



Automating the Enterprise with Foundation Models

Michael Wornow*
Stanford University
mwornow@stanford.edu

Avanika Narayan*
Stanford University
avanikan@stanford.edu

Krista Opsahl-Ong
Stanford University
kristao@stanford.edu

Quinn McIntyre
Stanford University
qam@stanford.edu

Nigam Shah
Stanford University
nigam@stanford.edu

Christopher Ré
Stanford University
chrismre@stanford.edu

ABSTRACT

Automating enterprise workflows could unlock \$4 trillion/year in productivity gains. Despite being of interest to the data management community for decades, the ultimate vision of end-to-end workflow automation has remained elusive. Current solutions rely on process mining and robotic process automation (RPA), in which a bot is hard-coded to follow a set of predefined rules for completing a workflow. Through case studies of a hospital and large B2B enterprise, we find that the adoption of RPA has been inhibited by high set-up costs (12-18 months), unreliable execution (60% initial accuracy), and burdensome maintenance (requiring multiple FTEs). Multimodal foundation models (FMs) such as GPT-4 offer a promising new approach for end-to-end workflow automation given their generalized reasoning and planning abilities. To study these capabilities we propose **ECLAIR**, a system to automate enterprise workflows with minimal human supervision. We conduct initial experiments showing that multimodal FMs can address the limitations of traditional RPA with (1) near-human-level understanding of workflows (93% accuracy on a workflow understanding task) and (2) instant set-up with minimal technical barrier (based solely on a natural language description of a workflow, **ECLAIR** achieves end-to-end completion rates of 40%). We identify human-AI collaboration, validation, and self-improvement as open challenges, and suggest ways they can be solved with data management techniques.

PVLDB Reference Format:

Michael Wornow, Avanika Narayan, Krista Opsahl-Ong, Quinn McIntyre, Nigam Shah, and Christopher Ré. Automating the Enterprise with Foundation Models. PVLDB, 17(11): 2805 - 2812, 2024.
doi:10.14778/3681954.3681964

PVLDB Artifact Availability:

The source code, data, and/or other artifacts have been made available at <https://github.com/HazyResearch/eclair-agents>.

1 INTRODUCTION

Digital workflows are the core of our modern economy, with 92% of jobs now requiring digital skills [10]. Many workflows can and should be automated — across industries, workers average 3 hrs/day

* Equal contributions

This work is licensed under the Creative Commons BY-NC-ND 4.0 International License. Visit <https://creativecommons.org/licenses/by-nc-nd/4.0/> to view a copy of this license. For any use beyond those covered by this license, obtain permission by emailing info@vldb.org. Copyright is held by the owner/author(s). Publication rights licensed to the VLDB Endowment.

Proceedings of the VLDB Endowment, Vol. 17, No. 11 ISSN 2150-8097.
doi:10.14778/3681954.3681964

doing repetitive digital workflows tangential to their core jobs [2], a phenomenon referred to as "death by 1,000 clicks" [67].

For example, consider a large B2B enterprise with tens of thousands of customers (full case study in Section 3.2). A key accounting workflow is ingesting customer contracts into a centralized database. After receiving a contract via email, an analyst must manually extract relevant data and enter it into an enterprise resource planning system (e.g. NetSuite, SAP, etc.) before they can conduct downstream analyses. Doing this at enterprise scale requires hundreds of individuals, each potentially taking 40 minutes per contract.

As illustrated above, most enterprise workflows involve data integration, ingestion, and transformation, and the market for process management is projected to reach \$65 billion by 2032 [23]. **Decades of data management work has gone into understanding and managing these workflows [15, 27, 34, 37, 66, 88], yet the ultimate vision of end-to-end automation — from understanding to execution to monitoring — has remained elusive.**

The current best-in-class solution is Robotic Process Automation (RPA), in which workflows identified via process mining get manually encoded into a fixed set of rules for a program to follow [41, 50, 51]. Narrowly scoped RPA deployments can have ROIs of 30-200% and 2x the speed of workflows [18, 42]. However, more widespread adoption of RPA has been limited by three key failure modes [65] which surfaced in interviews with technology leaders at a hospital and B2B enterprise (full case studies in Section 3):

- **High set-up costs:** The desired workflow must be *demonstrated* to the RPA bot. This is costly, as it requires a trained specialist to map workflows, write automation scripts, and integrate with IT infrastructure [34, 50, 58]. A leading RPA vendor estimates that 3-6 months of experience is needed to become proficient in RPA [72]. In our B2B case study, it took 12 months to go from project kickoff to deployment.
- **Brittle execution:** The RPA bot must *execute* the workflow. Since RPA relies on hard-coded rules, bots cannot adapt to slight variations in input (e.g., a button changing location on a screen, or a form field being renamed) [16, 79, 87]. This leads to a "death by a thousand cuts" as the space of possibilities is essentially unbounded. In our B2B case study, the RPA bot was initially only 60% accurate and took 6 months of improvement to reach 95%.
- **Burdensome maintenance:** RPA deployments often require human oversight to *validate* outputs and fix edge cases. In our B2B case study, the bot required continual monitoring by 2 full-time equivalents (FTEs).

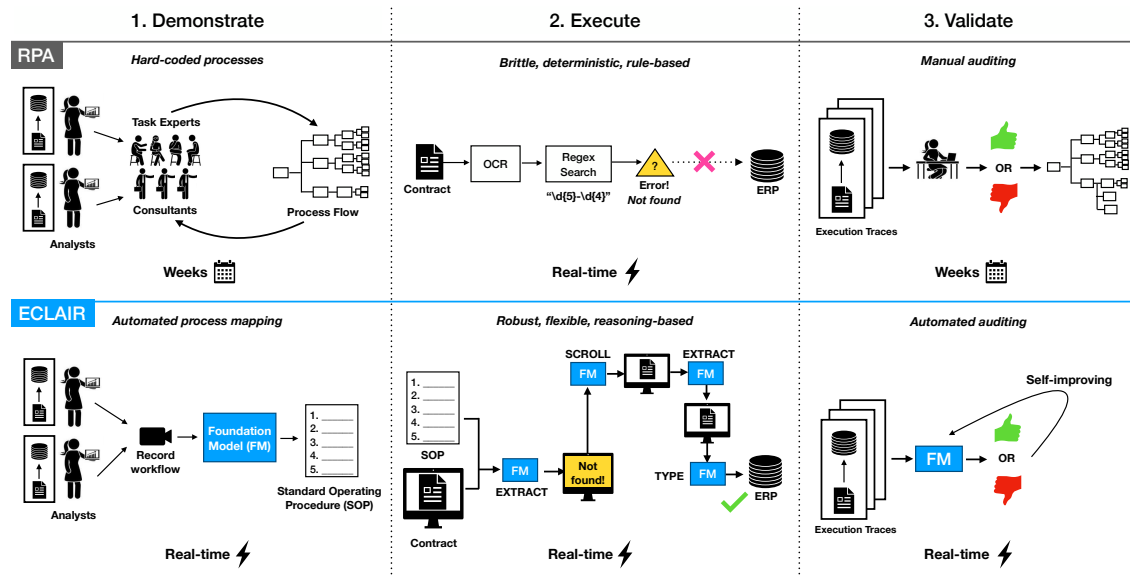


Figure 1: Differences between ECLAIR and traditional RPA. ECLAIR uses FMs to learn expertise via video demonstrations (left), navigate GUIs given written documentation (center), and audit completed workflows (right).

