From Perception to Reasoning: Enhancing Vision-Language Models for Mobile UI Understanding

Settaluri Lakshmi Sravanthi[*†], **Ankit Mishra**[*†], **Debjyoti Mondal**[‡],
Subhadarshi Panda[‡], **Rituraj Singh**[‡], **Pushpak Bhattacharyya**[†]
†Indian Institute of Technology Bombay, Mumbai, India
‡Samsung Research Institute, Bangalore
{sravanthi, ankitmishra}@cse.iitb.ac.in,
{d.mondal, subha.darshi, rituraj.s}@samsung.com
pb@cse.iitb.ac.in

Abstract

Accurately grounding visual and textual elements within mobile user interfaces (UIs) remains a significant challenge for Vision-Language Models (VLMs). Visual grounding, a critical task in this domain, involves identifying the most relevant UI element or region based on a natural language query—a process that requires both precise perception and context-aware reasoning. In this work, we present - **MoUI**, a light-weight mobile UI understanding model trained on **MoIT**, an instruction-tuning dataset specifically tailored for mobile screen understanding and grounding, designed to bridge the gap between user intent and visual semantics. Complementing this dataset, we also present a human-annotated reasoning benchmark **MoIQ** that rigorously evaluates complex inference capabilities over mobile UIs. To harness these resources effectively, we propose a two-stage training approach that separately addresses perception and reasoning tasks, leading to stronger perception capabilities and improvement in reasoning abilities. Through extensive experiments, we demonstrate that our **MoUI** models achieve significant gains in accuracy across all perception tasks and *state-of-the-art* results on public reasoning benchmark ComplexQA (78%) and our MoIQ (49%). We will be open-sourcing our dataset, code, and models to foster further research and innovation in the field. Code and data are available in the repo [1]

1 Introduction

Mobile user interfaces (UIs) are structured systems of visual and textual elements that facilitate digital interactions such as navigation, communication, and data retrieval. Automating the perception and interaction within these interfaces has the potential to significantly simplify how users achieve their goals (Edwards et al., 1995).

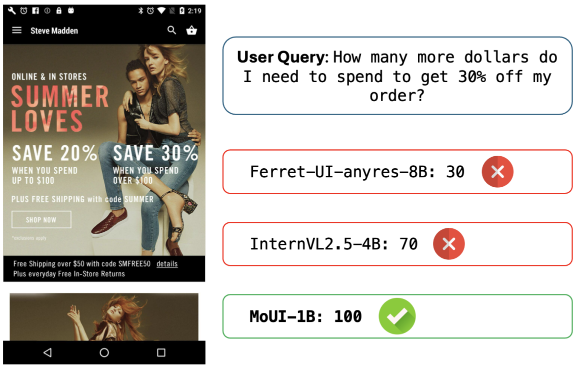

Steve Madden
ONLINE & IN STORES
SUMMER LOVES
SAVE 20% WHEN YOU SPEND UP TO $100
SAVE 30% WHEN YOU SPEND OVER $100
PLUS FREE SHIPPING with code SUMMER
SHOP NOW
Free Shipping over $50 with code SMFREE50 details
Plus everyday Free In-Store Returns
User Query: How many more dollars do I need to spend to get 30% off my order?
Ferret-UI-anyres-8B: 30
InternVL2.5-4B: 70
MoUI-1B: 100

Figure 1: Comparison of **MoUI** between other VLMs. Both Ferret-UI-anyres (8B) and InternVL2.5-4B answer wrongly, while MoUI model generates the correct output. This is an example from ComplexQA

However, the hierarchical layouts, design complexities, and functional dependencies of UIs present unique challenges that differ from those encountered in natural images.

Vision-Language Models (VLMs) (Zhang et al., 2024a; Dai et al., 2023; Chen et al., 2024c; He et al., 2024), while highly effective in tasks such as image captioning and cross-modal retrieval, often struggle with mobile UI understanding due to their training on datasets predominantly composed of natural images. These models typically lack the ability to handle the context-aware interactions and specialized semantics inherent to UIs. One of the central challenges in this domain is visual grounding, which involves aligning textual queries with corresponding UI elements—a task further complicated by limited structured UI datasets and the need to infer functional relationships between components. Consequently, traditional VLMs fall short in capturing the nuanced semantics of interactive interfaces, underscoring the need for domain-specific solutions. Recent research has made progress in tackling these challenges, with several models showing promise. For example, ScreenAI (Baechler et al.,

[*] Equal Contributions
[1] Code and Datasets

Findings of the Association for Computational Linguistics: ACL 2025, pages 25250–25269
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

2024) employs a flexible patching strategy and diverse datasets to enhance its understanding of UIs and infographics.

Similarly, Ferret-UI (You et al., 2023; Zhang et al., 2024b) is a multimodal large language model tailored for mobile UI screens, incorporating advanced capabilities for referring, grounding, and reasoning. Another noteworthy approach, SeeClick (Cheng et al., 2024), focuses on GUI grounding to develop intuitive visual GUI agents capable of effectively interacting with screen elements. Despite these advancements, most datasets and models in this domain remain inaccessible or constrained by limited availability. Furthermore, many existing models are computationally large, making them unsuitable for deployment as UI assistants on edge devices. To address these limitations, we propose Mobile UI Instruct (MoUI-IT), an instruction-tuning dataset specifically designed for mobile screen grounding tasks. MoUI-IT provides detailed examples that align natural language queries with UI elements, enabling models to better understand user intent within structured environments. In addition to this dataset, we introduce a human-written reasoning test set aimed at evaluating VLMs' ability to perform complex inferential tasks over mobile UIs. This benchmark serves as a critical resource for assessing higher-order reasoning capabilities in this domain. To fully leverage these resources, we propose a two-stage training framework that decouples perception from reasoning tasks. The first stage focuses on developing robust perception skills for accurately detecting UI elements, while the second stage enhances the model's ability to perform higher-order reasoning over these detected elements. This structured approach ensures that perception and reasoning are optimized independently yet cohesively. Using this framework, our **1-billion-parameter model** achieves state-of-the-art results on publicly available mobile UI reasoning benchmarks, demonstrating its effectiveness in advancing mobile UI understanding. Our contributions are:

- **MoUI**: A series of lightweight models-1B, 2B, and 4B-designed for complex reasoning tasks on mobile UI screens, achieving state-of-the-art performance on public reasoning benchmarks (Section 6).

