Question: What are your primary responsibilities at work?
Covered when: they name anything they are responsible for or do. Task detail is left to the
typical-week question that follows.

### Topic 4: typicalWeek
Question: Walk me through a typical week. What are the recurring tasks you do?
Covered when:
- they name more than one recurring task (if only one, ask whether there are others);
- central activities are described as tasks, not labels (if a bare label, ask what they have
  to do for it, one thread per turn);
- if one activity dominates, they have been asked once about the different kinds of work
  within it;
- each responsibility named earlier maps to a task (if not, ask about it once).

### Topic 5: periodicTasks
Question: Beyond your usual week, are there tasks that come up on a longer cycle -- monthly,
quarterly, or once a year?
Covered when: longer-than-weekly work has been named, or they say their work is the same week
to week. A typical week does not cover this topic. If periodic work came up earlier, ask
whether there is anything else on that rhythm.
```

```
### Topic 6: outputs
Question: What do you produce or deliver in your work?
Covered when: at least one concrete thing they produce or deliver is named.

### Topic 7: stakeholders
Question: Who do you do your work for or with -- the people, teams, or clients you deal with?
Covered when: the most central party they name carries one concrete communication or
coordination task (running the weekly sync, writing the client update). If that party is a
bare label, ask once how they interact with them. Go deeper on one relationship, not wider.

### Topic 8: importantTasks (asked verbatim)
Question: Of everything you've described, which of those do you spend the most time on?
Covered when: they name at least one task that takes a lot of their time. This is about time,
not importance; a rough answer is enough.

### Closing question (asked once the guide is covered)
One last thing -- if someone shadowed you for a couple weeks, what would they catch you doing
that hasn't come up yet?
```

P3 is the system message of a call to GPT-4o mini at temperature 0 with a JSON-object reply; the user message lists the tasks covered so far and the latest answer. It runs on every turn, in parallel with the question planner, and once over all answers when a session is resumed.

Prompt 3: Per-Turn Task Extraction

```
{# system #}
Count the NEW, concrete, recurring WORK TASKS an interview answer reveals that are NOT already
in the covered list (compare meaning, not wording; ignore vague/non-task statements). Return
JSON {"tasks":["..."],"count":<int>}.


{# user #}
COVERED:
{% for task in covered %}- {{ task }}
{% else %}(none)
{% endfor %}
ANSWER:
{{ latest_answer }}

List only the genuinely new tasks.
```

P4 is the system message of a call to GPT-5.4 at temperature 0.4 with a JSON-object reply. The user message lists the remaining topics with their criteria, the conversation so far, the questions already asked, and the covered tasks. It runs on every turn while guide topics remain; the interview asks at most 17 and at least 5 questions.

Prompt 4: Coverage Judgment and Next Question

```
{# system #}
You run a work-task interview. Do BOTH of these in ONE step:

PART A -- COVERAGE: For each TOPIC listed, decide whether the conversation now ADEQUATELY
satisfies that topic's criteria. Be strict -- covered only if the criteria are genuinely MET by
what the participant actually said, not merely touched on in passing. EXCEPTION: if the
participant DECLINED or gave a non-answer to a topic ("skip", "nothing", "pass", "n/a", "don't
know"), mark it COVERED -- they won't engage, so it must not be re-asked.
SECOND EXCEPTION -- THE QUESTION DIDN'T LAND: if the participant asked what a question MEANS or
said it doesn't fit their field ("can you elaborate?", "I'm not sure what you mean", "I'd need
clarification", "that doesn't really apply to X"), you get ONE explanation -- see WHEN THEY ASK
WHAT YOU MEAN below for how to give one that works. If they say it again on that SAME topic,
mark the topic COVERED and move on: a second clarification request is not a coverage gap you
can close by asking again, and a third attempt reads as not listening. Judge this on the LAST
TWO turns of the topic, not on a single confused answer.

PART B -- NEXT QUESTION: Among the topics NOT yet adequately covered, pick the MOST VALUABLE and
ask the single best next question to advance it toward what its criteria still need. Keep
```

```
working the same topic across turns until it's covered; drill into concrete sub-tasks rather
than asking a shallow one-liner.

QUESTION RULES (hard):
- OPEN-ENDED -- invite them to describe their work in their own words; NEVER a yes/no stem ("Do
  you", "Can you", "Could you", "Are there", "Is there", "Have you", "Did you", "Would you") --
  and not buried mid-sentence either ("..., do you spend much time on X?"). The whole question
  must be answerable only by describing, never yes/no. Prefer an imperative or open-wh phrasing
  over a "Can you..." stem.
- NON-LEADING -- never name or hint at a specific task/answer you hope to hear; no examples or
  menus ("like X, Y, or Z").
- NOT REPETITIVE -- never repeat or rephrase an already-asked question (listed below), AND never
  ask about a task or activity the participant has ALREADY described anywhere in the
  conversation (even under a different question). Only open genuinely new ground or push
  meaningfully deeper.
- ACKNOWLEDGE & PIVOT ON A MISS -- if you're re-asking a topic because the LAST answer was
  off-target or incomplete (e.g. they named TOOLS, systems, or a vague label instead of the
  actual TASKS), do NOT repeat the same generic question. Briefly reference what they DID say
  (one thing, a few words) and ask specifically for the missing piece -- the concrete tasks
  (e.g. "we use Jira and Airflow" → "On the release side, what do you actually do start to
  finish?").
- CONVERSATIONAL -- like a warm, curious colleague chatting; plain everyday language,
  contractions fine; do NOT tack "in your role as a [job title]" or "as a [role]" onto the
  question. One sentence.
- VARY THE STRUCTURE -- the #1 tell of a robotic interviewer is question after question in the
  SAME shape. Each question must use a clearly different sentence structure and opener from the
  ones just before it; don't reuse the same template even when the topic changes. Before you
  write, glance at your recent questions and deliberately pick a different structure. When you
  refer back to something they said, ROTATE how you do it -- "Let's go back to X", "Coming back
  to X", "Earlier you said X", "About the X you mentioned", or just ask about X directly. "You
  mentioned X ..." is only ONE of those options, not the default -- don't lean on it, and never
  lead with the same referring-back phrasing two turns running. Rotate openers freely overall
  ("What goes into X?", "Tell me about the X side of things", "When you X, what happens next?",
  a brief lead-in, or just ask about X directly).
- BE CLEAR & PLAIN -- the participant should INSTANTLY get what you're asking. Avoid abstract
  framings, vague verbs, and hedgy time-scaffolding ("what do you turn out from your work over
  the course of a project or a normal stretch of time?" → just "What do you produce or deliver
  in your work?"). A plain "in your work" / "week to week" beats "over the course of a project
  or a normal stretch of time". Short, concrete, instantly understandable.
- DECOMPOSING A BROAD ACTIVITY -- when you break a broad activity (e.g. "writing code", "writing
  papers") into its pieces, ask DIRECTLY: fold the activity straight into the question and ask
  for the different KINDS of that work or its concrete steps ("What kinds of coding work do you
  do?", "What different kinds of writing come up?", "What does writing a paper actually
  involve, start to finish?"). Cut roundabout scaffolding like "the different kinds of work
  that make up that part of your week" -- name the activity and ask what kinds of it there are.
  Just make sure it's clear you're after the actual tasks/steps, not the topic.
- WHEN THEY ASK WHAT YOU MEAN, EXPLAIN -- DON'T REWORD. A clarification request ("can you
  elaborate?", "I'm not sure what you mean", "I'd need clarification for my field") means the
  FRAMING didn't reach them. Restating the same question with synonyms is what fails: "what do
  you produce or deliver?" → "what do you hand off, write, or otherwise deliver?" is the same
  question again, and gets the same blank. Instead: say plainly what you were after, DEFINE the
  abstract word you used, and put it in terms of the work THEY have already described. A real
  example that worked: "what comes up on a longer cycle?" → "I meant work that comes up less
  often than your usual week -- monthly, quarterly, or once a year?". Defining your own term is
  NOT leading; naming the answer you expect is. So explain the KIND of thing you're asking
  about, never the specific task you hope to hear.
- SKIP MEANS MOVE ON -- if their LAST answer was a non-answer or refusal ("skip", "nothing",
  "pass", "n/a", "don't know", or the like), they won't engage with that area: do NOT re-ask it
  from any angle -- move to genuinely different ground (or, in the emergent phase, set done).
- KEEP IT TIGHT & NATURAL -- when you reference what they said, name AT MOST one thing in a few
  words; never recap a multi-item list. Avoid stilted scaffolding like "What does your work
  involve around X" -- say it the plain way ("What do you do for X?", "What's the X side of your
  work like?"). To keep a guess non-presuming, a light tail ("..., if that's part of your work?")
  is fine -- don't reach for the same hedge every time. One clean ask, no piled-up trailing
  clauses.
- PLAIN, NOT FOLKSY -- natural means plain everyday language, not chatty idioms. NEVER use
  figurative expressions like "lands on your plate", "bread and butter", "wear many hats", or
  "day in the life" -- say it straight. And do NOT tack the participant's ROLE onto the
  question: "in your PhD work", "in your nursing work", "in your [role] work", "as a [role]"
  all read awkwardly since their role is already known. A plain "in your work" is fine; the
  role-specific version is not.

Return JSON {"covered":["<field of each adequately-covered topic>"],"gap":"<the uncovered area
you target this turn>","topic":"<the field this question
advances>","question":"...","done":<bool>}. Set done true only if EVERY topic is already
```

```
covered.


{# user #}
TOPICS -- field: text, with criteria:
{% for t in remaining_topics %}- {{ t.field }}: {{ t.text }}
   criteria: {{ t.criteria | join(' | ') }}
{% endfor %}
Conversation so far:
{{ history }}  {# Q:/A: pairs #}

Already-asked questions (do NOT repeat or rephrase):
{% for q in asked %}- {{ q }}
{% endfor %}
Tasks already covered:
{% for task in covered %}- {{ task }}
{% endfor %}
Judge coverage, then advance the most valuable uncovered topic with the single best next
question.
```

P5 is the system message of a call to GPT-5.4 at temperature 0.4 with a JSON-object reply; the user message carries the conversation, the tasks gathered so far, all questions asked, and the last two questions. It runs on every turn after the guide is covered. The interview ends when the model reports done, when two consecutive emergent turns add no task (after at least five questions), or at the 17-question budget.

Prompt 5: Emergent Gap-Driving

```
{# system #}
You are finishing a work-task interview. From the conversation, infer the participant's ROLE
and what they've already described, then:
0) PRIORITIZE THE PARTICIPANT'S MOST IMPORTANT TASKS. Your goal is to fully cover the tasks
   THIS participant signalled matter most -- the ones they named when asked what matters most,
   plus the ones they emphasize or spend the most time on -- before any peripheral area.
   DECOMPOSE THE CORE FIRST: look at the CENTRAL / most-important activities they ALREADY
   named. If a dominant one is still just a bare label -- "write code", "design experiments",
   "see patients", "write papers" -- with none of its distinct sub-tasks surfaced, that
   under-decomposed core IS the biggest gap. A dominant activity is worth SEVERAL tasks, not
   one: ask them to break it into the different KINDS of work it involves (e.g. "writing code"
   for a researcher → implementing models, running experiments, debugging, analyzing results,
   reviewing others' code). Drill the richest bare-label activity this way before moving on to
   peripheral missing areas -- a thin, single-task core is a worse gap than a missing minor
   area. ONE DECOMPOSITION PASS PER ACTIVITY: break a given activity into its kinds/steps ONCE;
   after that pass it is DECOMPOSED -- do NOT return to the same activity from another angle. An
   activity's sub-phases are the SAME activity, not new areas (e.g. for "user interviews":
   "before/during/after", then "setting one up", then "the conversation itself" are all ONE
   activity; for "writing a paper": "drafting", then "revising", then "the writing process" are
   all ONE). Spread depth ACROSS the participant's important tasks, not endlessly into one --
   once an important task has had its one pass, move to the NEXT important task or set done.
1) BUILD THE ROLE'S TASK INVENTORY -- but anchor it on their SPECIFIC described focus, NOT the
   generic job title. From what they've actually told you they do, list the recurring tasks
   someone in THAT specific role routinely does. CRITICAL GUARD: if their described work is
   narrower than the title suggests (e.g. a "BI Specialist" who is clearly
   RELEASE/PLATFORM-focused), inventory only the work they've SIGNALLED (deployments,
   monitoring, environment upkeep, access management) -- do NOT inventory generic tasks for the
   title (dashboard-building, data analysis) they've given NO sign of doing. Inferring from the
   bare title instead of their described focus is the #1 cause of asking about work they don't
   do.
2) DIFF & TARGET -- mark which inventory tasks are already gathered; among the unsurfaced ones
   (or bare-label ones per step 0), target the MOST IMPORTANT area that is PLAUSIBLY part of
   THEIR work given everything they've said. IMPORTANCE IS THE PARTICIPANT'S, NOT THE TITLE'S --
   weight by what THIS participant said matters most and by what they emphasize or spend time
   on, not by generic role-importance; make sure their most important tasks are fully covered
   and decomposed before spending a question on anything peripheral. Go by importance, not just
   "any gap".
   - NEVER RE-ASK AN AREA you (or any earlier question) already asked about -- even reworded
     from a new angle. An activity you have ALREADY decomposed once counts as already asked:
     its sub-phases (setup vs. during vs. after; planning vs. execution vs. follow-up) are NOT
     separate areas, so do not come back to the same activity to break it down again. If a
     prior question on an area got a THIN or vague answer, that means they don't have more to
```

```
     give there; take it as covered and move to a genuinely DIFFERENT area (or set done).
     Re-probing the same area twice (e.g. asking about study design / the research question two
     turns in a row, or user interviews three turns in a row) is the #1 repetition failure --
     check the already-asked list before you target.
   - GROUNDED & SPECIFIC, NOT VAGUE -- the area you target must be a SPECIFIC slice of work that
     connects to something concrete they ALREADY described, and your question must NAME it.
     Anchor it to a thing they said: they deploy between environments → ask what they do when a
     deployment fails or to catch issues before prod; they prepare Airflow schedules → ask how
     they keep those jobs running / what they do when one breaks; they review release requests
     → ask what they check for. Do NOT fall back to a vague catch-all like "what are your daily
     responsibilities?" or "what do your daily operations involve?" -- those re-ask the whole
     job and surface nothing new. If you can't name a SPECIFIC grounded area worth one
     question, set done.
   - GUARD AGAINST PRESUMING -- if an area is typical for the TITLE but they've shown NO signal
     of doing it, do NOT probe it; that's a guess that will miss. Only probe areas adjacent to
     what they've ACTUALLY described. Frame every question so "that's not really part of my
     job" is a perfectly natural answer.
   - COMMONLY-OMITTED CHECK -- run this BEFORE you set done, EVEN WHEN the gathered list already
     looks complete. Some tasks are NEAR-UNIVERSAL for a role yet routinely go UNMENTIONED
     because they feel like background rather than "real work" -- and they are DISTINCT tasks,
     NOT implied by what's already gathered. The PATTERN (not a checklist to read off): keeping
     current with the developments or methods the role depends on; sharing, presenting, or
     disseminating one's own work to others; evaluating or giving feedback on peers' work. If a
     near-universal, currently-unsurfaced task like this plausibly exists for THIS specific
     role, probe ONE before stopping. When such work is CENTRAL to the role it is NOT "generic
     filler" (see the filler rule's own exception below). STRICT LIMITS: NEAR-UNIVERSAL only --
     the presuming guard still fully applies to title-plausible work that varies by the
     person's focus; probe at most one or two such gaps, then set done.
   - BAN GENERIC FILLER. Do NOT probe surrounding-work areas that apply to almost any job --
     record-keeping, documentation, "staying up to date", general admin, tool/environment
     upkeep, generic "compliance" -- UNLESS that area is clearly CENTRAL to this specific role.
     For an individual contributor especially, those are usually marginal; skip them.
   - You have a LIMITED number of questions, so spend them on the highest-value gaps -- the
     participant's MOST IMPORTANT tasks first. Once those important tasks are well covered and
     only MINOR or peripheral areas remain, set "done": true and omit the question -- do NOT
     spend a question on something marginal.
3) Otherwise ask ONE question about that most-important gap.

QUESTION RULES (hard):
- OPEN AND NATURAL, NOT REPETITIVE -- ask an open question that invites them to describe their
  work in this area in their own words. The LAST TWO questions you asked are shown below; use a
  clearly DIFFERENT sentence structure and opener than those, so it doesn't read as a template
  (don't echo their wording or pattern).
- DON'T PRESUME HEAVY INVOLVEMENT -- the gap is a guess, so frame it so that "that's not really
  part of my job" or "not much" is a perfectly natural answer. AVOID "How do you handle ___?" --
  it assumes they do a lot of it.
- NON-LEADING -- name the AREA in plain words; never the specific task or examples you hope to
  hear.
- NO META / PEER-COMPARISON -- NEVER ask the participant to compare themselves to others or to
  name their own gaps ("how do your tasks differ from others in your profession?", "anything
  you do that they don't, or that they do that you don't?", "what's unique about your work?").
  That offloads YOUR diff-against-the-inventory job onto them and surfaces nothing concrete.
  You name the specific grounded area; ask about THAT directly.
- CONVERSATIONAL -- ask it the way a warm, curious colleague would in a chat, NOT like a survey:
  plain everyday language, slightly casual, contractions fine; no stiff "in your role as a
  [title]" tails or piled-up clinical noun-phrases. One sentence.
- VARY THE STRUCTURE -- the #1 tell of a robotic interviewer is question after question in the
  SAME shape. Each question must use a clearly different sentence structure and opener from the
  ones just before it; don't reuse the same template even when the topic changes. Before you
  write, glance at your recent questions and deliberately pick a different structure. When you
  refer back to something they said, ROTATE how you do it -- "Let's go back to X", "Coming back
  to X", "Earlier you said X", "About the X you mentioned", or just ask about X directly. "You
  mentioned X ..." is only ONE of those options, not the default -- don't lean on it, and never
  lead with the same referring-back phrasing two turns running. Rotate openers freely overall
  ("What goes into X?", "Tell me about the X side of things", "When you X, what happens next?",
  a brief lead-in, or just ask about X directly).
- BE CLEAR & PLAIN -- the participant should INSTANTLY get what you're asking. Avoid abstract
  framings, vague verbs, and hedgy time-scaffolding ("what do you turn out from your work over
  the course of a project or a normal stretch of time?" → just "What do you produce or deliver
  in your work?"). A plain "in your work" / "week to week" beats "over the course of a project
  or a normal stretch of time". Short, concrete, instantly understandable.
- DECOMPOSING A BROAD ACTIVITY -- when you break a broad activity (e.g. "writing code", "writing
  papers") into its pieces, ask DIRECTLY: fold the activity straight into the question and ask
  for the different KINDS of that work or its concrete steps ("What kinds of coding work do you
  do?", "What different kinds of writing come up?", "What does writing a paper actually
```

```
  involve, start to finish?"). Cut roundabout scaffolding like "the different kinds of work
  that make up that part of your week" -- name the activity and ask what kinds of it there are.
  Just make sure it's clear you're after the actual tasks/steps, not the topic.
- WHEN THEY ASK WHAT YOU MEAN, EXPLAIN -- DON'T REWORD. A clarification request ("can you
  elaborate?", "I'm not sure what you mean", "I'd need clarification for my field") means the
  FRAMING didn't reach them. Restating the same question with synonyms is what fails: "what do
  you produce or deliver?" → "what do you hand off, write, or otherwise deliver?" is the same
  question again, and gets the same blank. Instead: say plainly what you were after, DEFINE the
  abstract word you used, and put it in terms of the work THEY have already described. A real
  example that worked: "what comes up on a longer cycle?" → "I meant work that comes up less
  often than your usual week -- monthly, quarterly, or once a year?". Defining your own term is
  NOT leading; naming the answer you expect is. So explain the KIND of thing you're asking
  about, never the specific task you hope to hear.
- SKIP MEANS MOVE ON -- if their LAST answer was a non-answer or refusal ("skip", "nothing",
  "pass", "n/a", "don't know", or the like), they won't engage with that area: do NOT re-ask it
  from any angle -- move to genuinely different ground (or, in the emergent phase, set done).
- KEEP IT TIGHT & NATURAL -- when you reference what they said, name AT MOST one thing in a few
  words; never recap a multi-item list. Avoid stilted scaffolding like "What does your work
  involve around X" -- say it the plain way ("What do you do for X?", "What's the X side of your
  work like?"). To keep a guess non-presuming, a light tail ("..., if that's part of your work?")
  is fine -- don't reach for the same hedge every time. One clean ask, no piled-up trailing
  clauses.
- PLAIN, NOT FOLKSY -- natural means plain everyday language, not chatty idioms. NEVER use
  figurative expressions like "lands on your plate", "bread and butter", "wear many hats", or
  "day in the life" -- say it straight. And do NOT tack the participant's ROLE onto the
  question: "in your PhD work", "in your nursing work", "in your [role] work", "as a [role]"
  all read awkwardly since their role is already known. A plain "in your work" is fine; the
  role-specific version is not.

Return JSON {"gap":"<the specific missing task area you are targeting, or
'none'>","question":"...","done":<bool>}.


{# user #}
Conversation so far:
{{ history }}

Tasks GATHERED so far:
{% for task in covered %}- {{ task }}
{% endfor %}
Already-asked questions (do NOT repeat or rephrase):
{% for q in asked %}- {{ q }}
{% endfor %}
LAST TWO questions asked -- your question MUST use a clearly DIFFERENT sentence structure than
these:
{% for q in last_two %}- {{ q }}
{% endfor %}
Name the biggest gap in their task inventory and ask about it -- or set done if it's well
covered.
```

A.2 Task Generation and Verification

Tasks are first extracted from the whole transcript (`task-extract`), then consolidated and gap-filled as described in Section 3.1. Workers are instructed to review the proposed tasks against the work they personally perform, not the occupation as a whole (Figures 6d and 7a).

A.2.1 Prompts for Task Generation

P6 is the system message of a call to GPT-5.4 at temperature 0.7 with one JSON object per line; the user message gives the occupation title, the extracted tasks, and the full transcript. At most 10 additions are kept, de-duplicated against the consolidated list and tagged as inferred; an error skips this step.

Prompt 6: Task-Statement Generation: Gap-Fill (inferred tasks)

```
{# system #}
You are adding a FEW inferred "you might also do this" tasks to a participant's task list. They
have already described their job; the list of tasks they explicitly named is below. Your job:
add ONLY the additional recurring tasks they very likely do but did NOT mention, AND that you
can tie to something specific they said.

10 IS A HARD CAP, NOT A QUOTA. Most participants warrant only a FEW such tasks, and often ZERO.
Do NOT pad the list to approach 10 -- a short, well-anchored list is the goal, and returning
nothing is a valid and common answer. It is far better to return 1-2 tasks you are sure of than
to reach 10 by adding generic role-typical filler.

THE HARD RULE -- INFER FROM THIS SPECIFIC PERSON, NOT FROM A GENERIC JOB TITLE. Anchor every
task you add in something THIS participant actually told you -- and use BOTH:
  (a) their described WORK -- the responsibilities, weekly tasks, and activities they named; and
      (b) their stated BACKGROUND -- their seniority and level, the years and roles of
      experience they mention, the specific industries / domains / clients / institutions they
      name, and the full portfolio of practices they say they operate (e.g. someone who advises
      boards, spans consulting + teaching + coaching, or has 25 years in senior leadership).
      Use that background RICHLY: surface the tasks that a person with THAT background,
      operating at THAT level, doing THIS described work, very likely also does. A good
      addition is a likely NEIGHBOR, NEXT-STEP, or LEVEL-APPROPRIATE counterpart of what they
      described or who they are -- e.g. a senior leader who names board members and investors
      among their stakeholders very likely does board-level / governance advisory, even if they
      didn't list it as a task. PITCH AT THEIR LEVEL. Match the seniority and framing of their
      actual work: for a senior, strategic, or advisory professional the inferred tasks should
      be senior and strategic (shaping strategy, advising leadership/boards, governance), NOT
      junior execution. What to DROP: generic filler whose ONLY justification is "people with
      this title usually do this", with no anchor in who THIS person is or what they described.
      The test is specificity to THEM (their work AND their background), not avoidance of
      inference.
- ✓ Runs experiments and writes papers → "Respond to peer-review feedback and revise papers for
  resubmission." (next-step of submitting papers)
- ✓ Background: 25 years senior leadership, names board members/investors among stakeholders →
  "Provide board-level and governance advisory on strategic decisions to senior leadership."
  (implied by their seniority + stated board interactions)
- × Generic filler with no anchor in their work OR background: "Keep records up to date",
  "Attend company meetings", "Respond to email" -- DROP unless they specifically pointed at it.

CONSTRAINTS:
- MUTUALLY EXCLUSIVE -- each addition must be a genuinely DISTINCT activity: not a duplicate,
  restatement, kind/case, or subset of (a) any task already in their list, OR (b) any other
  task you add here. Tasks serving the SAME underlying activity or objective count as
  overlapping even when the channel, medium, or occasion differs (e.g. emailing parents vs.
  meeting parents is ONE task; routine coaching vs. competition-day prep is ONE task). When
  candidates overlap, keep only the single most important and drop the rest.

TASKS ALREADY IN THEIR LIST -- do NOT repeat, restate, or add a KIND/CASE of any of these:
{% for task in consolidated %}- {{ task }}
{% endfor %}- SPECIFIC and same shape as the rest: O*NET style, verb-led, 8-18 words, sentence
case, terminal period, plain language. Name the actual artifact / audience / output -- match the
specificity of their other tasks, do NOT go vague.
- MOST IMPORTANT ONLY -- QUALITY OVER QUANTITY. 10 is a CEILING, not a target. Rank candidates
  by how central they are to THIS person's role and return only the most important,
  well-anchored, mutually exclusive ones; if only two qualify, return two. Zero is fine. Before
  adding EACH task, ask yourself: can I point to the exact thing this participant said that
  makes this task likely? If not, DROP it -- never include a task just to reach 10. The moment
  you find yourself reaching for generic, role-typical tasks to lengthen the list, STOP and
  return only what you have.

OUTPUT: one JSON object per line (JSONL), no array, no commentary:
{"name": "..."}


{# user #}
Job: {{ occupation_title }}
TASKS THE PARTICIPANT EXPLICITLY MENTIONED (every one must be COVERED by exactly one task in
your output -- absorbed/merged as needed, never copied in verbatim or dropped):
{% for task in mentioned %}- {{ task }}
{% endfor %}

FULL INTERVIEW (their own words -- read it for background, seniority, domains, and stakeholders
like boards/investors that the task list above drops):
{{ transcript }}
```

```
Add only the inferred tasks they likely do but did NOT mention that you can tie to something
specific above -- at most 10, but fewer (or none) is expected. Do NOT pad to reach 10.
```

A.3 Task Processing

Each verified list is processed before aggregation. Long entries that a worker added or edited are split into single activities (`split`). A filter then keeps only valid tasks under O*NET's definition (P7), removing pure goals, outcomes, broad duties, skills, tools, vague statements, and off-topic text (`filter`). The remaining tasks are rewritten into action-on-object form without adding information (`normalize`), brand names are removed (`brand`), and exact duplicates within a worker's list are dropped.

A.3.1 Prompts for Task Processing

P7 is a shared text spliced into `filter` at its `{% include "task-definition.txt" %}` line.

Prompt 7: Task Definition (shared)

```
WHAT IS A TASK (the structural test). A task is a discrete, observable work ACTIVITY: a
concrete action verb + an object, together naming something a worker can be observed doing.
Coordination and communication activities DO count when a counterpart or named process is given
(attend a pre-shift crew briefing, participate in a site walkthrough, present findings to
stakeholders, review a repair estimate with a customer). A task MAY carry a goal/purpose clause
("... to ⟨goal⟩") -- that does not disqualify it; only a statement that is NOTHING BUT a goal is
excluded.

SCOPE (O*NET): a task is the SMALLEST UNIT OF ACTIVITY WITH A MEANINGFUL OUTCOME. This bounds
both ends -- too BROAD = not the smallest unit, a duty spanning many tasks ("Manage
operations"); too NARROW = no meaningful outcome on its own, a single step or keystroke ("Press
save", "Pick up a wrench"). These are NOT tasks:
  - a goal / outcome / state, when the whole statement is the goal ("Ensure jobsite safety",
    "Improve performance"). A concrete action + object that also names a goal is STILL a task
    ("Replace worn brake pads to restore stopping distance", "Recalibrate the packaging line to
    reduce waste" -- both tasks).
  - a broad duty or responsibility area ("Manage operations", "Own the customer relationship")
  - a skill, knowledge, or ability ("Proficient in Excel", "Strong communicator")
  - a tool or technology ("a forklift", "the scheduling system")

WEAK VERBS -- judge the LEAD verb only; a weak verb inside a "to ..." purpose clause does NOT
count (the lead verb decides -- "Audit records to ensure compliance" leads with "audit", a real
task). A lead verb is weak, with no concrete action following, when it is either:
  (a) RELATIONAL -- names a relationship to the work, not the work itself: manage, oversee,
      supervise, own, ensure, enable, be responsible for, support, be involved in, help with,
      contribute to (and, with a vague object ONLY, handle, drive, deal with, facilitate --
      "facilitate collaboration" is weak, but "facilitate a training session" or "facilitate a
      team meeting" is a real coordination task);
  (b) CONTENTLESS -- so generic it reduces to "do": do, work on, use. ("use" is weak only as a
      bare lead ("Use the scheduling system"); in "Use ⟨tool⟩ to ⟨action⟩" the real verb is
      ⟨action⟩ and the task is "⟨action⟩ using ⟨tool⟩" -- "Use routing software to plan delivery
      stops" = "Plan delivery stops using routing software", a task.)

SPECIFIC OBJECT -- a recognizable artifact, output, or category a worker would recognize
("invoices", "repair orders", "irrigation schedules", "wiring", "shipping manifests"). NOT
specific: the placeholders things, stuff, tasks, work, items, or a missing object; nor a bare
abstract process noun (automation, improvements, solutions, tooling, systems, processes,
services, functionality, changes, updates, operations) unless paired with a domain qualifier.

COMPOUNDS -- for an "X and Y" statement, judge the STRONGEST clause: if any clause is a discrete
task, the statement IS a task (it can be split downstream); do NOT reject the whole for a weak
lead verb -- "Manage and resolve customer complaints" is a task ("resolve customer complaints"),
"Audit and manage inventory records" is a task ("audit ... records").
```

A.4 Task Aggregation

Clustering. Each task in the pool is filed under at most one cluster or left unassigned (P11). Clusters with tasks from fewer than two workers return their tasks to the pool.

Synthesis. Each draft statement (P12) is checked for coverage of its member tasks (P13) and for writing defects (`writing-only`), and failing drafts are repaired (`synthesize-repair`). A statement judged to fuse two duties is split (P14).

Partitioning. Only statement pairs above a cosine-similarity threshold are judged for overlap (P15). The overlapping pairs form a graph, and each connected component goes, with its member tasks, to a model that merges statements describing the same work or nested within one another (P16). Statements that overlap but each cover distinct work are instead reworded to separate their scopes (`sharpen`).

Coverage and iteration. A statement covers a task when performing the task means performing the activity the statement names, in whole or in part (P13). Tasks that fail this check, and those left in the pool, go to a judge that sees the full list and either refiles them under an existing statement or returns them to the pool (P17). Rounds repeat on the pool while each yields new tasks reported by at least two workers, up to three rounds.

A.4.1 Prompts for Aggregation

P8 is a shared text spliced into P11, P14, P15, P16, `sharpen`, and P17 at their `{% include "task-dimensions.txt" %}` line.

Prompt 8: Task Dimensions (shared grouping doctrine)

```
WHAT DIVIDES ONE TASK FROM ANOTHER

A task inventory is a taxonomy: every entry exists because something distinguishes it from
every other entry. Below are the dimensions two pieces of work can differ along, and whether a
difference along each one earns a SEPARATE ENTRY or belongs INSIDE one as an example.

ONE TEST GOVERNS ALL OF THEM. Does changing this dimension change WHAT THE WORKER DOES -- the
steps, the judgement, the skill it takes? Not what the thing is called, not who is served, not
when it happens. If the work is the same work, it is one entry however unlike the two instances
look.

CUT HORIZONTALLY: THE ACT IS THE ENTRY, NOT THE LARGER JOB IT SERVES. Where the same act -- the
same steps, the same judgement, the same skill -- recurs inside several of the occupation's jobs
or workstreams, it is ONE entry across all of them, with those jobs as its examples. Do not
bury an act's instances as steps of the larger duties they feed: for an electrician, testing
circuits and recording the readings is one entry whether the test belongs to a new
installation, a fault call, or a scheduled safety check -- the installation and fault-call
entries keep only the work that is theirs alone. A cut that gives each larger job its own copy
of a shared act hides the occupation's most widespread work inside vertical bundles, and the
bundles then read as overlapping. When choosing between naming the recurring act once and
distributing it into the jobs it serves, name the act.
THE HORIZONTAL CUT NEVER CROSSES A DIVIDING DIMENSION. It gathers INSTANCES of one act across
jobs; it does not gather DIFFERENT acts under one banner. The two sides of an exchange are two
acts -- each side is named once across all jobs, never one entry spanning both sides; and acts
split by a dividing trigger stay split, each cut horizontally on its own.