The common cause behind these shortcomings is the difficulty of encoding "tacit" (i.e. difficult to define) human workflow expertise into a rule-based system like RPA [5, 14, 51]. Automating most workflows requires planning a sequence of actions and executing them on a graphical user interface (GUI). This requires (a) visual understanding, e.g. identifying a button on a screen; (b) real-time decision making, e.g. knowing to scroll to locate a form field that has shifted location; and (c) common sense to error correct, e.g. hitting escape when an irrelevant pop-up appears.

Multimodal foundation models (FMs) such as GPT-4 [53] have demonstrated visual understanding [7, 76, 90] and generalized reasoning abilities [1, 75, 78, 86] for automating simple digital workflows [29, 33, 82, 83, 85, 89, 91]. This offers the possibility of sidestepping the failure modes of traditional RPA, just as deep learning eclipsed rule-based approaches over the past decade in machine learning. We thus ask the question: **Can multimodal foundation models automate enterprise workflows?**

We take a first natural step in studying the opportunities and challenges of applying multimodal FMs across all three stages of traditional RPA by proposing ECLAIR – "Enterprise sCaLe AI for woRkflows". As shown in Figure 1, our system is defined as follows:

- (1) **Demonstrate:** ECLAIR uses multimodal FMs to learn from human workflow expertise by watching video demonstrations and reading written documentation. This *lowers set-up costs* and *technical barriers to entry*. Initial experiments show that ECLAIR can identify every step of a workflow based on screenshots from a demonstration with 93% accuracy.
- (2) **Execute:** ECLAIR observes the state of the GUI and plans actions by leveraging the reasoning and visual understanding abilities of FMs [7, 53, 86]. Based solely on written documentation of a workflow, ECLAIR improves end-to-end completion rates over an existing GPT-4 baseline from 0%

to 40% on a sample of 30 web navigation tasks [92]. However, this is still far from the accuracy needed for enterprise settings, and we identify opportunities to close this gap.

- (3) **Validate:** ECLAIR utilizes FMs to self-monitor and error correct. This *reduces the need for human oversight*. When classifying whether a workflow was successfully completed, ECLAIR achieves a precision of 90% and recall of 84%.

Our initial evaluations also identify several patterned failure modes for future research. In **Execute**, ECLAIR has difficulty decomposing higher-level steps into discrete actions (e.g. breaking "Search XXX" into the sequence of "click", "type XXX", "press enter") and grounding actions to specific GUI elements (e.g. differentiating two buttons with the same label). In **Validate**, ECLAIR's lack of heuristics for navigating GUIs make step-level validation challenging (e.g. checking that a text field is first focused before typing).

While there is still progress to make, we are excited by the potential of ECLAIR to automate entirely new categories of workflows requiring real-time decisions, interaction with GUIs, and "tacit" domain knowledge [5], as outlined in Figure 2. McKinsey estimates this could double the amount of automatable knowledge work [17].

The rest of the paper is structured as follows. In **Section 2**, we discuss related work on process mining, RPA, and applying FMs to workflow automation. In **Section 3**, we provide case studies of a hospital and large B2B enterprise which highlight the limitations of RPA. In **Section 4**, we outline how ECLAIR can address these shortcomings. We conclude in **Section 5** with a discussion of future work and opportunities for the data management community.

Our contributions are: (1) two case studies highlighting the limitations of process mining / RPA, (2) a framework, ECLAIR, for achieving end-to-end enterprise workflow automation with multimodal FMs, (3) evaluations of ECLAIR on 30 workflows involving

enterprise web applications, and (4) proposals for applying data management techniques to workflow automation.

2 BACKGROUND

We survey related work on RPA, process mining, data management tools for workflow automation, and foundation models (FMs), then detail the problem setting that **ECLAIR** seeks to solve.

2.1 Related Work

Significant effort has gone into developing tools for understanding and automating workflows.

Foundation Models (FMs) are deep learning models trained on large datasets which can be adapted to a broad range of downstream tasks [13]. They have demonstrated robust world knowledge [38, 63], reasoning [78, 86], and planning abilities [1, 21, 69, 75], and have achieved state-of-the-art results on data processing tasks such as integration and cleaning [39, 52].

Process Mining is the identification and improvement of workflows based on observational data [4, 61, 73]. Recent works applied FMs to process mining tasks such as Petri net generation, workflow understanding, and process improvement [11, 22, 23, 28, 51, 62, 74], but were limited to small case studies and unimodal models.

Robotic Process Automation (RPA) is the leading approach for automating enterprise workflows [19, 36]. In RPA, a human manually defines a set of rules that a bot then follows to accomplish a specific workflow [16, 36, 79]. These fixed rulesets make RPA brittle (e.g. failing if a form changes the ordering of its fields) and difficult to maintain [24, 50]. FMs offer a compelling alternative due to their robust reasoning capabilities [1, 86] and ability to navigate GUIs [33]. Initial FM-based approaches [20, 29, 47] utilized Large Language Models (LLMs) to act on websites. Since LLMs can only understand text, these works relied on scraping a webpage’s HTML as input to the model. This prevented their application to native desktop and virtualized software. Multimodal FMs address this limitation by attaching a vision model to the base LLM [90], which enables them to directly reason over screenshots of a GUI [7, 26, 33, 35, 68]. Multimodal FMs have already shown promise in navigating websites [30, 85, 91], mobile apps [83], and desktop applications [82, 89]. We aim to design a system that helps bridge the gap between these proof-of-concepts and enterprise-level solutions.

Data Management for Workflow Automation has been studied for nearly two decades, with works ranging from business process management [15, 27, 60, 66, 88] to workflow automation and understanding [34, 37]. All of this work, however, pre-dates multimodal FMs. As a result, the challenges around which these systems were designed differ substantially from modern systems. Our work aims to build on these prior efforts from the data management community by developing a system that integrates FMs.