- **MoIT**: A 150k instruction-tuning dataset tailored for mobile UI grounding, enhancing the alignment between user queries and UI elements (Section 3.1).

- **MoIQ**: A 3k human-written reasoning evaluation benchmark to assess the complex inference capabilities of VLMs over mobile UIs (Section 3.2).

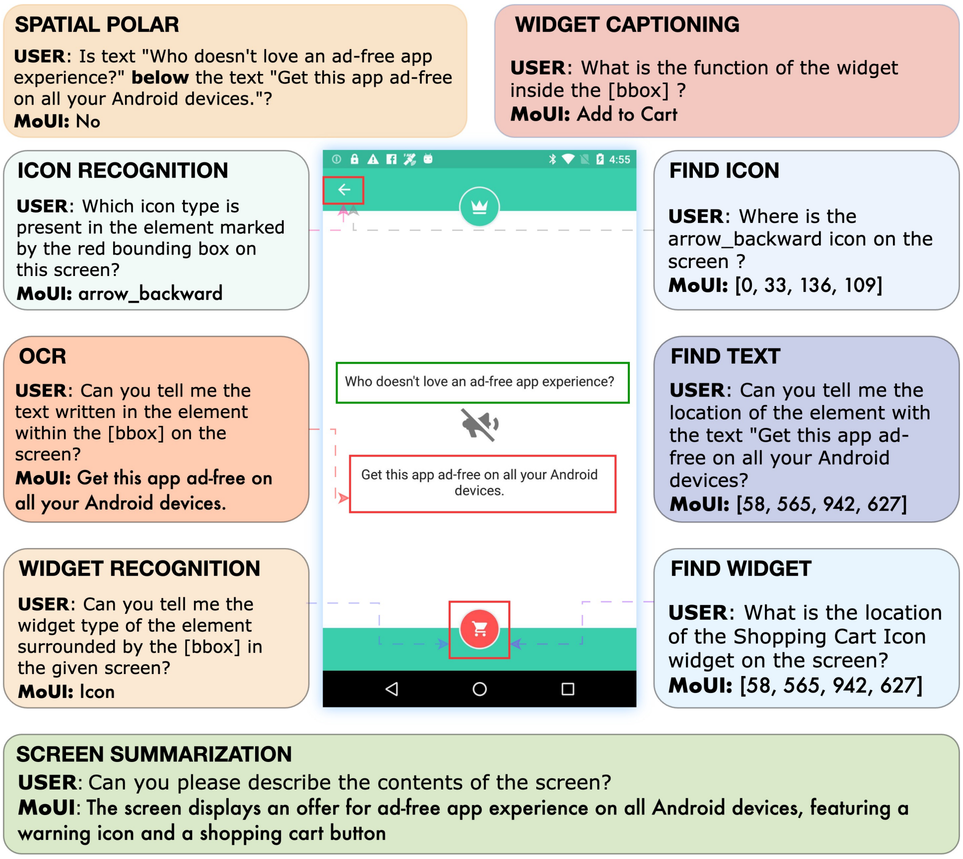
- A two-stage training pipeline that decouples the learning of perception and reasoning tasks, leading to more effective model adaptation. Comprehensive experiments demonstrate that MoUI models achieve *state-of-the-art* results on public reasoning benchmark ComplexQA (78%) and MoIQ (49%) (Section 6).

By open-sourcing our dataset, code, and models, we aim to drive further research in mobile UI understanding and enhance the capabilities of VLMs in digital mobile environments.

2 Related Work

Perception with VLMs Recent advances in integrating vision encoders with LLMs have enhanced their reasoning in vision-language tasks (Alayrac et al., 2022; Liu et al., 2023; Zhu et al., 2023; Dai et al., 2023; Chen et al., 2024c; Zhang et al., 2024a; Zhou et al., 2024). However, VLMs struggle with geometric and numerical interpretation, requiring pixel-wise analysis (Li et al., 2024a). To address this, some approaches incorporate bounding-box regression via quantized coordinates (Chen et al., 2021; Peng et al., 2023; You et al., 2023; Zhang et al., 2024b; Wang et al., 2023; Zang et al., 2024), while others use auxiliary perception modules to enhance visual understanding (Zhang et al., 2024a; Wu et al., 2024a; Pi et al., 2024).

Mobile UI Datasets and Grounding on GUIs Recent efforts have focused on developing benchmarks for assessing grounding and interpretative capabilities in mobile UI screenshots.The RICO dataset (Deka et al., 2017a), a foundational resource, contains over 66k annotated screens and has inspired various expansions. Building on this foundation, later studies such as RICO Semantics (Sunkara et al., 2022), MoTIF (Burns et al., 2022), GUI-WORLD (Chen et al., 2024a), and Mobile-Views (Gao et al., 2024a) have broadened dataset types and coverage, further advancing research in GUI agents. Initial works (Bai et al., 2021; He et al., 2021) on adapting transformer-based models



SPATIAL POLAR

USER: Is text "Who doesn't love an ad-free app experience?" **below** the text "Get this app ad-free on all your Android devices."?
MoUI: No

WIDGET CAPTIONING

USER: What is the function of the widget inside the [bbox] ?
MoUI: Add to Cart

ICON RECOGNITION

USER: Which icon type is present in the element marked by the red bounding box on this screen?
MoUI: arrow_backward

FIND ICON

USER: Where is the arrow_backward icon on the screen ?
MoUI: [0, 33, 136, 109]

OCR

USER: Can you tell me the text written in the element within the [bbox] on the screen?
MoUI: Get this app ad-free on all your Android devices.