THESE DIVIDE
  · THE ACT. Different verbs on the same object are different entries -- inspecting a roof and
  repairing one, drafting a document and approving one, loading a machine and servicing it. ·
  THE SIDE OF AN EXCHANGE. Issuing a request and answering one received, sending work out for
  another party's decision and deciding on work they send in, ordering supplies and filling an
```

order that arrives -- these are different acts EVEN WHEN THE SAME WORKERS routinely do both. Each side has its own trigger (the counterpart's move arrives on its own schedule) and its own product, so a shared plan -- "handle the exchange end to end" -- does not unite them.
AN EXCHANGE NEEDS AN OPPOSING OR INDEPENDENT PARTY, one acting in its own interest: a request served on them, their demand answered, their submission ruled on. Question-and-answer inside the worker's own operation is not an exchange -- answering staff questions, directing their work and unblocking their tasks are the upkeep of one supervisory responsibility (the CHORES bullet below). · WHAT THE RECEIVER MUST BE TOLD, where it changes what the worker produces. Persuading one receiver, seeking a second's approval before the work can proceed, informing a third after the fact -- those are three acts, and they stay three whether the telling is written, spoken face to face, or called in over a radio. It is the verb that divides them, not who receives the work: reach for the act, not for the audience.
· WHAT IT TAKES TO PRODUCE IT. Where two outputs demand DIFFERENT JUDGEMENT, they are two acts: an output that carries the worker's own analysis or diagnosis -- reasoning only someone trained in the work could produce -- and one that relays status or logs what happened are different work, however alike "writing things up" or "keeping records" makes them sound. The medium is not the point: a written report, a spoken handover and a tag hung on a machine can each carry either. Likewise, entries that record the WORK ITSELF divide from entries that run the workplace's own business -- time, supplies, money owed. This is NOT the old output-function rubber stamp: two outputs taking the SAME judgement (two kinds of binding instrument, two kinds of filing, two sizes of the same fitting) remain one act with the kinds as examples.
· THE TRIGGER, WHERE IT CHANGES THE ACT. What prompts the work can change the work. A TRIGGER THAT IS MERELY THE ARRIVAL OF A JOB TYPE OR CUSTOMER SEGMENT IS THE SUBJECT MATTER, NOT A TRIGGER -- phrasing a segment as "when such jobs arrive" does not make it divide. For a maintenance technician, attending a machine because a fault was reported and going round the machines to a schedule are two entries: one starts from a complaint and ends when it is resolved, the other starts from a date and ends when the round is done, and the findings feed different decisions.

THESE DO NOT DIVIDE -- they belong inside an entry, as its examples
· THE COUNTERPART'S IDENTITY. Which customer, which agency, which colleague, which body -- and equally whether the person advised is a walk-in customer, a fleet manager, or an insurer, which is usually the EMPLOYER differing rather than the work. The same act for a different person is the SAME entry, and one entry may name several counterparts in a line: explaining a repair estimate to the vehicle's owner and to the workshop manager is one entry, not two.
· SUBJECT MATTER, FIELD, JOB TYPE, CUSTOMER SEGMENT. The act is unchanged by what it is applied to, and workers specialising in different subjects perform the same activities on them.
AND THE TRADE'S NAME FOR THE ACT CHANGES WITH THE SUBJECT -- do not be fooled by it. Every specialism dresses a shared act in its own vocabulary, so the same moves arrive wearing different names, each setting naming its own equipment, its own paperwork and its own occasions. Strip the vocabulary and ask what the worker DOES, move by move; where only the subject and its trade name change, the versions are ONE entry with each setting's subjects as its examples. An inventory that keeps every setting's renamed copy publishes the division of labor, not the work.
· THE KIND OF ARTIFACT. One act of composing, checking, or handing on, applied to many kinds of product, is one entry with the kinds as its examples -- however differently the products are named and used.
THE OUTPUT'S FORM AND DESTINATION ARE PART OF ITS KIND: a formal written instrument and an outline prepared for live use, one filing form and another, a part fitted on a new build and the same part fitted as a replacement, are kinds of the same made thing where producing them takes the same judgement. Only a difference in the JUDGEMENT of producing it -- different moves, not a different template or a different size -- marks two acts.
· THE STAGE IN A PROCESS, for the same act BEFORE THE SAME COUNTERPART. The check that opens a job and the one that signs it off, both reported to the same recipient, take the same moves. A stage is a property of the job, not of the work. But check the recipient first: if the later point in the process means a DIFFERENT authority applying a different standard, that is a counterpart and it divides. · THE TOOL, SYSTEM, SETTING, OR MEDIUM.
· THE CHORES OF ONE ONGOING RESPONSIBILITY. Where a worker is answerable for one continuing thing -- a helper's work, a budget they hold, a stock of supplies, the records of their own jobs -- the acts its upkeep demands are ONE entry, even though each is articulable, has its own occasion, and would pass an act test taken alone: recording entries and later chasing the ones that have not resolved, directing the day's work and periodically appraising it, counting what is on hand and reordering what has run down. No one of these acts is anybody's job on its own: asked what they do, a worker names the responsibility -- "I keep the stores", "I supervise the helpers" -- and the acts are how it is discharged. KEEPING ONE SET OF RECORDS is the case this bullet most often decides: entering what happened, reviewing what is outstanding, and reconciling or reporting the totals are its chores, however many kinds of record the set contains and however far apart the entry and the review fall.
TEST: write the act alone as an entry. If it reads as a duty someone could hold apart from the responsibility, it divides; if it reads as a chore of the responsibility, it stays inside.
THIS DOES NOT REACH A DELIVERABLE AND ITS AFTERLIFE: producing a thing others use and

```
    maintaining it later are still two entries.

A TRIGGER IS NOT A STAGE. A trigger is the event that makes work start -- a fault reported, a
request received, a date arriving; a stage is a position in a sequence the work runs through. A
different trigger often means a different act; a different stage almost never does.

OCCUPATIONAL HETEROGENEITY SETS THE LEVEL OF CLOSE CALLS. An occupation spans workplaces and
specialisms, and two alike-looking entries are often one act refracted through that
heterogeneity. Where the difference between them tracks the occupation's DIVISION OF LABOR --
different settings each doing their variant -- they are ONE entry at the covering abstraction,
the variants its examples; splitting there publishes specialisms, not tasks. Corpus worker
sets, where known, measure this: disjoint
workers say the settings divide the act and it merges up.

WHEN DIMENSIONS DISAGREE
  · THE ACT WINS over names, artifacts, and stages: a difference in what the worker does is
  decisive; a difference in what the product is called, or where it sits in a process, is not.
  · THE SIDE OF AN EXCHANGE WINS over a shared plan or phase, and over the CHORES bullet: an
  exchange with an opposing party divides even where one worker handles it end to end. · THE
  CHORES OF ONE RESPONSIBILITY WIN over the act test taken alone: acts that are all the upkeep
  of one responsibility stay one entry however cleanly each can be articulated.
```

P9 is a shared text spliced into P12, P16, and `sharpen`.

Prompt 9: O*NET Statement Forms (shared)

```
FORM -- pick the shape that best fits the content. Do NOT default to "such as"; leaning on it
for every statement is a formulaic tell, and real O*NET lines VARY: · BARE:          "<verb>
<specific object>"                          -- when the object alone already says it · OBJECTS:
"<verb> <A>, <B>, or <C>"                                -- alternative objects (Water lawns, trees,
or plants) · METHOD:       "<verb> <object>, using <means>"                    -- when the means
is the informative part · PURPOSE:      "<verb> <object> to <result>" / "for <result>"     --
when a shared result matters · STANDARD:     "<verb> <object> to ensure <standard>"
-- ONLY when the ensured thing is a SEPARATELY VERIFIABLE condition that could be FALSE while
the activity is still performed (a code or policy it must conform to, a checkable state of the
object). "Inspect fire-safety equipment to ensure it meets code" -- equipment can be inspected
and still fail code, so the clause draws a real line. If the "standard" merely restates the
activity's own aim ("coordinate deliveries to ensure timely delivery"), use BARE instead. ·
PEOPLE/TOPIC: "<verb> with <whom> about <topic>, <topic>, and <topic>" · EXAMPLES:     "<verb>
<object>, such as <kind> or <kind>"          -- ONLY for genuine example KINDS of a category
head, and TWO OR THREE of them, never more

THESE ARE PARTS, NOT SEVEN EXCLUSIVE TEMPLATES. The normal shape of an O*NET line is an object
slot plus AT MOST ONE modifier clause: "<verb> <A>, <B>, or <C> to <result>", or "<verb>
<object>, such as <kind> or <kind>, to
<result>". Take the object slot you need -- bare, a short list, or a category with examples --
and add at most one modifier: method, purpose, or standard -- OR NONE, WHICH IS THE COMMONEST
SHAPE OF ALL.

START BARE AND ADD NOTHING YOU DO NOT NEED. The object slot on its own is already a complete
task statement, and more than a third of real O*NET lines are exactly that: a verb, an object,
a full stop. A modifier is not the finishing touch that makes a statement look finished -- it is
a claim that a reader could not tell WHAT the work is without it. Write the bare sentence
first. Keep a modifier only if removing it would leave the activity genuinely unclear, and
delete it whenever the object already says the thing. "Water lawns, trees, or plants" needs
nothing after it; "to maintain healthy growth" only restates why anyone waters anything. A
purpose clause that any reader could have supplied is padding, and padding is the most common
way these statements go wrong.

NEVER TWO MODIFIERS. "Review records, using case-management software, to decide next steps
before each hearing" stacks means, purpose and occasion, and the sentence stops being readable.
Pick the one modifier carrying the most information -- usually the purpose, which is the
commonest modifier in real O*NET lines by a wide margin -- and if none of them carries much,
drop them all. Let the phrasing be natural.

## HOW THE SENTENCE IS BUILT

- ONE NATURAL, GRAMMATICAL SENTENCE. If it contains a list, the items must be PARALLEL -- the
  same grammatical form, each self-contained. Prefer noun phrases over gerunds ("permits,
  inspections, and site surveys", not "pulling permits, doing inspections").
- ONE TASK, AT MOST TWO CLOSELY-LINKED VERBS -- a single action, or two facets of one activity
  ("review and approve", "diagnose and repair"). NEVER a 3-or-more-verb compound: "mow, edge,
```

```
  and fertilize lawns" is "maintain lawns"; "analyze, build, and report metrics" keeps the ONE
  dominant verb or their common generalization.
- NEVER GRAFT A SEPARATE ACTIVITY ON AS A TRAILING MODIFIER. "...interpreting diagnostic
  trouble codes using a compression tester" fuses reading codes with running a compression
  test. A tail may only elaborate HOW the head's own action is carried out. If an activity does
  not fit cleanly, drop it.
- TWO OR THREE EXAMPLES AFTER "SUCH AS", NEVER MORE. "Such as" says the list is ILLUSTRATIVE,
  so a long one contradicts its own opening: seven items is an inventory wearing an
  illustration's clothes, and a reader takes it as a claim that the set is complete. Name the
  two or three a practitioner would recognise as the most typical and stop. Nothing is lost by
  leaving the rest out -- "such as" has already said the list is partial, which a bare list of
  objects has not.
- NO "BY <VERB>ING" AND NO "THROUGH <VERB>ING". A means clause is a NOUN PHRASE after "using"
  -- never a gerund clause. Write "Inspect residential electrical panels, using circuit testers
  and thermal imaging", not "Inspect residential electrical panels by testing circuits,
  scanning for heat, and logging readings". A "by ...ing" tail invites a chain of sub-steps: it
  turns one task into a procedure, buries the activity behind its method, and reads as a
  workflow rather than a duty. IF THE METHOD IS THE INTERESTING PART, MAKE IT THE
  VERB -- "Test circuits and scan panels for overheating" says more than "Perform panel
  inspections by testing circuits and scanning for overheating".
- NO BRAND, PRODUCT, OR PROGRAMME NAMES. Write what the thing IS, not what it is called: "a
  spreadsheet application", not the vendor's name for one; "a powered pallet truck", not the
  manufacturer's model; "a federal nutrition-assistance programme", not its initials. A trade
  name reads as a claim that the work depends on that one product, it dates the statement, and
  it localises a duty that exists wherever the same work is done under another name.
  THE TEST: would a worker doing this job in another state, country, or firm read the sentence
  as describing THEIR work? If the name is the thing that makes them say no, replace it with
  the category it belongs to. This holds even where the name has become the ordinary word for
  the object -- a genericised trademark is still a trademark, and the generic term always
  exists because that is what the trademark was coined against.
- SENTENCE CASE, ONE SENTENCE. Capital letter, imperative verb first, nothing else capitalised
  except proper nouns the rule above still permits, full stop at the end (acronyms are written
  out in full -- see 2.2). Never Title Case, never open in lower case, never lead with a
  gerund.
- PRESENT TENSE, IMPLIED PLURAL SUBJECT ("Prepare reports", not "Prepares reports"). Say "to
  <result>", never "in order to"; prefer "or" over "and"; no semicolons, forward slashes,
  "and/or", or "etc."; spell out any acronym.
- LENGTH: aim for ~15 words. Go to 25 only when the activities genuinely span several DISTINCT
  KINDS the statement has to name -- never pad, never exceed 25.

## HOW MUCH TO ENUMERATE

- TWO KINDS OF LIST, TWO DIFFERENT LIMITS. Do not confuse them:
    · ILLUSTRATIONS -- a "such as A, B, or C" list of EXAMPLES. Cap 3, each ONE broad KIND
    rather than a raw instance ("brake, suspension, and exhaust repairs", NOT "brake pads,
    rotors, calipers, and hoses"). These are a sample; a fourth adds nothing and the reader
    knows the list is partial. · OBJECT / SCOPE -- the things the work ACTS ON, ASSESSES,
    PRODUCES, or the parties it runs BETWEEN. Up to 5, and name every kind whose omission would
    misrepresent the work. This is CONTENT, not verbosity: "assess vehicle condition from road
    tests, diagnostic scans, fluid checks, tire measurements, and driver reports" is five
    objects and all five are the work; cutting to three would say the mechanic does not
    road-test. When you must shorten, cut ILLUSTRATIONS first, then means and tools.
- ONE LIST PER STATEMENT. NEVER both a multi-item object list AND a "such as" list: "Prepare
  reports, analyses, statements, packets, or presentations, such as variance analyses,
  management updates, or dashboards" is wrong twice over. Pick ONE place to enumerate -- either
  the object names the kinds, or it names the category and the "such as" list gives them.
  Moving items from one into the other is not a fix; the total is what is capped.
- SPEND THE ITEMS ON OBJECTS, NOT INPUTS OR TOOLS. Name WHAT THE WORK ACTS ON or PRODUCES, not
  the material it consumes or the software it uses. "Prepare breads, pastries, and celebration
  cakes" beats "Prepare baked goods, using flour, yeast, and proofing schedules": the second
  spends every slot on inputs and leaves the reader not knowing what is produced. Name inputs
  or tools only if slots remain.
- KEEP SHARED SPECIFIC KINDS -- over-generalizing is a failure, not a virtue. Where genuine
  object KINDS or output/decision TYPES are shared, keep them ("such as sedans, trucks, or
  motorcycles", NOT "vehicles"; "to approve, deny, or escalate", NOT "to decide"). An enabler
  ("using <means>") must characterize the WHOLE set -- if the means differ, drop it rather than
  name one instance.
- A PURPOSE OR ENABLER CLAUSE MUST APPLY TO EVERY ITEM IT FOLLOWS. "Order parts and shelve
  deliveries to keep the bins stocked" is WRONG -- ordering does not stock the bins. If the
  clause is true of only one item, drop the clause or drop the items it does not cover.

## WHAT THE WORDS MUST NOT BE

- NO PLACEHOLDER NOUNS. "items", "information", "data", "materials", "activities", "content",
  "documentation" name nothing. Every noun must be a KIND a practitioner would recognise:
```

```
  "replace brake pads, rotors, or calipers", NOT "replace parts or components". The same goes
  for invented compounds ("installation controls", "readiness criteria") -- if you cannot point
  to the wording that supports a term, do not write it.
- NEVER A VAGUE UMBRELLA ("perform duties", "provide services", "handle matters", "manage
  operations"); name the concrete kind of work.
- NO INDUSTRY, EMPLOYER, CLIENT TYPE, PRODUCT, BRAND, VENDOR, OR PLATFORM NAME, and no "for
  <field>" / "in <area>" / "such as <subfield>" tail naming which specialty the work happens to
  serve. Name the generic tool category or drop it ("using spreadsheet software", not the
  product). The statement has to read for EVERY worker in this occupation. For a welder, "Weld
  shipyard hull seams" is "Weld structural seams" -- and if removing the sector word leaves the
  statement saying less, that word was doing the OBJECT's job, so name the object better.
- A DELIVERY CHANNEL IS NOT A MEANS. "by phone, by video, in person, by email, in writing" is
  HOW a conversation reached someone, not what the work is, and the same judgement made in a
  meeting and made over the phone is one task -- so naming the media identifies nothing while
  spending room a purpose clause needs. The METHOD shape is for a means that changes the work
  (an instrument, a technique, a class of tool), not the medium a message travelled through.
  Say who the work is done WITH and what it is FOR instead. Keep a channel only when the medium
  IS the work.
- NAME PEOPLE AS PEOPLE. When the work is done WITH or FOR people, say so ("counsel clients
  about ..."), do not flatten them into a detached noun ("client information").

## CHOOSE THE MOST SPECIFIC VERB THAT COVERS THE WORK

A carrier verb -- perform, conduct, provide, manage, handle, coordinate, carry out, do,
undertake, give, exchange, ensure, oversee, support, facilitate -- carries no action of its own.
Whether it is acceptable depends entirely on the OBJECT it takes, and THE TEST IS WHETHER THE
OBJECT IS A DISGUISED VERB.

FORBIDDEN -- the object is a nominalisation, so the real verb is already sitting there. Promote
it. For a mechanic, "perform brake inspections" is "inspect the brakes"; "give a quote" is
"quote the repair"; "conduct a road test" is "road-test the vehicle". For a baker, "conduct
dough proofing" is "proof the dough". If you can turn the object back into a verb, you must.

ACCEPTABLE -- the object is a genuine CLASS of work that is not itself an action word, and the
carrier verb is how practitioners speak. For a mechanic, "perform scheduled maintenance" is
fine: maintenance is the recognised name of a body of work, not a verb wearing a noun ending.

So the question is never "is this verb generic?" but "does the object already contain the
verb?" If it does, rewrite. If the object is a real class of work, a carrier verb is correct --
do not force a narrower one that covers only part of it.

Examples (UNRELATED occupations -- copy the FORM only, never the content -- and note the VARIED
shapes):
- bare:      "Sharpen chef's knives" / "Sharpen paring knives" -> "Sharpen kitchen knives."
- objects:   "Water the lawn" / "Water the shrubs" / "Water potted plants" -> "Water lawns,
  shrubs, or potted plants."
- method:    "Log hours in the tracker" / "Check the timesheet against the roster" -> "Track
  labor hours, using scheduling software."
- standard:  "Check smoke alarms" / "Test the extinguishers" -> "Inspect fire-safety equipment
  to ensure it meets code."   (the code is verifiable and can fail -- that is what licenses "to
  ensure")
- such as:   "Change tires" / "Jump-start batteries" / "Unlock cars" -> "Provide roadside
  assistance, such as tire, battery, and lockout service."
```

P10 is a shared text spliced into P12, `writing-only`, and `sharpen`.

Prompt 10: Writing Principles (shared)

```
WRITING PRINCIPLES FOR A TASK STATEMENT. These govern HOW the statement is written -- not what
work it describes, and not who owns that work, which are separate questions settled elsewhere.

## 1. VERB SELECTION

1.1 THE VERB MUST NAME THE ACTUAL AUTHORITY RELATION. "Advise" implies non-binding expertise
supplied on request; use it only where the worker has no authority over the outcome. For
downward or peer-directed communication, the verb is "direct", "instruct", "assign", or
"update".
    TEST: name the recipient, then ask whether the worker can be overruled by them. If the
    recipient is a client, a court, a board, or anyone the worker serves, "advise" is right. If
    the recipient is the worker's own staff or peers -- anyone who takes the worker's word as
    the operative instruction -- pick one of the four verbs above. The tell is a recipient
    INSIDE the worker's own operation. 1.2 THE VERB MUST NOT UNDERSTATE THE DELIVERABLE. If the
```

```
    worker produces the change, the verb is "develop", "implement", or "revise". "Recommend"
    terminates the worker's involvement at the proposal and is wrong wherever the work
    continues past it.
    TEST: ask who acts next. If the sentence ends at a proposal and SOMEONE ELSE would have to
    act for the work to be finished, "recommend" is right. If nobody else appears -- if the
    worker is the one who would carry it out -- the verb has stopped short of the deliverable.
    Say what they produce. 1.3 THE VERB MUST NOT CLAIM AN OUTCOME THE WORKER DOES NOT OWN.
    Verbs naming an organisational or political result -- advance, secure, win, drive --
    attribute a multi-party outcome to one worker. Re-anchor the verb on the worker's own
    contribution and demote the outcome to the purpose clause. 1.4 PREFER THE
    LOWEST-ABSTRACTION VERB THAT IS STILL OCCUPATION-STANDARD. An umbrella verb is acceptable
    only if a practitioner at any employer would use the term unprompted. Otherwise decompose
    into the concrete action.
    THE USUAL OFFENDERS NAME NO ACT AT ALL: "manage", "handle", "conduct", "oversee", "deal
    with", "pursue", "ensure", and the synonyms of "do" -- "perform", "execute", "use",
    "utilize". Each is a container for whatever follows it, which is why statements led by them
    always grow an enumeration behind -- the vague head forces the tail to do the work. Replace
    the container with what the worker actually does; if no single concrete verb covers the
    activities, that is evidence of TWO tasks, not license for an umbrella. 1.4a LEAD WITH THE
    RESPONSIBILITY, NOT ITS FIRST CHORE. Where a statement covers the several acts that upkeep
    one continuing responsibility -- opening a new file, running a stocktake, supervising staff,
    keeping one set of books -- the HEAD VERB names the responsibility, and the chores follow in
    AT MOST ONE subordinate participial clause carrying a noun list ("Open new tenancy files,
    preparing reference checks and inventory records..."), never as a flat list of coordinate
    verbs with the responsibility buried or absent, and never as stacked "-ing" clauses --
    1.4b's one-clause cap holds for this form too.
    TEST: read the head verb and object alone. If they name a step ("send reference requests")
    rather than the thing the worker is answerable for ("open new tenancy files"), re-anchor
    the sentence on the responsibility.
1.4b GERUNDS ARE A LAST RESORT. A statement carries AT MOST ONE participial ("-ing") clause.
Chores and steps beyond that are DROPPED -- the members carry the detail, and a statement need
not enumerate its chores -- or written another way. DROP THE CHORES, KEEP THE TEXTURE: the
trimmed statement keeps a short tail naming the object kinds, the counterpart, or two or three
OCCASIONS the duty serves ("for X, Y, or Z", "such as A or B"). A statement stripped to a bare
verb and object has been over-trimmed -- the target register is a finite verb (or two), an
object list, a findable counterpart, and an occasions tail: a genuine second step takes a
finite verb ("Prepare X and monitor Y"), an enumeration becomes a "such as" NOUN list, an
outcome goes in the purpose clause. Stacked gerunds are the enumeration defect wearing grammar
-- two or more "-ing" clauses in one statement is the same smell as a verb chain, whatever
introduces them. ONE "by X-ing" MEANS CLAUSE IS ALLOWED. Where the method carries the act's
judgement or skill, a statement may carry one "by"-clause naming it -- the standard O*NET shape,
and it counts as the one permitted participial, not a stack, even when it coordinates TWO
gerunds ("by taking X and reviewing Y"). TWO GERUNDS IS THE CEILING: three or more in the
clause is the enumeration defect wearing "by" -- keep the one or two moves that carry the
judgement; the members hold the rest. The clause names the means of the MAIN verb; a gerund
that is its own errand is a second duty and stays out. 1.4c A STATEMENT IS ABOUT 15 WORDS. That
length holds one verb, one object list with an example or two, and one clause -- which is all
one duty needs. A statement pushing past ~20 words is carrying a second duty, a redundant
clause, or an enumeration that belongs in a "such as" list cut to two or three items. Trim to
what a reviewer can take in at a glance; the members carry the detail. 1.4d THE VERB MUST NAME
THE WORK, NOT THE MANDATE. "Represent", "serve", and "act for" name the RELATIONSHIP under
which work happens, not the work: everything an agent does for the principal who engaged them
is representation, so a represent-led sentence says only who the worker was for. Lead with the
concrete act performed under the mandate and demote the mandate to the purpose clause: "Survey
boundaries, stake corners, and file the plat TO ACT FOR owners in a dispute" names the work and
keeps the relationship. EXCEPTION: where standing in the role is itself the act -- entering an
appearance, covering a hearing where presence is the duty, or carrying a matter through an
engagement whose span is the task ("from data calls to negotiated findings") -- the role verb
stands. 1.5 THE VERB MUST NAME THE WORK, NOT THE FORUM IT HAPPENED IN. "Meet with", "discuss",
"check in with", "attend" and "talk to" name an occasion, not an accomplishment -- they tell a
reader where the worker was, not what they did or what changed. Name the act the occasion
exists to perform and let the setting go, or demote it to an occasion clause where it is what
makes the task distinct. For a building surveyor, "meet with contractors about defects" names
the meeting; "agree remedial work with contractors" names the duty.
    TEST: ask what is different afterwards. If the only answer is "the conversation happened",
    the verb has named the forum. A verb that survives names something checkable -- a decision
    reached, an instruction given, information obtained, an approval granted.
    NOT THIS: a conversational duty is not banned. Where the talking IS the work, use the
    specific verb for it -- "interview", "negotiate", "counsel" -- not the generic container.

## 2. VOCABULARY

2.1 NO HOUSE JARGON. The test: would this term appear in a job posting from a different
employer, using a different methodology, in a different sector? If not, replace it with the
plain equivalent. 2.2 NO CLIPPED OR ABBREVIATED FORMS, AND NO BARE ACRONYMS. Write the full
```

```
word, and write out what the initials stand for.
    TEST: a CLIPPING is one word with its end cut off -- restore the letters and you recover
    ordinary English ("admin" -> "administrative", "spec" -> "specification"). An ACRONYM is
    the initials of several words. Both are written out; they differ only in how.
    NO EXCEPTION AND NO PARENTHETICAL. Not "<expansion> (<acronym>)": the initials add length
    where length is scarcest and tell a reader who already knew them nothing. Where the
    expansion alone reads badly the term is house jargon under 2.1 and the fix is a plainer
    phrase, not a shorter one. THERE IS NO EXEMPTION FOR NAMES. A named method, standard,
    programme or statute whose initials stand for words is written out like any other. Nor for
    clipped names: "C-section" is "caesarean section". The only question is whether the short
    form stands for something longer. A CLIPPING INSIDE A COMPOUND IS STILL A CLIPPING, and the
    compound is where it hides, because it reads as the name of a thing rather than as a
    shortened word: "app store", "in-app purchase", "admin panel", "spec review", "config
    file". Restore the letters.
    NOT A DEFECT: a capital that stands for no word at all -- "X-ray", "type 2", a model
    number. There is nothing to expand.

2.3 TERMS MUST BE NATIVE TO THE OCCUPATION. A term borrowed from an adjacent field signals the
statement was drafted from the wrong template. Replace it with the occupation's own term.
    TEST: take each noun in turn and ask whose vocabulary it belongs to. A noun naming another
    trade's subject matter -- its livestock, its castings, its permits, its tolerances -- is
    borrowed even when the surrounding sentence is impeccable, and even when this occupation
    sometimes encounters the thing. Name what THIS occupation tracks about it instead.
2.4 STATEMENTS MUST BE COMPLETE. No mid-word or mid-clause truncation. Every statement is a
grammatical sentence ending in a period.
2.5 A TERM MET ONLY IN ONE SPECIALTY DOES NOT BELONG IN AN OCCUPATION-LEVEL STATEMENT, whatever
slot it fills -- object, artifact, counterpart, procedure, statute, system, or setting. The
statement describes the occupation; a term only one practice area's practitioners would use of
their own work narrows the statement to that subfield and silently excludes everyone else.
    TEST: ask which practitioners could use this term of their own work. If only those in one
    practice area, sector, or client base could, replace it with the covering term at the
    occupation's level. AN ENUMERATED LIST OF SPECIALTY TERMS IS THE SAME DEFECT IN PLURAL, and
    the commoner shape, because it looks like coverage: each item is one practice area's
    instance, and the list silently excludes every specialty not on it.
    THE FIX KEEPS THE SHARED CORE BY NAME AND COVERS THE REST WITH ONE OPEN TAIL. An item every
    practitioner recognises stays; the specialty items fold into one covering phrase ("and
    other property owners", "or other regulated filings"). Where no item on the list is
    shared-core, the covering term replaces the whole list. The covering term must still be
    concrete -- a role, an artifact kind, a category a reader can point to -- never a collective
    that names nobody (2.1 and 5.2 still apply).
    NOT THIS: a concrete example inside a shared-core list is not a specialty term. "Such as"
    examples every practitioner recognises make a statement better (3.1a); the test is whether
    the term is met only inside one subfield, not whether it is specific.

## 3. SCOPE -- ONE TASK, ONE ACTIVITY

3.1 OBJECT LISTS MUST BE CO-HYPONYMS. Every item in an "A, B, or C" list must name a thing the
SAME activity is applied to. If one item is produced by a genuinely different process, it
belongs in its own statement.
    TEST: substitute each item into the head on its own and ask whether the worker is doing the
    same thing each time. Same activity on different instances is ONE list, however varied the
    items look and however differently each reaches its recipient. A different activity splits
    out. Do NOT additionally require the items to match on workflow and on counterpart: that
    three-way test fails almost every real list and is the main source of wrong splits. 3.1a A
    LIST OF NEAR-SYNONYMS IS ONE ITEM. Co-hyponymy is necessary, not sufficient. Where two or
    three items name the same thing at slightly different scope or duration, they cost length
    and carry no information, and the sentence starts to read like a form rather than like the
    work. Collapse them to the plainest one and spend the words on a concrete example instead:
    one example a practitioner would recognise says more than three abstract near-equivalents.
    TEST: read the list in order and ask what item 2 tells a reader that item 1 did not. If the
    only answer is that it can be bigger, longer-term, more formal or more detailed, it is the
    same item twice -- keep one.
    NOT THIS: this does not license dropping a genuinely different object to shorten a list.
    The items being collapsed have to be the same thing described twice, not two things the
    same activity is applied to. 3.1b OBJECT LISTS JOIN WITH "OR", NOT "AND". THE STATEMENT
    DESCRIBES AN OCCUPATION, NOT A PERSON, and that settles the conjunction. The list was
    assembled from activities reported by SEVERAL different workers, so it names what different
    practitioners do -- one member handles the first kind of work on that list, another the
    second, a third neither. "A, B, and C" asserts that whoever does this task does all three,
    every time it occurs, which is false of an occupation by construction. "A, B, or C" says
    the activity applies to any of them, which is what a task inventory means.
    IT ALSO STOPS THE LIST READING AS SEPARATE DUTIES. A conjunction is how two duties are
    written, so "and" invites the reader to count the items; "or" tells them these are
    alternatives within one duty.
    TEST: put "any of" in front of the list. If the sentence is still a true description of the
```

```
    occupation's work, the list is disjunctive and takes "or". At occupation level almost every
    object list is -- including ones a single worker really does handle together on a single
    occasion, because the next worker does not. THIS IS ABOUT OBJECTS, NOT VERBS. Verbs joined
    by "and" are governed by 3.2, which permits them where they are successive steps of one
    process -- "draft and file", "record and submit". Do not change those.
    NOT THIS: the only objects that keep "and" are ones that are not alternatives at all --
    inseparable parts of a single artifact or act, where naming one without the others
    describes nothing anyone does.

3.2 VERB LISTS MUST NOT CONJOIN DISTINCT ACTIVITIES, AND SUCCESSIVE STEPS OF ONE PROCESS ARE
NOT DISTINCT. Two shapes are acceptable. "Verb1 ... or Verb2 ..." where the verbs are
ALTERNATIVE realisations of one activity, either of which finishes the job. And "Verb1 and
Verb2 ..." where they are SUCCESSIVE STEPS of one process over the SAME object, ON ONE OCCASION
-- drafting a document and then checking it before it goes out, deciding an outcome and then
telling whoever needs to know, fitting a part and then testing that it works. A worker who did
the first and not the second has left one job half done, and a statement naming both describes
that job completely rather than fusing two. Otherwise split the statement, or raise it to a
genuine superordinate verb.
    NOT SUCCESSIVE, THOUGH IT SOUNDS IT: setting a rule and later applying it, making a plan
    and later working to it, producing a thing and later maintaining it. Each pair is one act
    done once and another done repeatedly afterwards, on occasions of its own -- the second is
    not the finish of the first, it is the job the first made possible. Those split.
    BUT THE CHORES OF ONE ONGOING RESPONSIBILITY STAY TOGETHER, and this bounds the paragraph
    above: where the acts are the upkeep of one thing the worker is answerable for -- recording
    entries and chasing the unresolved ones, directing daily work and periodically appraising
    it -- no one of them is a duty on its own, and naming two or three of them describes the
    responsibility completely rather than fusing tasks. SUPERVISING ONE'S OWN STAFF IS THE
    COMMONEST SUCH RESPONSIBILITY, and its chores span both directions and all consequences:
    assigning and directing their work, answering their questions, unblocking their tasks,
    appraising their performance and settling what follows from the appraisal. A statement
    naming several of these is one supervisory duty described completely -- question-and-answer
    inside the worker's own operation is not an exchange of sides, and an appraisal's
    consequences are not a second act.
    The produce-then-maintain split is for deliverables that leave the worker's own bookkeeping
    and are used by others.
    TEST: ask whether the second verb would ever be performed by someone who had NOT done the
    first, on a different occasion and for its own reason. If yes, they are separate duties and
    they split. If the second exists only because the first was done, they are one duty and
    they stay.
    NOT THIS: doing the work and WRITING ITS RESULT INTO A RECORD are not steps of one process.
    The recording is nearly always already a task of its own, so a statement naming both
    annexes that task's act -- which is how one line comes to read as a duplicate of another.
3.3 REACTIVE AND GENERATIVE WORK ARE SEPARATE TASKS. Handling reported problems is a different
activity from eliciting new requirements, even when both occur in the same conversation with
the same counterpart.
3.4 OBJECT LISTS MUST BE DRAWN FROM THE OCCUPATION'S SHARED CORE, NOT ONE SPECIALTY. Where an
occupation spans practice areas, specialty-specific objects narrow the statement to a subfield;
generalise the list to the category that covers all of them.

## 4. PURPOSE, OCCASION AND METHOD CLAUSES

4.1 A PURPOSE CLAUSE MUST BE INFORMATIVE. Drop it where the verb already entails it.
    THE EASIEST CASE TO MISS IS THE MORPHOLOGICAL ONE: the clause is built on a word already
    standing in the head, so it restates it without repeating it. For a building surveyor,
    "measure floor areas to take measurements" and "schedule inspection visits to arrange
    scheduling" both say one thing twice, and neither reads as a restatement until you line the
    words up. CHECK THE STEMS, not the sense: if a noun or verb in your clause shares a root
    with a word in the head, the clause is the head with "to" in front. Say what the work FEEDS
    or DECIDES, or drop it.
    TEST: delete the clause and read the sentence again. If nothing was lost -- if a reader
    could have predicted the clause from the verb and object alone -- it was empty and it goes.
    Clauses naming what happens to anything sent anywhere ("for review", "for approval", "for
    their records") almost always fail this: submissions are reviewed by definition. Keep the
    clause only where it names the decision or outcome at stake, which a reader could not have
    guessed.
4.2 A PURPOSE CLAUSE MUST BE TRUE OF EVERY ITEM IN THE OBJECT LIST. If it holds for only part
of the list, either drop the clause or split the statement.
4.2a AN OCCASION CLAUSE MUST NARROW WHEN THE WORK HAPPENS. "as needed", "when required", "in
the course of the work", "on an ongoing basis", "during or after the working day" -- these name
no occasion at all. They are true of every moment the task could occur, so they add length and
tell a reader nothing about when to expect the work.
    TEST: name an occasion the clause EXCLUDES. "before each shift" excludes mid-shift; "after
    an inspection finding" excludes the routine case; "at intake" excludes everything later. If
    you cannot name a time the clause rules out, delete it -- the sentence is stronger without
    it. KEEP THE CLAUSE where the occasion is what makes this task different from a
```

```
    neighbouring one: the same act performed at a different trigger can be a different duty,
    and then the trigger is load-bearing and belongs in the sentence.

4.3 A METHOD CLAUSE MUST INFORM, AND AN INFORMATIVE ONE IS WELCOME. A "by ..." or "through ..."
tail naming HOW the act is performed -- the one or two moves that carry its judgement or skill
("by reviewing records and examining the patient") -- is the standard O*NET shape for method and
is LEGITIMATE even where the head alone already names the work: the method is the texture a
reviewer wants. The clause is a DEFECT in exactly two cases: it REPEATS the head's stems
("measure areas by taking measurements" -- check the roots, as 4.1 does), or it SMUGGLES A
SECOND DUTY -- a means item that is not a constituent move of the MAIN act but its own errand
with its own occasion. IF THE HEAD GOES EMPTY without the clause -- the "by ..." was carrying
the only real content -- the head was a category and the clause's contents are the actual tasks:
promote one of them to the verb. NOT THIS, AND THE EXCEPTION IS NARROWER THAN IT SOUNDS. Naming
a shared end and putting the activities after "through" is the repair for ONE defect: a
statement whose activities need DIFFERENT VERBS and would read as two or three separate tasks
if written as coordinate verb phrases. It does NOT apply to one verb over several objects --
"prepare A, B, and C for X" is a plain object list and stays one.
    TEST THE EXCEPTION BEFORE CLAIMING IT: ask whether a single verb covers every activity. If
    one does, there was no defect to repair, and the "through" clause has traded a clean list
    for a category -- which is the defect this principle exists to catch, arrived at from the
    other side. Never use the shape for a manner or a step.
    AND THE EXCEPTION IS NEVER A GERUND CHAIN. "by A-ing, B-ing, and C-ing" strings three
    activities behind the head the way a conjunction strings duties in front of it -- the shape
    hides an enumeration, it does not repair one. Where the exception genuinely applies, name
    at most TWO activities after "through", or use the participial responsibility form of 1.4a;
    three or more "-ing" clauses after "by" or "through" is the defect itself, whatever the
    head says.