2.2 Problem Formulation

We aim to achieve end-to-end automation of enterprise workflows at minimal cost. We have a workflow $w \in \mathcal{W}$ which consists of a sequence of alternating states $s \in \mathcal{S}$ and actions $a \in \mathcal{A}$, such that $w = (s, a, s', a'', \dots)$. Each workflow w is done by a set of workers during the course of business operations. These workers follow a

standard operating procedure (“SOP”), a form of written documentation which outlines all of the steps and actions of the workflow. Once the workflows are executed, an auditing process validates (either manually or programmatically) whether each workflow w was completed successfully. The goal is to automate workflow w by learning from human demonstrations and written documentation (e.g. SOPs). We aim to directly operate on the GUI, as many workflows do not have APIs or must be executed in native desktop/virtualized applications that can only be observed visually [33]. This necessitates a vision-based approach [3, 89, 91].

3 CASE STUDIES

We conducted interviews with business leaders across a number of industries who led RPA projects. We select two organizations — a hospital and large B2B enterprise — for case studies. Both took over a year to deploy RPA pilot projects, and both declined to expand RPA to additional workflows due to its high set-up costs (\$100k’s and months of development), unreliable execution (accuracies started around 60% and peaked at 95%), and maintenance requirements.

3.1 Hospital Revenue Cycle Management (RCM)

Given the complexity of healthcare reimbursement, most hospitals have a dedicated Revenue Cycle Management (RCM) department to ensure that timely payment is collected for services delivered [40]. Example RCM workflows include verifying a patient’s insurance eligibility, obtaining prior authorization, and processing claims [40]. It is estimated that RCM processes cost roughly 15% of every dollar of revenue gained [8]. Additionally, 90% of health systems face RCM staffing shortages [59]. Despite increased interest in automating RCM workflows, most remain highly manual: roughly 94% of claims submissions and 76% of eligibility verifications involve manual labor [31]. In conversations with hospital IT leaders who considered automating certain RCM workflows, but only managed a limited deployment, the following weaknesses of RPA surfaced:

(1) High Set-Up Costs. Integration with a hospital’s IT infrastructure is costly, as is retraining staff [40]. Hospital leadership estimated that it took about 18 months and \$10k’s to develop and deploy their RPA bot. Significant back-and-forth with the vendor occurred as each workflow had to be manually mapped and coded into a set of well-defined, “always true” actions.

(2) Brittle Execution. It was estimated that ~ 0.2 FTEs worth of effort was saved with the RPA bot, as it could only handle two narrowly-scoped workflows involving one payer and one department. Quarterly updates to the hospital’s electronic health record and constant changes to payers’ websites would break the bot. Eventually, this required the hospital to develop a custom API that the bot could use to increase its reliability.

(3) Burdensome Maintenance. Given staffing constraints, the hospital outsourced continued oversight of the bot to the vendor as a managed service. Despite this outsourcing, RCM managers still manually reviewed outputs to ensure compliance.

3.2 B2B Enterprise Invoice Processing

Large B2B enterprises must ingest a wide range of complex contracts and process the information into systems of record such as NetSuite or SAP. These workflows can vary widely depending on

| Sample Workflow | Workflow Properties | | | RPA | ECLAIR |
|--|--------------------------|-----------------|---------------------|-----|--------|
| | Enumerable seq. of steps | Decision making | Knowledge intensive | | |
| Sending a templated post-visit follow-up email | ✓ | ✗ | ✗ | } | ECLAIR |
| Digitizing insurance claim documents | ✓ | ● | ✗ | | |
| Verifying a patient's insurance eligibility | ✓ | ✓ | ✗ | | |
| Ordering proper medication dosages | ✓ | ✓ | ● | | |
| Coordinating post-surgery recovery plan | ✓ | ✓ | ✓ | | |

Key

Does workflow have this property?

✓ Yes

● Somewhat

✗ No

Figure 2: ECLAIR can automate entirely new categories of workflows, such as those that contain hard-to-describe steps, require complex decision making, or are knowledge intensive. Listed examples are real-world hospital workflows (see Section 3.1).

the specific product being invoiced, the end purchaser, and the time the contract was written [64]. Given its importance to enterprise financial planning, invoice processing has also attracted attention from RPA vendors. We interviewed the head of revenue operations at a large B2B enterprise which tried to implement RPA for a single invoice processing workflow. After a year of development, 5 FTEs working with the RPA bot were able to successfully accomplish a workflow that previously took 20 FTEs. Despite this apparent success, however, the enterprise chose not to expand RPA to other workflows, as the implementation proved to be too painful:

(1) **High Set-Up Costs.** Going from initial contract with the RPA vendor (\$150k) to production deployment took over 12 months. External consultants (\$100k) and 3 FTEs were required to integrate the bot with existing IT systems. Challenges included (a) accurately defining the workflow; and (b) the steep learning curve for programming bots, which required a proprietary development toolkit.

(2) **Brittle Execution.** The RPA bot was initially only 60% accurate post-deployment and took 6 months of iteration to reach a final accuracy of 95%. Beyond invoice processing, only ~50% of desired workflows seemed feasible to automate with RPA.

(3) **Burdensome Maintenance.** The cost of a wrongly processed invoice is high (\$10k's), so 2 FTEs were allocated to continuously monitor the bot to debug issues, add new input formats, and inspect outputs. An outside firm also conducted manual batch reviews.

4 PRELIMINARY EVALUATIONS

In this section, we describe how ECLAIR can leverage multimodal FMs across all three stages of the traditional RPA pipeline.

We provide preliminary experiments assessing the feasibility of this approach. For our evaluations, we subsample 30 workflows from the WebArena benchmark [92]. WebArena provides a set of interactive websites in which an AI agent must complete complex workflows specified via natural language. Specifically, we choose 30 workflows from the Gitlab and Adobe Magento environments which their GPT-4 baseline model failed to complete [29]. We have human annotators record themselves completing each workflow and write a step-by-step guide ("SOP") on the steps they took.

4.1 Demonstrate

ECLAIR aims to learn from passively collected human demonstrations, with no updates to the underlying FM's weights. This limits the cost of labeling data, simplifies deployment, and avoids known biases that arise when humans try to articulate their work processes [44]. Our experiments show that GPT-4 can accurately identify the

steps of a workflow based on visual observation of a human demonstration, with step-level precision of 0.94 and recall of 0.95.

4.1.1 *Can ECLAIR determine the steps of a workflow by viewing raw video demonstrations?* This would enable ECLAIR to substantially improve the effectiveness and scalability of process mining.

Hypothesis: A multimodal FM can generate accurate SOPs based on screenshots taken at key frames from a video recording.