FIND TEXT

USER: Can you tell me the location of the element with the text "Get this app ad-free on all your Android devices?
MoUI: [58, 565, 942, 627]

WIDGET RECOGNITION

USER: Can you tell me the widget type of the element surrounded by the [bbox] in the given screen?
MoUI: Icon

FIND WIDGET

USER: What is the location of the Shopping Cart Icon widget on the screen?
MoUI: [58, 565, 942, 627]

SCREEN SUMMARIZATION

USER: Can you please describe the contents of the screen?
MoUI: The screen displays an offer for ad-free app experience on all Android devices, featuring a warning icon and a shopping cart button

4:55
Who doesn't love an ad-free app experience?
Get this app ad-free on all your Android devices.

Figure 2: Illustrations of conversational QA pairs in MoIT dataset for all the perception tasks. The data construction process is elaborated in the section 3.1

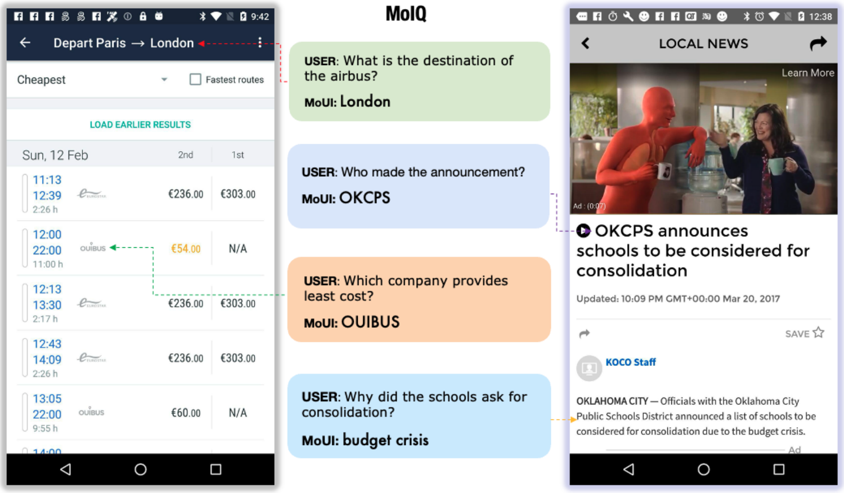
that jointly learn from image and text representations of UIs. Studies such as Gou et al. (2024); Gao et al. (2024b); Li et al. (2025) introduce text instructions, while Cheng et al. (2024) extends the dataset beyond mobile-specific use cases. Others (Zhang et al., 2024c; Rawles et al., 2023; Li et al., 2020a) focus on multi-step tasks, and Lu et al. (2024) explores cross-app navigation. However, existing datasets lack structural details. Our work addresses this gap by integrating view hierarchies for single-screen queries across diverse tasks. Recent VLMs enhance UI understanding, such as Cheng et al. (2024) predicting actions and Baechler et al. (2024) incorporating annotations and QA tasks. You et al. (2024); Li et al. (2024b) apply LLMs for reasoning and grounding, though their training datasets remain unavailable. Other methods improve perception and representation learning, including Burns et al. (2024) that introduces a new pretraining objective, which trains the model to generate a description of a future screen image based on an action performed in the current visual state, Jiang et al. (2024)'s Universal Proposal Network and Wu et al. (2024b)'s multistage pre-training for Chinese UIs. Our approach advances these works with a two-stage framework that separates perception and reasoning while maintaining a unified architecture.

3 Dataset Construction and Tasks

To train a model with a comprehensive understanding of mobile user interfaces, we define a set of UI tasks designed to enhance both its perception and reasoning abilities. Perception tasks help the model develop a holistic understanding of UI design and its components, while advanced reasoning tasks assess its ability to interpret relationships between multiple elements on the screen. In the following sections, we provide an overview of each task category and describe the data collection process.

3.1 Perception Tasks

The primary objective of perception tasks is to help the model develop a deeper understanding of various UI elements, their functions, and their positions on the screen.



Sun, 12 Feb		2nd	1st
11:13 12:39 2:26 h		€236.00	€303.00
12:00 22:00 11:00 h	OUIBUS	€54.00	N/A
12:13 13:30 2:17 h		€236.00	€303.00
12:43 14:09 2:26 h		€236.00	€303.00
13:05 22:00 9:55 h	OUIBUS	€60.00	N/A

USER: What is the destination of the airbus?
MoUI: London

USER: Who made the announcement?
MoUI: OKCPS

USER: Which company provides least cost?
MoUI: OUIBUS

USER: Why did the schools ask for consolidation?
MoUI: budget crisis

OKCPS announces schools to be considered for consolidation

Updated: 10:09 PM GMT+00:00 Mar 20, 2017

KOCO Staff

OKLAHOMA CITY — Officials with the Oklahoma City Public Schools District announced a list of schools to be considered for consolidation due to the budget crisis.

Figure 3: Examples from the MoIQ dataset illustrating reasoning tasks. The questions are designed in such a way that the model needs good perception, language comprehension and reasoning abilities to answer correctly

3.1.1 Elementary Tasks

We incorporate standard elementary tasks in UI domain (You et al., 2023; Li et al., 2024b) to evaluate the model's referring and grounding capabilities. Using the **view hierarchy** data associated with each image in the RICO dataset (Deka et al., 2017b) we parse information like text, icon and widget type along with their bounding box location and generate template based conversational QA pairs for each of the below tasks. For each task base template, we generate 10 variants of them for a diverse set of questions.

Referring tasks enable the model for precise identification and classification of UI elements. We define tasks for text, icon and widget extraction/-classification under this category. The **OCR** task entails asking the model to extract the text inside an element surrounded by a bounding box on the screen, **Icon Recognition** task expects the model to predict the type of icon enclosed within a specified bounding box on the screen and the **Widget Recognition** task asks the model to classify the type of widget located within the given bounding box on the screen.

The **Grounding tasks** are a reverse formulations of the referring tasks in which the model is expected to generate the bounding box coordinates of the target element like text, icon and widget. Formally, we define these tasks as **Find Text** where the model needs to identify and locate the UI element that contains a given text string, **Find Icon** task in which the model determines the position of an icon of a specified type and **Find Widget** task to locate a widget of a specified type on the screen.

For each task, we generate QA pairs for the referring task, then reverse input and output labels to create grounding QA pairs, helping the model associate element type and location.