4.4 A STATE IS A GOAL, NOT A TASK. Some statements put a lasting condition where the object
belongs -- a disposition of people toward the worker or each other, a climate, a standing --
something no single act produces, that persists between acts, and that no one could be watched
doing. Work acts on an object a reader can see acted on; a condition like that is what the work
is FOR. A statement whose object is such a condition has promoted its goal into the object slot
and dropped the work: for a hotel concierge, "Build rapport with returning guests" names
nothing a reader could watch. Write the observable act as the verb and object, and the
condition as the purpose clause: "Greet returning guests by name and check in during their stay
to build rapport." The activities name the acts; take the one or two that carry the duty.
    TEST: ask what the worker was seen DOING. If the sentence names no observable act, the work
    is missing and a goal is standing in its place.

## 5. COUNTERPARTS

5.1 INHERENTLY COORDINATIVE TASKS NAME THE COUNTERPART. Use published occupation titles where
the counterpart is a specific occupation, and add "and other staff" to keep the statement open.
5.2 A GENERIC COLLECTIVE IS NOT A COUNTERPART. "stakeholders", "other parties", "personnel",
"relevant teams", "decision-makers", "the business" -- these name nobody. They tell a reader
that somebody was involved without saying who, and they are to a counterpart what a bare
category is to an object list: the reach without the evidence. Name the role, the function or
the occupation; where the activities genuinely span several, name the two or three that recur
and add "and other staff" to keep it open. A GENERIC WORD IS FINE AS THE OPEN TAIL, NEVER AS
THE WHOLE COUNTERPART. 5.1 asks for that tail on purpose, so "the site foreman and other
trades" is right while "other trades" alone is not -- exactly as a category may head an object
list but not replace it. Before flagging, check whether at least ONE named party appears beside
the generic one. If one does, the sentence has its counterpart and the tail is doing its job.
    TEST: could a reader go and find the person? For a building firm, "the site foreman" and
    "the parts supplier" are findable; "stakeholders" is not. If the word survives being
    swapped for any other group in any other trade, it is naming nobody.
    THIS ONE IS WORTH ITS OWN LINE because a reviewer rejects it on sight: it reads as a
    statement written without asking who the work was for, and one such word can sink an
    otherwise sound sentence. 5.3 A COUNTERPART MUST BE SHARED-CORE, NOT SPECIALTY. This is 2.5
    applied to the counterpart slot: a counterpart met only in one specialty's cases narrows
    the statement to that subfield the same way a specialty object does (3.4). Keep the
    counterpart every practitioner meets by name and fold specialty counterparts into the open
    tail (5.1); where none is shared-core, the covering role replaces the list. For a
    veterinarian, "negotiate treatment plans with dairy herd owners, racehorse trainers, or
    kennel operators" is three specialties, not the occupation -- "with animal owners" covers
    them and everyone unlisted. The tail must still ride on a findable named party, never a
    collective (5.2).
```

P11 is sent as one user message to GPT-5.4 at medium reasoning effort, with the occupation title and the numbered tasks of the pile in its slots. The reply is one line per family, a heading followed by the task numbers. The output limit is at least 32,000 tokens and a truncated reply is retried with a larger limit and then at low effort. Families with tasks from fewer than two workers return to the pile.

Prompt 11: Clustering by Goal

```
You are performing a HIERARCHICAL TASK ANALYSIS for a worker in the occupation: {{ occupation
}}.

Below are the specific tasks observed for this occupation. Group them into the TASK FAMILIES
that make up the job -- the mid-level functional groups an occupational task list (like O*NET)
uses as headings. You are naming the PARENTS; the specific tasks are the leaves filed under
them. Make each family as SPECIFIC as its shared plan allows.

{% include "task-dimensions.txt" %}

A task FAMILY is a GOAL accomplished by a SHARED PLAN over its tasks -- the tasks are STEPS
toward one outcome. Whether tasks belong to ONE family is decided by that shared plan, NOT by
surface similarity. The guards against grab-bag families:
1. SAME THEME / TOOL / PHASE / SETTING / AUDIENCE ≠ SHARED PLAN. Tasks that merely sound
   similar, use the same tool, happen in the same phase, or serve the same people but are
   DIFFERENT ACTIONS on DIFFERENT objects belong to DIFFERENT families. · mechanic: "inspect
   brakes" + "order parts" -- both in the shop, but inspecting ≠ purchasing → DIFFERENT
   families. · electrician: "pull new wiring" + "inspect the panel for code violations" -- both
   electrical, but installing ≠ auditing → DIFFERENT families. · letting agent: "show a flat to
   applicants" + "chase a late rent payment" -- both serve the same tenants, but showing ≠
   collecting → DIFFERENT families. A shared AUDIENCE or POPULATION served (tenants, customers,
   members) is NOT a shared plan; name the specific ACTION, not the population it is aimed at.
   That cuts one way only: a SHARED audience never JOINS two different acts, but a DIFFERENT
   audience can still SPLIT one act, where the reader changes what the worker actually
   produces. A shared OVERSIGHT or LEADERSHIP domain across the leaves -- one program, function,
   or area the worker leads -- is likewise a THEME, not a shared plan: under a single umbrella
   verb, leaves that each carry out a DIFFERENT action are DIFFERENT families; split by the
   distinct action, not by the shared domain. · maintaining the machinery that DOES a kind of
   work is not the same plan as USING that machinery to do it -- building/maintaining a tool,
   pipeline, or system ≠ operating it for a specific instance of output. Likewise authoring
   reusable instructions, scripts, or materials ≠ executing them on a specific occasion.
2. COMPOUND-VERB RULE: a family heading may join at most TWO verbs, and ONLY when they are two
   VERY CLOSELY RELATED steps of ONE activity toward a single result -- a tightly-linked pair (a
   do-then-confirm, a review-then-comment). THREE OR MORE verbs, or two LOOSELY-related
   actions, means the heading is lumping distinct goals: name the ONE specific action each
   family shares. THE TEST FOR "CLOSELY RELATED", and it is the same test the fusion judge
   downstream applies, so a heading that fails it will be taken apart later: WOULD THE SECOND
   STEP HAPPEN AT ALL IF THE FIRST HAD NOT BEEN DONE? If it would not -- the thing produced must
   then be checked, submitted, or handed on -- the two are one activity finished, and the pair
   is allowed. If each could be done on its own occasion, for its own reason, in either order,
   they are TWO families however natural the phrase sounds. Two acts that merely BOTH concern
   the same field fail this: one prompted by a request and one done to a schedule are two
   families, not a pair.
3. THE PLAN-OR TEST: write the family's one-line plan in your head. If it needs an "OR" across
   unrelated triggers/contexts ("...when a machine breaks OR when planning the week's
   schedule"), the tasks are DIFFERENT families.
4. NEVER a vague catch-all ("Provide services", "Coordinate work", "Manage operations", "Ensure
   quality", "Handle issues").
5. A FAMILY IS A KIND OF WORK, NEVER A SUBJECT MATTER OR SPECIALTY. Do not make a family out of
   the case type, product line, programme, client segment, OR KIND OF DOCUMENT the work is
   applied TO. Workers who specialise in different subjects still perform the SAME activities
   on them. THE DOCUMENT TYPE IS THE COMMONEST FORM OF THIS AND THE EASIEST TO MISS, because
   each type has its own name and its own body of knowledge, so a per-type family looks
   specific rather than fragmentary. One act of authoring, applied to a dozen kinds of paper,
   is ONE family with the kinds as examples -- not a family per kind. The same holds for
   reviewing them, filing them, and preparing them. If two candidate families share their verb
   and differ only in which artifact it produces or acts on, they are one family: name the act,
   and let the artifacts be the examples underneath it. TEST BEFORE YOU SPLIT ON AN ARTIFACT:
   ask what the worker does differently, step by step, for the two kinds. If the steps are the
   same and only the subject matter of the paper changes, it is one family however differently
   the two documents are named and however separate the two specialisms feel. · repair shop:
   "handle motorcycle jobs" + "handle van jobs" are NOT two families -- diagnosing, quoting,
   ordering parts and road-testing are the families, and the vehicle type is merely WHICH
   INSTANCE of each one a given mechanic happens to see. · training provider: "run the forklift
   programme" + "run the first-aid programme" are NOT two families -- writing course material,
   scheduling sessions, assessing candidates and issuing certificates are. A SUBJECT-MATTER
   FAMILY ALSO HIDES THE WORK EVERYONE SHARES, which is worse than merely being untidy. Every
   specialist's version of one common activity gets filed under their own subject, so a task
   that all of them perform appears once in each subject family and never looks shared -- the
   most widespread activity in the occupation can end up looking like nobody's. If two workers
   with different specialties would describe an activity in nearly the same words, that
   activity is a family and their specialties are not.
```

```
ADEQUACY (least upper bound): each family must be general enough to be its tasks' shared goal
but NO BROADER -- performing all of its tasks should FULLY accomplish it, with nothing left
over. Every word of the heading must trace to a grouped task.

MUTUAL EXCLUSIVITY: no task should plausibly belong to two families. NEVER emit two families
that name the same kind of work at slightly different angles ("Coordinate the team" vs "Manage
team workflow") -- merge them into one.

ON GENUINE DOUBT, KEEP FAMILIES SEPARATE RATHER THAN MERGING -- but a false split is NOT free.
One act divided by which moment it happened at and who read the result yields headings that
collide in the published list, and nothing downstream repairs them: merging is refused because
the activities under each heading genuinely differ, splitting cannot help because they are
already split, and rewording cannot separate two headings that name the same act. So the
asymmetry below is the right default and a reason to look hard at a close call -- not a licence
to treat splitting as costless. A false merge is not symmetric: once two distinct activities
are folded into one heading, the distinction is gone and nothing downstream can recover it. If
naming ONE shared plan for two candidate families would mean dropping a verb or object either
one names, they are not one family -- keep both, and sharpen the wording so neither reads as a
duplicate of the other. Never resolve a close call by writing a heading BROADER than either
candidate -- that is averaging, not judgment, and it fails guard 1 and guard 4 at once.

Use as many families as the work genuinely has -- no fixed number -- but each must be specific,
distinct from the others, and cover at least two of the tasks below.

FILE THE TASKS YOURSELF, AND LEAVE OUT THE ONES THAT DO NOT FIT. You name the families AND say
which numbered tasks belong to each, because you are the only step that sees the whole list at
once. Two rules about what you file: · A TASK GOES UNDER AT MOST ONE FAMILY. If it plausibly
fits two, your families are not exclusive -- fix the families, do not file it twice.
  · A TASK THAT FITS NO FAMILY IS OMITTED, NOT FORCED. Leaving it out is the correct answer and
  costs nothing: a later step looks specifically at what you left behind and can name a task
  for it there. Stretching a family to swallow an activity it does not describe is the
  expensive mistake, because the family's statement then has to widen to cover it and the
  distinct work disappears for good. Omit freely.

Return ONE family per line, in this exact form -- the heading, then "::", then the numbers of
its tasks:

  <Verb> <specific work> :: 3, 7, 12, 41

Nothing else: no prose, no bullets, no explanation, no line for the omitted tasks.

Each task may show "(reported by N)" -- how many workers reported it. Let the COMMON activities
anchor the families; rarer ones attach to the family whose work they share.

Tasks:
{{ items }}
```

P12 is sent as one user message to Claude Sonnet 4.6 with up to 12,000 output tokens and the model's default sampling; the static part of the prompt is cached. Its slots carry the occupation, the member tasks tagged by worker, and the sibling headings; the family heading itself is withheld. The call is skipped when members and prompt are unchanged, and an adopted statement whose member set overlaps by at least 60 % enters the gate as the draft.

Prompt 12: Statement Synthesis (writer)

```
You are writing ONE task statement for the occupation: {{ occupation }}.

YOU ARE DESCRIBING AN OCCUPATION, NOT A PERSON, and every rule below follows from that. The
activities were reported by SEVERAL different workers doing this job in different places, with
different caseloads and different employers. The statement has to be true of the occupation as
they collectively practise it -- not a portrait of any one of them. So the objects it names are
alternatives across the workforce rather than a bundle one person handles, the vocabulary it
uses has to be recognisable to a practitioner who has never seen the others' work, and a detail
that belongs to a single account is a detail about that workplace, not about the occupation.

The specific activities below were grouped as ONE task family by an earlier step. That step's
heading is deliberately NOT shown to you: it was written to name a shelf, and every statement
written from it inherited its verb and its omissions. The activities are the evidence. Write a
single O*NET-style task statement for the work they describe.
```

```
{% include "onet-statement.txt" %}

THE STATEMENT ALONE MUST TEACH WHAT THE ACTIVITIES TEACH. Almost everyone who reads it will
never see the activities below it, so the statement is not a label for them -- it stands in for
them, permanently. Whatever a reader would picture from your sentence alone, these activities
should show.

WHAT YOU ARE OPTIMISING. The statement has to READ NATURALLY and REPRESENT THE OCCUPATION WELL
-- a practitioner should recognise it as a description of their work and be able to say "yes, I
do that". That is the test, and it has to pass TWICE: a second practitioner, in a different
workplace with a different caseload, has to be able to say it too. A sentence only the first
one would claim describes a job, not an occupation.

COVER THE CORE OF WHAT THESE ACTIVITIES DO, IN THEIR WORDS. Read them as a group and name the
work most of them are doing, as concretely as they describe it. The statement's job is to be
RECOGNISABLE to most of the workers under it, not defensible against all of them; a member or
two doing something else does not get a vote on the wording. The core is what the statement
owes; the edges are not:
  - WHEN BREADTH AND CONCRETENESS CONFLICT, CONCRETENESS WINS. For an appraiser, sixteen
    activities spanning residential valuation, dispute reviews, mortgage write-ups and
    portfolio work written as "Assess property viability, risks, and likely outcomes to decide
    and recommend valuation or review strategy" lose every concrete noun the members carry --
    condition, comparable sales, repair cost, site access -- because those words are not true of
    all sixteen. What survives is true of all of them precisely because it is empty. Written
    for the majority it becomes "Evaluate condition, comparable sales, and repair cost to value
    a property" -- narrower, and worth reading. Do not climb to a word that spans a minority
    strand.
  - A minority strand that will not fit a readable sentence is better left out than crammed in.
    Covering everything is what produces noun piles.
  - BUT COUNT BEFORE YOU CALL SOMETHING A MINORITY STRAND, AND COUNT TAGS, NOT ACTIVITIES. The
    line above is about OUTLIERS: one activity, or several from a single worker, or work that
    plainly belongs to another task. It is not licence to drop a distinct act that several
    activities name and MORE THAN ONE worker reports. A third of the group going unnamed is not
    an edge left out, it is a statement describing less work than it was given -- and every
    activity you leave uncovered comes back later as its own thin task, so the list ends up
    carrying the work twice: once implicitly, in a statement that never mentions it, and once
    as a two-line afterthought.
    WHEN YOU FIND ONE, GIVE IT A VERB rather than dropping it. A second verb costs three words
    and 3.2 permits verbs joined by "and" where they are successive steps over the SAME object
    -- a set-up, tidy and upkeep cycle on one space is one duty described completely, not two
    crammed together.
    TEST: list the activities your draft does not cover, and count the DISTINCT TAGS behind
    them. One activity, or one tag, is an edge -- leave it. Several activities under several
    tags is a strand, and a strand that will not fit means your sentence needs another verb,
    not a shorter object list.
  - The activities are a SAMPLE, not an inventory -- what a few interviewed workers happened to
    mention. You may name something none of them states when it plainly belongs to this same
    work as this occupation does it, and you need not confine the task to the one setting they
    described.
  - What you may NOT do is name a DIFFERENT KIND OF WORK: a second activity carried by one
    stray member, or something belonging to another task or another occupation. Over-reach
    means wrong work, not unmentioned detail.

EXPECT OUTLIERS, AND IGNORE THEM. The activities were grouped by an earlier automated step, and
that step is not perfect: a list of six often contains one or two that do not belong -- a
different kind of work filed here by mistake, or one worker's specialism that shares only a
loose relationship with the rest. You are not obliged to describe them, and you should not let
them shape the sentence. HOW TO SPOT ONE. Read the activities and ask what kind of work MOST of
them are. An outlier is the one whose object, counterpart, or setting is of a different kind
from that answer -- not merely a variation of it. For a facilities engineer, three activities
servicing chillers in offices, plants and hospitals are one kind (maintaining building
systems); advising inspectors on fire-code certification beside them is a different kind, even
if all four are "handling building systems". WHAT TO DO. Write the statement as if the outlier
were not in the list. Do not name it, do not add a fourth list item for it, and above all do
not widen the head or the object so that it fits -- a word broad enough to cover the outlier is
a word too broad to describe the rest. There is no invented superordinate: if a strand will not
fit the sentence the core deserves, it is not this task's strand. Nothing is lost by leaving it
out: the activity stays attached to this task, and the sentence describes what most of them
are. A MINORITY STRAND HAS THREE FACES, and the second and third are easy to miss because they
look cheap to obey. · A DIFFERENT KIND OF WORK -- the obvious case, above.
  · A SUBJECT AREA. THE FIELD IS NOT THE WORK. Where the activities show the same action
  applied across different subject areas, name the action and the kind of object, not one area
  and not all of them. Each area belongs to whoever happened to be interviewed: naming one
  tells every other worker this task is not theirs, and listing four makes a checklist that
```

```
  still misses the fifth. Eleven activities inspecting for defects -- in lifts, boilers,
  pressure vessels and cranes, one worker each -- written as "Inspect lifts and cranes for
  defects" present two workers' corner as the whole scope; "Inspect equipment for defects and
  report the remedial work required" is the same length and covers all eleven. THE TEST: could
  a practitioner in a DIFFERENT part of this occupation, doing this same activity on their own
  subject matter, read the sentence and recognise their work? If naming the field is what makes
  the sentence concrete, put the concreteness in the ACTION and the OBJECT instead.
  · AN EXTRA RECIPIENT. A second party costs nothing to bolt on -- one more noun, no vaguer head
  -- and still does not belong. Where one activity names another recipient alongside work the
  counterpart most activities name already covers, leave it out: "Update tenants on repair
  progress" already serves an activity reading "update landlords and tenants on claim status".
  Adding "or block managers" spends the sentence on one worker's recipient. ADD A SECOND PARTY
  ONLY WHEN ACTIVITIES WHOSE WORK IS OTHERWISE UNCOVERED DEPEND ON IT.

TAKE THE LEAD VERB FROM THE ACTIVITIES. Lead with the verb most of them lead with, or use two
where they genuinely divide between two actions. COUNT, do not impressionise: twenty activities
of which ten begin "Install" and one begins "Design" is an installing task, and leading with
"Design and maintain ..." makes one activity's verb the head and demotes the other ten. Fifteen
activities that ALL begin with a meeting verb, written up under a planning verb, read as a
duplicate of the neighbouring task that really is about planning. If the activities' own verb
is a carrier (perform, conduct, provide, manage, handle, coordinate) attached to an object that
is really a verb in noun form -- an inspection, an approval, a repair -- promote that verb and
drop the carrier: "perform X evaluations" becomes "evaluate X".

EACH ACTIVITY CARRIES A WORKER TAG, AND IT DECIDES ONE THING ONLY: WHETHER A NOUN MAY STAND AS
THE WHOLE OBJECT, OR ONLY AS ONE ITEM AMONG OTHERS. "[W1]" and "[W2]" are different people
reporting their own work; the tags mean nothing beyond same-worker-or-not.
  - A NOUN UNDER SEVERAL TAGS CAN CARRY THE STATEMENT. More than one person named it, so it is
    the occupation's and not one workplace's.
  - A NOUN UNDER ONE TAG MAY STILL APPEAR -- as one item in a list, or after "such as". What it
    must not do is become the ONLY object, because then the sentence describes one person's
    caseload while claiming to describe everyone who does this work. Keep it, and put it beside
    something the group shares. EXCEPT A SPECIALTY TERM, WHICH DOES NOT RIDE IN ON ONE WORKER'S
    TAG. A counterpart, object, or setting that only one practice area's practitioners would
    name (2.5 in the principles below) does not earn a place in the list from one person's
    activities: it goes into the covering term or the open tail, beside the parties or items
    every practitioner shares. One specialist's counterpart in an occupation-level list is that
    person's caseload wearing the occupation's sentence.
  - A CATEGORY IS WELCOME AS THE HEAD. IT IS NOT A SUBSTITUTE FOR THE ITEMS. Generalising a set
    of one worker's nouns to the class that covers them is the right move -- but the class goes
    in FRONT of them, with the concrete items following as examples, not in place of them.
    "<class>, such as A, B, or C" keeps both the reach and the evidence; "<class>" alone has
    deleted the evidence and kept only the reach.
    TEST: after generalising, can a reader still name one thing this task is done to? If the
    sentence now contains no concrete object at all -- a bare class noun with a filler
    qualifier, of the "other parties", "various items", "relevant matters", "applicable
    requirements" kind -- the category has replaced the work rather than covering it, and it
    fails the test this prompt opens with. Put two or three of the items back after "such as".
    AND IF THE ONLY CONCRETE VOCABULARY BELONGS TO ONE WORKER, USE IT ANYWAY, beside the class:
    a specific sentence resting on one account is worth more than a vague one resting on none.
  - WHERE NO VOCABULARY IS SHARED ACROSS TAGS AT ALL, GENERALISE -- DO NOT CONCATENATE. A small
    group can hold two activities from two workers with no noun in common. The wrong answer,
    and the tempting one, is to list both of their nouns joined by "or": that sentence reads as
    one task for the first specialist and a second task for the other, and describes the
    occupation as neither of them. Find the ACT they share and the CATEGORY both objects belong
    to, and write that -- one act, one class of object, with the specifics dropped rather than
    doubled. A statement covering two workers' work as a class is worth more than one that
    quotes each of them in turn.
    THE CATEGORY STILL HAS TO NAME SOMETHING, so this is not licence to retreat to a filler
    word: if the only class you can find covering both is a placeholder, keep the more widely
    reported of the two and let the other go uncovered -- an honest narrow statement beats a
    wide meaningless one.
  - WHERE ONE TAG OWNS MOST OF THE ACTIVITIES, that worker's vocabulary IS this group's
    vocabulary and this rule has nothing to say. Count before you generalise: a group of
    fifteen activities where twelve carry the same tag is that worker's work, described by the
    people who also do a little of it -- not a shared core with a specialty bolted on. Write the
    most concrete sentence the activities support.

EVERY OBJECT THE ACTIVITIES SHARE BELONGS IN YOUR STATEMENT. Naming a group is lossy, which is
why the heading is withheld: a heading routinely omits a concrete term that TWO OR MORE of its
own activities named. One group held three activities about preparing "crews and site plans",
"crews, instruments, targets, and field books" and "instruments, benchmark records, and
levelling notes" -- every one mentioning materials -- and the statement written from its heading
mentioned only the people. WRITE SHORT AND FINITE. The statement is ABOUT 15 WORDS -- one verb,
one object list with at most two or three examples, at most ONE "-ing" clause. Never stack
```

```
gerunds ("...ing, ...ing, and ...ing") and never chain three verbs: if the sentence needs them,
it is carrying more than one duty's worth of content, and the members -- not the sentence -- are
where detail lives.

So read the ACTIVITIES for the objects. If an object, counterpart, document, or setting appears
in two or more activities, it belongs in your statement -- that is not over-reach, it is the
evidence. UNLESS EVERY ACTIVITY NAMING IT COMES FROM ONE SPECIALTY. Repetition inside one
practice area is evidence about that specialty, not about the occupation: a counterpart,
object, or setting that only one subfield's practitioners would name goes into the covering
term with the open tail (2.5 in the principles below), beside whatever parties or items every
practitioner shares. Ask of each recurring term: would a practitioner in a DIFFERENT practice
area ever meet this in their own work? If not, name its covering role or class and let the tail
hold it. BUT THE THRESHOLD DOES NOT OVERRULE YOUR JUDGEMENT, AND A TERM THAT DOES NOT FIT THIS
GROUP'S ACTIVITY IS YOURS TO LEAVE OUT. An object can appear in two activities and still be the
wrong word for this statement. Two ways that happens: it names work a SIBLING statement is
dedicated to, so carrying it makes two lines read alike; or it is a word that means one thing
in this trade and another thing elsewhere, so a reader cannot tell which sense you intend. The
test is 3.1's -- is the term a CO-HYPONYM of the rest of your list, the same activity applied to
another instance? If it is not, omit it. A statement that reaches for every noun its activities
mention stops describing one activity and starts reading like an inventory of words. WHAT YOU
MAY NOT OMIT IS A DISTINCT ACT that two or more workers report: dropping an act drops a duty,
and the activity comes back later as its own thin task. This licence covers a stray OBJECT that
pulls the sentence toward a neighbour -- never a verb.

STAY AT THE LEVEL YOU WERE GIVEN. These activities sit at one level of a task hierarchy and
sibling tasks cover the neighbouring work. Do not climb to the general category -- that is a
different task's job, and two siblings that both climb become duplicates. Do not descend to
just one activity either. Match the statement's breadth to what the activities COLLECTIVELY do:
broaden to their shared object where they spread wider than any one of them, and tighten to the
concrete kind of work where they all name the same one.

AND CHECK IT AGAINST THESE WRITING PRINCIPLES. Your sentence has to satisfy all of them.

{% include "writing-principles.txt" %}

A QUALIFIER MUST NOT MISREPRESENT THE MAJORITY. The rules above let a MINORITY strand go
uncovered; they do not license a narrowing that makes the core look like something it is not.
Narrowing the party, occasion, setting or DIRECTION changes who or when, and a reader takes the
omission as real: "hand over finished vehicles at shift end" hides that half the activities
TAKE IN arriving ones, and "schedule repairs for walk-in customers" says a booked appointment
is out of scope. Where MOST activities run both ways, name both -- "to or from", "from owners or
fleet managers" -- because two coordinate words cost almost nothing and leaving them out is a
factual error about the core, not concision. The distinction from the minority rule: drop a
strand ONE OR TWO activities carry, never a direction or party that HALF of them do.

## WRITE FROM THE ACTIVITIES, NEVER TOWARD ANOTHER TASK LIST

No published task list, O*NET's included, has any say over the CONTENT. Do not reach for the
phrasing some other inventory uses for adjacent work, do not widen an object so it will line up
with a category you recall, and do not narrow one to avoid overlapping something. Those lists
are what this work is LATER MEASURED AGAINST, and a statement written to match one makes that
measurement circular: the comparison would report the reaching, not the work. A statement that
reads oddly beside an external list but says exactly what these workers described is correct,
and the mismatch is a finding to keep.

ACTIVITIES:
{{ items }}

Answer with ONLY the statement.

MUTUAL EXCLUSIVITY WITH THE OTHER TASKS IN THIS LIST. These are the OTHER families being
written up alongside yours:
{{ siblings }}

Your statement must not describe work one of those owns. Two ways that happens, and both are
errors:
- AN EXAMPLE that belongs to another family. If one of your "such as" items is the core work of
  a family above, drop that item and pick a different one from your OWN activities -- the other
  statement covers it.
- AN OBJECT CATEGORY that swallows another family. If your object names a whole class of work
  another family is dedicated to ("... with medications, procedures, or interventions" when
  separate families do the procedures), narrow the object to what YOUR activities actually
  describe. A category broad enough to contain a sibling makes your statement read as its
  parent.
```

```
Do NOT solve this by going vague, by deleting your specifics wholesale, or by inventing work
your activities do not describe. Say precisely what YOUR activities are, and leave the rest to
the tasks that own them.
```

P13 is sent as one user message to GPT-5.4 mini with up to 700 output tokens and a JSON-object reply, followed by the pair as `ATOM:` and `LABEL:` lines. Only the yes/no verdict is used; an unparseable reply is retried rather than cached.

Prompt 13: Coverage Judge*

```
You are judging whether one specific worker task (the ATOM) falls under a broader O*NET-style
task statement (the LABEL), for tasks within a single occupation.

The ATOM FALLS UNDER the LABEL when performing the atom is ONE WAY of doing the label -- an
instance, a sub-type, or a constituent step of the SAME activity, and no broader than the
label.

THE DEFAULT ANSWER IS YES. Below is the COMPLETE list of things that make it NO. Work through
the two reading steps, then check the four disqualifiers. If none of the four applies, answer
YES. Do not invent a fifth.

READING STEP A -- SPLIT EACH SIDE INTO ACT, OBJECT, AND PURPOSE.
  ACT = the verb, what the worker does. OBJECT = what it is done to. PURPOSE = everything after
  "to ...", "for ...", "so that ..." -- the reason it is performed. Do this for the label and for the
  atom.

READING STEP B -- NOW DISCARD BOTH PURPOSE CLAUSES.
  Labels are written to stand alone and almost always state a purpose. Atoms are terse notes
  from a worker and usually leave it implicit. The atom's silence about purpose is therefore
  evidence of NOTHING, and a purpose it does state in unlike words is evidence of nothing
  either.
    · "Review medical and accident records for PIP arbitrations" and "Review investigative and
    medical records to assess case facts and damages" are the SAME ACT ON THE SAME OBJECT; one
    states a purpose and one does not, and that difference is invisible here.
  Never answer NO because the atom omits the label's purpose, lacks its purpose verb ("assess",
  "evaluate", "determine", "improve", "ensure"), or gives a different-sounding reason for the
  same act. Judge ACT and OBJECT.

THE FOUR DISQUALIFIERS. Check each against the ACT and OBJECT you extracted.

D1 -- A DIFFERENT ACT. The atom's act is not the label's act and is not a constituent step of
it. Sharing a tool, a topic, a field or a client is not sharing an act.
   NOT a different act: a label joining several acts with "and", "or" or a comma list is
   satisfied by ANY ONE of them -- never require every conjunct, and never require every listed
   object. A "such as ..." / "including ..." list is ILLUSTRATIVE of one CATEGORY, not a closed
   checklist: an atom of the same kind qualifies even when unlisted ("operate shop machinery,
   such as saws" covers a lathe). A constituent step qualifies: a physical exam is part of
   "evaluate patients".

D2 -- THE ATOM IS BROADER. The atom's act or scope exceeds the label's, so the label describes
only part of it. Specificity runs ONE WAY only: an atom naming a NARROWER counterpart, object,
instrument, setting, specialty or client type still falls under the label. The label is not
required to anticipate the instance. Only the reverse -- a broader atom -- disqualifies.

D3 -- THE OTHER SIDE OF AN EXCHANGE. The label puts the worker on one side (performing,
requesting, sending, authoring) and the atom puts them on the other (checking someone else's
performance, granting, receiving, approving). However much vocabulary they share, these are
different work.

D4 -- A DIFFERENT PRODUCT, AND THIS ONE IS CONDITIONAL. It applies ONLY where the label's act
produces a THING: a document, a delivered service, a configured system, a released build. Then
an atom producing an artifact of a different kind, or work the label never reaches, is
disqualified; an atom producing an input to or an earlier stage of that thing is NOT.
   D4 DOES NOT APPLY WHERE THE LABEL'S PRODUCT IS A JUDGEMENT. If the label's act is analysing,
   assessing, reviewing, evaluating or advising, its product IS a conclusion, an opinion, a
   recommendation or a decision. An atom that reaches a conclusion, decides what to do next, or
   recommends a course is then producing EXACTLY what the label produces. Do not disqualify it.

VERDICT. If D1, D2, D3 and D4 all fail to apply, the answer is YES -- including when you are
unsure, because work described in two registers, a worker's terse note and a written task
statement, is more often the same work than different work.
```

```
Respond ONLY with JSON:
{"label_act":"<the label's verb>","label_object":"<what it acts on>","atom_act":"<the atom's verb>","
atom_object":"<what it acts on>","same_act":"<yes|no> -- is the atom's act the label's act, a synonym, or a
sub-type?","deciding_rule":"<D1|D2|D3|D4|none -- the disqualifier that applied, or none>","why":"<one sentence
>","verdict":"<yes|no>"}
```

P14 is sent as one user message to GPT-5.4 mini at low reasoning effort with up to 2,000 output tokens and a JSON-object reply; the static prefix is cached. Its slots carry the occupation, the statement, and its member tasks. A failed call counts as one duty and is not cached; verdicts are kept in a ledger keyed by member set, and a statement the writer flagged as fused is heard once more.

Prompt 14: One-Duty Judge (split hearing)*

```
You are a trained occupational analyst deciding whether ONE proposed task statement names ONE
duty, for a worker in the occupation: {{ occupation }}.

You are given the statement and the activities filed under it, as reported by workers. Decide
whether the statement names a single duty, or whether it has FUSED several duties into one
line.

WHAT ONE DUTY MEANS. A task is a GOAL accomplished by a SHARED PLAN over its activities. Doing
all the activities should accomplish one outcome -- not several outcomes that happen to be
listed together.

ENUMERATION IS NOT FUSION, AND THIS IS THE DISTINCTION THAT DECIDES MOST CASES: · ONE ACTION
OVER A LIST OF OBJECTS IS ONE DUTY. A statement may name many things the SAME action is applied
to -- the materials produced, the media used, the settings, the audiences, the document types. A
long list of objects is a statement being specific about the same work, which is desirable, not
a defect. If deleting the list would leave the same action with the same purpose, it is one
duty. · SEVERAL ACTIONS WITH SEPARATE OUTCOMES IS FUSION. If the statement names actions that
are pursued for DIFFERENT reasons, on different occasions, producing different results -- so
that doing one tells you nothing about whether the others were done -- it has fused duties,
however smoothly the sentence reads. · RESPONDING TO REPORTED PROBLEMS AND SOLICITING NEW WORK
ARE SEPARATE DUTIES, EVEN IN ONE CONVERSATION WITH ONE COUNTERPART. This is the case the "same
occasion" exemption below most often swallows wrongly, because the two really do happen in the
same meeting, with the same people, in the same breath. They are still two duties: handling
what has gone wrong is triggered by the problem being reported and ends when it is resolved,
while gathering what should be built next is a standing obligation with no trigger and no
completion. For a landscape gardener, "discuss plant losses and next season's planting ideas
with clients" is TWO duties -- the losses arrive as complaints and the proposals are solicited --
and the shared occasion does not join them. Neither does a shared verb: "discuss", "meet about"
and "talk through" name the setting, not the duty, so a statement whose only unifying element
is the conversation itself is FUSED.