Set-Up: We prompt GPT-4 to generate an SOP for a workflow given a human demonstration. We ablate various ways to provide the demonstration: just the workflow description (WD); the workflow description and screenshots of key frames from a video recording of the demonstration (WD+KF); or the workflow description, key frames, and a textual action log of each click and keystroke (WD+KF+ACT). Using a manually-written SOP as reference, a human annotator calculates GPT-4's precision ("What percent of steps in the GPT-4 SOP are in the true SOP?"), recall ("What percent of steps in the true SOP are in the GPT-4 SOP?"), and correctness ("By following the GPT-4 SOP, can I complete the workflow?").

Table 1: (Demonstrate) GPT-4 generation of SOPs. Metrics averaged across all 30 workflows.

| Method | # of Steps in SOP | | | Accuracy of SOP | | |
|--------------|-------------------|-----------|-------|-----------------|-------------|-------------|
| | Missing | Incorrect | Total | Precision | Recall | Correctness |
| WD | 1.57 | 3.58 | 13.67 | 0.75 | 0.81 | 0.60 |
| WD+KF | 0.67 | 1.05 | 10.17 | 0.89 | 0.92 | 0.90 |
| WD+KF+ACT | 0.63 | 0.57 | 9.63 | 0.94 | 0.95 | 0.93 |
| Ground truth | 0 | 0 | 8.70 | 1 | 1 | 1 |

Takeaways: As shown in Table 1, the SOPs generated by GPT-4 using the WD+KF+ACT strategy are judged as sufficiently "correct" to complete 93% of workflows. Even providing GPT-4 with screenshots alone (WD+KF) achieves 90% correctness. However, WD+KF experiences almost twice as many hallucinations (1.05 incorrect steps per SOP versus only 0.57 when the action trace is included). Note that we do not do any workflow-specific prompt engineering or data labeling to achieve these results. Additionally, we preprocess our video demonstrations into a sequence of key frames using imperfect heuristics (i.e. alignment with clicks and keystrokes). Future work may benefit from using a model that can directly process video (rather than just images).

4.2 Execute

After defining the workflow, ECLAIR must execute a sequence of steps to accomplish it. We divide each step into two phases:

(1) action suggestion, i.e. planning what action to take; (2) action grounding, i.e. turning the plan into actual clicks/keystrokes of GUI elements. We find that providing domain knowledge to **ECLAIR** via SOPs doubles workflow completion rates, while general models (e.g. GPT-4) lag smaller models fine-tuned on GUIs for action grounding.

4.2.1 Can **ECLAIR accurately suggest the next action to take in a workflow?** We investigate whether multimodal FMs can predict the next action in a workflow based on the current state of the GUI and action history. We evaluate if providing the FM with an SOP for the workflow increases overall workflow completion rates.

Hypothesis: Providing high-level natural language guidance to the model via an SOP improves workflow completion rates.

Set-Up: At each step of the workflow, the model takes as input the ground truth history of actions, full SOP, and current GUI, and is expected to generate the next action to take. We measure the accuracy of the model’s suggested next action using a human annotator to evaluate whether it is semantically equivalent to the corresponding "ground truth" action for the workflow.

Table 2: (Execute) GPT-4 average accuracy on next action suggestion with and without SOP guidance.

| SOP | Next Action Suggestion Acc. | Overall Workflow Completion Acc. |
|-----|-----------------------------|----------------------------------|
| ✗ | 0.83 | 0.17 |
| ✓ | 0.92 | 0.40 |

Takeaways: Our results in Table 2 demonstrate that SOPs improve overall workflow completion rates by up to 23 points. While the SOP boosts the accuracy of each step’s action suggestion to 0.92, the model struggles to associate these suggested actions with the appropriate GUI elements, thereby deflating its overall completion rate (recall that the model must ground its actions to successfully complete the workflow). For example, when the action is "Click on the profile button", but the HTML element for the icon is identified as "svg" rather than "button", the model fails to select this element.

4.2.2 Can **ECLAIR accurately "ground" its actions suggestions to GUI elements?** Once **ECLAIR** suggests an action to take (i.e. "Click on the "Submit" button"), it must then map the action into mouse/keyboard commands which specify the pixel location of the GUI element to interact with [91]. Multimodal FMs are known to have difficulty with this, so we evaluate several grounding strategies [91].

Hypothesis: Providing bounding boxes for each element in a GUI improves a multimodal FM’s ability to ground elements.

Set-Up: We sample a total of 120 and 302 webpages from the WebUI [80] and Mind2Web [20] datasets, respectively. We create a natural language description for one element on each page, then prompt a model to generate a bounding box (BB) for that element given a natural language description of the BB and screenshot of the webpage. We measure "accuracy" as the percentage of predicted BBs whose center is within the element’s true BB (i.e. *If the model clicked on the center of its prediction, would it successfully hit the target element?*). We evaluate GPT-4 [53] and CogAgent [33] as two state-of-the-art closed/open source multimodal FMs for GUI navigation. While CogAgent directly outputs BBs, GPT-4 does not. Thus, we use "set-of-marks" prompting for GPT-4, in which we

overlay a unique numeric label on top of every element in the webpage screenshot provided to GPT-4, and have it output the number of a labeled element [84]. We generate these labels either directly from the webpage’s HTML ("HTML") or from a YOLONAS object detection model ("YOLO") finetuned on 7k WebUI webpages [80]. The latter simulates the setting where HTML is not available.

Table 3: (Execute) Accuracy on grounding actions to GUI elements. "S | M | L" is accuracy on small, medium, and large elements. Note: We found that HTML bounding boxes were not accurate in Mind2Web, so it is excluded.

| Model | Bbox Source | Mind2Web | | | WebUI | | | | |
|----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | | S | M | L | Overall | S | M | L | Overall |
| GPT-4 | – | 0.01 | 0.03 | 0.16 | 0.07 | 0.00 | 0.03 | 0.14 | 0.05 |
| GPT-4 | YOLO | 0.38 | 0.68 | 0.80 | 0.62 | 0.50 | 0.58 | 0.69 | 0.58 |
| GPT-4 | HTML | – | – | – | – | 0.56 | 0.58 | 0.67 | 0.60 |
| CogAgent | – | 0.55 | 0.71 | 0.87 | 0.71 | 0.71 | 0.64 | 0.75 | 0.70 |

Takeaways: The 18-billion parameter CogAgent outperforms GPT-4 on action grounding. This suggests that smaller models purpose-built for GUI navigation can be more effective than larger, general purpose multimodal FMs. However, overall accuracy peaks at 70%, and interacting with smaller-sized elements remains challenging. For GPT-4, using the bounding boxes generated by the YOLONAS model performs similarly to the "ground truth" HTML boxes. This suggests that detecting elements on a GUI with a vision model is not the bottleneck, but rather choosing which of those detected elements is the desired element to interact with.