In addition to the above elementary tasks, we also introduce a new **Spatial** task **Spatial Polar** which evaluate the model's ability to reason about the spatial relationships between UI elements which is essential for screen navigation and interaction for agents. Given the 2-dimensional nature of UI screens, we restrict ourselves to spatial relations like "left", "right", "above" and "below". In Spatial Polar task, given two elements with their bounding boxes and a specified spatial relationship, the model must determine whether the relationship holds with a Yes/No. We leverage the Screen Annotation dataset for generating template based QA pairs on the above tasks.

3.1.2 Spotlight Tasks

We consider the following tasks defined in Spotlight (Li and Li, 2023, 2022) for improving the model's UI comprehension and contextual understanding. **Screen summarization** requires the model to summarize the contents of the screen in a concise manner. In **Widget Captioning** task, given a bounding box around any element on the screen, the model is tasked to generate a brief summary about the functionality of the element.

To train on these tasks, we leverage the Screen2Words (Wang et al., 2021) and Widget Captioning (Li et al., 2020b) datasets, which provide annotated images from the RICO dataset. We convert these annotations into conversational QA pairs in an instruction-tuning format. Specifically, we employ the open-sourced InternVL2-40B model to generate structured QA conversations between a user and a digital assistant using the existing annotations as guidance. This guided approach enhances the quality of QA pairs while minimizing hallucinations. The generated dataset is human verified and filtered to ensure quality. Figure 2 provides the complete detail of all the perception tasks along with the examples based on the UI screen. Appendix F gives a statistical overview of the MoIT dataset generated.

Dataset Name	Training Samples
Stage I (Perception Phase)	
Screen Summarization	8,000
Widget Captioning	15,480
OCR	15,032
Icon Recognition	15,135
Widget Recognition	15,421
Find Text	15,032
Find Icon	15,135
Find Widget	15,421
Spatial Polar	15,407
Stage II (Reasoning Phase)	
Complex ScreenQA	6,347
ScreenQA Short	10,000
Total	146,410

Table 1: Training Mixture Statistics. Screen Question-Answering datasets are generated using VLMs, Elementary task datasets are generated using template based methods from View Hierarchies and we also use a subset of the publicly available ScreenQA datasets

3.2 Advanced Reasoning Tasks

For the advanced reasoning tasks, we use the publicly available mobile UI reasoning datasets, Complex ScreenQA and ScreenQA Short introduced in ScreenAI. We use the training split of 6K QA pairs from the Complex ScreenQA dataset and select a subset of 10K QA pairs from the 80K ScreenQA Short dataset to maintain a balanced training mix.

We also introduce a new reasoning benchmark **MoIQ** for mobile UI screens, consisting of *3k* human-annotated complex reasoning questions spanning **885** unique mobile screens from the RICO dataset with most answers consisting of 2-3 words. Annotators are instructed to create questions that require engaging with multiple elements on the screen and involve at least one step of reasoning to arrive at the answer. The questions evaluate the models on the functionality, positions, color and the spatial relationships between the various UI elements on the screen. Annotation details for MoIQ are described in Appendix A. Unlike Complex ScreenQA, which is model-generated, our benchmark is entirely human-annotated, ensuring higher-quality and more challenging questions. figure 3 shows a few examples from our MoIQ benchmark.

4 Training Strategy

To effectively address the challenges of mobile UI grounding and reasoning, we adopt a two-stage training strategy as shown in the figure 4. This approach ensures that the model first learns strong perceptual representations before transitioning to reasoning-intensive tasks, enabling a more structured learning process.

Stage I: Grounding, Referring, and Spatial Understanding

The first stage focuses on enhancing the model's perceptual understanding of mobile UI screens by training it to associate visual elements with textual descriptions accurately. This phase involves three key tasks: **Grounding:** The model learns to **locate** the most relevant UI element based on a natural language query. **Referring:** The model resolves references to specific UI elements, especially when multiple similar components exist on the screen. **Spatial Understanding:** The model is trained to interpret spatial relationships between UI elements, such as proximity, alignment, and hierarchical structure.

To achieve this, we adopt an instruction-tuning approach, where the model is exposed to a diverse set of tasks requiring it to *ground*, *refer,* and *spatially reason* about UI elements based on textual queries. The model is trained using Next Token Prediction (NTP) loss, ensuring it learns to generate accurate responses by integrating both visual and textual information. Given a mobile UI image I and an instruction q, the model generates a sequence of tokens $Y = (y_1, y_2, \ldots, y_T)$. The training objective is formulated as:

$$\mathcal{L}_{\text{NTP}} = -\sum_{t=1}^{T} \log p(y_t | I, q, y_{<t})$$

where y_t is the predicted token at timestep t, I is the input UI image, q is the textual query or instruction,

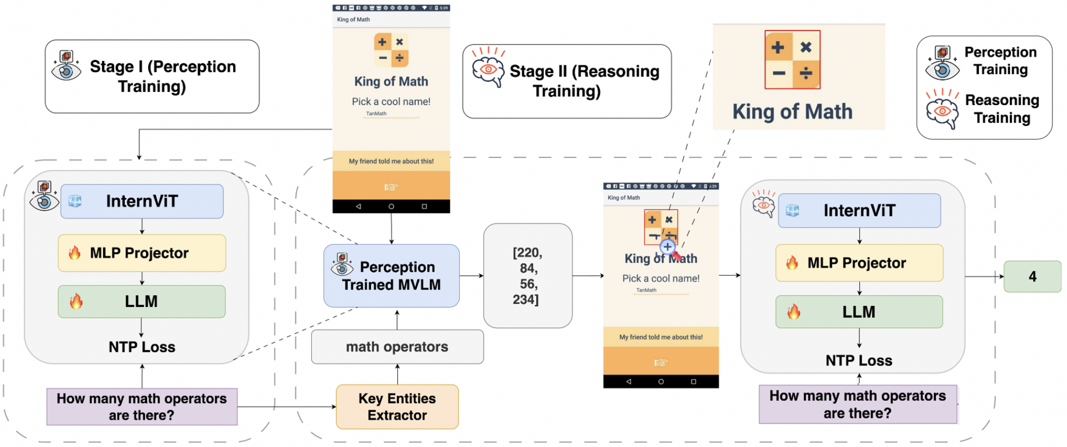


Figure 4: Overview of our two-stage training. In *Stage-I* training the model learns to refer to and ground UI elements, comprehend screens, and infer widget functionalities. For the *Stage-II* training, the key entities are extracted from question and the corresponding *bounding boxes* are annotated on the image. We then train the perception aware model with these annotated images on Complex ScreenQA and ScreenQA Short datasets for enhancing reasoning abilities of the model. This two-stage training pipeline equips MoUI models with strong perception skills, thereby enhancing their reasoning capabilities.