TWO STEPS OF ONE TRANSACTION ARE ONE DUTY, AND THIS EXEMPTION COMES BEFORE THE TESTS BELOW.
Work often has a step that exists only because the first step happened, on the same occasion,
about the same case: deciding and then notifying whoever needs to know, doing and then
confirming, handing something over and taking it up. · PRODUCING SOMETHING AND THEN
TRANSMITTING IT IS THE COMMONEST CASE OF THIS, AND THE DIRECTION TEST BELOW WILL TRY TO SPLIT
IT -- DO NOT LET IT. Making a document and then lodging, sending, submitting or serving it is
ONE duty: the transmitting step exists only because the thing was made, it goes out about the
same matter, and a worker who drafted it and did not lodge it has not finished the job. This
OUTRANKS the direction test's "producing is different from submitting", exactly as the
recurring-exchange bullet below outranks it. THE DISCRIMINATOR IS WHETHER THE SECOND STEP
CARRIES ITS OWN JUDGEMENT ON ITS OWN OCCASION. Ask what the worker must decide at the second
step. If the answer is nothing -- the content was settled when the thing was made, and what
remains is getting it where it goes -- it is the completion of one duty, however many days later
it happens. If instead the second step needs its own preparation and its own judgement in front
of somebody, so that a worker could do it well or badly independently of how the thing was
made, it is a second duty. For a draughtsman, drawing a plan and submitting it to the authority
is one duty; drawing it and then defending it at a public hearing is two. BUT RECORDING WHAT
WAS DONE IS ITS OWN DUTY. Performing work and writing the result into a record are two acts:
they need different attention, they are often done at different moments, and in a real
inventory the recording is nearly always ALREADY A TASK OF ITS OWN -- entering the figures into
the system, updating the register, filing the return. A statement that names the work AND the
recording annexes that other task's act, which is how one line comes to read as a duplicate of
another. A statement of the form "<do the work> and record the results to <purpose>" annexes
the recording task's act, and the two then read as duplicates. If the statement names an act
and then names writing its output down, that is FUSED -- say so, and give the recording its own
```

```
outcome. THIS DOES NOT REACH A STATEMENT THAT MERELY NAMES ITS OWN PURPOSE OR RESULT: naming
what the work produces is one duty described completely, and a statement is better for naming
it. The line is whether a SEPARATE ACT OF RECORDING is named. Apply this test: · SAME OCCASION,
AND CAUSED BY THE FIRST. If the second part happens on the same occasion as the first and would
not happen at all unless the first had been done, it is the same duty. The worker is not
choosing between two jobs; they are finishing one. · BOTH ENDS OF ONE RECURRING EXCHANGE ARE
ONE DUTY, AND THIS OUTRANKS THE DIRECTION TEST BELOW -- do not let "different people" or
"opposite directions" pull it apart. Where work is routinely handed between people, giving and
receiving are the two halves of a single exchange the worker performs on both sides in the
ordinary course. For a locksmith, handing keys to incoming tenants and taking them back from
outgoing ones is ONE key-control duty seen from both ends, not two. The same holds for any
procedure a worker conducts on one occasion and submits to on another: it is one competence,
one plan, one duty. Do not split them. · A PURPOSE OR MEANS CLAUSE IS NOT A SECOND DUTY. A
clause saying what the work is for, or by what means it is carried out, describes the same duty
even though it contains another verb. · BUT TWO MEANS A READER WOULD COUNT SEPARATELY ARE TWO
DUTIES, and this narrows the exemption just above. That exemption is for ALTERNATIVE means --
interchangeable techniques for ONE instance of the work, where the worker picks one and the job
is done. For a locksmith, "open a locked door by picking the cylinder or drilling it" is one
duty: either way the door opens once. It is NOT for CUMULATIVE means, where BOTH happen, to the
same recipient, each with its own preparation, its own occasion, or its own artifact. For a
building surveyor, "report findings through a written schedule and a client walkthrough" is TWO
duties: the schedule is drafted at a desk and the walkthrough is conducted on site, both are
owed, and doing one does not discharge the other. THE TELL IS USUALLY "AND" RATHER THAN "OR".
Alternatives are joined by "or" because one suffices; separate duties are joined by "and"
because both are owed. Do not decide on the conjunction alone -- ask whether one means
substitutes for the other on a given occasion, or whether both are performed. If both are
performed and each carries its own preparation, they are two duties however single the purpose
they serve, and a statement naming both in a "through ..." or "by ..." clause is FUSED. · WHAT
STILL COUNTS AS FUSED: parts that arise on SEPARATE OCCASIONS, from separate triggers, or as
separate standing obligations the worker could be doing in any order and for unrelated reasons.
Two record-keeping obligations owed to different bodies are two duties, and so is recording the
thing you just did when the statement names both the doing and the recording.

{% include "task-dimensions.txt" %}

THAT TAXONOMY GOVERNS WHICH DIFFERENCES MAY COUNT AS SEPARATE OUTCOMES HERE, and it is the same
one the merge and coverage judges are held to, so a verdict grounded outside it puts this check
at odds with every step downstream. Naming two outcomes is open to you only when they differ
along a dimension that DIVIDES. Two parts of a statement are therefore NOT two duties merely
because they sit at different STAGES of the same matter before the same counterpart -- seeking
something and later reporting on it, opening a matter and closing it -- nor because the same
act reaches its counterpart through a different MEDIUM, TOOL or SETTING. Where the
DOES-NOT list covers the only difference you can find, answer one_duty: true and say which
bullet it was.
THE CUMULATIVE-MEANS RULE ABOVE SURVIVES THIS, BUT ONLY THROUGH THE ACT. Two means are two
duties when they are two different ACTS -- composing a document at a desk and conducting a
session in person take different steps, different judgement and different skill, which is why
the surveyor's schedule and walkthrough divide. They are NOT two duties when one act is merely
carried over two media. Ask which it is before you split.

TESTS, IN ORDER:
  1. THE PLAN TEST. State the one-line plan the statement implies. If writing it needs an "or"
     across unrelated triggers, occasions, or purposes, the statement is fused.
  2. THE DIRECTION TEST. Ask who acts on whom, and to what end. Work the worker performs ON
     THEMSELVES OR THEIR OWN OBLIGATIONS is a different duty from work they perform ON OTHER
     PEOPLE, even when both are described with the same verb and both concern the same subject
     matter. Producing something is a different duty from reviewing or receiving it. THIS TEST
     DOES NOT REACH PRODUCING-AND-THEN-TRANSMITTING: the transaction exemption above settles
     those, and it comes first. Do not cite this test to split making a thing from lodging,
     sending, submitting or serving it.
  3. THE MEMBER-PARTITION TEST. Look at the activities filed under the statement. If they fall
     into groups that share no plan with each other -- each group serving its own outcome -- the
     statement is covering for a grouping that was never one duty. If instead every activity is
     a variant of the same action, it is one duty however varied the wording. A GROUP OF ONE IS
     A MISFILED ACTIVITY, NOT A FUSED DUTY. An earlier automated step did the filing and it is
     not perfect, so a long list routinely carries one stray that belongs elsewhere. One
     activity out of many, pointing somewhere the STATEMENT ITSELF does not go, is evidence
     about the CLUSTER and not about the sentence -- the statement never claimed that work, so
     splitting it changes nothing the statement says. Require a real partition: each side needs
     several activities, or a single activity the statement actually names. Do not split a
     statement to accommodate one outlier.
  4. THE SUBSTITUTION TEST. Could a worker do everything in one part of the statement and
     nothing in another, and still be doing their job as the statement describes it? If yes,
     the parts are separate duties. INDEPENDENCE IS NOT FUSION, AND THIS TEST IS THE ONE MOST
     OFTEN MISREAD. Almost any statement naming two objects passes a bare "could you do one
```

```
     without the other" -- that is true of every list ever written, and answering on it alone
     splits statements that are plainly one duty. The question is not whether the parts CAN
     occur apart; it is whether they SERVE UNRELATED ENDS. Ask what each part is FOR. If both
     feed the same outcome -- the same case, the same delivery, the same obligation being
     discharged -- they are one duty even when a worker could do one on a day they do not do the
     other. Only answer "fused" when doing one leaves the other's outcome untouched and
     unadvanced.

WHAT DOES NOT MATTER HERE: length, wording, how many nouns appear, whether the statement is
broad or narrow, whether it duplicates some other task, and whether every activity is well
covered. Judge ONLY whether it names one duty or several.

BE CONSERVATIVE. A statement that is merely long, or that lists many objects or examples, is
ONE duty. Answer "fused" only when you can name the two or more separate outcomes and say which
activities belong to each.

THE UNIT OF THE INVENTORY IS A DUTY A PRACTITIONER WOULD LIST, NOT AN ACT YOU CAN ARTICULATE.
Almost every real duty decomposes into acts with their own objects and occasions, so being able
to name two parts is not yet evidence of fusion. Before answering "fused", write each part as
its own entry and ask whether it reads as a duty of the occupation someone could hold on its
own. If a part reads instead as a chore of the same ongoing responsibility as the others -- the
recording that belongs to its accounts, the appraisal that belongs to its supervision, per the
taxonomy's chores-of-one-responsibility bullet -- the statement is ONE duty and the parts are
its examples. This outranks "WHAT STILL COUNTS AS FUSED" above wherever the separate occasions
are all the upkeep of one responsibility.

YOUR OUTCOMES ARE WORKING LABELS, NOT PUBLISHED PROSE. A split part that survives loses its
statement and has its head REDRAWN by the writer from its members (apply_splits strips the
stamps precisely so that happens), so do not labour the wording: one plain line per outcome,
verb first, naming the act and its object clearly enough that a reader can tell the parts
apart. What you MUST get exactly right is the MEMBERS -- the partition is the evidence, the
label is a sketch. HOW A STATEMENT IS WRITTEN NEVER DECIDES THE VERDICT, and the distinction
matters because this check gates every merge in the pipeline: a statement is NEVER fused
because it uses house jargon, picks a weak verb, carries an empty purpose clause or omits a
counterpart. Those are defects for another step to fix. If the only thing wrong with a
statement is how it is written, answer one_duty: true. Return ONLY this JSON:
{"one_duty": true|false, "outcomes": [{"statement": "<one plain line naming the separate outcome this fused
part serves>",
                "members": ["<every activity from the list below that belongs to this part, copied VERBATIM,
                            character for character -- do not paraphrase, shorten, or merge
                            lines>"]}], "reason": "<one sentence naming the test or bullet that
                            decided it -- REQUIRED for both verdicts, including one_duty: true,
                            where it should name the exemption or DOES-NOT-DIVIDE bullet relied
                            on>"} `outcomes` is [] if one_duty. EVERY activity you assign must
                            be an exact copy of a line from the list below. AN ACTIVITY THAT
                            SERVES MORE THAN ONE OUTCOME IS LISTED UNDER EACH of them -- a
                            conjunctive activity spanning two parts is evidence the parts share
                            it, not evidence it belongs to neither, and listing it once under
                            the wrong part or nowhere loses it. Leave an activity out of every
                            outcome ONLY when none of them describes it.

STATEMENT:
{{ statement }}

ACTIVITIES FILED UNDER IT:
{{ members }}
```

P15 is sent as one user message to GPT-5.4 with up to 2,000 output tokens, followed by the two statements as bullet lines; the free-text verdict is parsed to yes or no. Only pairs with embedding cosine of at least 0.44 (`text-embedding-3-large`) are judged.

Prompt 15: Overlap Judge (router)*

```
Two task statements from ONE occupation's published task list were flagged as possibly
overlapping. Decide ONE thing: do these two lines' CLAIMS overlap -- is there an act that
appears on both? You are ROUTING, not merging. Whether an overlapping pair is one duty the
inventory owes once, or two duties whose boundary needs rewording, is decided further down the
chain BY THE MEMBER ACTIVITIES under each label, not by you: you see two sentences and nothing
else, which is the right instrument for the routing question and the wrong one for the merge
question. Answer the routing question and stop.
```

```
THE TEST IS HOW THE TWO LINES READ. These statements ARE the deliverable -- almost every reader
will see them and never see the activities beneath them. So ask: reading these two lines side
by side, would a worker in this occupation say "that is the same job duty, written twice"? If
yes, say YES. Do not ask how long the list is or whether it "has room" for both; a duplicate
reads as a duplicate in a list of ten or a hundred. Ask it about the WORK, not the sentences.
Two statements can share almost no words and still be one duty, because each writer reached for
different instances, different paperwork, or a different moment in the cycle.

{% include "task-dimensions.txt" %}

DECIDE IN THIS ORDER. STEPS 1-2 ARE A GATE, NOT FACTORS TO WEIGH.

STEP 1 -- LIST THE ACTS EACH LINE NAMES. A line can name more than one: read its lead verb AND
any act in an "and ..." clause or a trailing clause. Collapse near-synonyms into one act before
comparing (record, log, enter, file, maintain and document are ONE act; plan, prepare and
develop are ONE act; assess, evaluate and grade are ONE act).

STEP 2 -- IF NO ACT APPEARS ON BOTH LISTS, ANSWER NO AND STOP. This is the case this judge gets
wrong most often: measured against hand labels it answered YES to seven pairs in a row whose
acts were disjoint. A shared OBJECT, a shared POPULATION served, a shared DATA SOURCE, a shared
SETTING or a shared PURPOSE is NOT a shared act. Delivering a service and inspecting one are
two acts. Working out what is wrong and instructing someone are two acts. Informing a client
and building rapport with them are two acts. Producing a figure and advising someone what to do
about it are two acts. In every one of those the two lines share their object and still
describe different work, and a published inventory owes both lines.

WHEN YOU GENUINELY CANNOT TELL, ANSWER YES. You are the FIRST of two judges. A wrong YES costs
one look at the pair's member activities downstream, where the two lines are read against the
concrete work beneath them and the default is to keep both -- a wrong YES publishes nothing. A
wrong NO is FINAL: the pair is never looked at again, and if the claims do overlap the
inventory ships the overlap. The two errors are not symmetric, so uncertainty resolves to YES.
THIS IS NOT A LICENCE TO SKIP STEP 2, AND IT DOES NOT SOFTEN IT. The tie-break applies ONLY
where you cannot tell whether an act appears on both lists. Where you CAN tell and it does not
-- where the only thing shared is an object, a population served, a data source, a setting, a
tool, a document format, or a carrier verb -- the answer is NO and it is not close. Answering
YES on act-disjoint pairs is the failure this judge is measured on: against hand labels it once
answered YES to seven such pairs in a row. Doubt about a SHARED ACT resolves to yes; the
absence of one does not.

STEP 3 -- IF AN ACT APPEARS ON BOTH LISTS, THE ANSWER IS YES. If ONE side names an act the other
also names, that is enough, even when each side names a second act of its own -- the overlap is
what matters, not the surplus. The shapes below are what a shared act LOOKS LIKE; they are not
a further test to pass. They fall in two families and BOTH ANSWER YES, because both are pairs a
reader could file the same activity under.

SAY YES -- one duty written twice:
* THE SAME ACT ON DIFFERENT SPECIALTIES, PRACTICE AREAS, CASE TYPES, OR CLIENT SEGMENTS. This
  list describes an OCCUPATION, not a person, so the subject the work is applied to is which
  INSTANCE a given member happens to see -- not a second duty. Two lines that name one act and
  differ only in the field it is aimed at are one line the inventory owes once, with the fields
  as examples. It does not matter that the two specialisms need different bodies of knowledge,
  or that no single member does both: an occupation-level inventory that carries one line per
  specialty hides the activity they share, and the most widespread work in the job then appears
  nowhere as itself.
* SAME ACTION, SAME OBJECT, different WORDS FOR THE SAME SET. For a groundskeeper, "Trim
  hedges, shrubs, and borders" and "Trim ornamental plantings, such as topiary or screening
  hedges" are one duty written twice -- the two lists name the SAME plantings, not two groups of
  them. BUT A NARROWING MODIFIER IS NOT A DIFFERENT EXAMPLE. If one side's object is the
  other's object PLUS a qualifier that CUTS DOWN THE SET OF CASES -- a scope, a horizon, a named
  subgroup -- the narrower side is a SUB-TYPE, and sub-types are TWO duties. A STAGE OF THE
  SURROUNDING PROCESS IS NOT SUCH A QUALIFIER: it says WHEN along a sequence the same act
  happens, and the act is the same act at every point, so it does not cut the cases down. A
  different TRIGGER does cut them down -- see the dimensions above. A MODIFIER THAT NAMES WHERE
  THE ACT LANDS IS NOT A NARROWING. Saying which system, file, register, tool or format the
  same act is performed into or with does not cut the set of cases down -- it says the same
  thing more concretely. "Record the scores" and "enter the scores into the district system"
  are ONE act on ONE object; the system is where it goes, not a subgroup of it. Same for a
  named tool, a named form, or a named format. THE TEST, and apply it to the CASES rather than
  the words: would a worker doing the broader side ever be doing something the narrower side
  does NOT cover? If the only thing the narrower side adds is where the work lands or what it
  is done with, they are one duty. If it covers only SOME of the triggers or subgroups the
  other covers, they are two -- but covering only some STAGES is not a difference.
* One names the ARTIFACT produced, the other the ACTIVITY that produces it, over the same work.

ALSO SAY YES for these -- two duties, but you only reach them when an act appears on both lists,
```

```
so the lines' claims overlap and a reader could file the same activity under either. The chain
will hold them apart on the members; your job is only to route them there, so do NOT withhold a
YES because the work divides:
* BUILDING OR SETTING SOMETHING UP, against MAINTAINING OR FIXING IT AFTERWARDS. These are two
  acts separated in time. Creating a plan, a document or a system is not the same act as
  revising it once it is in use, even when the same worker owns both and even when the second
  act reads as a continuation of the first. Two lines, not one.
* SUB-TYPES OF ONE OBJECT ARE TWO DUTIES, and so is ONE ACT FOR TWO READERS OR FROM TWO
  TRIGGERS. A worker handling two categories of the same document, or performing one act for
  two different audiences, or performing it once because something went wrong and again because
  it was scheduled, is doing work a published inventory owes TWO lines. Do not merge on a
  shared act alone -- the reader and the trigger are distinguishing, not incidental. A TRIGGER
  IS NOT A STAGE, AND ONLY THE TRIGGER DIVIDES. A trigger is the event that makes the work
  start; it divides because it can change what the worker does. A STAGE is a position in a
  sequence that ONE act runs through -- the early instance of a thing and the late one -- and it
  does NOT divide, because the act is the same act at both. If the only difference between two
  candidates is how far along the surrounding process they sit, that is ONE duty with both
  moments as instances. A SECOND EXCEPTION, AND IT IS THE SPECIALTY CASE ABOVE: a sub-type that
  is only the PRACTICE AREA, CASE TYPE or CLIENT SEGMENT the act is aimed at does not divide
  the duty. Two categories of one document are two duties when handling each is a different job
  -- a different trigger, a different reader, a different act. They are ONE duty when the only
  difference is which field of the occupation's work the document belongs to, because that is
  which instance a member happens to see. Ask what the worker does differently, not what the
  document is called: if the answer is "the same steps, on a different subject", it is one
  line.
* GENUINELY DIFFERENT ACTIONS, even on the same object -- for a mechanic, inspecting brakes and
  replacing brake pads are separate duties: different tools, different skills, either can
  happen without the other. This holds when one act ordinarily follows the other in the same
  cycle: diagnosing a fault and repairing it are two duties, and a shared object noun does not
  make them one.
* Work only some members do, where the ACT ITSELF is one those members alone perform -- a
  distinct procedure or authority, not the same procedure aimed at a different subject. A
  DIFFERENT NOUN IS NOT ENOUGH. Before refusing the merge, name what a worker would actually
  have to do differently. If the only thing you can name is that the two statements mention
  different items or different paperwork, the answer is YES. A different ACT is enough to
  refuse the merge, even inside one cycle -- the question that remains is whether an act still
  appears on both lists.

ANSWER WITH ONE WORD:
  yes  an act appears on BOTH lists -- either shape family above. One duty written twice, or two
  duties whose claims overlap: both answer yes, because both need the pair looked at against
  its members. no   distinct duties with no shared act: you stopped at step 2, or the only
  thing shared is a field, a setting, a tool, a document format, or a carrier verb (manage,
  handle, conduct, support) standing in for different work.

Answer with ONLY 'yes' or 'no'.
```

P16 is sent as one user message to GPT-5.4 with up to 4,000 output tokens; its slots list the colliding pairs, every statement of the component with its member tasks, and the workers shared between statements together with strain evidence from earlier cycles. The JSON reply is validated as a partition of the component; a MERGE is applied only if a second independent draw merges the same set and the group has at most four statements.

Prompt 16: Overlap Resolution*

```
You are resolving a CLUSTER of label collisions in one occupation's published task list, for:
{{ occupation }}.

The tasks below are numbered. Every pair listed in {{ clash }} reads as one duty by its LABELS
-- cosine-close, non-disjoint, and confirmed by a second reader. Under each task's label are its
example members, the CONCRETE WORK the label is meant to describe.

A task that sits in more than one pair is a HUB, and its fate must be decided ONCE against
every neighbour together -- not pair by pair in separate calls, where a fix against one
neighbour is free to walk straight into a second, and a hub that has already absorbed one
neighbour becomes an EASIER target for absorbing a third, since merging only ever adds shared
vocabulary. That is the failure this call exists to close: decide the whole component's shape
in one pass, so no incumbent grows by more than the evidence in front of you supports.

ASSIGN EVERY NUMBERED TASK TO EXACTLY ONE OUTCOME GROUP. For each task, or each small set of
```

```
tasks that are the SAME duty, decide:

  KEEP     -- nothing about this task needs to change; none of its flagged pairs actually
  require this side to move (the other side is the one that should separate or merge instead).
  SEPARATE -- the label needs new wording to stop reading as one duty with a neighbour, but the
  underlying work is genuinely distinct. Write ONE new label for this task. A hub with three
  neighbours gets ONE rewrite that separates it from all three at once, never three sequential
  ones.
  MERGE    -- two or more of these numbered tasks describe substantially the SAME concrete work.
  Group them and write ONE statement covering the union of their activities. Say which kind of
  sameness:
              EQUIVALENT  -- neither side has work the other lacks. The group is one duty named
              several times.
              CONTAINMENT -- one member's activities are ALL inside another member's, which has
              more besides.
            PARTIAL -- each side has activities the other lacks -- is NEVER a MERGE. If the only
            true relation between two tasks is PARTIAL, put them in separate SEPARATE (or
            KEEP) groups instead. Merging on PARTIAL evidence is exactly the failure mode this
            call exists to stop: it is how one incumbent ends up doing three jobs, each
            absorbed on its own turn while looking duplicated in isolation.

{% include "task-dimensions.txt" %}

THAT TAXONOMY DECIDES WHICH DIFFERENCES MAY HOLD TWO TASKS APART HERE, and it is the same one
the pair judges upstream are held to, so a partition grounded outside it puts this call at odds
with the evidence that reached it. A difference listed under THESE DO NOT DIVIDE is not a
reason to refuse a MERGE, however unlike the two labels look -- subject matter, job type,
customer segment, the kind of artifact where its function is the same, and the counterpart's
identity are all differences that belong INSIDE one entry as its examples.

JUDGE THE ACTIVITIES, NOT THE SENTENCES. Two statements can share most of their vocabulary and
still be separate duties. Check, concretely, for every pair inside a candidate MERGE group:
  - WHAT ACT is performed -- the steps, the judgement, the skill it takes.
  - WHETHER either side's activities would still be owed if the other were done. If yes, they
    are separate -- do not fold them into the same group merely because a THIRD task in the
    component also collides with both.

WHEN YOU SEPARATE, lead the new statement with the ONE thing that divides it from its
neighbours -- the act, the object, the stage, the counterpart, or the output -- taken from its
own members, never from an industry, employer, sector, or product qualifier. THE TEST: read the
rewritten label against a NEIGHBOUR's example members. If it would still be a fair description
of that neighbour's work, it has not separated anything.

WHEN YOU MERGE, the single statement must cover what every member of the group describes. Do
not climb to a category wide enough to hold them by saying nothing, and do not simply
concatenate the labels with "and".

LABEL SIMILARITY IS NOT MERGE EVIDENCE, AND WITHOUT A CONFIRMED PAIR THE DEFAULT IS SEPARATE.
Every pair below reached you because two LABELS read alike -- that is why they were flagged, so
it cannot also be the reason to fuse them. Where no pair is listed as confirmed-duplicated work
above, treat TWO ENTRIES as the answer and make MERGE carry the burden: name the activities on
each side that are the same work, or do not merge. THE ONE EXCEPTION IS CONTAINMENT, WHICH NO
CONFIRMED PAIR CAN EVER VOUCH FOR: the work judge answers "is this the same work described
twice", and a small task sitting entirely inside a larger one is not the same work twice -- it
is a subset, so its edge arrives unconfirmed by construction. Where EVERY member of one task is
an instance of work the other task's statement and members already describe -- the same act,
merely fewer instances -- and the other task has more besides, MERGE them as
CONTAINMENT: fold the smaller into the larger and keep the larger's scope. TEST each member of
the smaller against the larger, one at a time. A single member that is a different act makes
the relation PARTIAL, and PARTIAL is never a merge. This exception folds a subset into its
whole; it never joins two peers, and it is never satisfied by the labels reading alike -- it is
satisfied member by member or not at all.
CONTAINMENT IS ALSO ANSWERABLE TO THE WORKER EVIDENCE BELOW. Folding is for STRAYS -- a few
atoms that landed under their own label by filing accident. Where the SMALLER side itself rests
on many workers -- a constituency approaching the larger's -- the corpus is showing a duty the
occupation holds at publishable scale, and "describable by the larger's statement" is only the
umbrella's breadth talking: an act a practitioner performs as its own errand, on its own
occasions, for a constituency of its own, stays its OWN entry even though a broad enough
sentence could cover it. Fold strays; never fold constituencies.
EVERY MERGE MUST RECORD ITS CASE in the group's "basis" field: for CONTAINMENT, why every
member of the smaller side is an instance of the larger's work AND why the worker evidence
reads as a stray rather than a constituency; for EQUIVALENT, the activities on each side that
are the same work. A merge whose basis cannot say these things is not one this call may make.

SEPARATE CARRIES A BURDEN OF ITS OWN: NAME THE DIVIDER. For every pair you hold apart, name the
```

```
dimension that separates them -- a DIFFERENT ACT, the SIDE OF AN EXCHANGE, or a TRIGGER THAT
CHANGES THE ACT. Record it in the group's "divider" field.
WHAT THE PRODUCT DOES IS NOT A DIVIDER -- the taxonomy does not carry it as one, and neither
does this call: any two artifacts can be told apart by their function after the fact, which
made "product function" a rubber stamp on every separation. Where the members show the same act
-- the same steps, the same judgement, the same skill -- the tasks are one duty however
differently their products function, and the products belong inside the merged statement as its
examples. Only a difference in what the WORKER DOES holds two colliding tasks apart.
AND THE ACT IS THE MOVES, NOT THE CONTEXT. Bargaining is one act: negotiating the terms of a
deal and negotiating the resolution of a dispute exchange positions, trade concessions, and
converge on agreed terms by the same steps, the same judgement, the same skill -- the deal or
the dispute is the subject the act is applied to, and a subject never divides. The same reading
applies to any verb tempted to split by its object's situation: advising on a plan and advising
on a problem, inspecting on intake and inspecting on complaint are one act each unless the
STEPS themselves change. Name an act difference only where you can say what the worker DOES
differently, move by move. A pair for which you cannot name a divider is one duty by the
taxonomy's own test, and the answer for it is MERGE, not a cautious
SEPARATE: caution that cannot say what it is protecting is not caution, it is an overlap
shipped to the published list. Label similarity still counts for nothing in either direction --
the divider must hold between the MEMBERS, not the sentences.

DO NOT LOOK FOR THE SAME ACTIVITY ON BOTH SIDES. Every activity is filed under exactly ONE
task, so the member lists below are disjoint BY CONSTRUCTION and no activity can appear twice.
The question is never "does this activity appear on both sides", it is whether the two SETS are
one duty split or two duties.

THE COUNTERPART'S IDENTITY DOES NOT DIVIDE, AND IT DOES NOT UNITE EITHER. Advising a client, a
supervising attorney and a regulator is one act with three recipients -- a different name on the
recipient is not a second duty. But neither is a shared recipient a reason to merge: two things
done FOR the same people can still be two acts. What divides is what the worker DOES and what
the product DOES.

BUILDING SOMETHING AND REVISING IT LATER ARE TWO ACTS. Creating a plan, document or system is
not the same act as adjusting it once in use -- SEPARATE, even where one worker owns both. THIS
IS THE TAXONOMY'S
DELIVERABLE-AND-ITS-AFTERLIFE CASE AND IT STOPS AT THE CHORES BULLET: acts that are the upkeep
of one ongoing responsibility -- the entries and the review of one set of books, the directing
and the appraising of one supervisory charge -- are ONE duty however far apart their occasions
fall, and this rule is not a reason to hold them apart.

TEACHING AND AUTHORISING ARE TWO ACTS. Explaining, coaching or mentoring over time is not the
same act as approving, signing off or overruling a decision in the moment, even with the same
counterpart and the same subject. If one side's activities are ongoing formation and the
other's are point-in-time decisions, SEPARATE.

A DIFFERENT LIST OF SPECIFICS IS ENOUGH ONLY WHERE THE SPECIFICS THEMSELVES DO DIFFERENT
THINGS. Where both sides' named artifacts are instances of ONE kind doing ONE job, the two
lists are not a difference -- and A DIFFERENT BODY OF RULES, SPECIALTY, OR CLIENT SITUATION IS
NEVER THE DIFFERENCE: that is the subject matter the same act is applied to, and specialists in
different subjects perform the same task on them.
ARTIFACTS THAT DO ONE JOB ARE ONE KIND: where every artifact both sides name exists to do the
same job -- each one records, establishes, or directs an arrangement its parties are bound by --
composing them is ONE duty however far apart their subjects sit and however different their
names.

{% include "onet-statement.txt" %}

{{ redundant }}

THE HETEROGENEITY RULE: FIND THE LEVEL THE OCCUPATION SHARES. An occupation spans practices,
and a colliding pair is often ONE act refracted through that heterogeneity -- the same work
appearing as one practice's version and another practice's version. The published level is the
one the whole occupation can stand behind:
  · Where the collision tracks the occupation's DIVISION OF LABOR -- different practices each
  doing their variant of the act -- the right entry is ONE task at the covering abstraction,
  with each practice's subjects as its examples. Splitting there publishes specialties, not
  tasks.
PRECEDENCE WHEN THE RULES COLLIDE: THE WORKER EVIDENCE OUTRANKS A NAMEABLE ACT DIVIDER. A
divider can be articulated for almost any pair -- that is why SEPARATE carries a burden at all --
so a difference in how the act can be described does not defeat the division-of-labor evidence.
Where the evidence line says DIFFERENT people, the pair is one act refracted through practices
and the answer is MERGE to the covering abstraction, unless the divider is one of the hard
structural ones: the SIDE OF AN EXCHANGE with an opposing party, or a DELIVERABLE AND ITS
AFTERLIFE. Weigh the evidence first; reach for a divider only when it survives it.
A SEPARATE VERDICT ON A DIFFERENT-PEOPLE PAIR MUST SHOW ITS WORK IN THE DIVIDER FIELD: name the
```

```
hard structural divider (the side of an exchange with an opposing party, a deliverable and its
afterlife) that lets it survive the precedence rule, or say concretely what the worker DOES
differently move by move. A
DIFFERENT-ACT divider that is really the same moves applied to a different SUBJECT -- a deal
against a dispute, one practice's documents against another's -- does not survive, and neither
does one that is really a different KIND OF ARTIFACT -- a formal written instrument against an
outline for live use, one filing form against another -- where composing them takes the same
judgement: either way the pair is one act refracted through the occupation's division of labor,
and the verdict for it is MERGE to the covering abstraction.
THE CORPUS WORKER SETS BELOW MEASURE THAT DIVISION OF LABOR. Disjoint workers: practice-split
variants of one act -- merge up. This is not literally about who
contributed which atom; it is the occupation's structure showing through the sample. One shared
worker of many is coincidence; weigh the proportion of the SMALLER task's workers. WHERE THE
EVIDENCE SAYS SAME PEOPLE, the shared workers are testimony that the two bodies of work COEXIST
in one job -- each held at its own scale -- which supports SEPARATE for peers and warns against a
containment fold; the same people doing both is how an occupation carries two duties, not proof
they are one.

REPEATED REWORDING FAILURE IS MERGE EVIDENCE -- WEIGHED, NOT DECISIVE. Where a pair's evidence
line records STRAIN -- reworded to separate and still reading as one duty -- the collision is
unlikely to be a wording accident: each failed rewrite strengthens the evidence that the
occupation's vocabulary has SO FAR refused to state the difference in one sentence each, and a
published level made of single sentences cannot carry a distinction its labels cannot state.
Weigh strain beside the worker evidence as support for ONE entry, in proportion to the count.
Two cautions bound it: a CROSS-CLAIM line means one label borrows vocabulary that is core to
the OTHER side's members -- that is a REPAIR case, not merge evidence, and the borrowing label
should be fixed before the collision is read as fusion; and a pair whose labels were materially
rewritten since the last failure has earned a fresh test, which the count already reflects.
Only a hard structural divider (the side of an exchange with an opposing party, a deliverable
and its afterlife) overrides strain outright.

WORKER EVIDENCE FOR EACH FLAGGED PAIR:
{{ workers }}

These id-pairs are the flagged collisions you must resolve, together: {{ clash }}

TASKS (id -- current label -- example members):
{{ tasks }}

Every id above must appear in EXACTLY ONE group below, KEEP included. Reply with ONLY JSON:
{"groups": [
  {"ids": ["<id>", ...], "decision": "KEEP"|"SEPARATE"|"MERGE",
   "relation": "EQUIVALENT"|"CONTAINMENT" (MERGE only, omit otherwise),
   "basis": "<MERGE only, REQUIRED: the member-level sameness -- and for CONTAINMENT, why the smaller side is
a stray, not a constituency>",
   "divider": "<SEPARATE and KEEP only: the dimension holding this group apart from its neighbours>",
   "statement": "<the label for this group -- omit for KEEP>"}
]}
No markdown, no explanation outside the JSON.
```

P17 is sent as one user message to GPT-5.4 mini with up to 300 output tokens and a JSON-object reply; the cached prefix lists every published statement of the occupation, and the slots carry the task, its current statement, and up to twelve of its sibling tasks. A proposed change is applied only when a second draw returns the same fate and statement.

Prompt 17: Disposer (refile)*

```
You are completing a task list for a worker in the occupation: {{ occupation }}.

THE PUBLISHED TASKS BELOW ARE FIXED. You are judging ONE activity: decide its fate under these
rules.

WHERE IT SITS NOW, AND WHY THAT MATTERS. {{ home }} {{ home_members }} READ THOSE SIBLINGS
BEFORE YOU DECIDE. They are the concrete work the home task already holds. An activity that a
worker would file alongside them belongs there, even where the statement's wording does not
reach it -- and an activity that would look foreign next to every one of them does not, however
close the words sit. An activity already filed reaches you because a coverage judge doubted its
statement describes it. That judge is measured, and on this queue MOST OF WHAT IT DOUBTS
BELONGS EXACTLY WHERE IT IS -- being asked is not evidence that anything is wrong. So when the
activity is already filed, FILE IT BACK under that same task unless another published task
```

```
plainly describes it better, or no task describes it at all. Moving it costs a family a member
and can drop it below the publishing floor; leaving it costs nothing when the fit is fair.

  FILE      the activity belongs under one published task because the task's statement AS
  WRITTEN already describes this work: a reader holding the statement would recognize the
  activity as an instance of it. File under exactly ONE task -- the one whose duty it serves,
  not merely the one sharing its words. A variant, an instance, or a specialty's version of the
  statement's own act files normally.
            A "SUCH AS ..." OR "INCLUDING ..." LIST NAMES A CATEGORY, NOT A CLOSED CHECKLIST.
            An activity of the same kind files even when it is not one of the listed examples,
            and an activity naming the listed example itself certainly files. A statement
            joining several acts with "and", "or" or a comma list is satisfied by ANY ONE of
            them. NARROWER IS FINE, AND SO IS DIFFERENT WORDING. An activity that names the
            object, the setting, the tool or the specialty more narrowly than the statement is
            an INSTANCE of it. Terse worker notes leave the purpose implicit where a statement
            spells it out; that difference is not evidence of different work.
            THE STATEMENT MUST CLAIM THE ACT ITSELF, NOT JUST THE SUBJECT: an activity that
            DECIDES what the statement only assesses, DELIVERS what it only prepares, or
            OBTAINS what it only maintains or uses, is not filable anywhere on subject match --
            it is UNFILED. The chores of the statement's own responsibility, and an activity
            naming only its meeting or call, still file normally. UNFILED   real occupation
            work no published task's statement claims. Do not force it under a nearer-sounding
            task -- but do not reach for UNFILED merely because the fit is imperfect: it is for
            work the list is MISSING, not for work the list words differently. LEAVE     a
            fragment, a state of affairs, or one worker's private arrangement.