4.3 Validate

There are several levels of validation that **ECLAIR** must provide. At the individual step-level, the agent should validate that its action suggestions are (a) feasible to execute and (b) making progress towards accomplishing the workflow. At the workflow-level, the agent should understand (a) whether the workflow was successfully completed and (b) whether the steps taken to achieve the workflow were sensible. We find that GPT-4 struggles with the former lower-level details but performs well at the latter higher-level reasoning.

4.3.1 Can **ECLAIR self-monitor at the individual step-level?** Identifying errors in individual actions would allow **ECLAIR** to perform error correction in real-time. This would improve the reliability of the Execution stage by enabling the model to avoid undesirable states and backtrack where appropriate.

Hypothesis: A multimodal FM can detect if an action will succeed or fail based on visual observation of changes in screen state.

Set-Up: First, we test the model’s ability to detect if an action failed (e.g. typing had no effect because no text field was first focused). We sample (s, a, s') traces from our dataset (positive examples) and generate tuples where $s' = s$ (negatives). We prompt GPT-4 to identify if the action a in (s, a, s') was successfully executed, given screen shots for s and s' . We sample three negatives for each positive example. Second, inspired by prior work on data cleaning [6, 9], we create a set of "integrity constraints" defining whether an action is viable at a particular state. For example, an "integrity constraint" for clicking a button is that the button is visible and not disabled. We annotate constraints for all actions in our dataset,

then prompt GPT-4 with (c, s) pairs where c is the constraint for the action directly after state s (positive examples) and (c, s') pairs where s' is a random state occurring before s (negatives).

Table 4: (Validate) Performance of GPT-4 on self-validation tasks. Metrics averaged across all 30 workflows.

| Eval Type | Precision | Recall | F1 |
|----------------------|-----------|--------|------|
| Actuation | 0.95 | 0.85 | 0.90 |
| Integrity Constraint | 0.67 | 0.36 | 0.47 |
| Workflow Completion | 0.90 | 0.84 | 0.87 |
| Workflow Trajectory | 0.88 | 0.83 | 0.85 |

Takeaways: The results are shown in rows "Actuation" and "Integrity Constraint" in Table 4. GPT-4 has a high precision (0.95) and recall (0.85) when assessing whether an action was successfully executed. However, the low integrity constraint scores indicate that it struggles to identify which actions are viable given the state of the GUI. This could be due to only observing static screenshots, which makes it difficult to discern animations such as a blinking cursor. These results suggest that current multimodal FMs cannot adequately self-monitor for enterprise use cases.

4.3.2 Can multimodal FMs self-monitor at the overall workflow-level? Understanding whether a workflow was successfully completed or not can help the agent (a) at runtime to know when to stop execution, and (b) post-deployment for self-auditing.

Hypothesis: A multimodal FM can determine if a demonstration correctly achieved a workflow based on visual observation of changes in screen state.

Set-Up: First, we test if the model can determine if it successfully completed a workflow. To evaluate this, we sample full traces of $(s^1, a^1, \dots, a^{n-1}, s^n)$ from our dataset (positive examples) and truncate some by a random number of frames to get $(s^1, a^1, \dots, a^{k-1}, s^k)$ (negatives) where $k < n$. Given the trace and workflow description, we prompt GPT-4 to provide a binary assessment of whether the workflow was successfully completed. Second, we investigate whether the model understands the proper *trajectory* of actions in a workflow – i.e. it is not sufficient to merely complete the workflow, but the steps taken to complete it must align with its SOP. We sample full traces $(s^1, a^1, \dots, a^{n-1}, s^n)$ from our dataset (positives) and either (a) randomly shuffle or (b) randomly delete frames from this trace (negatives). We provide GPT-4 with the SOP for the workflow, the workflow description, and the trace, and have it output a binary assessment of whether the trace exactly followed the SOP.

Takeaways: The F1 scores of 0.87 and 0.85 in rows "Workflow Completion" and "Workflow Trajectory" of Table 4 suggest that GPT-4 can self-monitor higher-level properties of a workflow.

5 DISCUSSION

To be deployed, **ECLAIR** must meet a minimum level of performance. While this is highly workflow-dependent, **ECLAIR** must be accurate enough such that the cost of correcting its errors is outweighed by the efficiency gains of using it (which might not require 100% accuracy). In our Section 3.2 case study, for example, an accuracy of 95% was sufficient for the RPA bot to be deployed. Defining such success criteria can be done via interviews with stakeholders,

financial modeling, and other enterprise planning methods [41] We discuss considerations for deploying **ECLAIR** below.

Error Handling and Monitoring. We envision a multi-tiered system which combines (1) self-validation, (2) programmatic heuristics, and (3) limited human intervention to provide error correction. (1) Per our Section 4.3 experiments, a repository of integrity constraints – which have successfully enhanced the quality of database schemas [6, 9] – could improve the accuracy of FM self-monitoring. Though FMs can exhibit non-deterministic behavior, their reliability can be improved by setting their temperature to 0, repeatedly querying [70] and ensembling predictions [43], or eliciting confidence scores to surface cases where intervention is necessary [71]. (2) Programmatic heuristics can also be used to detect failures, e.g. deviation from the average time to execute a workflow. (3) Existing methods for auditing human workflows can be re-applied, such as random spot checks or screenshots of confirmation screens. However, we envision such monitoring to be more limited – namely, to close the knowledge gap between **ECLAIR** and a domain – as the outputs from monitoring can be used to improve **ECLAIR** [54].

Human-ECLAIR Collaboration. While the long-term vision for **ECLAIR** is to require minimal human interaction – i.e. only the Demonstrate phase (Section 4.1) needing human involvement – we acknowledge that human supervision may be necessary for certain workflows. For example, a physician sign-off before prescribing medications or tasks involving user authentication. To accomplish this, the SOP could mark steps where the model transfers control to a human. Alternatively, a whitelist of sensitive actions can be compiled to automatically force transfer of control to a human when triggered, similar to how kernels use interrupts to handle control flow [49]. Finally, as mentioned in the prior section, generated human-**ECLAIR** execution traces can be used to improve **ECLAIR** performance via fine-tuning or few-shot prompting [54].

Self-Improvement. As **ECLAIR** repeatedly executes a workflow, it can observe the effects of its actions on the environment. By documenting these observations, **ECLAIR** can compile a database of common "skills" that can later be transferred to different workflows [25, 55, 77, 82, 85]. Applying principles from self-driving databases, which aim to continuously improve performance by implementing sequences of "actions" (i.e. changes to their configurations) based on utilization patterns [48, 56, 57], could provide a principled approach for such a self-improving workflow automation system.