$y_{<t}$ represents previously generated tokens. This stage enables the model to develop a structured representation of mobile UIs, aligning natural language instructions with their corresponding visual components effectively.

Stage II: Reasoning through perception guidance

Once the model has developed a strong perceptual foundation, the second stage fine-tunes the model for reasoning-intensive tasks that demand complex inference over UI elements. The process begins by extracting key entities from user queries using an entity recognition function:$\mathcal{E} = g(q) = \{e_1, e_2, \ldots, e_n\}$, where $g(q)$ identifies relevant entities e_i from the query q. Next, the perception model from **Stage I** retrieves the corresponding bounding box coordinates b_i for each entity: $b_i = f_{\text{perception}}(I, e_i)$, where $f_{\text{perception}}$ is the model's learned function for grounding entities in the UI image I. Instead of masking irrelevant regions, the bounding boxes are superimposed onto the original image to explicitly highlight the relevant UI components: $I' = \text{Superimpose}(I, \{b_i\}_{i=1}^{n})$, where *Superimpose* denotes a function that overlays the bounding boxes b_i onto the original image I while preserving the rest of the content. Rather than extracting each region separately and passing them to the model, we superimpose them to ensure the model perceives the context and relationships between key region elements on the screen. The grounded images I' are then paired with text prompts and used to further fine-tune the perception model. The training continues with Next Token Prediction (NTP) loss, ensuring that the model generates reasoning-aware responses by integrating perceptual understanding with semantic inference:

$$\mathcal{L}_{\text{reasoning}} = -\sum_{t} \log p(y_t | I', q, y_{<t})$$

where y_t is the predicted token at timestep t, conditioned on the grounded image I', query q, and previously generated tokens $y_{<t}$. This stage emphasizes reasoning over extracted entities and their relationships within the UI context, refining the model's ability to answer complex UI-based queries.

5 Experiments

5.1 Training Configurations

We build upon existing Internvl architecture with pre-trained InternViT-300M-448px (Chen et al., 2024c) as vision encoder, Qwen 2.5 (Yang et al., 2024) and IntenLM 2.5 (Cai et al., 2024) language models and a 2-layer MLP projection layer. Unlike existing opens source VLMs, Internvit employs a dynamic high-resolution strategy that segments images into 448×448 tiles which is essential accommodating diverse image resolutions. For Stage I training we use the perception data and for Stage II training we use Complex QA and Short QA from the training data mix 1. In both the stages, we freeze the vision encoder and train both the projection layer and LLM. We train all our MoUI for one epoch in both the stages with InternVL 2.5 (Chen et al., 2024b) as the base model. We set the training batch size to 128, with a learning rate of $4e-5$ and a warmup ratio of 0.03. Training for both stages takes 1b, 2b, 4b models take 60, 80, 160 minutes each on 4-H100 GPUs. Instructions and prompts used for each task in the training are given in the sections D and C.

5.2 Evaluation

To tackle the deployment challenges that can arise on edge devices, we use *LMDeploy* (Contributors, 2023) for efficient inference. To quantify its impact on inference latency, we evaluated the model's performance on the ComplexQA test set with and without LMDeploy. With LMDeploy, MoUI-4B processes all 759 samples in approximately 193 seconds, whereas without it, the same task takes around 573 seconds. To ensure reproducibility and obtain more reliable results, all evaluations are conducted using greedy decoding. Scores for reasoning tasks on Ferret-UI model are obtained using the evaluation script provided in their repository. For the perception tasks, we report the the maximum of the average scores achieved on the elementary tasks as mentioned in the Ferret-UI paper and the scores achieved on our test set. This is done for fair benchmarking as their test set is not publicly available and the Ferret-UI model might not be familiar with our icon/widgets. In any case we have considered the maximum score obtained on these tasks by Ferret-UI model. Results on ScreenAI are reported from the paper (Baechler et al., 2024) as their model is not released publicly. BLIP-2-Finetuned results are reported from the paper (Burns et al., 2022) where the model is finetuned on each task separately (i.e. there are two seperate models for Screen2Words and Widget Captioning). The prompts for each task are detailed in Appendix C.

6 Results and Analysis

We compare the MoUI models against the latest InternVL 2.5 series, the Ferret UI-anyres model, and the ScreenAI model. These selections are based on each model's unique strengths: among the open-source VLMs; InternVL 2.5 models have excelled in grounding and referring tasks, while the Ferret UI-anyres and ScreenAI models are specifically optimized for mobile UI understanding.

We present the detailed results for the perception and reasoning tasks in Tables 2 and 3, respectively.

6.1 Results on Perception Tasks

Our elementary tasks (3.1.1) are designed to assess the models' capabilities in referring, grounding, and spatial reasoning. As shown in Table 2, MoUI consistently improves upon all baseline scores across perception tasks. For OCR-related tasks (OCR and Find Text), our two-stage training strategy yields a substantial performance gain of over 26% compared to baselines. Additionally, MoUI surpasses the Ferret-UI-anyres model by 10% on these tasks. For icon and widget grounding, baseline models perform poorly due to their pre-training mixtures, which lack mobile UI datasets. However, after training on MoIT, a dataset specifically designed for mobile UI understanding, we observe significant improvements of 36.5% and 47% for icon and widget-related tasks, respectively, compared to baseline scores. Moreover, MoUI outperforms Ferret-UI-anyres by ≈10% on these tasks, further demonstrating its effectiveness in mobile UI perception.