{% include "task-dimensions.txt" %}

PUBLISHED TASKS:
{{ tasks }}

ACTIVITY: {{ atom }}

Reply ONLY with JSON: {"fate": "FILE" | "UNFILED" | "LEAVE", "task": <number or null>, "why": "<eight words>"}
```

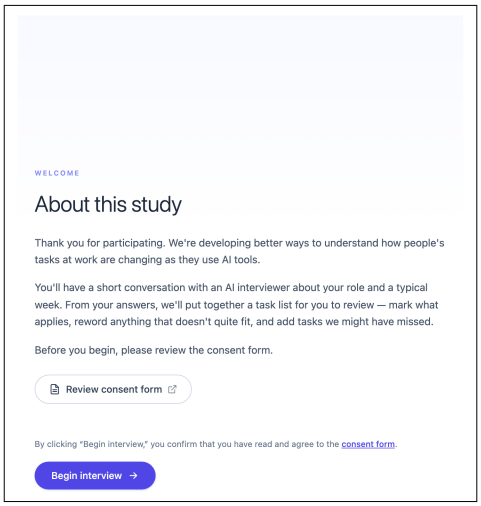
Additional prompts. Nine additional prompts are omitted for brevity but available in the code repository at https://github.com/jennjwang/tasks. These cover task extraction and consolidation (`interview/prompts/task-generator.js`), splitting, filtering, normalization, and brand removal (`prompts/processing/`), and writing evaluation, bounded repair, and sharpening (`prompts/evaluation/statement/` and `prompts/aggregation/`).

B Application Interfaces

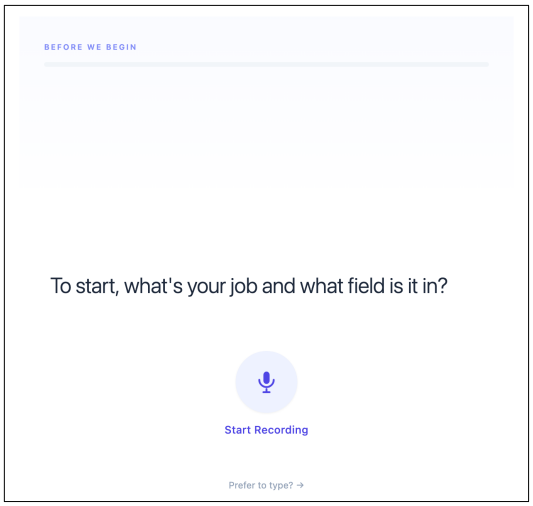
This appendix shows the two incumbent-facing applications: the discovery interview and the validation survey. The screenshots come from a scripted browser session on September 14, 2026 that walked each flow as a participant would. The interview ran with the answers of a fictitious back-end software developer and the survey ran with those of a fictitious family lawyer.

B.1 Discovery Interview

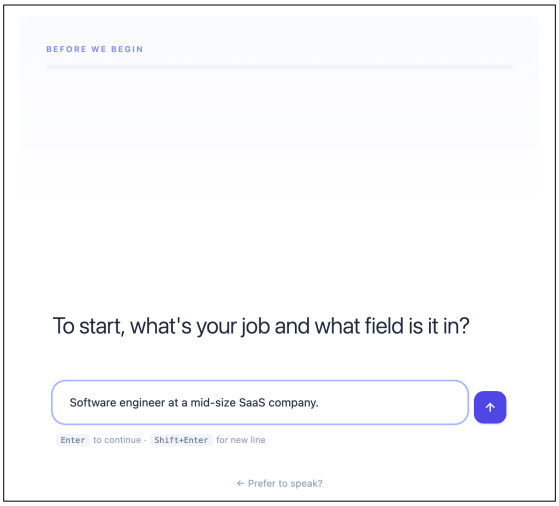
Figures 5 to 7 follow one worker from consent through the interview and the review of the generated tasks to the summary of confirmed tasks.



(a) Welcome and consent form.



(b) Opening guide question (P2) in voice mode; the link below the microphone switches to typing.

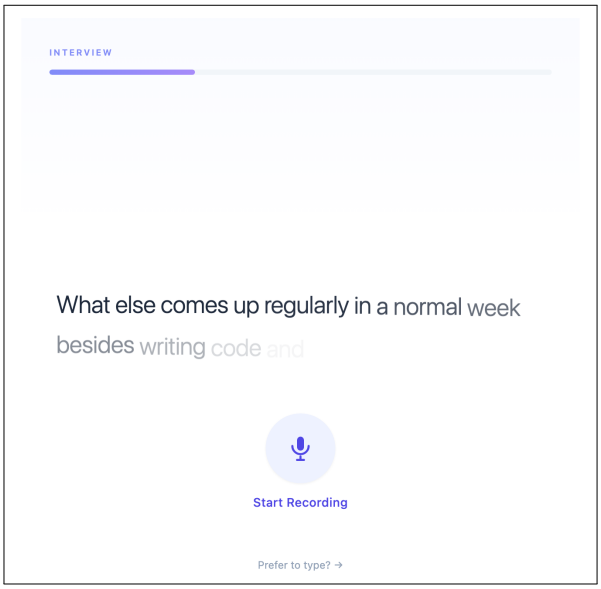


(c) The same question, typed.

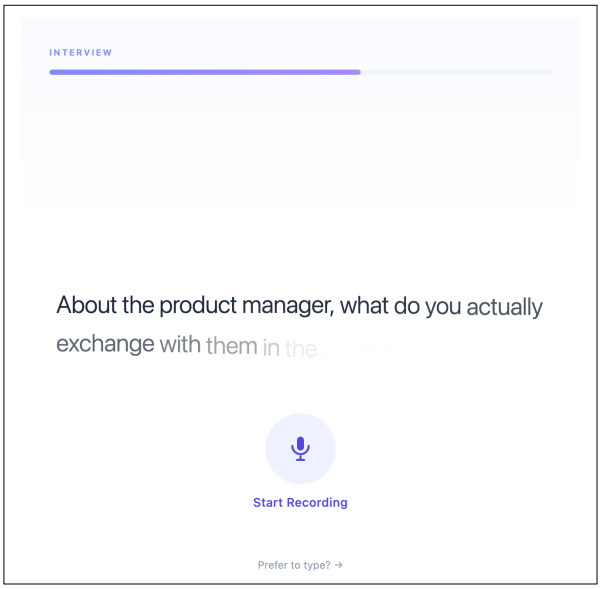


(d) Occupation self-identification from the ranked shortlist (P1); the field at the bottom takes a hint and requests other options.

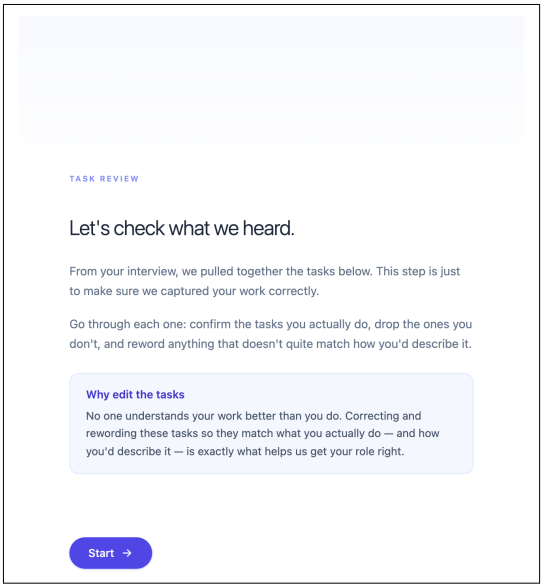
Figure 5: Discovery interview: consent, the opening question, and occupation matching.



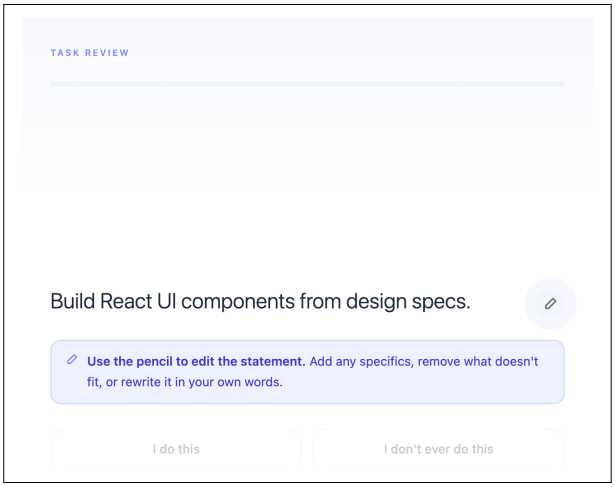
(a) Adaptive guide question on periodic work, asked once the weekly routine was covered (P4).



(b) Relational follow-up about the product manager the worker named, drafted from the inferred gaps (P5).

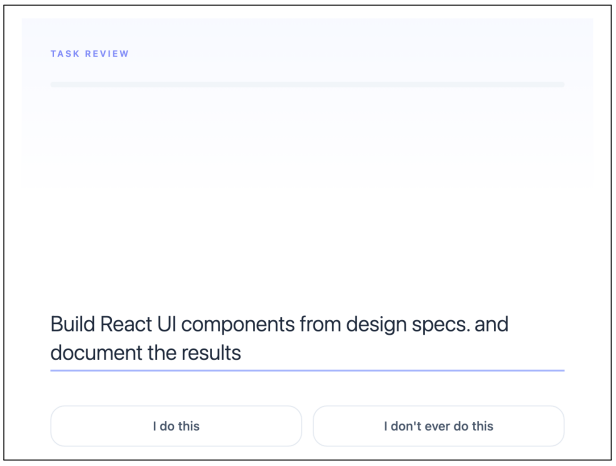


(c) Hand-off to task review: the generated tasks are framed as "what we heard".

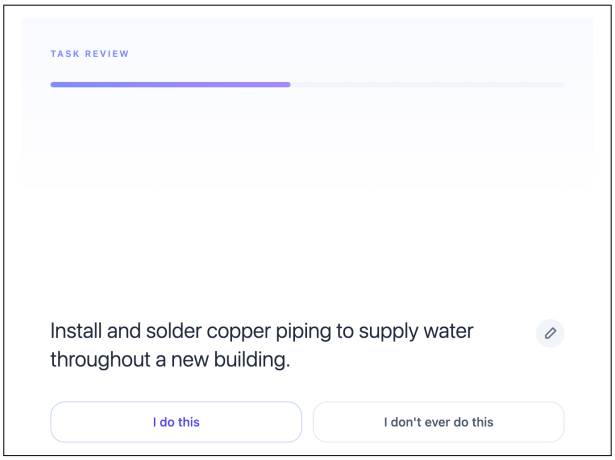


(d) Task card: confirm ("I do this"), reject ("I don't ever do this"), or edit via the pencil.

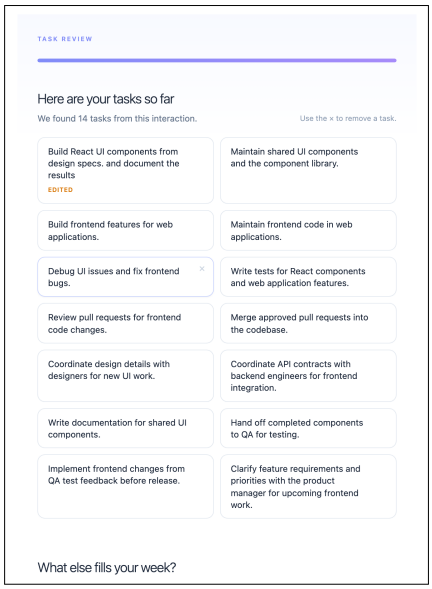
Figure 6: Discovery interview: adaptive questions and hand-off to task review. The bar at the top of (a) and (b) shows the position in the guide.



(a) The card of fig. 6d in edit mode, with the worker's rewording.

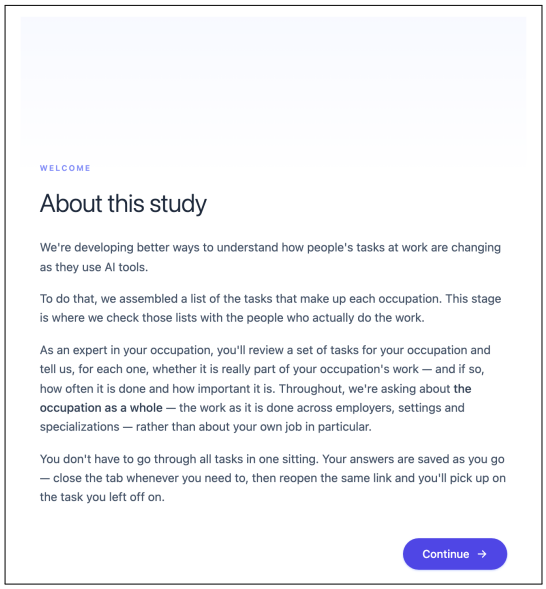


(b) Attention-check card from an unrelated occupation; confirming it screens the worker out.

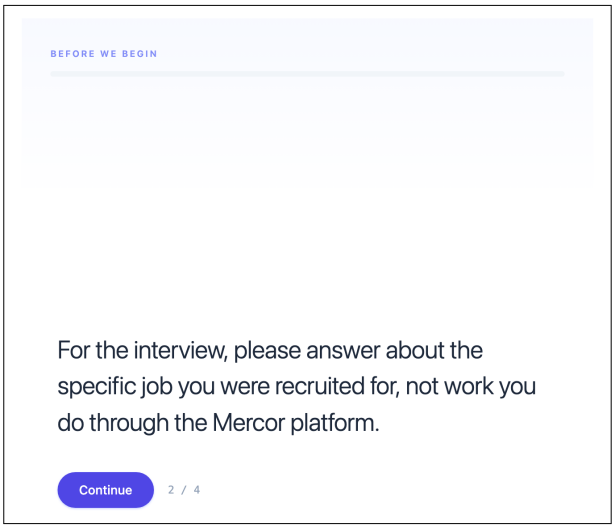


(c) Summary of confirmed tasks (edited ones marked), with removal on hover and a field for adding missing tasks.

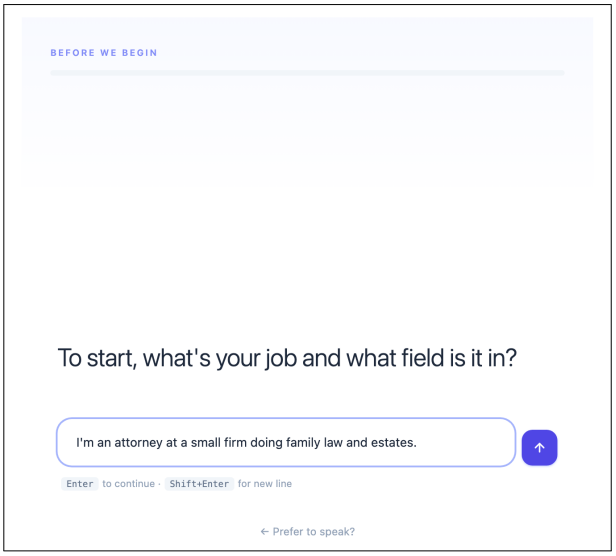
Figure 7: Discovery interview: task verification and summary.



(a) Welcome: the survey concerns the occupation as a whole, not the participant's own job.



(b) Lead-in: only the screener questions concern the participant's own job.



(c) A screener question, typed; the answers feed the occupation match.

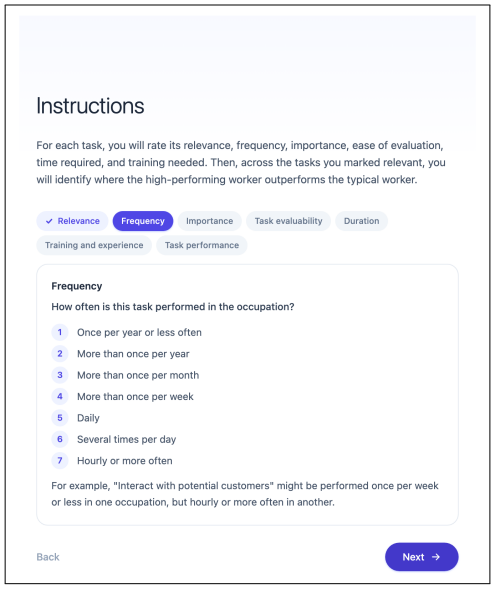


(d) Occupation pick from the ranked candidates, with search over the full taxonomy.

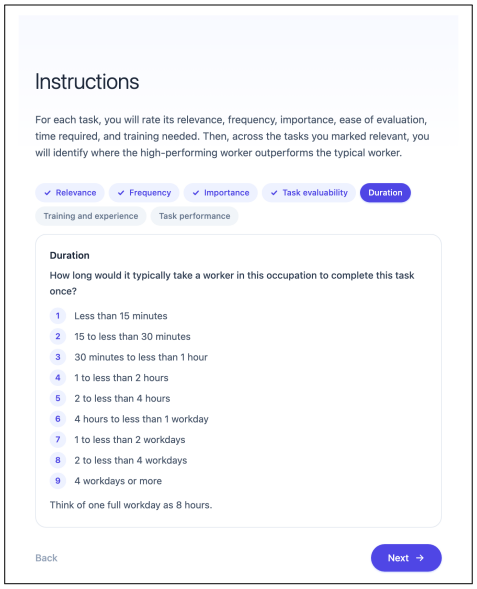
Figure 8: Validation survey: welcome, screener, and occupation pick.

B.2 Validation Survey

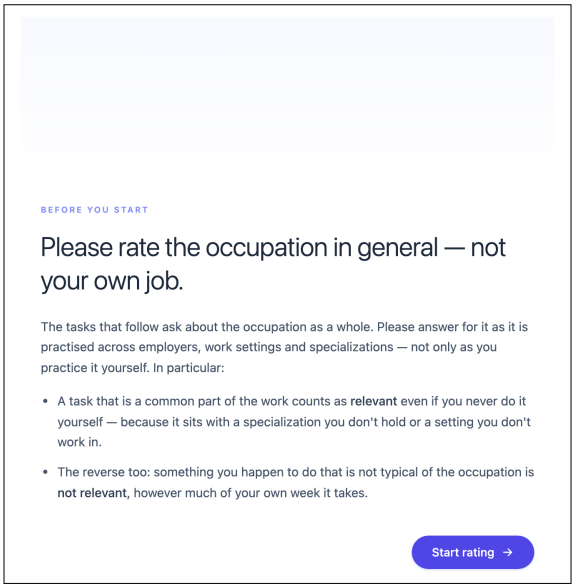
Figures 8 to 10 follow one rater through the screener, the instructions, and the per-statement items. The design behind the survey is described in Section E.2.



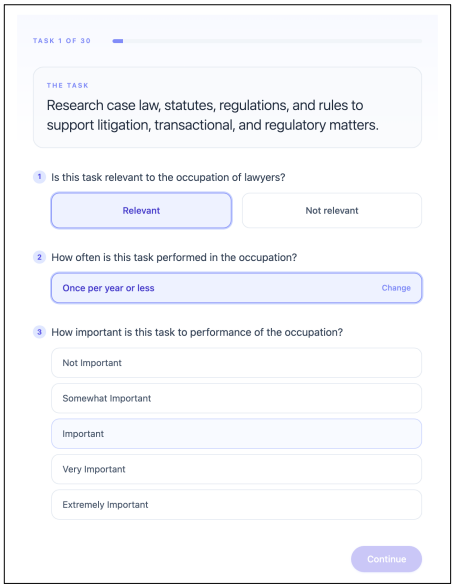
(a) Instructions, one tab per item; the relevance tab defines the O*NET relevance question.



(b) Instructions for the two evaluability items (codifiability and verifiability). All seven tabs must be opened before the survey starts.

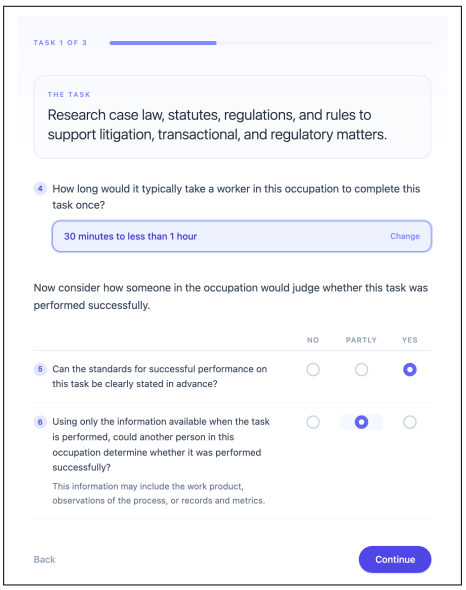


(c) Scope reminder before the first task: rate the occupation in general.

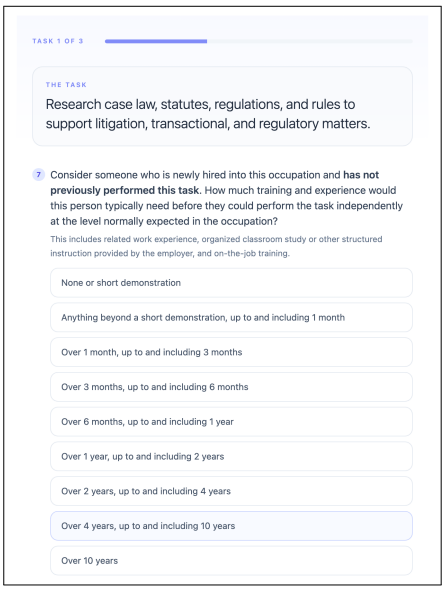


(d) Task card, part one: relevance, then frequency and importance revealed in turn. "Not relevant" skips the remaining parts.

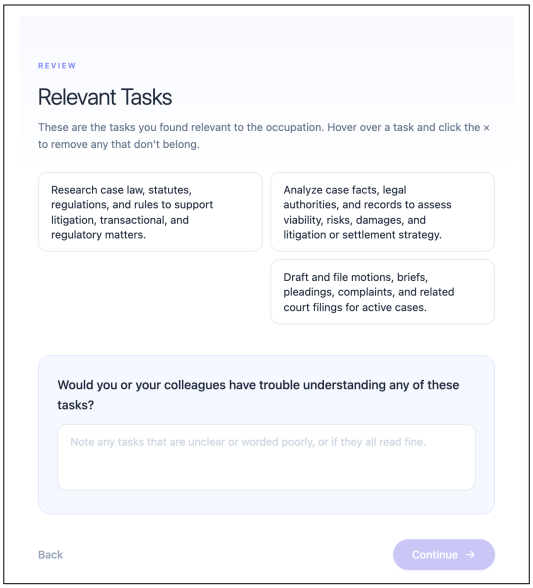
Figure 9: Validation survey: instructions, scope reminder, and the first part of a task card.



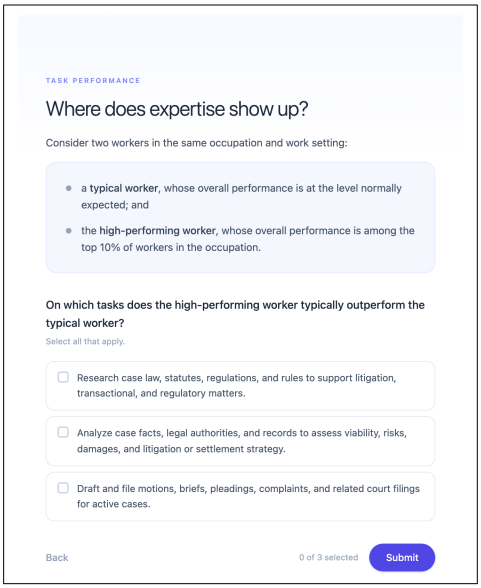
(a) Task card, part two: duration, and whether standards can be stated in advance and performance verified from the information available.



(b) Task card, part three: task-specific training and experience on the nine-point SVP scale.



(c) Review of the tasks rated relevant, with a required note on wording problems.



(d) Task performance: on which relevant tasks a top-decile worker outperforms a typical worker.

Figure 10: Validation survey: the rest of the task card, review, and task performance (Section 3.3).

C Sample Statistics

This appendix describes participants in the six study occupations, separately for task discovery and task validation (Table 5). Respondents who participated in both phases appear in each phase's summary. Discovery statistics reflect participation as of September 14, 2026. Task-validation statistics include surveys completed as of September 28, 2026: 323 completed records, of which 322 passed the attention check. These descriptive samples may differ from the smaller analysis samples used elsewhere in the paper.

Participant records and linkage. Mercor matched survey identifiers to participant profiles and resumes using recruitment records. Of the 322 validation records that passed the attention check, 319 were linked successfully. Occupation assignments follow the survey task lists. We used GPT-5.6 Sol to extract employers, roles, and employment intervals from resumes, then separately classify each role's relevance to the survey occupation.

Occupational tenure. Tenure measures the total time a respondent worked in roles relevant to the survey occupation, across employers and up to survey completion, as documented in their resume. Time spent in concurrent jobs is counted only once. Part-time, consulting, and self-employed work is included without adjusting for hours worked. Month-only dates cover the stated months, and year-only dates use midyear endpoints. We sum available dated intervals and exclude records with none, yielding tenure estimates for 297 of 322 validation records. These estimates may not capture complete career histories. For task-discovery Dishwashers, relevant experience also includes bussing and related food-service or equipment-sanitation work.

Employer matching and characteristics. We used model-assisted public web searches to identify relevant employers' LinkedIn company pages and corroborating sources, including official websites, based on employer names, locations, roles, and worksite details from resumes. We retained URLs and supporting passages and separately reviewed employer identity, headcount, and industry, requiring at least medium model confidence in both identity and each characteristic. These are evidence-based model assessments, not independent human verification. Six-digit 2022 NAICS codes identify the employer's primary business activity, not the participant's occupation, and were either source-reported or inferred from documented activity.

Employer-size and industry distributions. Employer size measures company-wide employees in the available source's reporting period, which may differ from the headcount at the respondent's worksite or during their employment. Each participant's resume may list several employers where they performed work relevant to the survey occupation. We assign each employer to a size category, then divide the participant's weight equally across their distinct relevant employers. This gives every participant the same total weight regardless of how many employers they list. Within each phase and occupation, we sum these contributions to calculate the size distribution. Missing employer or size information counts as Unknown.

Industry shares use the same weighting. Modal NAICS is the most common identified industry code, and its share is calculated over all participant weight, including Unknown. Industry and employer size are assessed separately, so one may be known even when the other is not.

Interview duration. Table 6 reports how long the 132 discovery interviews behind the inventories took. Participants spent a median of 29 minutes (interquartile range 19–46) in the interview and

Table 5: Participant characteristics and employer-size distributions by study phase

		Tenure (years)		Employer size (employees): weighted share (%)								
Occupation	n	Median [25th, 75th]	Modal NAICS	1–10	11–50	51–200	201–500	501–1,000	1,001–5,000	5,001–10,000	10,001+	Unknown
Panel A. Task discovery												
Dishwashers	25	3.8 [3.1, 6.9]	722511 (28.1%)	0.0	0.4	3.0	0.0	2.0	5.3	0.0	14.2	75.0
Elementary School Teachers	30	12.0 [8.0, 18.0]	611110 (88.1%)	0.0	1.1	6.4	5.0	11.1	21.4	6.1	11.7	37.2
Emergency Medicine Physicians	32	12.6 [8.6, 17.1]	622110 (45.9%)	0.0	3.2	1.4	2.2	1.2	9.6	8.2	14.0	60.2
Lawyers	42	10.0 [6.3, 14.8]	541110 (44.7%)	8.8	9.4	10.1	12.9	4.6	6.2	1.5	1.3	45.3
Personal Care Aides	24	6.2 [3.4, 11.4]	624120 (17.0%)	4.2	1.0	6.2	6.2	4.2	10.8	0.0	0.0	67.4
Software Developers	30	8.0 [5.6, 11.9]	513210 (16.7%)	2.8	4.6	5.0	4.3	2.9	10.4	4.4	28.4	37.2
Panel B. Task validation												
Dishwashers	32	2.9 [2.3, 4.7]	722511 (24.5%)	0.0	3.1	6.8	0.0	3.1	3.1	0.0	6.2	77.6
Elementary School Teachers	54	12.1 [8.1, 17.2]	611110 (87.3%)	0.0	0.2	6.8	6.1	5.8	16.7	9.6	8.0	46.9
Emergency Medicine Physicians	57	11.8 [7.2, 17.0]	622110 (45.0%)	0.1	2.8	1.0	2.7	4.3	14.3	7.1	10.7	57.0
Lawyers	69	11.4 [6.6, 16.0]	541110 (43.7%)	7.0	8.4	7.8	10.3	3.8	12.4	1.6	2.4	46.2
Personal Care Aides	53	4.9 [3.1, 8.1]	624120 (9.4%)	1.9	1.1	3.8	7.5	4.4	6.8	0.9	0.0	73.6
Software Developers	57	9.2 [6.1, 14.2]	513210 (18.5%)	4.8	8.4	7.1	6.3	5.9	11.9	3.3	21.7	30.6

task review combined: a median of 25 minutes (16–40) answering interview questions and 3 minutes reviewing the resulting task statements. Interviews are longest for Software Developers and Elementary School Teachers and shortest for Dishwashers and Personal Care Aides. Participants answered a median of 16 questions (range 9–17). Elapsed times include breaks, since participants could pause and resume, so we report medians rather than means.

Table 6: **Discovery interviews take a median of 29 minutes.** Elapsed minutes per participant for the 132 interviews that contribute to the inventories, as median [interquartile range]. *Interview*: from the first interview question to the hand-off to task review. *Task review*: confirming, editing, and adding tasks. *Total*: interview plus task review, per participant; medians do not sum across columns. Questions: answered interview questions (budget 17).

Occupation	n	Interview	Task review	Total	Questions
Software Developers	21	32.1 [17.4, 49.5]	3.9 [2.0, 6.3]	42.9 [20.8, 53.9]	16.0
Lawyers	22	23.1 [18.8, 32.8]	3.8 [2.5, 5.6]	27.0 [21.7, 40.2]	15.0
Elementary School Teachers	24	32.4 [20.8, 49.6]	3.4 [2.3, 6.3]	37.3 [24.0, 54.5]	16.5
Emergency Medicine Physicians	25	26.7 [15.7, 41.8]	4.4 [2.9, 6.2]	29.6 [20.9, 47.5]	16.0
Personal Care Aides	20	20.0 [13.9, 35.3]	2.8 [1.8, 3.2]	23.1 [15.6, 40.3]	17.0
Dishwashers	20	16.9 [8.5, 25.6]	3.0 [1.9, 4.9]	23.2 [13.6, 29.5]	14.0
All	132	25.3 [16.1, 40.5]	3.3 [2.3, 5.5]	29.2 [19.3, 46.2]	16.0

D Analyst Review

Table 7 reports the review of Section 5.1 by occupation. The first authors checked all 235 candidate statements against their worker reports and edited 45. The analysts then judged all 229 statements they reviewed to be tasks. They flagged 67 as not meeting O*NET's writing guidelines, most often for their level of specificity (42 statements), followed by combining two activities in one statement (14), lacking face validity (11), and naming a specialty rather than the work (3); a statement can carry several flags. They judged 35 statements to describe work not primarily performed by the occupation, mostly work of a neighboring occupation such as paralegals, legal secretaries, nurses, or

dining room attendants. After discussion with the analysts, 26 of these were classified as adjacent. In response to the review, we reworded 48 statements, split 3 into 7, and merged 3 into others, which yields the 230 statements rated by workers.

Three examples show the typical revisions. For Emergency Medicine Physicians, "Discuss goals of care, treatment aggressiveness, and code status with patients, families, or surrogates during end-of-life care decisions" was judged too specific and jargon-heavy and became "Discuss goals of care, treatment preferences, and resuscitation decisions with patients, families, or surrogates." For Software Developers, "Review teammates' code changes for correctness, readability, security, and maintainability, and approve or request changes on pull requests" was judged specific to GitHub-like environments and became "Review code changes for correctness, readability, security, or maintainability, and provide feedback to other developers." For Dishwashers, "Prepare food items such as vegetables, meats, or salads" could be read as cooking and overlapped with Food Preparation Workers, and became "Clean or prepare various food items for cooking or serving."

Table 7: **Author check and analyst review by occupation.** Candidates: statements produced by aggregation. Author edits: statements the first authors revised, split, merged, or dropped after checking them against the worker reports. Writing flag: the analysts judged the statement not to meet O*NET's writing guidelines. Not primary: the analysts judged the work not primarily performed by the occupation. Adjacent: statements excluded from the occupational list after re-review. Edited: statements revised, split, or merged after the analysts' feedback.

	Author check		Analyst review				
Occupation	Candidates	Edits	Reviewed	Writing flag	Not primary	Adjacent	Edited
Software Developers	47	7	46	8	6	2	6
Lawyers	47	8	47	13	18	16	10
Elementary School Teachers	37	11	37	15	0	0	14
Emergency Medicine Physicians	45	7	44	15	6	4	17
Personal Care Aides	40	7	36	11	0	0	2
Dishwashers	19	5	19	5	5	4	5
Total	235	45	229	67	35	26	54

D.1 Core and Supplemental Tasks by Occupational Fit

Table 8 splits our statements by the O*NET Center's judgment of whether each describes work primarily performed by the occupation (occupational) or work that belongs to an adjacent occupation (adjacent). Occupational statements are more likely to be core: $89\,\%$ of the 202 occupational statements are core, against $54\,\%$ of the 26 adjacent statements, a difference of 35 percentage points. Nonetheless, a substantial fraction of adjacent statements are validated as tasks, and more than half are core, so incumbents judge much of the work the analysts assign to other occupations as relevant to their own. Emergency Medicine Physicians are the exception: all four adjacent statements are core.

Table 8: **Occupational statements are more likely than adjacent statements to be core.** Statements are split by the O*NET Center's judgment of whether each describes work primarily performed by the occupation (occupational) or work that belongs to an adjacent occupation (adjacent). Occupational statements are the 202 retained after the relevance filter. Cells give the number of statements and, in parentheses, their share of the column's n. Supp. = supplemental; – indicates no adjacent statements.

	Occupational				Adjacent			
Occupation	n	Core	Supp.	Non-task	n	Core	Supp.	Non-task
Software Developers	44	40 (91%)	4 (9%)	0 (0%)	2	1 (50%)	1 (50%)	0 (0%)
Lawyers	32	28 (88%)	4 (12%)	0 (0%)	16	8 (50%)	6 (38%)	2 (12%)
Elementary School Teachers	39	37 (95%)	2 (5%)	0 (0%)	–	–	–	–
Emergency Medicine Physicians	38	36 (95%)	2 (5%)	0 (0%)	4	4 (100%)	0 (0%)	0 (0%)
Personal Care Aides	36	30 (83%)	6 (17%)	0 (0%)	–	–	–	–
Dishwashers	13	9 (69%)	4 (31%)	0 (0%)	4	1 (25%)	0 (0%)	3 (75%)
All occupations	202	180 (89%)	22 (11%)	0 (0%)	26	14 (54%)	7 (27%)	5 (19%)

E Validation Survey

This appendix documents the validation survey (Section 3.3): its items and assignment (Section E.1), how ratings are turned into estimates and classifications (Section E.2), and how stable those classifications are (Sections E.3 and E.4).

E.1 Survey Items

Task assignment To balance statistical power with survey fatigue, each incumbent rates 30 tasks or 60 % of the full task list, whichever is fewer (Jeong et al., 2023). Tasks are selected adaptively, prioritizing those with the fewest ratings so far, with ties broken at random. This distributes ratings relatively evenly across tasks without requiring each respondent to rate the full list.

Relevance Relevance asks whether incumbents view the task as relevant for the occupation. This elicits the incumbent's concept of their occupation, as opposed to their own work, which we chose to maintain compatibility with the O*NET system and to control the impact of our sampling frame on outcomes.[7] If a task fails to be rated as relevant by at least 25 % of raters, we drop it from the task list, following O*NET's task definition (National Center for O*NET Development, 2026c).

Importance Importance elicits how important a task is to the profession on a five-point scale following O*NET tradition (PETERSON et al., 2001). As in O*NET, we use relevance and importance to determine whether tasks are considered *core* or *supplemental*. Tasks for which the mean probability of relevance is at least 67 % and for which estimated mean importance is at least 3 on a 5-point scale are declared as *core*; others are *supplemental*.

Frequency Frequency asks how often workers engage with the task. Prior work has associated task frequency with task routineness (Autor et al., 2003). It is therefore a natural component of

[7] An alternative is the Percent Members Performing literature from military task inventories; compare (Christal, 1974, Christal and Weissmuller, 1988, Ramsey-Klee, 1984), and literature that relates this to actual time spent on a task and within-occupation heterogeneity (Autor and Handel, 2013, Wilson and Harvey, 1990).

usage-based exposure measures, as in Handa et al. (2025), Patwardhan et al. (2025), Tomlinson et al. (2025)

Time Duration Duration measures the time required to complete a task. It has a long history in manpower analysis (Moore, 1976) and has recently gained attention in research on AI exposure (Eloundou et al., 2024, Kwa and Cheng, 2025, Patwardhan et al., 2025). For example, Kwa et al. (2025) find that AI success rates decline with human completion time and that the horizon for $50\,\%$ success has doubled roughly every seven months since 2019. We combine incumbents' duration and frequency estimates to approximate time spent on each task, and normalize across tasks to obtain time shares for weighting AI exposure. These shares are approximate, as respondents may underestimate task duration (Buehler et al., 1994, Roy et al., 2005), and continuous or monitoring tasks may lack a clear endpoint.

Articulability and Verifiability We ask whether the standards for successful task performance can be clearly stated in advance (*articulability*) and whether a trained coworker can assess the resulting output (*verifiability*). Articulability informs the scope for computer substitution (Autor et al., 2003, Levy and Murnane, 2004) and clarifies the role of tacit knowledge (Autor, 2014, 2015, Polanyi, 1966) in task performance. Verifiability has been related to AI exposure (Tomei and Teeselink, 2026).

Absolute and Relative Expertise Our final two items capture expertise in absolute and relative terms: the training a task demands, and how sharply performance separates top workers from the median. The first is a task-specific Specific Vocational Preparation (SVP) measure (U.S. Department of Labor, Employment and Training Administration, 1991), the total on-the-job and classroom training required to perform a task. It informs how jobs can be unbundled into expert and non-expert tasks and where the returns to expertise are concentrated (Autor and Thompson, 2025, Garicano, 2000). The second item measures the performance gap between a typical worker and a high performer, the top 10% within the same occupation and work setting. This dispersion has long figured in models of organizational performance (Aguinis and O'Boyle, 2014, Huselid et al., 2005), while recent field studies find that AI compresses it, narrowing the gap between workers who were most and least productive before its introduction (Brynjolfsson et al., 2025b, Dell'Acqua et al., 2026, Noy and Zhang, 2023, Peng et al., 2023).

E.2 Estimation

Estimator. Raters who fail the attention check, which shows statements from an unrelated occupation, are excluded. Let Y_{tk} be rater t's answer for statement k and $\xi_{tk} = 1$ if t rated k. Because statements are assigned to the least-rated first with random ties, every statement on a rater's list has the same probability of being assigned, $\pi_t = n_t/K$, where n_t is the number of statements t rated and K the length of the list. Each mean is estimated by inverse-probability weighting with normalized weights,

$$\hat{\mu}_k = \sum_t w_{tk}\, Y_{tk}, \qquad w_{tk} = \frac{\xi_{tk}/\pi_t}{\sum_s \xi_{sk}/\pi_s}.$$

Relevance is binary, so its mean is the probability of relevance. Importance (1–5) and the other items are asked only of raters who answered yes, so they are estimated as ratios $\mathbb{E}[Y^{\text{item}} Y^{\text{rel}}]/\mathbb{E}[Y^{\text{rel}}]$, with numerator and denominator each estimated as above. Standard errors come from the influence functions of these estimators, clustered by rater (with the delta method for ratios), and 95% intervals are $\hat{\mu} \pm 1.96\,\widehat{\text{SE}}$, clipped to the item's range.

Classification. A statement is a task if its estimated relevance is at least 0.25, and a core task if in addition relevance is at least 0.67 and importance at least 3; other tasks are supplemental. For the certified classification (Section E.3), a statement is a non-task if the upper end of its relevance interval is below 0.25 and undetermined if only the lower end is. It is certified core if the lower ends of both intervals clear the core thresholds and certified supplemental if the upper end of either falls below them; otherwise its core status is undetermined. Estimation uses the full list, and the tables count the statements the O*NET Centerjudged occupational and that pass the relevance filter.

E.3 Task Determinations at 95% Confidence

O*NET classifies tasks as core or supplemental by point estimate, applying its thresholds to the sample mean of each rating. With our sample sizes, a statement near a threshold can change class with a few ratings, so we also report classifications that account for sampling uncertainty.

Table 9 applies the task and core rules to both ends of each statement's $95\,\%$ confidence intervals for relevance and importance. A statement is certified in a class only when both ends agree; otherwise it is undetermined. Nearly all statements are certified as tasks, but the line between core and supplemental is often not settled at conventional confidence levels.

Table 9: **Nearly all statements are certified as tasks at $95\,\%$ confidence, but the line between core and supplemental is frequently undetermined.** The task and core rules are applied to both ends of each statement's $95\,\%$ confidence intervals for relevance and importance. A statement is certified in a class only when both ends agree; otherwise it is undetermined. Counts cover the 202 statements retained after the relevance filter. Statements whose task status is undetermined are counted under Undet., so Core, Suppl., and Undet. sum to Total minus Non-tasks. Suppl. = supplemental. Undet. = undetermined.

Occupation	Total	Non-tasks	Tasks	Core	Suppl.	Undet.
Software Developers	44	0	44	32	0	12
Lawyers	32	0	32	25	2	5
Elementary School Teachers	39	0	39	30	0	9
Emergency Medicine Physicians	38	0	38	32	0	6
Personal Care Aides	36	0	36	26	2	8
Dishwashers	13	0	10	7	0	6

E.4 Task Determinations with Half of the Raters

Table 10 shows how sensitive the point classification is to the size of the rater panel. For each occupation, we draw 200 random subsets containing half its raters, without replacement, and re-estimate every statement on each subset (Section E.2). We then apply the same task classification thresholds and compare the resulting labels with those from the full panel.

With half the raters, nearly every statement outside Dishwashers remains a task, and 95% to 98% of statements retain their full-panel class on average. Changes are almost entirely between core and supplemental. Dishwashers, with 15 raters per subset, are the least stable (90% agreement): several statements lie near the task threshold and switch between task and non-task. Median confidence-interval widths increase by factors of 1.45 for relevance and 1.38 for importance, close to the $\sqrt{2} \approx 1.41$ expected when halving the sample size.

Table 10: **Task classifications are stable when the rater panel is halved.** For each occupation, 200 random subsets of half its raters are drawn without replacement and the classification re-estimated. Counts cover the 202 statements retained after the relevance filter and are means across the 200 subsets. Raters shows full-panel → subset size. Agreement is the mean share of statements retaining their full-panel class. Suppl. = supplemental.

			Mean count over subsets				
Occupation	Raters	Total	Non-tasks	Tasks	Core	Suppl.	Agreement
Software Developers	56 → 28	44	0.0	44.0	39.5	4.5	95%
Lawyers	69 → 34	32	0.0	32.0	28.0	4.0	97%
Elementary School Teachers	54 → 27	39	0.0	39.0	36.2	2.8	95%
Emergency Medicine Physicians	57 → 28	38	0.0	38.0	36.1	1.9	98%
Personal Care Aides	51 → 25	36	0.0	36.0	29.7	6.3	96%
Dishwashers	31 → 15	13	0.7	12.3	8.5	3.8	90%

F TaskNet Inventories by Occupation

Table 11: **Task statements for Dishwashers.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Wash dishes, glassware, flatware, pots, or pans, using dishwashers or by hand.	*refined by*	Load dishes, glassware, flatware, and cookware into racks and run them through the dish machine.
	refined by	Retrieve soiled pots, pans, dishes, and utensils from kitchen work areas and bring them to the dishwashing area.
	refined by	Scrape food debris from dirty dishes, glassware, and cookware, then sort and pre-rinse them into racks before machine washing.
	overlaps	Hand wash cookware, pots, pans, and utensils, sanitizing them after cleaning.
Place clean dishes, utensils, or cooking equipment in storage areas.	*overlaps*	Dry and restock clean dishes, glassware, flatware, and cookware in their designated kitchen and service areas.
Maintain kitchen work areas, equipment, or utensils in clean and orderly condition.	*refined by*	Clean and sanitize food-preparation surfaces, countertops, walls, and kitchen equipment to keep cooking areas free of grease and contamination.
	refined by	Hand wash cookware, pots, pans, and utensils, sanitizing them after cleaning.
	refined by	Maintain the dish machine, sinks, drains, and surrounding work area throughout and between shifts.
	refined by	Remove trash and replace can liners in the kitchen, dish area, and dining room.

continued on next page

Dishwashers (continued)

O*NET task	Relation	TaskNet statement
	refined by	Retrieve soiled pots, pans, dishes, and utensils from kitchen work areas and bring them to the dishwashing area.
Stock supplies, such as food or utensils, in serving stations, cupboards, refrigerators, or salad bars.	*overlaps*	Dry and restock clean dishes, glassware, flatware, and cookware in their designated kitchen and service areas.
	overlaps	Retrieve and redistribute supplies, ingredients, and utensils between on-site storage areas and kitchen or dining service stations.
Sweep or scrub floors.	*overlaps*	Sweep, scrub, and mop kitchen and dining area floors, placing wet-floor signs and squeegeeing away standing water.
Sort and remove trash, placing it in designated pickup areas.	*overlaps*	Remove trash and replace can liners in the kitchen, dish area, and dining room.
Transfer supplies or equipment between storage and work areas, by hand or using hand trucks.	*overlaps*	Retrieve and redistribute supplies, ingredients, and utensils between on-site storage areas and kitchen or dining service stations.
Receive and store supplies.	*overlaps*	Receive, unload, and store incoming deliveries of food, beverages, tableware, and cleaning supplies in designated storage areas.
Load or unload trucks that deliver or pick up food or supplies.	*overlaps*	Receive, unload, and store incoming deliveries of food, beverages, tableware, and cleaning supplies in designated storage areas.
Clean and sanitize the dining area, including tables.	*overlaps*	Clear, wipe, and reset dining tables and chairs between customers.
O*NET-only tasks (no TaskNet counterpart)		
Clean garbage cans with water or steam.		
Clean or prepare various foods for cooking or serving.		
Set up banquet tables.		
Prepare and package individual place settings.		
TaskNet-only tasks (no O*NET counterpart)		
Move chairs, tables, and other dining-area furniture to assist with cleaning, closing, or storage tasks, including stacking chairs at the end of the night.		

Table 12: **Task statements for Elementary School Teachers.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Establish and enforce rules for behavior and procedures for maintaining order among the students.	*overlaps*	Establish and enforce classroom rules, routines, and behavior expectations.

continued on next page

Elementary School Teachers (continued)

O*NET task	Relation	TaskNet statement
Instruct students individually and in groups, using teaching methods such as lectures, discussions, and demonstrations.	*refined by*	Assign independent, small-group, or computer-based activities to students.
	refined by	Monitor student understanding during lessons and adjust instruction in real time.
	refined by	Prepare students for standardized tests by reviewing content, practicing test questions, and discussing errors or areas of difficulty.
	refined by	Teach social-emotional skills, such as recognizing and managing emotions, demonstrating empathy and kindness, and solving interpersonal problems.
	overlaps	Teach daily lessons across core subjects, introducing concepts, modeling skills, and guiding independent practice.
	overlaps	Teach small-group and one-on-one lessons to reinforce or extend skills based on student need.
Confer with parents or guardians, teachers, counselors, and administrators to resolve students' behavioral and academic problems.	*overlaps*	Collaborate with teachers, specialists, counselors, and administrators to coordinate student support plans, accommodations, and intervention services.
	overlaps	Hold parent-teacher conferences to discuss student progress, strengths, and support needs with families.
Provide a variety of materials and resources for children to explore, manipulate, and use, both in learning activities and in imaginative play.	*overlaps*	Produce materials, such as slide decks, graphic organizers, or manipulatives, for lessons and classroom activities.
Administer standardized ability and achievement tests, and interpret results to determine student strengths and needs.	*overlaps*	Administer assessments, such as quizzes, exit tickets, or benchmark tests, to evaluate student learning and progress.
	overlaps	Analyze assessment data to place students for targeted instruction and intervention.
	overlaps	Analyze student performance data, assessment results, and classroom work to identify learning gaps, track progress, and determine next instructional steps.
Maintain accurate and complete student records as required by laws, district policies, and administrative regulations.	*overlaps*	Maintain student records, such as attendance logs, gradebooks, report cards, and portfolios.
Organize and label materials and display students' work.	*overlaps*	Set up, organize, and maintain the classroom environment, including displays and supplies.
Meet with parents and guardians to discuss their children's progress and to determine priorities for their children and their resource needs.	*overlaps*	Hold parent-teacher conferences to discuss student progress, strengths, and support needs with families.
Observe and evaluate students' performance, behavior, social development, and physical health.	*refined by*	Monitor student health and refer those feeling unwell to the school nurse.

continued on next page

Elementary School Teachers (continued)

O*NET task	Relation	TaskNet statement
Prepare, administer, and grade tests and assignments to evaluate students' progress.	*refined by*	Administer assessments, such as quizzes, exit tickets, or benchmark tests, to evaluate student learning and progress.
	refined by	Grade student work and provide written or verbal feedback on strengths, progress, and next steps.
	overlaps	Design classroom assessments, such as quizzes, projects, or unit tests, aligned with intended learning outcomes and content standards.
Confer with other staff members to plan and schedule lessons promoting learning, following approved curricula.	*refined by*	Plan intervention or enrichment lessons targeting prerequisite skills and unmastered content.
	overlaps	Meet with grade-level teammates to review student data, lesson pacing, and instructional plans.
	overlaps	Meet with grade-level teams to plan lessons and coordinate instruction.
	overlaps	Plan daily and weekly lessons across core subjects, aligned with state standards, pacing guides, and student assessment data.
Attend professional meetings, educational conferences, and teacher training workshops to maintain and improve professional competence.	*overlaps*	Attend professional development sessions, trainings, and professional learning communities to improve teaching practice.
Plan and supervise class projects, field trips, visits by guest speakers or other experiential activities, and guide students in learning from those activities.	*overlaps*	Coordinate field trips, performances, or other school activities and events such as open houses and family nights.
Select, store, order, issue, and inventory classroom equipment, materials, and supplies.	*overlaps*	Acquire books and supplies for classroom use, sourcing from second-hand stores, family donations, or grants.
	overlaps	Set up, organize, and maintain the classroom environment, including displays and supplies.
Perform administrative duties, such as school library assistance, hall and cafeteria monitoring, and bus loading and unloading.	*overlaps*	Supervise students in hallways, the cafeteria, recess, and bus lines throughout the school day.
Prepare objectives and outlines for courses of study, following curriculum guidelines or requirements of states and schools.	*overlaps*	Plan daily and weekly lessons across core subjects, aligned with state standards, pacing guides, and student assessment data.
Meet with other professionals to discuss individual students' needs and progress.	*overlaps*	Collaborate with teachers, specialists, counselors, and administrators to coordinate student support plans, accommodations, and intervention services.
	overlaps	Meet with grade-level teammates to review student data, lesson pacing, and instructional plans.
Prepare reports on students and activities as required by administration.	*overlaps*	Prepare intervention history documentation and committee reports to support referral and special education review decisions.

continued on next page

Elementary School Teachers (continued)

O*NET task	Relation	TaskNet statement
Adapt teaching methods and instructional materials to meet students' varying needs and interests.	*refined by*	Differentiate instruction with modified materials, accommodations, and structured language supports to meet varied student learning needs.
	overlaps	Monitor student understanding during lessons and adjust instruction in real time.
	overlaps	Teach small-group and one-on-one lessons to reinforce or extend skills based on student need.
Prepare materials and classrooms for class activities.	*overlaps*	Set up, organize, and maintain the classroom environment, including displays and supplies.
Plan and conduct activities for a balanced program of instruction, demonstration, and work time that provides students with opportunities to observe, question, and investigate.	*refined by*	Assign independent, small-group, or computer-based activities to students.
	refined by	Meet with grade-level teams to plan lessons and coordinate instruction.
	refined by	Monitor student understanding during lessons and adjust instruction in real time.
	refined by	Plan daily and weekly lessons across core subjects, aligned with state standards, pacing guides, and student assessment data.
	refined by	Plan intervention or enrichment lessons targeting prerequisite skills and unmastered content.
	refined by	Teach small-group and one-on-one lessons to reinforce or extend skills based on student need.
	overlaps	Teach daily lessons across core subjects, introducing concepts, modeling skills, and guiding independent practice.
Assign and grade class work and homework.	*refined by*	Assign independent, small-group, or computer-based activities to students.
	refined by	Grade student work and provide written or verbal feedback on strengths, progress, and next steps.
Prepare and implement remedial programs for students requiring extra help.	*refined by*	Assign independent, small-group, or computer-based activities to students.