Multi-Agent Collaboration. Applying multiple agents to the same task can improve accuracy [43, 45], as seen in recent work on multi-agent software engineering [32] and chatbot applications[81]. Such an approach could be utilized within the **ECLAIR** framework to create specialized agents for distinct subtasks or digital environments. Prior work on collaborative data processing tools offers a reference for how **ECLAIR** can be scaled to multi-agent (and multi-human) workflows with shared resources. [12, 46].

6 CONCLUSION

We are excited for the potential of multimodal FMs to reimagine how work gets done. By addressing the three main shortcomings of traditional process mining and RPA (high set-up costs, brittle execution, and burdensome maintenance), the realization of **ECLAIR** can help achieve the promise of enterprise workflow automation.

REFERENCES

- [1] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. 2022. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691* (2022).
- [2] Automation Anywhere. 2020. <https://www.automationanywhere.com/company/press-room/global-research-reveals-worlds-most-hated-office-tasks>
- [3] Rim Assouel, Tom Marty, Massimo Caccia, Issam H Laradji, Alexandre Drouin, Sai Rajeswar, Hector Palacios, Quentin Cappart, David Vazquez, Nicolas Chapados, et al. 2023. The Unsolved Challenges of LLMs as Generalist Web Agents: A Case Study. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.
- [4] Adriano Augusto, Raffaele Conforti, Marlon Dumas, Marcello La Rosa, Fabrizio Maria Maggi, Andrea Marrella, Massimo Mecella, and Allar Soo. 2018. Automated discovery of process models from event logs: Review and benchmark. *IEEE transactions on knowledge and data engineering* 31, 4 (2018), 686–705.
- [5] David Autor. 2014. *Polanyi's paradox and the shape of employment growth*. Technical Report. National Bureau of Economic Research.
- [6] Ioannis G Baltopoulos, Johannes Borgström, and Andrew D Gordon. 2011. Maintaining database integrity with refinement types. In *European Conference on Object-Oriented Programming*. Springer, 484–509.
- [7] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Saġnak Taşlar. 2023. Introducing our Multimodal Models. <https://www.adept.ai/blog/fuyu-8b>
- [8] Matthew Bayley and Ed Levine. 2013. Hospital revenue cycle operations: opportunities created by the ACA. *Management* (2013).
- [9] Michael Benedikt, Julien Leblay, and Efthymia Tsamoura. 2015. Querying with access patterns and integrity constraints. *Proceedings of the VLDB Endowment* 8, 6 (2015), 690–701.
- [10] Amanda Bergson-Shilcock and Roderick Taylor. 2023. Closing the Digital "Skill" Divide: The Payoff for Workers, Business, and the Economy. *National Skills Coalition* (2023).
- [11] Alessandro Berti and Mahnaz Sadat Qafari. 2023. Leveraging Large Language Models (LLMs) for Process Mining (Technical Report). *arXiv preprint arXiv:2307.12701* (2023).
- [12] Anant Bhardwaj, David Karger, Harihar Subramanyam, Amol Deshpande, Sam Madden, Eugene Wu, Aaron Elmore, Aditya Parameswaran, and Rebecca Zhang. 2015. Collaborative data analytics with DataHub. In *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, Vol. 8. NIH Public Access, 1916.
- [13] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* (2021).
- [14] Erik Brynjolfsson, Danielle Li, and Lindsey R Raymond. 2023. *Generative AI at work*. Technical Report. National Bureau of Economic Research.
- [15] Fabio Casati and Ming-Chien Shan. 2000. Process automation as the foundation for e-business. In *VLDB*. Citeseer, 688–691.
- [16] Tathagata Chakraborti, Vatche Isahagian, Rania Khalaf, Yasaman Khazaeni, Vinod Muthusamy, Yara Rizk, and Merve Unuvar. 2020. From Robotic Process Automation to Intelligent Process Automation: –Emerging Trends–. In *Business Process Management: Blockchain and Robotic Process Automation Forum: BPM 2020 Blockchain and RPA Forum, Seville, Spain, September 13–18, 2020, Proceedings 18*. Springer, 215–228.
- [17] M Chui, E Hazan, R Roberts, A Singla, K Smaje, A Sukharevsky, L Yee, and R Zimmel. 2023. The economic potential of generative AI The next productivity frontier The economic potential of generative AI: The next productivity frontier.
- [18] Cristiano André da Costa, Uélison Jean Lopes dos Santos, Eduardo Souza dos Reis, Rodolfo Stoffel Antunes, Henrique Chaves Pacheco, Thayná da Silva França, Rodrigo da Rosa Righi, Jorge Luis Victória Barbosa, Franklin Jebadoss, Jorge Montalva, et al. 2023. Intelligent methods for business rule processing: State-of-the-art. *arXiv preprint arXiv:2311.11775* (2023).
- [19] Laila Dahabiyeh and Omar Mowafi. 2023. Challenges of using RPA in auditing: A socio-technical systems approach. *Intelligent Systems in Accounting, Finance and Management* (2023).
- [20] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. 2023. Mind2Web: Towards a Generalist Agent for the Web. *arXiv:2306.06070* [cs.CL]
- [21] Norman Di Palo, Arunkumar Byravan, Leonard Hasenclever, Markus Wulfmeier, Nicolas Heess, and Martin Riedmiller. 2023. Towards a unified agent with foundation models. *arXiv preprint arXiv:2307.09668* (2023).