Spotlight tasks, such as Screen2Words and Widget Captioning, require a deeper understanding of the various components and the functions on a mobile screen. After training, our models exhibit significantly improved screen comprehension, with an increase of approximately 30 CIDEr points in screen summarization and 75 CIDEr points in widget captioning. Notably, the MoUI-1B model achieves competitive performance on these tasks, closely matching the results of the much bigger FerretUI-anyres-8B and ScreenAI-5B models. We further observe a considerable improvement in spatial understanding task, with our models achieving around 98% accuracy on the spatial polar task, outperforming the baseline across all models.

As shown in Table 10, MoUI models are trained on substantially less data compared to ScreenAI and FerretUI, yet they outperform on six perception tasks and achieve competitive results on the remaining three. We attribute this performance to the high quality of our training data, the majority of which is generated from view hierarchies, ensuring minimal errors.

6.2 Results on Reasoning Tasks

The reasoning tasks 3.2, including Complex QA, Short QA, and MoIQ, evaluate the ability of models to perform complex inference over mobile UIs, requiring multi-step reasoning and contextual un-

Model Size	Screen2 Words	Widget Capt.	OCR	Icon Recog.	Widget Recog.	Find Text	Find Icon	Find Widget	Spatial Polar
InternVL2.5									
1B	79.68	76.32	43.13	23.56	15.44	76.54	64.18	53.21	49.34
2B	75.06	69.31	52.21	37.31	29.11	79.18	71.07	49.56	58.10
4B	85.56	83.00	77.09	57.12	32.45	84.23	73.31	63.42	72.89
Ferret-UI-anyres									
8B	115.6	140.3	82.4	82.4	**82.4**	76.5	**76.5**	76.5	42.50
ScreenAI									
5B	120.8	**156.4**	—	—	—	—	—	—	—
BLIP-2 Finetuned									
1.2B	**127.9**	128.0	—	—	—	—	—	—	—
MoUI (ours)									
1B	108.48	147.76	89.25	86.37	80.81	86.32	73.64	82.01	98.44
2B	100.06	140.56	87.71	87.32	81.95	85.95	73.11	81.07	98.28
4B	123.2	151.25	**90.63**	**87.15**	81.33	**89.14**	75.75	**83.51**	**98.81**

Table 2: Results of MoUI and baseline models on perception tasks. For Screen2Words and Widget Captioning, we report CIDEr scores. Exact match accuracy is used for referential elementary tasks (OCR, Icon Recognition, Widget Recognition) and the spatial polar task. For grounding elementary tasks (Find Text, Icon, and Widget), we report Acc@IoU=0.1 scores. **MoUI** models consistently outperform baseline models, surpassing existing mobile UI understanding models (Ferret-UI-anyres and ScreenAI) on the Screen2Words benchmark while achieving competitive results on other tasks.

derstanding. Across all tasks, MoUI consistently outperforms baseline models like InternVL2.5 and Ferret-UI-anyres, demonstrating the effectiveness of our two-stage training approach. figure 1 shows an example of MoUI-1B model's performance compared to 8x and 4x models.

For ComplexQA, MoUI achieves significant gains, with MoUI-1B scoring 0.66 and MoUI-4B reaching a state-of-the-art score of 0.78, far exceeding ScreenAI-5B at 0.43 and Ferret-UI-anyres-8B at just 0.29. In Short QA, MoUI (1B) and (2B) both achieve 0.86, while MoUI (4B) slightly improves to 0.89, closely rivaling ScreenAI (5B), which scores 0.95, and outperforming InternVL2.5 (4B) at 0.66 and Ferret-UI-anyres (8B) at 0.49. For the challenging MoIQ benchmark, which combines perception with logical inference, MoUI demonstrates steady improvements across model sizes, with both 1B and 2B scoring 0.43 and 4B achieving the highest score of 0.49, compared to InternVL2.5 (4B) at 0.35 and Ferret-UI-anyres (8B) at just 0.29. These results highlight MoUI's superior ability to leverage its enhanced perception capabilities for robust reasoning, achieving state-of-the-art performance in Complex QA and MoIQ while remaining competitive in Short QA against larger models like ScreenAI. Few, illustrations of generated bounding boxes used for stage 2 are provided in Figure K. We can observe that the bounding boxes are accurately generated for all the extracted key entities.

Model Size	Complex ScreenQA	ScreenQA Short	MoIQ
InternVL2.5			
1B	0.42	0.44	0.23
2B	0.49	0.52	0.26
4B	0.65	0.66	0.35
Ferret-UI-anyres			
8B	0.29	0.49	0.29
ScreenAI			
5B	0.43	**0.95**	_
MoUI (1-stage training)			
4B	0.74	0.83	0.42
MoUI (2-stage training)			
1B	0.66	0.86	0.43
2B	0.67	0.86	0.43
4B	**0.78**	0.89	**0.49**

Table 3: Results of MoUI and baseline models on reasoning tasks. We report the SQuAD F1 scores for all three reasoning tasks. **MoUI** outperforms ScreenAI and Ferret-UI-anyres (8B) on the complexQA benchmark and achieves competitive scores on the Short ScreenQA benchmark.

6.3 Qualitative Error Analysis

For OCR-related tasks, the model occasionally confuses text adjacent to the given bounding box with the intended text. In icon-related tasks, errors arise when distinguishing visually similar icon types, such as menu vs. list or *expand_more* vs. *arrow_downward*. Additionally, in some cases, the model incorrectly includes the red bounding box enclosing the icon as part of the icon itself, leading to misclassification. In widget recognition, the model struggles with fine-grained distinctions, often confusing text with text button or image with icon. Furthermore, since many widgets contain illustrations, they are sometimes misinterpreted as images. For reasoning tasks, the model faces challenges when multiple entities are involved, often failing to infer relationships between them and struggles to construct the logical reasoning necessary to answer the question correctly. We have added a few error cases in the Appendix G

6.4 Ablation Studies

Comparison with one stage training
In this setting, instead of training the model in two different stages for perception and reasoning tasks as shown in figure 4, we train the model only once on the entire data. This experiment helps us to establish the importance of perception guidance during reasoning training stage. We observe that in table 3 there is an increase in the accuracy of $\approx 4\%$ on all the three reasoning tasks. We conclude that integrating perception guidance enhances the reasoning capabilities of MoUI. As shown in Appendix, Section 8 we find that Stage II training not only improves the model's reasoning abilities but also enhances certain perception capabilities acquired during Stage I, resulting in overall better performance.