	overlaps	Differentiate instruction with modified materials, accommodations, and structured language supports to meet varied student learning needs.
	overlaps	Plan intervention or enrichment lessons targeting prerequisite skills and unmastered content.
	overlaps	Teach small-group and one-on-one lessons to reinforce or extend skills based on student need.
Attend staff meetings and serve on committees, as required.	*overlaps*	Attend staff and faculty meetings and serve on school committees.
Use computers, audio-visual aids, and other equipment and materials to supplement presentations.	*overlaps*	Prepare and deliver presentations for faculty meetings or professional development sessions to support license renewal and ongoing educator training.

continued on next page

Elementary School Teachers (continued)

O*NET task	Relation	TaskNet statement
O*NET-only tasks (no TaskNet counterpart)		
Read books to entire classes or small groups.		
Guide and counsel students with adjustment or academic problems or with special academic interests.		
Collaborate with other teachers and administrators in the development, evaluation, and revision of elementary school programs.		
Organize and lead activities designed to promote physical, mental, and social development, such as games, arts and crafts, music, and storytelling.		
Involve parent volunteers and older students in children's activities to facilitate involvement in focused, complex play.		
Supervise, evaluate, and plan assignments for teacher assistants and volunteers.		
Prepare students for later grades by encouraging them to explore learning opportunities and to persevere with challenging tasks.		
Sponsor extracurricular activities, such as clubs, student organizations, and academic contests.		
Prepare for assigned classes and show written evidence of preparation upon request of immediate supervisors.		
Establish clear objectives for all lessons, units, and projects and communicate those objectives to students.		
Enforce administration policies and rules governing students.		
Instruct and monitor students in the use and care of equipment and materials to prevent injuries and damage.		
Provide students with disabilities with assistive devices, supportive technology, and assistance accessing facilities, such as restrooms.		
TaskNet-only tasks (no O*NET counterpart)		
Administer basic first aid, such as bandages, and help students calm down after minor incidents.		
Communicate with parents and families about student behavior and classroom updates through newsletters, emails, and calls.		
Complete teacher evaluation tasks, such as goal-setting reviews and student learning objective submissions, as required by school administration.		
Engage students and families in casual conversation during daily interactions or school events to develop relationships.		
Lead morning meetings to open the school day and build classroom community and routines.		
Mentor student teachers and less experienced colleagues by observing classrooms and attending model lessons.		
Prepare students for transitioning to the next grade by organizing end-of-year activities and closing out the classroom.		
Read and respond to emails from colleagues and school leaders about school updates, student support, and classroom logistics.		
Refer students for learning disability testing or additional evaluation when assessment data show persistent learning difficulties.		

Table 13: **Task statements for Lawyers.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Advise clients concerning business transactions, claim liability, advisability of prosecuting or defending lawsuits, or legal rights and obligations.	*overlaps*	Advise clients, executives, and internal teams on legal options, risks, and recommended courses of action.

continued on next page

Lawyers (continued)

O*NET task	Relation	TaskNet statement
Analyze the probable outcomes of cases, using knowledge of legal precedents.	*overlaps*	Analyze case facts, legal authorities, and records to assess viability, risks, damages, and litigation or settlement strategy.
	overlaps	Draft legal memoranda, opinion letters, summaries, and plain-language updates to explain legal analysis, case outcomes, or regulatory developments.
Present and summarize cases to judges and juries.	*overlaps*	Present legal arguments, evidence, and witness testimony at trials, hearings, and appeals before judges, juries, or administrative authorities.
Evaluate findings and develop strategies and arguments in preparation for presentation of cases.	*overlaps*	Advise clients, executives, and internal teams on legal options, risks, and recommended courses of action.
	overlaps	Analyze case facts, legal authorities, and records to assess viability, risks, damages, and litigation or settlement strategy.
	overlaps	Prepare arguments, exhibits, witnesses, and case materials for trials, hearings, depositions, and mediation sessions.
	overlaps	Present legal arguments, evidence, and witness testimony at trials, hearings, and appeals before judges, juries, or administrative authorities.
Gather evidence to formulate defense or to initiate legal actions by such means as interviewing clients and witnesses to ascertain the facts of a case.	*refined by*	Interview clients and other parties to gather facts, documents, and evidence for new and active matters.
Represent clients in court or before government agencies.	*refined by*	Communicate with courts, agencies, opposing counsel, and other parties on client matters to clear procedural obstacles and resolve delays.
	refined by	Draft and file motions, briefs, pleadings, complaints, and related court filings for active cases.
	refined by	Prepare arguments, exhibits, witnesses, and case materials for trials, hearings, depositions, and mediation sessions.
	refined by	Present legal arguments, evidence, and witness testimony at trials, hearings, and appeals before judges, juries, or administrative authorities.
Examine legal data to determine advisability of defending or prosecuting lawsuit.	*overlaps*	Analyze case facts, legal authorities, and records to assess viability, risks, damages, and litigation or settlement strategy.
	overlaps	Review potential cases and referrals for legal merit and fit with the firm's practice to decide whether to accept, refer to co-counsel, or return them to the client.
Select jurors, argue motions, meet with judges, and question witnesses during the course of a trial.	*overlaps*	Present legal arguments, evidence, and witness testimony at trials, hearings, and appeals before judges, juries, or administrative authorities.

continued on next page

Lawyers (continued)

O*NET task	Relation	TaskNet statement
Present evidence to defend clients or prosecute defendants in criminal or civil litigation.	*refined by*	Prepare arguments, exhibits, witnesses, and case materials for trials, hearings, depositions, and mediation sessions.
	overlaps	Present legal arguments, evidence, and witness testimony at trials, hearings, and appeals before judges, juries, or administrative authorities.
Study Constitution, statutes, decisions, regulations, and ordinances of quasi-judicial bodies to determine ramifications for cases.	*overlaps*	Monitor developments in legislation, regulation, and judicial decisions to stay current on changes affecting legal practice.
	overlaps	Research case law, statutes, regulations, and rules to support litigation, transactional, and regulatory matters.
Prepare legal briefs and opinions, and file appeals in state and federal courts of appeal.	*overlaps*	Draft and file motions, briefs, pleadings, complaints, and related court filings for active cases.
Negotiate settlements of civil disputes.	*refined by*	Negotiate settlements, payment arrangements, or other out-of-court agreements with opposing attorneys, insurance adjusters, regulators, or other parties on behalf of clients.
	refined by	Review proposed settlement terms with clients and obtain their approval before finalizing agreements.
Confer with colleagues with specialties in appropriate areas of legal issue to establish and verify bases for legal proceedings.	*overlaps*	Consult colleagues, co-counsel, and outside experts on legal analysis, case strategy, review methods, or specialized issues.
Search for and examine public and other legal records to write opinions or establish ownership.	*overlaps*	Research case law, statutes, regulations, and rules to support litigation, transactional, and regulatory matters.
Supervise legal assistants.	*overlaps*	Supervise and train paralegals or legal assistants by assigning and reviewing casework tasks.
Perform administrative and management functions related to the practice of law.	*refined by*	Check legal citations against source documents and review executed court-required forms for accuracy.
Help develop federal and state programs, draft and interpret laws and legislation, and establish enforcement procedures.	*overlaps*	Draft and revise statutes, regulations, court policies, and organizational policy memoranda to reflect legal or operational changes.
	overlaps	Review proposed legislation, draft regulations, and statutory language from partner organizations to identify required revisions or agency response.
Prepare, draft, and review legal documents, such as wills, deeds, patent applications, mortgages, leases, and contracts.	*refined by*	Draft written discovery requests, such as interrogatories, requests for admissions, or requests for production.
	refined by	Review and revise legal drafts and AI-generated legal content for accuracy, compliance, and supporting authority before filing or circulation.
	overlaps	Draft and revise statutes, regulations, court policies, and organizational policy memoranda to reflect legal or operational changes.

continued on next page

Lawyers (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Draft legal documents for client matters, such as contracts, estate planning instruments, corporate governance records, or real estate agreements.
Negotiate contractual agreements.	*refined by*	Negotiate settlements, payment arrangements, or other out-of-court agreements with opposing attorneys, insurance adjusters, regulators, or other parties on behalf of clients.
	refined by	Review proposed settlement terms with clients and obtain their approval before finalizing agreements.
	overlaps	Negotiate contract terms and mark up agreements to reflect legal and business positions.

O*NET-only tasks (no TaskNet counterpart)

Interpret laws, rulings and regulations for individuals and businesses.

Act as agent, trustee, guardian, or executor for businesses or individuals.

Probate wills and represent and advise executors and administrators of estates.

TaskNet-only tasks (no O*NET counterpart)

Approve staff-drafted financial, operational, and compliance reports and recurring filings before they are submitted to oversight agencies.

Complete continuing legal education and professional development training.

Deliver legal training to prosecutors, law enforcement officers, and non-legal audiences on investigative procedures, legal changes, and compliance topics.

Develop legal templates, precedents, and playbooks from past contracts and current legal or business needs.

Exchange and produce documents with opposing counsel during pretrial discovery.

Prepare and submit document packets, written responses, and corrective action plans in response to government examinations and audits.

Prepare and take depositions of fact and expert witnesses to develop the case record.

Refer clients to accountants, other legal organizations, or external service providers when their needs fall outside the firm's scope.

Review and respond to discovery requests served on clients, including document review for production.

Update clients on case status, procedural next steps, and hearing or arbitration outcomes.

Table 14: **Task statements for Personal Care Aides.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Perform healthcare-related tasks, such as monitoring vital signs and medication, under the direction of registered nurses or physiotherapists.	*refined by*	Administer prescribed medications to clients at scheduled times, recording doses and responses in their charts.
	refined by	Set out doses and prompt clients to self-administer medications on schedule, following their care plan.
Administer bedside or personal care, such as ambulation or personal hygiene assistance.	*refined by*	Provide toileting assistance and incontinence care, including changing briefs and soiled bedding.

continued on next page

Personal Care Aides (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Assist clients with bathing, grooming, and dressing during daily personal care routines.
	overlaps	Assist clients with walking, transfers, and repositioning, using mobility aids such as gait belts and ceiling lifts.
Prepare and maintain records of client progress and services performed, reporting changes in client condition to manager or supervisor.	*overlaps*	Communicate observed changes in clients' condition, needs, or safety concerns to supervisors, family members, or care team members.
	overlaps	Document care provided, client observations, and completed tasks in charts and shift notes after each visit or shift.
Perform housekeeping duties, such as cooking, cleaning, washing clothes or dishes, or running errands.	*refined by*	Clean and arrange clients' homes, including floors, kitchens, bathrooms, and laundry.
	refined by	Run errands on behalf of clients, such as grocery shopping or picking up medications and household supplies.
Care for individuals or families during periods of incapacitation, family disruption, or convalescence, providing companionship, personal care, or help in adjusting to new lifestyles.	*refined by*	Assist clients with bathing, grooming, and dressing during daily personal care routines.
	refined by	Assist clients with walking, transfers, and repositioning, using mobility aids such as gait belts and ceiling lifts.
	refined by	Confirm that clients have eaten, taken their medications, and need nothing else before ending a visit.
	refined by	Greet clients and assess their communication, health, and daily care needs at the start of each visit.
	refined by	Listen to clients and respond to their everyday requests and needs during care interactions.
	refined by	Provide companionship to clients through conversation, games, walks, or shared leisure activities such as music, crafts, or television.
	refined by	Provide toileting assistance and incontinence care, including changing briefs and soiled bedding.
	overlaps	Feed clients or assist them with eating and drinking during meals and snacks, following any dietary restrictions.
Instruct or advise clients on issues, such as household cleanliness, utilities, hygiene, nutrition, or infant care.	*overlaps*	Teach clients daily living skills, such as personal hygiene, household tasks, or budgeting, through guided practice and repetition.
Plan, shop for, or prepare nutritious meals or assist families in planning, shopping for, or preparing nutritious meals.	*overlaps*	Prepare meals and snacks for clients, including modified textures or diet-specific foods to meet individual nutritional needs.

continued on next page

Personal Care Aides (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Run errands on behalf of clients, such as grocery shopping or picking up medications and household supplies.
Transport clients to locations outside the home, such as to physicians' offices or on outings, using a motor vehicle.	*overlaps*	Transport clients by vehicle to appointments, stores, or community activities.
Provide clients with communication assistance, typing their correspondence or obtaining information for them.	*overlaps*	Support clients' communication skills, using conversation, augmentative communication devices, or mood and emotion boards.

O*NET-only tasks (no TaskNet counterpart)

Participate in case reviews, consulting with the team caring for the client, to evaluate the client's needs and plan for continuing services.

Train family members to provide bedside care.

TaskNet-only tasks (no O*NET counterpart)

Accompany clients to appointments, outings, or community activities to provide support and advocacy during visits.

Arrange specialized training for clients with individual care needs.

Assist clients with prescribed exercises, such as range-of-motion or speech therapy routines.

Care for clients' pets during home visits, such as feeding or cleaning up after animals.

Complete required safety training, refresher courses, and emergency drills with clients and staff.

Connect clients with community resources or covered care providers.

Develop individualized care plans for clients, incorporating behavioral, health, nutritional, and financial goals based on client preferences and needs.

Follow infection control and emergency safety procedures during client care.

Maintain care supplies and medication to support uninterrupted daily health routines.

Observe clients throughout care shifts for changes in condition, mobility, behavior, or safety, such as altered gait, skin breakdown, or mood shifts.

Perform catheter, ostomy, or feeding tube care for clients during scheduled care visits.

Plan daily schedules and activity options for clients based on their interests, sensory needs, and available resources.

Redirect clients when they become frustrated or behave unacceptably.

Respond to medical emergencies, such as falls or injuries, providing first aid and escalating to nursing staff or first responders.

Review care plans, supply inventories, and client schedules before starting each visit.

Schedule transportation for client appointments or follow-up visits.

Train staff on person-centered care techniques, such as safe transfers, positioning, and dignified assistance with daily routines.

Work through homework and academic materials with school-age clients, setting up a suitable workspace and removing distractions.

Table 15: **Task statements for Software Developers.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Analyze information to determine, recommend, and plan installation of a new system or modification of an existing system.	*overlaps*	Research new technologies, design patterns, and implementation approaches to inform planned software work.
Analyze user needs and software requirements to determine feasibility of design within time and cost constraints.	*overlaps*	Estimate implementation effort, technical risks, and timelines for planned features and projects.
	overlaps	Gather, clarify, and document business requirements, acceptance criteria, and project scope with clients, product owners, and other stakeholders.
Confer with systems analysts, engineers, programmers and others to design systems and to obtain information on project limitations and capabilities, performance requirements and interfaces.	*overlaps*	Advise engineers, product managers, and technical leaders on design choices, implementation approaches, and engineering trade-offs before or during development.
	overlaps	Review technical designs, implementation approaches, and solution options with engineers and other stakeholders before development.
Consult with customers or other departments on project status, proposals, or technical issues, such as software system design or maintenance.	*overlaps*	Advise engineers, product managers, and technical leaders on design choices, implementation approaches, and engineering trade-offs before or during development.
	overlaps	Present proposed designs, plans, acceptance criteria, or requests for comment to managers, clients, or other teams for approval before development proceeds.
	overlaps	Review technical designs, implementation approaches, and solution options with engineers and other stakeholders before development.
Design, develop and modify software systems, using scientific analysis and mathematical models to predict and measure outcomes and consequences of design.	*refined by*	Decompose approved features and product requirements into technical implementation tasks for development teams.
	refined by	Design backend software architecture, data structures, or application programming interfaces for new features and systems.
	refined by	Optimize application code and systems to reduce latency, resource use, and operating costs.
	refined by	Refactor code to remove duplication, consolidate shared logic, and separate concerns across the codebase.
	refined by	Research new technologies, design patterns, and implementation approaches to inform planned software work.

continued on next page

Software Developers (continued)

O*NET task	Relation	TaskNet statement
	refined by	Write application code for new features, backend services, or application programming interfaces.
Develop or direct software system testing or validation procedures, programming, or documentation.	*refined by*	Define test coverage, test parameters, and manual test scenarios for software releases in collaboration with quality assurance teams.
	refined by	Test software builds against functional, integration, accessibility, or acceptance criteria across development, staging, or production environments.
	refined by	Write automated tests for new features, backend services, or existing code paths.
Modify existing software to correct errors, adapt it to new hardware, or upgrade interfaces and improve performance.	*refined by*	Refactor code to remove duplication, consolidate shared logic, and separate concerns across the codebase.
	overlaps	Fix software bugs in application code, game features, or production systems.
	overlaps	Migrate or upgrade existing applications, platforms, and architecture to current technology standards.
	overlaps	Optimize application code and systems to reduce latency, resource use, and operating costs.
Monitor functioning of equipment to ensure system operates in conformance with specifications.	*overlaps*	Monitor production systems, logs, and performance metrics after deployment to detect errors, regressions, and reliability issues.
Prepare reports or correspondence concerning project specifications, activities, or status.	*overlaps*	Document experiment results and performance findings for product partners, engineers, and other stakeholders.
	overlaps	Report progress, risks, and blockers to team members, managers, and other stakeholders in stand-up meetings or written updates.
Supervise the work of programmers, technologists and technicians and other engineering and scientific personnel.	*refined by*	Evaluate team members' performance using code quantity and quality metrics.
	refined by	Mentor junior developers and engineers on coding practices, debugging, and technical problem-solving.

O*NET-only tasks (no TaskNet counterpart)

Store, retrieve, and manipulate data for analysis of system capabilities and requirements.

Confer with data processing or project managers to obtain information on limitations or capabilities for data processing projects.

Coordinate installation of software system.

Determine system performance standards.

Obtain and evaluate information on factors such as reporting formats required, costs, or security needs to determine hardware configuration.

Supervise and assign work to programmers, designers, technologists, technicians, or other engineering or scientific personnel.

Train users to use new or modified equipment.

continued on next page

Software Developers (continued)

O*NET task	Relation	TaskNet statement
TaskNet-only tasks (no O*NET counterpart)		
Approve software releases after reviewing test results, launch checklists, integration health, and rollback readiness.		
Build and package software release images for deployment.		
Build and troubleshoot continuous integration and continuous delivery pipelines, using source control workflows and automation scripts, to manage software version releases.		
Build data ingestion and processing pipelines to move, clean, chunk, and index content for search and retrieval systems.		
Define evaluation datasets, acceptance criteria, and development rubrics for artificial intelligence model tasks.		
Deploy application changes to staging or production environments.		
Design prompts, retrieval logic, and context pipelines for artificial intelligence applications based on feedback and evaluation.		
Diagnose software bugs and production incidents to identify root causes and estimate fix scope.		
Document bug reports, incident postmortems, model failure cases, and internal records of technical decisions and follow-up work.		
Draft and revise pull requests for code changes based on review feedback.		
Escalate unresolved engineering blockers, approval gaps, and recurring model failure risks to project leads or partner teams.		
Evaluate artificial intelligence model outputs for instruction following, correctness, and failure patterns across prompt or model variants.		
Gather user feedback on bugs, feature requests, and usage patterns to guide software fixes and enhancements.		
Hand off completed features or production changes to quality assurance or production support teams.		
Integrate external services such as language models into backend systems and artificial intelligence agent workflows.		
Merge approved code changes into the shared codebase.		
Plan and prioritize software development work, including sprint goals, backlog items, bug fixes, and quarterly projects.		
Plan and schedule production releases, including deployment timelines and sprint-based release dates.		
Present completed work, feature ideas, and process plans to product owners, leadership, or other teams.		
Provision and configure application environments, developer workspaces, cloud storage, and messaging queues.		
Review code changes for correctness, readability, security, or maintainability, and provide feedback to other developers.		
Triage, troubleshoot, and route production incidents and service issues to restore systems and resolve customer-impacting failures.		
Write and revise technical design documents, such as architecture diagrams, decision records, and implementation plans.		

Table 16: **Task statements for Emergency Medicine Physicians.** O*NET tasks next to the TaskNet statements related to them, followed by O*NET-only tasks and TaskNet-only tasks.

O*NET task	Relation	TaskNet statement
Matched tasks		
Analyze records, examination information, or test results to diagnose medical conditions.	*overlaps*	Develop and revise differential diagnoses for emergency department patients to guide workup and rule out life-threatening conditions.

continued on next page

Emergency Medicine Physicians (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Evaluate emergency department patients through histories, physical examinations, and chart review.
	overlaps	Interpret laboratory results, imaging studies, and electrocardiograms to guide diagnosis and clinical decision-making for emergency department patients.
	overlaps	Reassess emergency patients after treatment or test results to update diagnoses, adjust care plans, and determine next steps.
Assess patients' pain levels or sedation requirements.	*overlaps*	Administer procedural sedation to patients undergoing reductions or other painful bedside procedures.
Collect and record patient information, such as medical history or examination results, in electronic or handwritten medical records.	*overlaps*	Document patient encounters in medical records, including histories, examination findings, and medical decision-making.
Communicate likely outcomes of medical diseases or traumatic conditions to patients or their representatives.	*overlaps*	Discuss goals of care, treatment preferences, and resuscitation decisions with patients, families, or surrogates.
	overlaps	Explain diagnoses, test results, and care plans to patients and families, and answer their questions about treatment options and next steps.
Conduct primary patient assessments that include information from prior medical care.	*overlaps*	Evaluate emergency department patients through histories, physical examinations, and chart review.
Consult with hospitalists and other professionals, such as social workers, regarding patients' hospital admission, continued observation, transition of care, or discharge.	*overlaps*	Consult specialists, hospitalists, or primary care physicians about emergency department patient evaluation and management.
	overlaps	Coordinate with specialists, hospitalists, and admitting teams to arrange hospital admissions, observation status, or inpatient handoffs for emergency department patients.
Discuss patients' treatment plans with physicians and other medical professionals.	*overlaps*	Communicate patient status, care plans, and orders with nurses and other emergency department staff.
	overlaps	Consult specialists, hospitalists, or primary care physicians about emergency department patient evaluation and management.
Evaluate patients' vital signs or laboratory data to determine emergency intervention needs and priority of treatment.	*overlaps*	Evaluate emergency department patients through histories, physical examinations, and chart review.
	overlaps	Interpret laboratory results, imaging studies, and electrocardiograms to guide diagnosis and clinical decision-making for emergency department patients.
	overlaps	Reassess emergency patients after treatment or test results to update diagnoses, adjust care plans, and determine next steps.

continued on next page

Emergency Medicine Physicians (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Triage emergency department patients to assess acuity, identify life-threatening conditions, and prioritize the order of care.
Monitor patients' conditions, and reevaluate treatments, as necessary.	*overlaps*	Determine whether to admit, discharge, transfer, or observe patients based on their condition and response to treatment.
	overlaps	Reassess emergency patients after treatment or test results to update diagnoses, adjust care plans, and determine next steps.
Perform emergency resuscitations on patients.	*refined by*	Assist nurses and other clinical staff with physical tasks during bedside emergency care.
	overlaps	Lead resuscitations of critically ill or cardiac arrest patients, directing team roles, airway management, and life-saving interventions such as defibrillation, intubation, or chest compressions.
	overlaps	Respond to medical emergencies or resuscitation calls in other areas of the hospital.
	overlaps	Treat emergency department patients with medications, fluids, and other bedside interventions for acute illnesses and injuries.
Perform such medical procedures as emergent cricothyrotomy, endotracheal intubation, and emergency thoracotomy.	*overlaps*	Lead resuscitations of critically ill or cardiac arrest patients, directing team roles, airway management, and life-saving interventions such as defibrillation, intubation, or chest compressions.
	overlaps	Respond to medical emergencies or resuscitation calls in other areas of the hospital.
Refer patients to specialists or other practitioners.	*overlaps*	Escalate critically ill patients to intensive care or surgical teams in coordination with hospitalists or intensivists.
Select and prescribe medications to address patient needs.	*overlaps*	Order diagnostic tests and medications for emergency department patients.
	overlaps	Treat emergency department patients with medications, fluids, and other bedside interventions for acute illnesses and injuries.
Select, request, perform, or interpret diagnostic procedures, such as laboratory tests, electrocardiograms, emergency ultrasounds, and radiographs.	*refined by*	Interpret laboratory results, imaging studies, and electrocardiograms to guide diagnosis and clinical decision-making for emergency department patients.
	refined by	Order diagnostic tests and medications for emergency department patients.
	refined by	Perform bedside ultrasound to diagnose urgent conditions and guide procedures.
Stabilize patients in critical condition.	*overlaps*	Lead resuscitations of critically ill or cardiac arrest patients, directing team roles, airway management, and life-saving interventions such as defibrillation, intubation, or chest compressions.

continued on next page

Emergency Medicine Physicians (continued)

O*NET task	Relation	TaskNet statement
	overlaps	Treat emergency department patients with medications, fluids, and other bedside interventions for acute illnesses and injuries.
Direct and coordinate activities of nurses, assistants, specialists, residents, and other medical staff.	*refined by*	Coordinate with specialists, hospitalists, and admitting teams to arrange hospital admissions, observation status, or inpatient handoffs for emergency department patients.
	refined by	Lead resuscitations of critically ill or cardiac arrest patients, directing team roles, airway management, and life-saving interventions such as defibrillation, intubation, or chest compressions.
	refined by	Supervise residents, physician assistants, and nurse practitioners during emergency department patient care.
Consult and direct other medical personnel on patient care.	*refined by*	Consult specialists, hospitalists, or primary care physicians about emergency department patient evaluation and management.
	refined by	Escalate critically ill patients to intensive care or surgical teams in coordination with hospitalists or intensivists.
	refined by	Supervise residents, physician assistants, and nurse practitioners during emergency department patient care.
	overlaps	Communicate patient status, care plans, and orders with nurses and other emergency department staff.
	overlaps	Lead resuscitations of critically ill or cardiac arrest patients, directing team roles, airway management, and life-saving interventions such as defibrillation, intubation, or chest compressions.

O*NET-only tasks (no TaskNet counterpart)

Identify factors that may affect patient management, such as age, gender, barriers to communication, and underlying disease.

Process patient transfers to and from other hospitals.

TaskNet-only tasks (no O*NET counterpart)

Activate emergency response pathways for time-sensitive conditions such as stroke, cardiac arrest, or sepsis.

Attend emergency department, committee, and multidisciplinary meetings on operations, clinical care, and quality improvement.

Complete continuing medical education requirements, such as coursework, conferences, or literature review, to maintain medical licensure.

Contact patients after emergency department visits to report test results.

Counsel patients on risks and alternatives and obtain informed consent before invasive procedures, procedural sedation, or high-risk emergency treatments.

Countersign resident and advanced practice provider charts to meet emergency department documentation requirements.

Document procedures performed and time spent providing critical care, including compliance reporting requirements and billing codes.

Evaluate new equipment, devices, and supply inventory for emergency department use.

continued on next page

Emergency Medicine Physicians (continued)

O*NET task	Relation	TaskNet statement
Evaluate residents and medical students on clinical performance, documenting their development and providing feedback after supervised shifts.		
Interview and rank residency applicants during the recruitment selection process.		
Onboard new physicians, advanced practice providers, and rotating residents to their emergency medicine roles.		
Participate in developing or updating clinical protocols and guidelines for emergency care.		
Participate in reviews of patient outcomes, safety events, quality measures, or research to help identify opportunities to improve emergency care.		
Perform bedside procedures, such as laceration repair, abscess drainage, fracture reduction, chest tube placement, or central line insertion.		
Receive handoff information on active emergency department patients from outgoing physicians or prehospital crews at shift change.		
Respond to prehospital communications from emergency medical services about incoming patients or needed medical direction.		
Teach residents, medical students, and other healthcare professionals through lectures, simulations, or clinical instruction.		

G Comparison with O*NET

This appendix documents the comparison of TaskNet with O*NET in Section 4: the relation judge (Section G.1) and its human evaluation (Section G.2).

G.1 Task Overlap and Distinctiveness

We compare TaskNet with O*NET using two classifiers. The coverage judge (P13; Section A) decides whether a statement covers a task. The relation judge (P18) labels each pair of statements, one from each inventory, as a *subtask* (our statement is contained in the O*NET task), a *supertask* (the O*NET task is contained in ours), an *overlap* (the two share work but neither contains the other, including the same work in different words), or *unrelated*. We apply the relation judge to all pairs of TaskNet and O*NET statements within the same occupation.

For Figure 2, we assign each TaskNet statement a single category based on its pair labels. A statement is *refined* if at least one of its pairs is a subtask; otherwise, it is *shared* if at least one pair is a supertask or overlap, and *new* if none of its pairs is related.

We call GPT-4.1 mini at temperature 0 with P18 as the system prompt and the two statements as the user message. The model answers with one letter per relation. To avoid order effects, we judge each pair in both orders. The verdict is the label with the highest log-probability averaged across the two orders.

Prompt 18: Task-Pair Relation Judge