- [22] Marlon Dumas, Fabiana Fournier, Lior Limonad, Andrea Marrella, Marco Montali, Jana-Rebecca Rehse, Rafael Accorsi, Diego Calvanese, Giuseppe De Giacomo, Dirk Fahland, et al. 2023. AI-augmented business process management systems: a research manifesto. *ACM Transactions on Management Information Systems* 14, 1 (2023), 1–19.
- [23] Dirk Fahland, Fabian Fournier, Lior Limonad, Inna Skarbovsky, and Ava JE Swevels. 2024. How well can large language models explain business processes? *arXiv preprint arXiv:2401.12846* (2024).
- [24] Dahlia Fernandez and Aini Aman. 2021. The challenges of implementing robotic process automation in global business services. *International Journal of Business and Society* 22, 3 (2021), 1269–1282.
- [25] Daocheng Fu, Xin Li, Licheng Wen, Min Dou, Pinlong Cai, Botian Shi, and Yu Qiao. 2024. Drive like a human: Rethinking autonomous driving with large language models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 910–919.
- [26] Hiroki Furuta, Ofir Nachum, Kuang-Huei Lee, Yutaka Matsuo, Shixiang Shane Gu, and Izzeddin Gur. 2023. Multimodal Web Navigation with Instruction-Finetuned Foundation Models. *arXiv preprint arXiv:2305.11854* (2023).
- [27] Diimitrios Georgakopoulos, Mark Hornick, and Amit Sheth. 1995. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and parallel Databases* 3 (1995), 119–153.
- [28] Michael Grohs, Luka Abb, Nourhan Elsayed, and Jana-Rebecca Rehse. 2023. Large Language Models can accomplish Business Process Management Tasks. In *International Conference on Business Process Management*. Springer, 453–465.
- [29] Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. 2023. A real-world webagent with planning, long context understanding, and program synthesis. *arXiv preprint arXiv:2307.12856* (2023).
- [30] Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. 2024. WebVoyager: Building an End-to-End Web Agent with Large Multimodal Models. *arXiv:2401.13919* [cs.CL]
- [31] Sarah Calkins Holloway, Michael Peterson, Andrew MacDonald, and Bridget Scherbring Pollak. 2018. From revenue cycle management to revenue excellence.
- [32] Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, et al. 2023. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352* (2023).
- [33] Wenyi Hong, Weihang Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. 2023. CogAgent: A Visual Language Model for GUI Agents. *arXiv preprint arXiv:2312.08914* (2023).
- [34] Richard Hull, Jianwen Su, and Roman Vaculin. 2013. Data management perspectives on business process management: tutorial overview. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*. 943–948.
- [35] Peter C Humphreys, David Raposo, Tobias Pohlen, Gregory Thornton, Rachita Chhparia, Alistair Muldal, Josh Abramson, Petko Georgiev, Adam Santoro, and Timothy Lillicrap. 2022. A data-driven approach for learning to control computers. In *International Conference on Machine Learning*. PMLR, 9466–9482.
- [36] Lucija Ivančić, Dalia Suša Vugec, and Vesna Bosilj Vukšić. 2019. Robotic process automation: systematic literature review. In *Business Process Management: Blockchain and Central and Eastern Europe Forum: BPM 2019 Blockchain and CEE Forum, Vienna, Austria, September 1–6, 2019, Proceedings 17*. Springer, 280–295.
- [37] Nicholas R. Jennings, Timothy J. Norman, and Peyton Faratin. 1998. ADEPT: An agent-based approach to business process management. *ACM Sigmod Record* 27, 4 (1998), 32–39.
- [38] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics* 8 (2020), 423–438.
- [39] Moe Kayali, Anton Lykov, Ilias Fountalis, Nikolaos Vasiloglou, Dan Olteanu, and Dan Suciu. 2023. CHORUS: Foundation Models for Unified Data Discovery and Exploration. *arXiv preprint arXiv:2306.09610* (2023).
- [40] Victor Kilanko. 2023. Leveraging Artificial Intelligence for Enhanced Revenue Cycle Management in the United States. *International Journal of Scientific Advances* 4, 4 (2023), 505–14.
- [41] Volodymyr Leno, Artem Polyvyanyy, Marlon Dumas, Marcello La Rosa, and Fabrizio Maria Maggi. 2021. Robotic process mining: vision and challenges. *Business & Information Systems Engineering* 63 (2021), 301–314.
- [42] Xavier Lhuier. 2016. The next acronym you need to know about: RPA (robotic process automation). (2016).
- [43] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. 2024. More agents is all you need. *arXiv preprint arXiv:2402.05120* (2024).
- [44] Toby Jia-Jun Li, Marissa Radensky, Justin Jia, Kirielle Singarajah, Tom M Mitchell, and Brad A Myers. 2019. Interactive task and concept learning from natural language instructions and gui demonstrations. *arXiv preprint arXiv:1909.00031* (2019).
- [45] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujun Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118* (2023).
- [46] Xiaozhen Liu, Zuozhi Wang, Shengquan Ni, Sadeem Alsudais, Yicong Huang, Avinash Kumar, and Chen Li. 2022. Demonstration of collaborative and interactive workflow-based data analytics in texera. *Proceedings of the VLDB Endowment* 15, 12 (2022), 3738–3741.
- [47] Zhiwei Liu, Weiran Yao, Jianguo Zhang, Le Xue, Shelby Heinecke, Rithesh Murthy, Yihao Feng, Zeyuan Chen, Juan Carlos Niebles, Devansh Arpit, et al. 2023. Bolaa: Benchmarking and orchestrating llm-augmented autonomous agents.