Comparison with other vision encoders
To enhance the perception and reasoning abilities of VLMS in use-cases like Mobile UI understanding, along with high quality instruction tuning data; a strong vision encoder that can accommodate images of any aspect ratio is important. To empirically verify this, we experimented with *TinyLLaVA-3.1B* model with SigLIP (Zhai et al., 2023) as vision encoder and Phi-2 (Li et al., 2023) as language decoder. As shown in Appendix E we can observe that even though there is an increase in the accuracy after finetuning on MoUI data the models are performing below par on perception tasks which results in lower performance in reasoning tasks as well.

7 Conclusion and Future Work

In this work, we introduce a series of lightweight MoUI models, along with the MoIT instruction-tuning dataset and the MoIQ reasoning benchmark. We also propose a two-stage training pipeline that first enhances the model's perception capabilities, and then leverages these improvements to strengthen its reasoning abilities. Our results demonstrate that the careful selection of a suitable vision encoder, high-quality training data, and the proposed training pipeline collectively contribute to achieving state-of-the-art or competitive performance on referring, grounding, and reasoning tasks—all while using smaller models. Despite these advances, the MoIQ benchmark remains challenging, with the current state-of-the-art model reaching only 49% accuracy. This highlights the need for more robust reasoning models in the UI domain. Looking ahead, we aim to close this gap by developing models capable of multi-command grounding, spatial reasoning, and understanding implicit user intent in complex UI contexts. This will involve extending our current instruction tuning paradigm to support multi-turn interactions and compositional tasks, while incorporating richer contextual cues from both visual and semantic modalities. By pushing the boundaries of reasoning in multimodal UI understanding, we hope to enable more intelligent and adaptable user interfaces. To foster continued research in this area, we open-source our datasets, models, and code.

8 Limitations

Current models are designed to process instructions in a single language. Extending them to support multiple languages would significantly enhance their applicability, enabling deployment to a broader, global audience. This multilingual capability would not only improve accessibility for users in non-English-speaking regions but also facilitate cross-lingual knowledge transfer, especially benefiting low-resource languages. Our dataset generation pipeline currently relies on view hierarchies or similar metadata about UI screens to construct meaningful instruction-response pairs. This structural information is crucial for grounding instructions to specific interface elements and understanding the semantic layout of the screen. At present, we use spaCy for entity extraction; however, more advanced models are available that offer improved accuracy, albeit at the cost of higher computational requirements.

9 Acknowledgments

Authors from Indian Institute of Technology, Bombay would like to thank Neysa for providing seamless GPU support.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Gilles Baechler, Srinivas Sunkara, Maria Wang, Fedir Zubach, Hassan Mansoor, Vincent Etter, Victor Cărbune, Jason Lin, Jindong Chen, and Abhanshu Sharma. 2024. Screenai: A vision-language model for ui and infographics understanding. *Preprint*, arXiv:2402.04615.

Chongyang Bai, Xiaoxue Zang, Ying Xu, Srinivas Sunkara, Abhinav Rastogi, Jindong Chen, et al. 2021. Uibert: Learning generic multimodal representations for ui understanding. *arXiv preprint arXiv:2107.13731*.

Andrea Burns, Deniz Arsan, Sanjna Agrawal, Ranjitha Kumar, Kate Saenko, and Bryan A Plummer. 2022. A dataset for interactive vision-language navigation with unknown command feasibility. In *European Conference on Computer Vision*, pages 312–328. Springer.

Andrea Burns, Kate Saenko, and Bryan A Plummer. 2024. Tell me what's next: Textual foresight for generic ui representations. *arXiv preprint arXiv:2406.07822*.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. 2024. Internlm2 technical report. *Preprint*, arXiv:2403.17297.

Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Liuyi Chen, Yilin Bai, Zhigang He, Chenlong Wang, Huichi Zhou, Yiqiang Li, et al. 2024a. Gui-world: A dataset for gui-oriented multimodal llm-based agents. *arXiv e-prints*, pages arXiv–2406.

Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. 2021. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024b. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024c. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. Seeclick: Harnessing gui grounding for advanced visual gui agents. *arXiv preprint arXiv:2401.10935*.

LMDeploy Contributors. 2023. Lmdeploy: A toolkit for compressing, deploying, and serving llm. `https://github.com/InternLM/lmdeploy`.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven CH Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. arxiv abs/2305.06500 (2023).

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017a. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual Symposium on User Interface Software and Technology*, UIST '17.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017b. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th annual ACM symposium on user interface software and technology*, pages 845–854.

W Keith Edwards, Elizabeth D Mynatt, and Kathryn Stockton. 1995. Access to graphical interfaces for blind users. *Interactions*, 2(1):54–67.

Longxi Gao, Li Zhang, Shihe Wang, Shangguang Wang, Yuanchun Li, and Mengwei Xu. 2024a. Mobileviews: A large-scale mobile gui dataset. *arXiv preprint arXiv:2409.14337*.

Longxi Gao, Li Zhang, Shihe Wang, Shangguang Wang, Yuanchun Li, and Mengwei Xu. 2024b. Mobileviews: A large-scale mobile gui dataset. *Preprint*, arXiv:2409.14337.

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. 2024. Navigating the digital world as humans do: Universal visual grounding for gui agents. *arXiv preprint arXiv:2410.05243*.

Muyang He, Yexin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. 2024. Efficient multimodal learning from data-centric perspective. *arXiv preprint arXiv:2402.11530*.

Zecheng He, Srinivas Sunkara, Xiaoxue Zang, Ying Xu, Lijuan Liu, Nevan Wichers, Gabriel Schubiner, Ruby Lee, and Jindong Chen. 2021. Actionbert: Leveraging user actions for semantic understanding of user interfaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5931–5938.