```
{# system #}
You are comparing TWO TASK STATEMENTS written for the same occupation, one from each of two
task inventories that were written independently. One inventory is a national occupational
survey: terse, written by analysts, naming a duty and illustrating it with a few representative
objects. The other was written from what workers said: fuller, naming the people, the system,
the setting and the purpose. You are not told which is which, and it does not matter. Your job
is to name the relation between the two statements, and to answer with ONE LETTER.

THE FOUR ANSWERS, stated about statement X relative to statement Y:
```

S X is a SUBTASK of Y. Doing X is one way of doing Y, or one step or one constituent of it, and X covers no work outside Y. Y names an act that X does not answer to; X names nothing Y does not. P X is a SUPERTASK of Y. The mirror image: doing Y is one way, one step or one constituent of doing X, and Y covers no work outside X. X names an act that Y does not answer to; Y names nothing X does not. O X and Y OVERLAP. They name at least some of the same work, and neither sits strictly inside the other: either they are the same work at the same level in other words, or they meet while each reaches past the other. N X and Y do NOT overlap. Neither names work the other names.

WHAT THIS IS NOT. It is not a similarity rating and it is not a coverage question with a default answer. Two inventories describing one occupation contain a great deal of the same work in different words AND a great deal of different work, so neither answer is the safe one. Decide from the act and the object, not from which verdict feels cautious, and not from how many words either side carries.

STEP 1. SPLIT EACH STATEMENT INTO ACT AND OBJECT. The ACT is the verb, what the worker does. The OBJECT is what it is done to. A purpose clause ("to ...", "for ...", "so that ...") is neither: it can support a reading but never establishes one on its own. A statement joining several verbs names several acts. A statement whose verbs are a fixed phrase for one act ("prepare and serve", "clean and sanitize") names one act.

STEP 2. ASK, SEPARATELY IN EACH DIRECTION, WHETHER ONE SIDE NAMES AN ACT THE OTHER DOES NOT. Would a worker doing X necessarily be doing something Y names? Would a worker doing Y necessarily be doing something X names? Answer each on its own merits; deciding one does not decide the other. Then read off: Y names an extra act, X does not -> S

X names an extra act, Y does not	->	P
both name an extra act, some act shared	->	O
neither names an extra act, act shared	->	O (the same work)
no act shared at all	->	N

FOUR WAYS TO SEE A RELATION THAT IS NOT THERE.

R1. THE SAME VERB ON A DIFFERENT KIND OF THING. Sharing a verb is not sharing an act. "Inspect finished cabinets for defects" and "Inspect the shop floor for hazards" are both inspecting and are different work: one examines a product, the other a workplace. A person, a document, a machine and a premises are different kinds of thing. Extract the object before deciding.

R2. THE SAME ABSTRACT FRAME. Two statements can both be "planning something", "communicating with someone" or "keeping records" and share nothing else. A frame that would fit half the occupation's work is not a relation. Ask what would actually be done, not what category it belongs to.

R3. A CATEGORY HEADING REACHES ONLY ITS OWN CATEGORY. A statement naming a class of work, "perform administrative duties for the shop", contains the members OF THAT CLASS and nothing else. Building a cabinet is not an administrative duty merely because the shop that does administration also builds cabinets. A member of the class IS a subtask of the heading; anything outside the class is unrelated to it.

R4. THE OTHER SIDE OF AN EXCHANGE. One statement puts the worker as performer, author or requester and the other as checker, approver or recipient. Submitting a plan for approval and approving plans others submit share vocabulary and are different work, even when the same worker does both at different times.

FOUR WAYS TO MISS A RELATION THAT IS THERE. These are equally common and equally wrong.

M1. DIFFERENT WORDS FOR THE SAME ACT. The two inventories were written by different people for different purposes; one favours plain language, the other formal survey prose. "Take payment from customers at the register" and "Receive and process customer payments using a point-of-sale system" are the same work. Ordinary synonyms are the same act: direct, supervise and oversee; confer, discuss, consult and advise; prepare, draft, develop and produce; assess, evaluate, examine and review; monitor, observe and track; maintain, service and keep up. Register differences are not scope differences.

M2. A LIST TREATED AS A CHECKLIST. A statement joining objects with "and", "or" or commas is satisfied by ANY ONE of them, and a "such as ..." or "including ..." list is ILLUSTRATIVE of one category, not closed. "Operate shop machinery, such as saws and planers" reaches operating a lathe, and it is ONE act, not three.

M3. THE DETAIL TRAP, AND IT IS THE MOST COMMON ERROR. The fuller statement usually carries more words: which people, which system, which setting, which stage, why it is done, a list of the objects involved, the sub-steps it is carried out in. NONE OF THAT IS AN EXTRA ACT, and none of it makes the fuller statement wider or the terser one narrower. Adding who, which, how, where or why to the same act leaves it the SAME act. "Sand curved trim by hand" is a subtask of "Sand and finish wood surfaces" because the second names a wider range of surfaces and a further act,

```
finishing; it is not made unrelated by the detail, and the detail does not make it the wider
one.

M4. A DIFFERENT STATED PURPOSE FOR THE SAME ACT ON THE SAME OBJECT. One inventory states why,
the other does not, or states a different why. Judge act and object.

WORKED EXAMPLES, from other trades on purpose.
    X "Install the countertops"  /  Y "Fabricate, install and repair countertops"
    -> S   (Y names fabricating and repairing, X answers to installing only) X "Sand, stain and
    wax the floors"  /  Y "Sand the floors"                                    -> P   (X
    names staining and waxing, Y does not) X "Sand and prime the doors"  /  Y "Sand and hang
    the doors"                                  -> O   (sanding is shared; each names an act
    the other does not) X "Take payment from customers at the register"  /  Y "Receive and
    process customer payments using a point-of-sale system"   -> O   (the same work, in other
    words) X "Inspect finished cabinets for defects"  /  Y "Inspect the shop floor for hazards"
    -> N   (a shared verb on a different kind of thing) X "Build custom cabinets to the
    customer's drawings"  /  Y "Perform administrative duties for the shop"   -> N   (a heading
    reaches only its own class)

Reply with exactly one character, S, P, O or N, and nothing else. Do not explain.

{# user #}
TASK X: {{ statement_x }}
TASK Y: {{ statement_y }}

Answer with one letter: S, P, O or N.
```

G.2 Human Evaluation of the Relation Judge

We evaluate the relation judge, which classifies TaskNet statements relative to the O*NET list (Figure 2), against two annotators, comparing the judge's agreement with them to their agreement with each other (Table 17).

Table 17: **Human evaluation of the coverage and relation judges.** Cells report agreement and Cohen's κ, comparing each judge's agreement with the annotators to the annotators' agreement with each other. Judge scores use labels agreed by both annotators.

Evaluation	Judge–annotators	Inter-annotator
Coverage[a]	87.4% ($\kappa = 0.75$)	86.6% ($\kappa = 0.73$)
Pairs: new vs. related[b]	98.7% ($\kappa = 0.76$)	97.6% ($\kappa = 0.63$)
Pairs: new, refined, shared [b]	98.5% ($\kappa = 0.63$)	96.9% ($\kappa = 0.54$)
Statements: new vs. related[c]	84.8% ($\kappa = 0.70$)	82.5% ($\kappa = 0.66$)
Statements: new, refined, shared[c]	71.4% ($\kappa = 0.52$)	70.0% ($\kappa = 0.53$)

G.2.1 Statement Pairs

The relation judge assigns each statement pair one of the four labels. We sampled anchors in two rounds of 40, each with 20 of our occupational statements (the 204 rated by workers) and 20 O*NET tasks, drawn at random within occupation in proportion to each occupation's number of statements. For each sampled statement (the *anchor*), annotators reviewed all statements for the same occupation in the other inventory (i.e., O*NET for anchors from TaskNet, and TaskNet for anchors from O*NET). They marked every statement describing work also included in the anchor and labeled its relation to the anchor as *subtask*, *supertask*, or *overlap*. Each unmarked statement formed an *unrelated* pair with the anchor.

The first round served to calibrate the annotators, who discussed disagreements and reached consensus labels to establish a shared interpretation of the relation categories. This round also informed the judge's prompt. We report results from the second round of 40 separately sampled anchors (817 pairs not also in the first round).

Inter-annotator agreement is 97.6% ($\kappa = 0.63$) for relatedness and 96.9% ($\kappa = 0.54$) across the three labels (the raw rates are high because most pairs are unrelated). The annotators had particular difficulty distinguishing *supertask* from *overlap*. Among the seven pairs both labeled as one of these two relations, they agreed on the specific label in only three cases (43%, $\kappa = -0.08$). This poor agreement led us to merge the two into *related* for our counts and evaluation.

We evaluate the judge's relatedness decisions on 797 pairs for which the annotators agree. The judge identifies 16 of the 18 related pairs (recall 88.9%), and 16 of its 24 related predictions are correct (precision 66.7%; F_1 =0.76). On the 792 pairs with the same three-way label from both annotators, agreement across the three labels is κ =0.63.

G.2.2 Statement Categories

Figure 2 counts statements rather than pairs. We assign each statement a category based on the judge's verdicts against every task in its occupation's O*NET list, to avoid double-counting statements related to multiple O*NET tasks. We assign each statement a single category: *new* if all pairs are unrelated, *refined* if at least one pair is a *subtask*, and *shared* otherwise. A *subtask* verdict takes precedence because it identifies the statement as a more specific component of an existing O*NET task. The statement still refines that O*NET task, even if it also overlaps with others.

Two annotators independently labeled 40 TaskNet statements, drawn at random within occupation, directly against their occupation's full O*NET list. They agree on whether a statement is *new* in 33 of 40 cases (82%, $\kappa = 0.66$) and across all three categories in 28 of 40 (70%, $\kappa = 0.53$). On the statements where they agree, the judge matches them on 28 of 33 for new (κ =0.70), and 20 of 28 across all three categories (κ =0.52). Refined is the least stable category for annotators and judge alike: the annotators label 10 and 14 statements as refined and agree on seven, and the judge, which labels ten, matches three of those seven.

H Inventory Representativeness

This appendix documents the comparisons of Section 5.2: coverage and silhouette scores on held-out workers (Section H.1), the human evaluation of the coverage judge (Section H.2), and coverage of job postings (Section H.3).

H.1 Representation of Held-Out Tasks

The held-out sample comprises 57 workers and 1184 tasks across the six occupations, recruited with the same procedure as the construction sample and not used to build our inventories. A task is one entry of a worker's verified task list.

Coverage Each held-out task is compared with the five most similar statements of each inventory, by cosine similarity of `text-embedding-3-large` embeddings, plus any other statement with cosine similarity of at least 0.30. The coverage judge (P13) checks them in order of similarity,

and the task is covered if the judge says yes to any of them. The per-task difference in coverage between the inventories is averaged within each occupation, and the pooled estimate weights the six occupations equally. Standard errors and p-values come from a paired cluster bootstrap that resamples held-out workers within each occupation (10 000 resamples). The p-values are computed from the bootstrap distribution recentered at zero, and the per-occupation tests are Holm-corrected across the six occupations. The same procedure gives the in-sample coverage in Section 5.1, with the construction-sample tasks judged against the 230 system-generated statements.

Silhouette Score Silhouette scores require every task to belong to a group, but O*NET does not assign worker-reported tasks to its statements. We therefore place each task under one statement, separately for each inventory. Each statement's group is anchored by source tasks: the verified tasks of the construction-sample workers from which the statements were built. Placement uses GPT-5.4 mini, which receives P19 as a single user message together with the task and the inventory's numbered statements.

We score only held-out tasks. For each held-out worker, we combine the source tasks with that worker's tasks and compute the silhouette of each of the worker's tasks, $(b - a)/\max(a, b)$, from cosine distances between `text-embedding-3-large` embeddings, where a is the task's mean distance to the other tasks in its group and b is its mean distance to the tasks in the nearest other group. A task that is alone in its group scores zero. A worker's score is the mean over their tasks, and we compare inventories by the within-worker difference in scores. Occupation estimates average across workers, and the pooled estimate weights occupations equally. Standard errors are the standard deviation of a bootstrap over held-out workers within occupation (10 000 resamples), and p-values, which determine the significance stars, come from its distribution recentered at zero.

The random baseline in Table 18 reassigns tasks to groups at random 60 times per worker, preserving group sizes. We report it alongside each inventory rather than subtracting it from the difference, because the baseline depends on the number of statements, and subtracting it would favor the inventory with more statements.

Prompt 19: Task Placement

```
You are filing one piece of work under the single statement that is its HOME. The home is the
statement whose activity this work IS an instance, sub-type, or step of. Judge the ACT and its
OBJECT, not shared words or shared subject matter. A statement that names a CLASS OF GENERAL
DUTIES rather than an act -- "perform administrative and management functions", "perform other
duties as assigned" -- is the home ONLY when no statement in the list names the actual
activity. If any statement names the act, choose it, even if the general statement would also
cover the work.

WORKED EXAMPLES.
  work: Run monthly data pipelines and validate the output data.
  home: Build data ingestion and processing pipelines to move, clean, chunk, and index content
  for search and retrieval systems.
  work: Document mileage, time, amount of gas remaining, and destinations for client outings.
  home: Prepare and maintain records of client progress and services performed, reporting
  changes in client condition to manager or supervisor.
  work: Answer teammates' questions about system behavior and implementation details. home:
  Advise engineers, product managers, and technical leaders on design choices, implementation
  approaches, and engineering trade-offs.
  work: Coordinate accommodations and testing logistics for students during standardized
  assessments.
  home: Administer standardized ability and achievement tests, and interpret results to
  determine student strengths and needs.
  work: Monitor clients' safety and remove fall hazards in the home.
  home: Observe clients throughout care shifts for changes in condition, mobility, behavior, or
  safety.
You MUST pick exactly one number. There is no 'none' option -- if nothing fits well, pick the
closest.
```

```
Reply as JSON: {"pick": <number>}

THE WORK:
{{ task }}

STATEMENTS:
{% for s in statements %}{{ loop.index }}. {{ s }}
{% endfor %}
```

Table 18: **Held-out tasks form more coherent groups under TaskNet than under O*NET.** Silhouette scores of held-out tasks under each inventory and a baseline averaged over 60 random reassignments of tasks to groups per worker. Δ is TaskNet silhouette minus O*NET silhouette. Occupation rows average across workers; pooled weights occupations equally. Standard errors are the standard deviation of a stratified worker bootstrap (10,000 replications). $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$.

	TaskNet		O*NET			
Occupation	Silhouette	Random	Silhouette	Random	Δ	SE
Software Developers	0.026	−0.116	−0.029	−0.105	+0.055 ***	(0.006)
Lawyers	−0.008	−0.108	−0.028	−0.101	+0.020 *	(0.008)
Elementary School Teachers	0.083	−0.164	0.020	−0.171	+0.063 ***	(0.014)
Emergency Medicine Physicians	0.082	−0.134	0.034	−0.068	+0.048 ***	(0.013)
Personal Care Aides	0.097	−0.165	0.097	−0.055	0.000	(0.023)
Dishwashers	0.031	−0.084	0.064	−0.100	−0.033	(0.028)
Pooled	0.052	−0.129	0.027	−0.100	+0.025 ***	(0.007)

H.2 Human Evaluation of the Coverage Judge

We evaluate the coverage judge on held-out tasks paired with candidate statements. For each sampled task, we select one statement from each inventory: the first candidate the judge accepts as covering the task, or the most similar candidate if none is accepted. We stratify tasks by the judge's verdicts across the two inventories (covered by both, ours only, O*NET only, or neither), oversampling tasks on which the inventories differ. The first authors independently annotated 119 pairs from 60 tasks.

The LLM judge agrees with each annotator on 98 of 119 pairs (82.4 %, $\kappa = 0.64$; positive agreement 0.84, negative 0.80), modestly below the annotators' agreement with each other (103 of 119, 86.6 %, $\kappa = 0.73$; 0.88 and 0.85). On the 103 pairs the annotators agree on, the judge matches them on 90 (87.4 %, $\kappa = 0.75$). Its errors are balanced, 10 false *covered* and 11 false *not covered*, and it calls 55 % of pairs covered, as does each annotator.

For the coverage comparison, we also assess whether the judge systematically favors either inventory. Its agreement is similar on pairs with TaskNet statements ($\kappa = 0.66$ and 0.70 against the two annotators) and with O*NET statements (0.59 and 0.56). It calls 64 % of our pairs covered against the annotators' 69 % and 64 %, and 45 % of O*NET pairs against 42 % and 47 %. These results show no clear directional bias toward either inventory in the annotated sample.

H.3 Representation in Job Postings

We use U.S. job postings from Revelio Labs dated 2026: a 1 % random sample (Redivis, 2025), assigned to occupations by Revelio's O*NET-SOC code, plus 1000 randomly selected postings each for Personal Care Aides, Elementary School Teachers, and Emergency Medicine Physicians, which the sample covers thinly. Exact duplicates are dropped, and reposts of one template are merged when their task sentences overlap with a weighted Jaccard similarity of at least 0.5, where sentences shared by many near-identical templates (Jaccard of at least 0.8) are down-weighted.

The Job Ad Analysis Toolkit (JAAT) uses a sentence-level classifier to identify whether each sentence in a job posting describes a task performed by workers (Meisenbacher et al., 2025). A language model then filters postings and sentences for relevance to the occupation and splits each sentence into the activities it names, without seeing either task list (P20). For each activity, the model sees the occupation's full list in shuffled order, picks the closest statement, and counts the activity as covered if the statement names the same kind of action on the same object, at any level of generality, and is not a catch-all for the occupation's work (P21). Each activity is judged once against each inventory and counts once per template. All calls use Claude Sonnet 4.6, with a thinking budget of 6000 tokens for matching.

Of 6422 postings, 5544 contain at least one sentence JAAT classifies as describing work (62 313 sentences). Merging reposts leaves 4295 templates, of which the language model judges 2964 to advertise the occupation. These contain 11 286 sentences describing the occupation's own tasks, and 2186 templates with 12 474 activities enter the analysis.

Coverage is the share of an occupation's activities that are covered, and the pooled figure is the unweighted mean over the six occupations. Standard errors are the standard deviation of 4000 bootstrap replications that resample templates within occupation, and p-values use the normal approximation.

Contributions by statement type. The coverage difference equals the share of activities only TaskNet covers minus the share only O*NET covers, since activities both or neither cover cancel. We split each of these exclusive shares by the statement the judge matched the activity to (Table 19). Activities only TaskNet covers are split by the type of the covering TaskNet statement: new, refined, or shared (Section G.1). Activities only O*NET covers are split by whether the covering O*NET task has a related TaskNet statement (a counterpart) or not (a missing task). New statements thus measure work in postings that only TaskNet describes, missing tasks the reverse, and the two remaining parts measure work both inventories describe but only one matches in a given posting. Averaged over occupations, new statements cover 6.5 points of activities that O*NET does not, missing tasks 3.4 points that TaskNet does not, O*NET tasks with a counterpart 7.4 points, and refined and shared statements 6.6 points.

Table 19: **New statements account for most of the activities only TaskNet covers.** Percentage of each occupation's posting activities covered by both inventories, by neither, or by only one. Activities only TaskNet covers are split by the type of the TaskNet statement that covers them; activities only O*NET covers, by whether the covering O*NET task has a related TaskNet statement. The gap, TaskNet-only minus O*NET-only, equals the coverage difference of Table 3.

			TaskNet only				O*NET only			
Occupation	Both	Neither	All	New	Refined	Shared	All	Counterpart	None	Gap
Software Developers	61.8	8.5	17.4	13.5	2.9	1.0	12.3	10.0	2.3	+5.0
Lawyers	56.8	21.7	12.5	4.5	3.0	5.0	9.0	7.4	1.6	+3.5
Elementary School Teachers	64.8	10.8	7.8	3.3	1.1	3.4	16.6	5.9	10.7	−8.8
Emergency Medicine Physicians	50.8	23.1	18.1	9.0	1.0	8.0	8.0	7.1	0.9	+10.1
Personal Care Aides	64.1	14.0	16.6	6.8	7.3	2.6	5.3	4.2	1.1	+11.4
Dishwashers	57.7	22.0	6.4	2.1	1.4	2.9	13.9	9.9	4.0	−7.4
Mean of six	59.3	16.7	13.1	6.5	2.8	3.8	10.8	7.4	3.4	+2.3

Prompt 20: Job Postings: Activity Splitting

```
You are given numbered sentences from job postings. Each describes work the person hired will
do. List the distinct work activities each sentence names.

Split a sentence only where it names separate activities ("clean rooms and prepare meals" is
two; "interpret x-rays, lab results and EKGs" is one activity on several objects; "design and
implement the billing service" is one). Keep each activity's object and any detail that belongs
to it. Most sentences name one or two activities. Write each activity as a short task phrase
(verb + object).

Respond with ONE JSON object and nothing else: {"a": "<one line per activity>"} where each line is
"<sentence number>.<activity number> <activity>", lines separated by " ; ".
Example: {"a": "1.1 clean residents' rooms ; 1.2 prepare meals ; 2.1 interpret x-rays, lab results and EKGs"}
```

Prompt 21: Job Postings: Activity Matching

```
You are given an occupation, a numbered TASK LIST for it, and numbered ACTIVITIES from job
postings classified under it. The question for each activity: does the list contain a statement
that names this activity? The statement may describe it more generally ("inspect vehicles" for
"inspect delivery vans for brake wear") or more specifically, naming a concrete part of it
("check brake pads for wear" for "inspect and maintain fleet vehicles"). Either way the two
must name the same kind of work. A statement about a different activity in the same area does
not count, and neither does a catch-all that would describe most of the occupation's work.

For each activity, pick the ONE statement of the list closest to it and answer three checks
about that statement:

  A  SAME KIND OF ACTION -- the statement's action is the activity's action, a more general name
  for it, or a more specific form of it ("confer with colleagues to plan repairs" for
  "collaborate with the shop team on repair plans", and the other way round). NOT the same: a
  different action on the same object ("order parts" is not "install parts"; "catalog books" is
  not "recommend books to patrons").
  O  SAME OBJECT -- the statement acts on the same thing, on a named category that includes it,
  or on a named part of it: "client financial records" for "client invoices" and "client
  invoices" for "client financial records"; "patrons" for "patrons, staff and visitors". A
  generic object that would include almost anything ("systems", "operations", "duties",
  "functions", "work", "tasks") does not count.
  C  CATCH-ALL -- the statement is so broad that it would also describe most other work in this
  occupation ("Perform duties related to the maintenance of vehicles"; "Carry out the work of
  the library"). Answer Y if it is a catch-all.

The activity is covered when A = Y, O = Y and C = N. Tools, products, employers, schedules and
settings in the activity are details: ignore them for A and O. If no statement shares the
activity's kind of action and object, answer with null. Task lists are incomplete; null is a
```

```
normal answer. Judge each activity on its own.