- arXiv preprint arXiv:2308.05960 (2023).
- [48] Lin Ma, Dana Van Aken, Ahmed Hefny, Gustavo Mezerhane, Andrew Pavlo, and Geoffrey J Gordon. 2018. Query-based workload forecasting for self-driving database management systems. In *Proceedings of the 2018 International Conference on Management of Data*. 631–645.
- [49] Pedro Mejia-Alvarez, Luis Eduardo Leyva-del Foyo, and Arnaldo Diaz-Ramirez. 2018. *Interrupt Handling Schemes in Operating Systems*. Springer.
- [50] Silvia Moreira, Henrique S Mamede, and Arnaldo Santos. 2023. Process automation using RPA—a literature review. *Procedia Computer Science* 219 (2023), 244–254.
- [51] Vinod Muthusamy, Yara Rizk, Kiran Kate, Praveen Venkateswaran, Vatche Isahagian, Ashu Gulati, and Parijat Dube. 2023. Towards large language model-based personal agents in the enterprise: Current trends and open problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. 6909–6921.
- [52] Avaniika Narayan, Ines Chami, Laurel Orr, and Christopher Ré. 2022. Can Foundation Models Wrangle Your Data? *Proceedings of the VLDB Endowment* 16, 4 (2022), 738–746.
- [53] R OpenAI. 2023. GPT-4 technical report. *arXiv* (2023), 2303–08774.
- [54] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.
- [55] Joon Sung Park, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–22.
- [56] Andrew Pavlo, Gustavo Angulo, Joy Arulraj, Haibin Lin, Jiexi Lin, Lin Ma, Prashanth Menon, Todd C Mowry, Matthew Perron, Ian Quah, et al. 2017. Self-Driving Database Management Systems. In *CIDR*, Vol. 4. 1.
- [57] Andrew Pavlo, Matthew Butrovich, Lin Ma, Prashanth Menon, Wan Shen Lim, Dana Van Aken, and William Zhang. 2021. Make your database system dream of electric sheep: towards self-driving operation. *Proceedings of the VLDB Endowment* 14, 12 (2021), 3211–3221.
- [58] Arif Perdana, W Eric Lee, and Chu Mui Kim. 2023. Prototyping and implementing Robotic Process Automation in accounting firms: Benefits, challenges and opportunities to audit automation. *International Journal of Accounting Information Systems* 51 (2023), 100641.
- [59] R1. 2022. Healthcare Financial Trends Report. <https://www.r1rcm.com/news/healthcare-trends-and-data-show-clinical-shortage-tip-of-the-iceberg>
- [60] Habibur Rahman, Saravanan Thirumuruganathan, Senjuti Basu Roy, Sihem Amer-Yahia, and Gautam Das. 2015. Worker skill estimation in team-based tasks. *Proceedings of the VLDB Endowment* 8, 11 (2015), 1142–1153.
- [61] Lars Reinkemeyer. 2020. Process mining in action. *Process Mining in Action Principles, Use Cases and Outlook* (2020).
- [62] Yara Rizk, Praveen Venkateswaran, Vatche Isahagian, Austin Narcomey, and Vinod Muthusamy. 2023. A Case for Business Process-Specific Foundation Models. In *International Conference on Business Process Management*. Springer, 44–56.
- [63] Tara Safavi and Danai Koutra. 2021. Relational world knowledge representation in contextual language models: A review. *arXiv preprint arXiv:2104.05837* (2021).
- [64] Sagar Sahu, Sania Salwekar, Atharva Pandit, and Manoj Patil. 2020. Invoice processing using robotic process automation. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol* 6, 2 (2020), 216–223.
- [65] Henriika Sarilo-Kankaanranta and Lauri Frank. 2021. The Slow Adoption Rate of Software Robotics in Accounting and Payroll Services and the Role of Resistance to Change in Innovation-Decision Process. In *Conference of the Italian Chapter of AIS*. Springer, 201–216.
- [66] Mehmet Sayal, Fabio Casati, Umeshwar Dayal, and Ming-Chien Shan. 2002. Business process cockpit. In *VLDB’02: Proceedings of the 28th International Conference on Very Large Databases*. Elsevier, 880–883.
- [67] Fred Schulte and Erika Fry. 2019. Death by 1,000 clicks: Where electronic health records went wrong. *Kaiser Health News* 18 (2019).
- [68] Peter Shaw, Mandar Joshi, James Cohan, Jonathan Berant, Panupong Pasupat, Hexiang Hu, Urvashi Khandelwal, Kenton Lee, and Kristina Toutanova. 2023. From Pixels to UI Actions: Learning to Follow Instructions via Graphical User Interfaces. *arXiv preprint arXiv:2306.00245* (2023).
- [69] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face. *arXiv:2303.17580 [cs.CL]*
- [70] Noah Shinn, Federico Cassano, Beck Labash, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language Agents with Verbal Reinforcement Learning (2023). *arXiv preprint cs.AI/2303.11366* (2023).
- [71] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975* (2023).
- [72] UiPath. 2022. UiPath Certified RPA Associate v1.0 - EXAM Description.pdf. <https://start.uipath.com/rs/995-XLT-886/images/UiPath%20Certified%20RPA%20Associate%20v1.0%20-%20EXAM%20Description.pdf>
- [73] Wil MP Van der Aalst. 2014. Process mining in the large: a tutorial. *Business Intelligence: Third European Summer School, eBISS 2013, Dagstuhl Castle, Germany, July 7-12, 2013, Tutorial Lectures 3* (2014), 33–76.
- [74] Maxim Vidgof, Stefan Bachhofner, and Jan Mendling. 2023. Large Language Models for Business Process Management: Opportunities and Challenges. *arXiv preprint arXiv:2304.04309* (2023).
- [75] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandolekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291* (2023).
- [76] Weihang Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. 2023. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079* (2023).
- [77] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinning Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, et al. 2023. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. *arXiv preprint arXiv:2311.05997* (2023).
- [78] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems* 35 (2022), 24824–24837.
- [79] Judith Wewerka and Manfred Reichert. 2020. Robotic Process Automation—A Systematic Literature Review and Assessment Framework. *arXiv preprint arXiv:2012.11951* (2020).
- [80] Jason Wu, Siyan Wang, Siman Shen, Yi-Hao Peng, Jeffrey Nichols, and Jeffrey P Bigham. 2023. WebUI: A Dataset for Enhancing Visual UI Understanding with Web Semantics. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–14.
- [81] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. 2023. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155* (2023).
- [82] Zhiyong Wu, Chengcheng Han, Zichen Ding, Zhenmin Weng, Zhoumianze Liu, Shunyu Yao, Tao Yu, and Lingpeng Kong. 2024. OS-Copilot: Towards Generalist Computer Agents with Self-Improvement. *arXiv preprint arXiv:2402.07456* (2024).
- [83] An Yan, Zhengyuan Yang, Wanrong Zhu, Kevin Lin, Linjie Li, Jianfeng Wang, Jianwei Yang, Yiwu Zhong, Julian McAuley, Jianfeng Gao, et al. 2023. Gpt-4v in wonderland: Large multimodal models for zero-shot smartphone gui navigation. *arXiv preprint arXiv:2311.07562* (2023).
- [84] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. 2023. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441* (2023).
- [85] Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. 2023. AppAgent: Multimodal Agents as Smartphone Users. *arXiv preprint arXiv:2312.13771* (2023).
- [86] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629* (2022).
- [87] Yining Ye, Xin Cong, Shizuo Tian, Jiannan Cao, Hao Wang, Yujia Qin, Yaxi Lu, Heyang Yu, Huadong Wang, Yankai Lin, et al. 2023. ProAgent: From Robotic Process Automation to Agentic Process Automation. *arXiv preprint arXiv:2311.10751* (2023).
- [88] Liangzhao Zeng, Boualem Benatallah, Phuong Nguyen, and Anne HH Ngu. 2001. Agflow: Agent-based cross-enterprise workflow management system. In *VLDB*. 697–698.
- [89] Chaoyun Zhang, Liqun Li, Shilin He, Xu Zhang, Bo Qiao, Si Qin, Minghua Ma, Yu Kang, Qingwei Lin, Saravan Rajmohan, et al. 2024. UFO: A UI-Focused Agent for Windows OS Interaction. *arXiv preprint arXiv:2402.07939* (2024).
- [90] Jingyi Zhang, Jiaying Huang, Sheng Jin, and Shijian Lu. 2023. Vision-Language Models for Vision Tasks: A Survey. *arXiv:2304.00685 [cs.CV]*
- [91] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024. GPT-4V(ision) is a Generalist Web Agent, if Grounded. *arXiv:2401.01614 [cs.IR]*
- [92] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, et al. 2023. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854* (2023).