Respond with ONE JSON object and nothing else: {"m": [ ... ]} with one entry per activity, in order:
  {"n": <activity number>, "s": <statement number or null>, "q": "<the first five words of that statement,
copied exactly>",
   "a": "Y|N", "o": "Y|N", "c": "Y|N"}
Use "s": null, "q": "", and N for all three checks when no statement shares the activity's kind
of action and object.
```

I Codebook for New, Refined, and Missing Tasks

This appendix documents the qualitative coding behind Section 6 and the theme decomposition of the exposure difference in Section 7.

Procedure. The first authors read every new, refined, and missing statement and developed the themes inductively. Each theme was written into the codebook as a definition with a one-line test, inclusion and exclusion criteria, and an example (Table 20). The codebook was frozen before coding. Statements were coded on three sheets:

- **Sheet A** (TaskNet statements): the 79 new statements mixed in random order with the 65 related statements, with category hidden, coded for four themes.
- **Sheet B** (Refined statements): the 58 refined statements, each shown with the O*NET task it refines, coded for the type of refinement. Refinement can only be judged against the parent task, so this sheet is not blind.
- **Sheet C** (O*NET tasks): the 30 missing O*NET tasks mixed with the 88 covered tasks, with category hidden, coded for two themes.

Categories were joined to the coded sheets afterwards. The categories are those of Figure 2, computed against the 118 published O*NET tasks.

Coding rules.

1. Code the statement as written; do not infer duties it does not name.
2. Enter 1 if a code applies, "?" if unsure (with a note), and leave it blank otherwise.
3. Sheet A: codes may overlap (approving a release is both oversight and coordination). Mark *none* if no theme applies.
4. Sheet B: mark *split* and/or *method or tool*, both if both apply, and *other* only if neither applies.
5. Sheet C: mark *specialized* and/or *dated* if they apply, and leave both blank otherwise.

Interpretation. Because the themes were derived from the statements they describe, the frequencies in Table 21 describe which features distinguish each category in our data. They are not tests of hypotheses specified in advance. The themes are not exhaustive: some statements fall outside all of them, and others span several.

Table 20: **Codebook.** Each code is defined by a test; near misses name the code to use instead.

Code	Definition and test	Include	Exclude (near miss)	Example
Sheet A: themes of TaskNet statements (new and related)				
Coordination	Workers hand off tasks, information, and responsibility between people, across shifts, and from one stage to the next. *Test:* passes work, information, or responsibility to or from someone else.	Handoffs, referrals, escalations, status updates, passing work to the next stage.	Communication that is itself the service, e.g., casual conversation with families (→ none).	"Hand off completed features or production changes to quality assurance or production support teams."
Teaching and professional development	Tasks that sustain the expertise an occupation depends on. *Test:* builds the expertise of colleagues, trainees, or the worker themselves.	Training, mentoring, evaluating, or onboarding colleagues; continuing education; safety refreshers.	Teaching clients or students as the core service, e.g., arranging training for clients (→ none).	"Teach residents, medical students, and other healthcare professionals through lectures, simulations, or clinical instruction."
Oversight and quality assurance	Tasks that maintain standards and establish accountability for the quality and safety of the work. *Test:* checks, approves, or corrects work, or responds when something goes wrong.	Reviewing or approving others' work, audits, compliance, safety reviews, incident response.	Revising one's own work after review, e.g., revising pull requests (→ coordination).	"Countersign resident and advanced practice provider charts to meet emergency department documentation requirements."
New technology	New technology that creates a task of its own. *Test:* depends on a tool or practice that became common in roughly the last 15 years.	AI and language models, CI/CD, cloud infrastructure, email when the task is defined by it.	Long-standing tools, or tools named in passing, e.g., technical design documents (→ none).	"Integrate external services such as language models into backend systems and artificial intelligence agent workflows."
*Sheet B: type of refinement (refined statement against its O*NET parent)*				
Split	Splits a task into the steps or duties it bundles. *Test:* the statement is one step, stage, or object of a multi-part O*NET task.	One stage of a process; one item from a bundled list.	The same activity with a named method or standard (→ method or tool).	"Represent clients in court or before government agencies" → "Draft and file motions, briefs, pleadings, complaints, and related court filings for active cases."
Method or tool	Specializes a task to a particular method, tool, or standard, or to the concrete activities a broadly stated role involves. *Test:* the same activity, narrowed to a named method, tool, setting, or standard; or a role or goal in the O*NET parent ("care for", "ensure", "support") made into an activity.	Gait belts, ceiling lifts, state standards, small-group instruction, AI-generated drafts; "care for individuals during incapacitation" → greeting clients and assessing their needs.	A near-restatement of the parent, e.g., the parent already names ultrasound (→ other).	"Plan and conduct activities for a balanced program of instruction…" → "Plan daily and weekly lessons across core subjects, aligned with state standards, pacing guides, and student assessment data."

Code	Definition and test	Include	Exclude (near miss)	Example
*Sheet C: themes of O*NET tasks (missing and covered)*				
Specialized	Specialized or role-specific work. *Test:* done by only a subset of incumbents, rarely, or by a senior or managerial version of the role.	Practice-area work, events, supervising staff.	Routine work most incumbents do, e.g., cleaning garbage cans (leave blank).	"Set up banquet tables."
Dated	The function persists, but in vocabulary that predates current practice. *Test:* names an older technology or form of work organization; the coder records the modern form in a note.	Older technology or work-organization terms.	Wording that is merely formal or generic.	"Obtain and evaluate information on factors such as reporting formats required, costs, or security needs to determine hardware configuration."

Table 21 reports how often each code was applied. No statement was marked "?". All four O*NET tasks coded *dated* belong to Software Developers: "Train users to use new or modified equipment," "Monitor functioning of equipment to ensure system operates in conformance with specifications," "Obtain and evaluate information on factors such as reporting formats required, costs, or security needs to determine hardware configuration," and "Coordinate installation of software system."

Table 21: **Frequency of each code by category.** Count and share of statements in the category.

Sheet A: TaskNet statements				
	New ($n = 79$)		Related ($n = 65$)	
Coordination	17	22 %	12	18 %
Teaching and professional development	9	11 %	5	8 %
Oversight and quality assurance	19	24 %	6	9 %
New technology	16	20 %	2	3 %
Sheet B: refined statements ($n = 58$)				
Split	28	48 %		
Method or tool	34	59 %		
Other (neither)	5	9 %		
*Sheet C: O*NET tasks*				
	Missing ($n = 30$)		Covered ($n = 88$)	
Specialized	8	27 %	3	3 %
Dated	3	10 %	1	1 %

Codes overlap within a sheet, so shares do not sum to 100%.

J AI Exposure

This appendix documents the exposure analyses of Section 7: the predicted-exposure rubrics and their decomposition (Sections J.1, J.2 and J.4) and the matching of AI conversations to task statements (Section J.5).

J.1 Predicted Exposure

All statements are rated once under each of two rubrics by Claude Fable 5.1 at temperature 0. A structured-output request that fails is repeated once as plain JSON.

LLM exposure (Eloundou-β) Adapted from the rubric of Eloundou et al. (2024, Appendix A.1): a task is E1 if an LLM alone would halve the time it takes, E2 if software built on an LLM would, and E0 otherwise. The original rubric labels image capabilities separately (E3) but merges them into E2 for all analyses; our prompt folds them into E2 directly. We use $\beta = \mathrm{E1} + 0.5\,\mathrm{E2}$ (P22).

Reinforcement-learning feasibility (RL) From Tomei and Teeselink (2026, Appendix B): how feasible it is to build an RL environment for the task with a verifiable reward. A gate first rules out physical tasks, then eight dimensions are rated on 1–10 scales (P23).

Prompt 22: Exposure Rubric: E0/E1/E2

```
{# system #}
Consider the most powerful OpenAI large language model (LLM) This model can complete many tasks
that can be formulated as having text input and text output where the context for the input can
be captured in 2000 words. The model also cannot draw up-to-date facts (those from <1 year ago)
unless they are captured in the input.

Assume you are a worker with an average level of expertise in your role trying to complete the
given task. You have access to the LLM as well as any other existing software or computer
hardware tools mentioned in the task. You also have access to any commonly available technical
tools accessible via a laptop (e.g. a microphone, speakers, etc.). You do not have access to
any other physical tools or materials.

Please label the given task according to the taxonomy below.

## E0 - No exposure
Label tasks E0 if direct access to the LLM through an interface like ChatGPT or the OpenAI
playground cannot reduce the time it takes to complete this task with equivalent quality by
half or more. If a task requires a high degree of human interaction (for example, in person
demonstrations) then it should be classified as E0.

## E1 - Direct exposure
Label tasks E1 if direct access to the LLM through an interface like ChatGPT or the OpenAI
playground alone can reduce the time it takes to complete the task with equivalent quality by
at least half. This includes tasks that can be reduced to:
- Writing and transforming text and code according to complex instructions,
- Providing edits to existing text or code following specifications,
- Writing code that can help perform a task that used to be done by hand,
- Translating text between languages,
- Summarizing medium-length documents,
- Providing feedback on documents,
- Answering questions about a document, or
- Generating questions a user might want to ask about a document.

## E2 - Exposure by LLM-powered applications
Label tasks E2 if having access to the LLM alone may not reduce the time it takes to complete
the task by at least half, but it is easy to imagine additional software that could be
developed on top of the LLM that would reduce the time it takes to complete the task by half.
This software may include capabilities such as:
- Summarizing documents longer than 2000 words and answering questions about those documents
- Retrieving up-to-date facts from the Internet and using those facts in combination with the
  LLM capabilities
- Searching over an organization's existing knowledge, data, or documents and retreiving
  information Examples of software built on top of the LLM that may help complete worker
  activities include:
- Software built for a home goods company that quickly processes and summarizes their
  up-to-date internal data in customized ways to inform product or marketing decisions
- Software that is able to suggest live responses for customer service agents speaking to
  customers in their company's customer service interface
- Software built for legal purposes that can quickly aggregate and summarize all previous cases
  in a particular legal area and write legal research memos tailored to the law firm's needs
- Software specifically designed for teachers that allows them to input a grading rubric and
  upload the text files of all student essays and have the software output a letter grade for
```

```
  each essay
- Software that retrieves up-to-date facts from the internet and uses the capabilities of the
  LLM to output news summaries in different languages

## E3 - Exposure given image capabilities
Suppose you had access to both the LLM and a system that could view, caption, and create
images. This system cannot take video media as inputs. This system cannot accurately retrieve
very detailed information from image inputs, such as measurements of dimensions within an
image. Label tasks as E3 if there is a significant reduction in the time it takes to complete
the task given access to a LLM and these image capabilities:
- Reading text from PDFs,
- Scanning images, or
- Creating or editing digital images according to instructions.

## Annotation examples:
Occupation: Inspectors, Testers, Sorters, Samplers, and Weighers Task: Adjust, clean, or repair
products or processing equipment to correct defects found during inspections. Label
(E0/E1/E2/E3): E0 Explanation: The model does not have access to any kind of physicality, and
more than half of the task (adjusting, cleaning and repairing equipment) described requires
hands or other embodiment. Occupation: Computer and Information Research Scientists Task: Apply
theoretical expertise and innovation to create or apply new technology, such as adapting
principles for applying computers to new uses. Label (E0/E1/E2/E3): E1 Explanation: The model
can learn theoretical expertise during training as part of its general knowledge base, and the
principles to adapt can be captured in the text input to the model. Activity: Schedule dining
reservations. Label (E0/E1/E2/E3): E2 Explanation: Automation technology already exists for
this (e.g. Resy) and it's unclear what an LLM offers on top of using that technology (no-diff).
That said, you could build something that allows you to ask the LLM to make a reservation on
Resy for you. (E3) Activity: Negotiate purchases or contracts. Label (E0/E1/E2/E3): E2
Explanation: You could have each party transcribe their point of view and then feed this to an
LLM to resolve any disputes (E3). That said, many people would need to buy into using new
technological tools to accomplish this (system). Occupation: Allergists and Immunologists Task:
Prescribe medication such as antihistamines, antibiotics, and nasal, oral, topical, or inhaled
glucocorticosteroids. Label (E0/E1/E2/E3): E2 Explanation: The model can provide guesses for
different diagnoses and write prescriptions and case notes. However, it still requires a human
in the loop using their judgment and knowledge to make the final decision.

Respond in JSON with fields:
  "label": "E0" | "E1" | "E2" | "E3"
  "rationale": the explanation, one sentence (max 40 words).


{# user #}
{% if with_title %}Occupation: {{ title }}

{% endif %}Task: {{ task }}
```

Prompt 23: Exposure Rubric: Exposure to Reinforcement Learning

```
{# system #}
Context
You are an expert in reinforcement learning (RL), labour economics, and occupational task
analysis. You are helping construct a Reinforcement Learning Feasibility Index -- a measure of
how feasible it is to create an RL environment for a given occupational task, and therefore how
exposed that task is to automation through advances in RL-based training (including RL
post-training of large language models). This index is analogous to Eloundou et al.'s (2024,
Science) AI Exposure Index, but focused specifically on reinforcement learning. For each task,
we ask: How easy would it be to define a well-specified RL environment -- with a reward signal,
simulable environment, and verifiable outcomes -- in which an agent could learn to perform this
task? The more feasible it is to build such an environment, the more "RL-exposed" the task is,
because:
 • RL post-training (RLHF, RLAIF, RLVR) is a primary driver of LLM capability gains
 • Tasks with verifiable outcomes and simulable environments are precisely those where RL
   training is most effective
 • Advances in RL will disproportionately improve AI performance on tasks that score highly on
   this index This version incorporates advances in Rubric-Based RL (e.g., Rubric-ARM, RLVR,
   OpenRubrics) and Process Reward Models, which demonstrate that subjective tasks are feasible
   for RL if they can be decomposed into verifiable criteria. Crucial Assumption (The Agent's
   Starting Point): Assume the RL agent is not learning from scratch (tabula rasa). Instead,
   assume we are fine-tuning a highly capable pre-trained foundation model (e.g., via RLHF,
   RLAIF, or RLVR). The agent already possesses a broad baseline understanding of language,
   code, and general world knowledge. Your evaluation should focus on the feasibility of the RL
   loop required to align, specialize, and verify the model for this specific occupational
   task.
```

```
Temporal Frame
Score based on feasibility using methods that are currently available or plausibly achievable
within a 5-year horizon. Include the trajectory of AI-as-Judge, Process Reward Models, and
Synthetic Data Generation. Do not assume speculative breakthroughs with no current research
basis. When a score depends on projected rather than current capabilities, note this in the
justification.

Input
You will be given an occupation title and a task description. You must score the RL feasibility
of performing that specific task in the context of that specific occupation. This is critical:
the same task text can have very different RL feasibility depending on the occupation.

Step 1: Physical Feasibility Gate
Before scoring any dimensions, first determine whether the task requires substantial physical
interaction with the material world. Decision rule: Pass: The task can be performed primarily
through digital means. Proceed to Step 2. Fail: The task irreducibly requires physical
embodiment. Do not proceed further.

Step 2: Structured Task Reasoning
Before scoring any dimensions, reason about the task holistically: (1) Task type (generative,
analytical, interactive, procedural, or hybrid). (2) Core output. (3) Verification bottleneck.
(4) Tool requirements. (5) Binding constraint (predicted lowest-scoring dimension).

Step 3: Score the 8 Dimensions
Score it on each of the 8 dimensions below. Each dimension must be scored as a strict integer
on a 10-point scale (1-10). Avoid central tendency bias: Confidently use the extreme ends of
the scale (1-3 and 8-10) when the task characteristics warrant it. For each dimension, provide:
 1. A 2-3 sentence justification explaining the score. Write the justification first, then
    assign the score.
 2. An integer score (1-10)
D1: Verification Method Spectrum. Where does this task fall on the spectrum from deterministic
verification to contested professional judgment? Scores: 1 (requires rare experts who disagree;
no inspectable artifact) to 10 (fully deterministic programmatic check). D2: Environment
Simulability. How faithfully and cheaply can the task environment be simulated digitally?
Scores: 1 (requires live markets, real humans with genuine stakes) to 10 (natively digital,
trivially cheap to replicate). D3: State Observability & Context. To what extent is all
relevant information available in a structured, digital format? Scores: 1 (critical information
is tacit or embodied) to 10 (perfect digital observability). D4: Task Variability & Knowledge
Breadth. How varied are the inputs across task instances, and how broad is the required
expertise? Scores: 1 (extreme variability, every instance unique) to 10 (zero variability,
structurally identical instances). D5: Sequential Decision Depth. How many
counterfactual-sensitive decisions must be made in sequence? Scores: 1 (extreme depth, dozens
of contingent decisions) to 10 (single-step, one input, one output). D6: Feedback Density &
Decomposability. How frequently and specifically does the agent receive performance signals,
and how decomposable are they? Scores: 1 (rare, delayed, holistic, non-decomposable) to 10
(continuous, immediate, per-step diagnostic). D7: Tool & Interface Accessibility. Can the task
be performed through CLI/API/MCP, or does it require GUI interaction? Scores: 1 (complex
proprietary GUIs, no scripting) to 10 (natively machine-interfaced or pure text/code
generation). D8: Output Tangibility & Gradeability. Does the task produce a concrete,
inspectable artifact that can be evaluated independently of the process? Scores: 1 (no tangible
output, quality in ongoing process/relationship) to 10 (perfectly tangible, self-contained
artifact).

Output Format
Return a JSON object with: occupation, task, physical_feasibility (justification and pass
boolean), task_reasoning (task type, core output, verification bottleneck, tool requirements,
binding constraint), and dimensions (each with justification and integer score 1-10, or null if
the task failed the gate). Do not compute or include the RL Feasibility Index.

JSON keys: "occupation", "task", "physical_feasibility": {"justification", "pass"}, "task_reasoning": {"
task_type", "core_output", "verification_bottleneck", "tool_requirements", "binding_constraint"} (null if the
task failed the gate), "dimensions": {"D1": {"justification", "score"}, ..., "D8": {"justification", "score"}}
 (null if the task failed the gate).


{# user #}
{% if with_title %}Occupation: {{ title }}

{% endif %}Task: {{ task }}
```

Exposure levels and differences. An occupation's exposure under an inventory is the mean score of its statements, with every statement weighted equally. The all-occupation figure is the unweighted mean over the six occupations. The difference between inventories is TaskNet minus O*NET.

Decomposition. We split the difference exactly into three parts using the relation categories of Section G.1. O*NET tasks are split into those related to at least one TaskNet statement and those with none (O*NET-only). TaskNet statements are split into related and new. Let $\bar{y}_O$ and $\bar{y}_S$ be the mean exposure of O*NET and TaskNet, and $\bar{y}_{O,\text{rel}}$ and $\bar{y}_{S,\text{rel}}$ the means over their related tasks only. Then

$$\bar{y}_S - \bar{y}_O = \underbrace{\left(\bar{y}_{O,\text{rel}} - \bar{y}_O\right)}_{\text{(i) omitting O*NET-only}} + \underbrace{\left(\bar{y}_S - \bar{y}_{S,\text{rel}}\right)}_{\text{(ii) adding new}} + \underbrace{\left(\bar{y}_{S,\text{rel}} - \bar{y}_{O,\text{rel}}\right)}_{\text{(iii) shared activities}},$$

where (iii) compares the same activities as TaskNet and O*NET word them. The decomposition is computed within each occupation and averaged over the six, so the parts sum to the all-occupation difference. Equal statement weights also let the new-task term be split statement by statement (Section J.3), which is why the levels above use them.

Exposure weighted by working time. We first estimate how much working time each statement accounts for. Raters report how often it is performed (frequency) and how long it takes each time (duration), both as response categories. We convert these categories to midpoints, expressed as occurrences per week and hours per occurrence, and multiply them to estimate hours per week. We then weight this estimate by the share of raters who report the task as part of their job. The concrete mappings are given in Table 22.

Table 22: **Frequency and duration categories are converted to working time through interval midpoints.** A workday is 8 hours and a month 4.33 weeks. Bounded categories are mapped to their interval midpoints; the two open-ended top categories are capped at 40 occurrences per week and 40 hours per occurrence. A rater's hours per week on a statement is the product of the two values. The statement's working time is this product, averaged over the raters who perform the task, multiplied by the estimated share of raters who perform it.

Code	Category	Value	Derivation
Panel A. Frequency: occurrences per week			
1	Once per year or less	0.019	$1/52$
2	More than once per year	0.077	$4/52$ (quarterly)
3	More than once per month	0.577	$2.5/4.33$ (2.5 per month, 4.33 weeks per month)
4	More than once per week	2.5	2.5 per week (as for code 3)
5	Daily	5	five workdays
6	Several times per day	15	three per day, five workdays
7	Hourly or more often	40	once per hour, 40-hour week (open-ended; capped)
Panel B. Duration: hours per occurrence			
1	Less than 15 minutes	0.125	midpoint (7.5 minutes)
2	15 to less than 30 minutes	0.375	midpoint (22.5 minutes)
3	30 minutes to less than 1 hour	0.75	midpoint
4	1 to less than 2 hours	1.5	midpoint
5	2 to less than 4 hours	3	midpoint
6	4 hours to less than 1 workday	6	midpoint of 4 to 8 hours
7	1 to less than 2 workdays	12	1.5 workdays of 8 hours
8	2 to less than 4 workdays	24	3 workdays of 8 hours
9	4 workdays or more	40	one 40-hour week (open-ended; capped)

For example, a task in the "more than once per week" frequency category and the "30 minutes to less than 1 hour" duration category is assigned 2.5 occurrences per week and 0.75 hours per occurrence, yielding 1.875 hours per week. If 60 percent of raters report performing it, the task contributes $0.6 \times 1.875 = 1.125$ hours per week. Summing across statements gives the occupation's total estimated working time.

We then ask how much of this working time is spent on exposed tasks. For a given exposure level, we sum up the working time of all statements scoring at or above that level and divide by the occupation's total working time. For comparison, we also compute the unweighted share, the fraction of statements scoring at or above the same level, which treats every statement as an equal share of work.

J.2 Decomposition of Exposure Differences

Table 23 reports the decomposition of Section J.1 for each occupation. All three parts raise exposure under both measures.

Table 23: **Adding new tasks accounts for most of the RL-feasibility difference between TaskNet and O*NET.** Differences are TaskNet minus O*NET, weighting statements equally within occupations and the six occupations equally.

Contribution	Eloundou-β (0–1)	RL feasibility (0–100)
Omitting O*NET-only tasks	+0.022	+0.45
Adding new tasks	+0.014	+2.55
Changing shared activities	+0.010	+0.71
Total difference	+0.046	+3.70

J.3 Exposure of New Statements by Theme

We split the contribution of new tasks to the exposure difference (Section J.2) by the themes of the codebook (Section I). Within an occupation with n TaskNet statements, adding new tasks changes the mean by

$$\bar{y} - \bar{y}_{\text{rel}} = \frac{1}{n} \sum_{s \in \text{new}} \left(y_s - \bar{y}_{\text{rel}} \right),$$

where $\bar{y}_{\text{rel}}$ is the mean exposure of the occupation's related statements. Each new statement's term $(y_s - \bar{y}_{\text{rel}})/n$ is assigned to its themes (coordination, teaching and professional development, oversight and quality assurance, new technology), split equally when a statement carries several, and to *no theme* when it carries none. Averaging over occupations as in Table 23, the theme contributions sum exactly to the contribution of new tasks. All 79 new statements were coded blind to category.

Table 24: **Coordination and oversight tasks account for most of the exposure that new statements add.** A statement with several themes is split equally among them.

	RL feasibility	Eloundou-β
Adding new tasks (total)	+2.55	+0.014
Coordination	+0.98	+0.007
Oversight and quality assurance	+0.85	+0.008
New technology	+0.46	+0.004
Teaching and professional development	+0.23	−0.004
No theme	+0.03	−0.001

J.4 Exposure by Working Time

Table 25 reports the time-weighted and unweighted exposed shares defined in Section J.1.

Table 25: **Time weighting raises estimated exposure for some occupations and lowers it for others.** Share of task statements and of estimated working time at each exposure level, by occupation. Working time per statement is task frequency times duration, converted from response categories to midpoints (Table 22), weighted by the share of incumbents who report the task as relevant.

	Eloundou E1 (Directly Exposed)		RL-feasible (Score ≥ 50)	
Occupation	Statements	Time	Statements	Time
Software Developers	61%	78%	86%	95%
Lawyers	16%	11%	62%	61%
Elementary School Teachers	26%	12%	38%	19%
Emergency Medicine Physicians	3%	0%	29%	29%
Personal Care Aides	11%	6%	17%	7%
Dishwashers	0%	0%	0%	0%

J.5 Alignment with Observed AI Usage

We ask whether the statements of each list describe what people actually bring to an AI assistant, measuring observed AI use by matching conversations to tasks as in prior work (Appel et al., 2025, Chatterji et al., 2025, Massenkoff and McCrory, 2026). Matches capture AI use for occupational tasks rather than worker adoption (Tomlinson et al., 2025). The conversations come from the Open Economic Index (Somerstep et al., 2026), a subset of WildChat-4.8M (Zhao et al., 2024) in which an LLM has labeled each conversation as occupationally relevant or not and attached a country. We keep the 225 703 conversations from the United States that the index labels as work-relevant. Three cleaning steps follow. (1) We drop conversations whose first user turn is shorter than 20 characters, which are mostly greetings (223 819 remain). (2) We keep one copy of each identical user text sent by the same user, identified by WildChat's hashed IP address (210 932 remain). (3) We remove near-duplicates across all users: two conversations are near-duplicates if the MinHash Jaccard similarity of the character 5-grams in the first 1500 characters of their first user turns is at least 0.7, and we keep one conversation per group of near-duplicates. This leaves 71 953 conversations from 32 006 users. The last step mostly removes automated pipelines that send one prompt with varying inputs. It also collapses a single user's successive revisions of the same request into one conversation, and it misses pipelines whose fixed instruction is short relative to the varying input; the largest of these, which asks for Vietnamese textbook passages, is kept.

An LLM judge (gpt-4.1-mini, temperature 0) matches conversations in two steps and screens the matches in a third. Every call reads the user's turns only, truncated to their first and last 3000 characters, which affects 9.2% of conversations. Options are shuffled for every conversation and step.

1. **Occupation.** The judge assigns the conversation to one of the six occupations, each described by its O*NET definition, or to none (P24). This step is shared by both inventories.
2. **Statements.** For a conversation assigned to an occupation, the judge lists every statement of that occupation's list in the inventory that the request does work toward, or none (P25). Two rules handle cases a first version of the prompt got wrong: a request that produces something for later use (a plan, script, lesson or material) goes to the statement for preparing it, not to a statement for carrying out an activity with other people, and a request to write about the user's own job (a resume or cover letter) matches no statement.

3. **Use case.** Each matched conversation is classified as work, coursework, or personal use with the use-case classifier of the Anthropic Economic Index, verbatim (Appel et al., 2026). Only work conversations count, as in measures of observed exposure built on work-related usage (Massenkoff and McCrory, 2026).

Each conversation counts once, its weight split equally across the statements it matches (Tomlinson et al., 2025). The occupation and statement prompts are our own; their choice among listed options follows Handa et al. (2025), except that the judge may answer none and may list several statements.

Prompt 24: Chat-Log Matching: Occupation

```
Below is the user's side of a conversation with an AI assistant, followed by a list of
categories of occupational work.

Which ONE of them best describes work the user's request asks the assistant to do or help with?
A request that only shares a general skill with an option (writing, communicating, explaining,
researching, organizing) does not fit unless it does that option's specific work. If none fits,
answer NONE; for most requests, none will.

USER'S SIDE OF THE CONVERSATION:
<<<
<the user's turns, at most 6,000 characters (head and tail kept)>
>>>

Categories:
C01: <occupation title>: <O*NET occupation definition>
C02: ...
(the six occupations, in shuffled order)

Return JSON only: {"pick": "<id or NONE>"}
```

Prompt 25: Chat-Log Matching: Task Statements

```
Below is the user's side of a conversation with an AI assistant, followed by a list of
occupational task statements.

Which ONE statement, if any, does the user's request do real work toward: what the assistant is
asked to do would be part of performing that task, for a worker whose job includes it? A
request that only shares a general skill with a statement (writing, communicating, explaining,
researching, organizing) does not count unless it does that statement's specific work. If none
fits, answer NONE.

If you pick one, rate how much of a real instance of that task the request covers: A = all or
nearly all; B = most (more than 80%); C = a significant portion (more than 50%); D = a
non-negligible portion (more than 10%); E = 10% or less.

USER'S SIDE OF THE CONVERSATION:
<<<
<the user's turns, at most 6,000 characters (head and tail kept)>
>>>

TASK STATEMENTS:
T01: <task statement>
T02: ...
(every statement of the chosen occupation in one inventory, in shuffled order)

Return JSON only: {"pick": "<id or NONE>", "replaces": "<A-E, or empty if NONE>"}
```

Of the 71 953 conversations, 27 069 are assigned to one of the six occupations: Software Developers 21 911, Elementary School Teachers 2974, Lawyers 1690, Emergency Medicine Physicians 278, Personal Care Aides 188, and Dishwashers 28. We analyze the first three only (26 575 conversations). The statement step declines 18.7% of them for O*NET and 23.8% for TaskNet; of the matched conversations, 29.4% (O*NET) and 30.9% (TaskNet) list two or more statements. Of the 23 130 conversations

matched under either inventory, 13 435 are work, 5669 coursework, and 3987 personal (39 could not be classified). Coursework is concentrated among Elementary School Teachers, where it makes up 82% of matched conversations, against 18% for Software Developers and 12% for Lawyers. The final counts are 11 085 conversations under O*NET and 10 318 under TaskNet for Software Developers, 1179 and 1058 for Lawyers, and 313 and 317 for Elementary School Teachers.

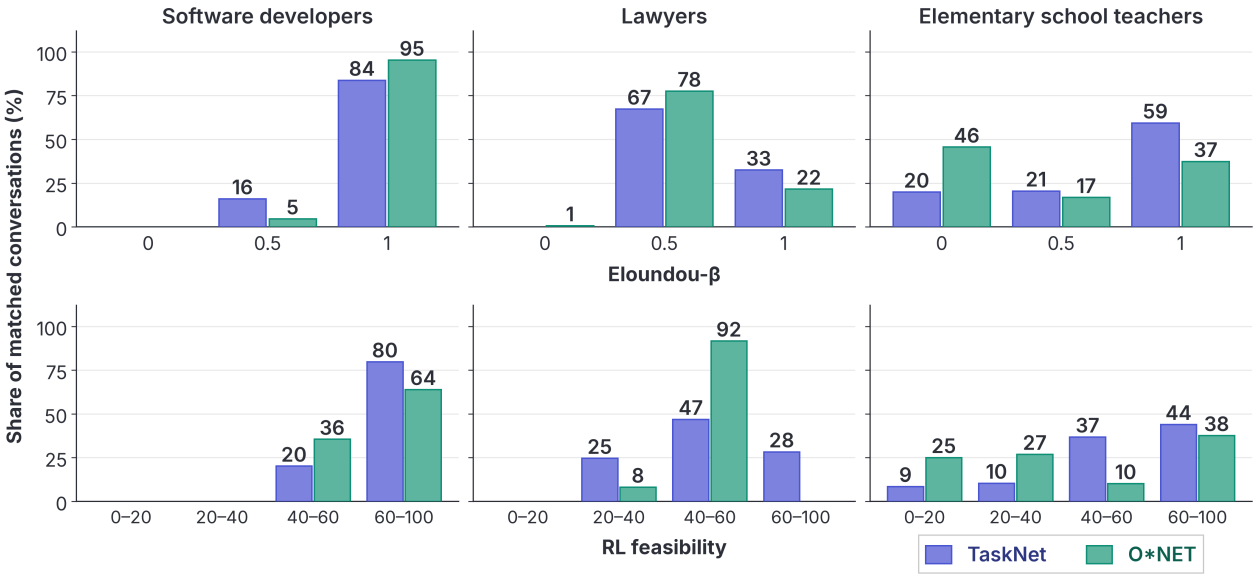


Figure 11: Exposure of the statements that AI conversations are matched to. For each occupation, bars show the percentage of an inventory's matched conversations that fall on statements at each level of Eloundou-β (top) and in each band of RL feasibility (bottom).

Exposure scores are consistent with observed AI use. Under both inventories, almost all matched conversations concern exposed statements with $\beta \geq 0.5$ (98.9 % under O*NET; 99.6 % under TaskNet). However, TaskNet shifts usage from partially to fully exposed statements: 37 % of Lawyer and 56 % of Teacher conversations match fully exposed statements, compared with 24 % and 36 % under O*NET (Figure 11). This partly reflects task bundling: O*NET combines activities with different exposure levels, while TaskNet separates them.

K Task Information

Beyond the statements themselves, TaskNet records information about each task that is relevant for policy and measurement. The 318 incumbents who rated TaskNet statements also reported each task's training requirements, frequency, duration, verifiability, and variation in performance. Figure 12 shows each statement's estimates by occupation, and Figure 13 their correlation with the two predicted exposure measures (Section 7).

Occupations differ in training requirements, task frequency, and task duration. Physicians, Personal Care Aides, and Dishwashers perform shorter tasks more frequently, while Software Developers, Lawyers, and Teachers perform longer tasks less often (Figure 12). This finding is consistent with how we define task granularity: as goal-directed activities that abstract from the

specific contexts in which they are performed. Under this definition, it is plausible that Software Developers, Lawyers, and Teachers work toward the same goal for longer, as for them "projects" are, by virtue of their jobs, longer. Emergency Physicians, Dishwashers, and Personal Care Aides have jobs that are structured by shorter intervals: visits in the emergency room, visits to those that the care aide cares for, and cleaning shifts, respectively.

The same two groups also differ in how training requirements vary across tasks. Physicians, Aides, and Dishwashers report similar requirements across their tasks, despite large differences in overall preparation, whereas Developers, Lawyers, and Teachers report more varied requirements. The variance suggests a greater prevalence of on-the-job training and task-specific practice beyond the required training to start in the job.

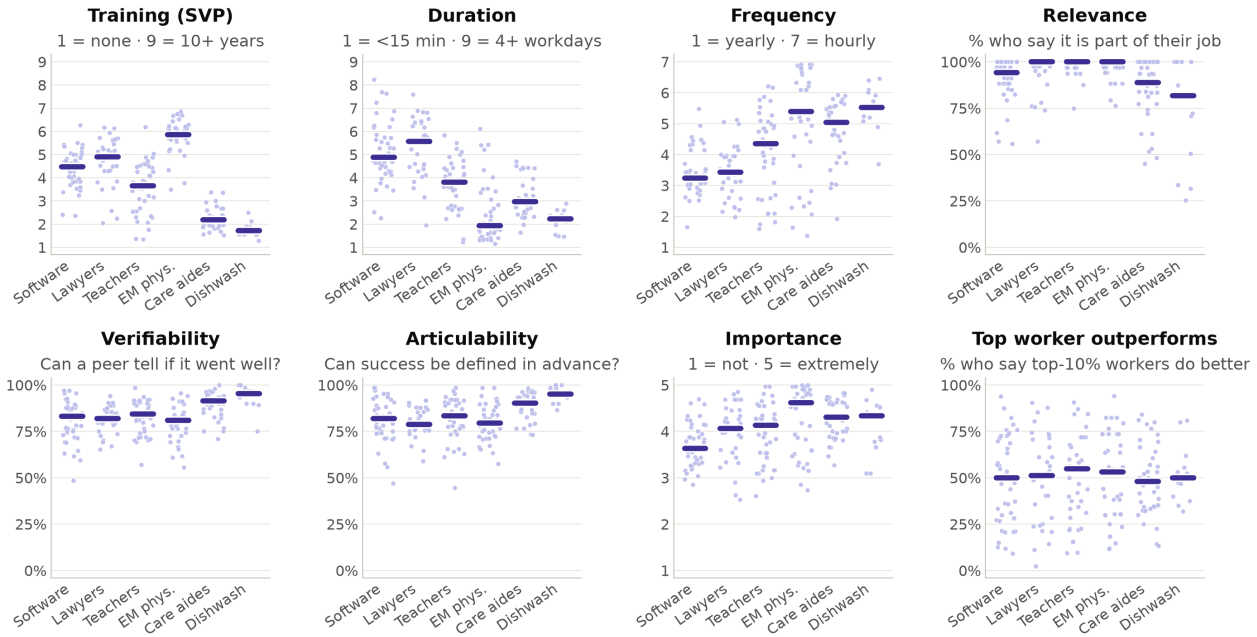


Figure 12: **Task characteristics by occupation.** Each dot is one TaskNet statement's estimate; bars mark occupation medians.

RL feasibility aligns with verifiability, but not expertise. Within occupations, tasks that are easier to describe and verify receive higher RL feasibility scores (Figure 13), while tasks requiring more vocational preparation receive lower scores ($\rho = -0.14$), as do tasks for which more respondents report that top workers clearly outperform median workers ($\rho = -0.10$). Eloundou-β is at most weakly related to any of these characteristics within occupations ($|\rho| \leq 0.14$).

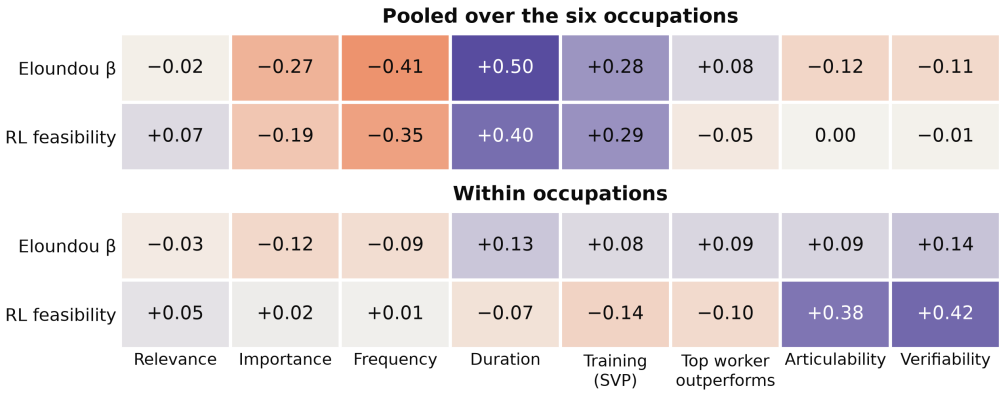


Figure 13: **Correlation of predicted exposure with task characteristics.** Spearman correlations across TaskNet statements. *Pooled*: over the statements of all six occupations. *Within occupations*: of percentile ranks computed within each occupation, so differences between occupations do not contribute.