Qing Jiang, Yuqin Yang, Yuda Xiong, Yihao Chen, Zhaoyang Zeng, Tianhe Ren, Lei Zhang, et al. 2024. Chatrex: Taming multimodal llm for joint perception and understanding. *arXiv preprint arXiv:2411.18363*.

Gang Li and Yang Li. 2022. Spotlight: Mobile ui understanding using vision-language models with a focus. *arXiv preprint arXiv:2209.14927*.

Gang Li and Yang Li. 2023. Spotlight: Mobile ui understanding using vision-language models with a focus. In *The Eleventh International Conference on Learning Representations*.

Guanzhen Li, Yuxi Xie, and Min-Yen Kan. 2024a. MVP-bench: Can large vision-language models conduct multi-level visual perception like humans? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13505–13527, Miami, Florida, USA. Association for Computational Linguistics.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. 2025. Screenspot-pro: Gui grounding for professional high-resolution computer use. Preprint.

Yang Li, Jiacong He, Xin Zhou, Yuan Zhang, and Jason Baldridge. 2020a. Mapping natural language instructions to mobile ui action sequences. *arXiv preprint arXiv:2005.03776*.

Yang Li, Gang Li, Luheng He, Jingjie Zheng, Hong Li, and Zhiwei Guan. 2020b. Widget captioning: Generating natural language description for mobile user interface elements. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5495–5510.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need II: phi-1.5 technical report. *CoRR*, abs/2309.05463.

Zhangheng Li, Keen You, Haotian Zhang, Di Feng, Harsh Agrawal, Xiujun Li, Mohana Prasad Sathya Moorthy, Jeff Nichols, Yinfei Yang, and Zhe Gan. 2024b. Ferret-ui 2: Mastering universal user interface understanding across platforms. *arXiv preprint arXiv:2410.18967*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.

Quanfeng Lu, Wenqi Shao, Zitao Liu, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, Yu Qiao, and Ping Luo. 2024. Gui odyssey: A comprehensive dataset for cross-app gui navigation on mobile devices. *arXiv preprint arXiv:2406.08451*.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. 2023. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*.

Renjie Pi, Lewei Yao, Jiahui Gao, Jipeng Zhang, and Tong Zhang. 2024. Perceptiongpt: Effectively fusing visual perception into llm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27124–27133.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. 2023. Androidinthewild: A large-scale dataset for android device control. In *Advances in Neural Information Processing Systems*, volume 36, pages 59708–59728. Curran Associates, Inc.

Srinivas Sunkara, Maria Wang, Lijuan Liu, Gilles Baechler, Yu-Chung Hsiao, Abhanshu Sharma, James Stout, et al. 2022. Towards better semantic understanding of mobile interfaces. *arXiv preprint arXiv:2210.02663*.

Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. 2021. Screen2words: Automatic mobile ui summarization with multimodal learning. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 498–510.

Weihan 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*.

Jiannan Wu, Muyan Zhong, Sen Xing, Zeqiang Lai, Zhaoyang Liu, Wenhai Wang, Zhe Chen, Xizhou Zhu, Lewei Lu, Tong Lu, et al. 2024a. Visionllm v2: An end-to-end generalist multimodal large language model for hundreds of vision-language tasks. *arXiv preprint arXiv:2406.08394*.

Qinzhuo Wu, Weikai Xu, Wei Liu, Tao Tan, Jianfeng Liu, Ang Li, Jian Luan, Bin Wang, and Shuo Shang. 2024b. Mobilevlm: A vision-language model for better intra-and inter-ui understanding. *arXiv preprint arXiv:2409.14818*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. 2023. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*.

Keen You, Haotian Zhang, Eldon Schoop, Floris Weers, Amanda Swearngin, Jeffrey Nichols, Yinfei Yang, and Zhe Gan. 2024. Ferret-ui: Grounded mobile ui understanding with multimodal llms. In *European Conference on Computer Vision*, pages 240–255. Springer.

Yuhang Zang, Wei Li, Jun Han, Kaiyang Zhou, and Chen Change Loy. 2024. Contextual object detection with multimodal large language models. *Preprint*, arXiv:2305.18279.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.

Hao Zhang, Hongyang Li, Feng Li, Tianhe Ren, Xueyan Zou, Shilong Liu, Shijia Huang, Jianfeng Gao, Leizhang, Chunyuan Li, et al. 2024a. Llava-grounding: Grounded visual chat with large multimodal models. In *European Conference on Computer Vision*, pages 19–35. Springer.

Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. 2024b. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*.

Jiwen Zhang, Jihao Wu, Yihua Teng, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. 2024c. Android in the zoo: Chain-of-action-thought for gui agents. *arXiv preprint arXiv:2403.02713*.

Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. 2024. Tinyllava: A framework of small-scale large multimodal models. *arXiv preprint arXiv:2402.14289*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Annotation Details

For MoUI we employed five annotators for writing QAs on MobileUI screens. They are experts in writing reasoning based datasets and have annotated several such datasets before. Three annotators are tasked for writing QA and two annotators are tasked for verification in a double-blinded manner. Each annotator is tasked to write 2-3 questions per screen. Each question will is verified by two other annotators. A question will be considered only if both the verifiers mark it as valid. A total of *4.5k* questions were written out of which *3k* are retained. The annotators were compensated as per the laws of the geographical location.

B Evaluation Metrics

We use the below evaluation metrics for evaluating our models on various tasks.

Task	**Metric Used**
Spotlight Tasks	
Screen Summarization	CIDEr
Widget Captioning	CIDEr
Elementary Tasks	
OCR	Exact Match
Icon Recognition	Exact Match
Widget Recognition	Exact Match
Find Text	Acc@IoU=0.1
Find Icon	Acc@IoU=0.1
Find Widget	Acc@IoU=0.1
Spatial Polar	Exact Match
Reasoning Tasks	
Complex ScreenQA	SQuAD F1
ScreenQA Short	SQuAD F1
MoIQ	SQuAD F1

Table 4: Evaluation Metrics Used

C Prompts for Data Generation and Zero-Shot Inference

C.1 Spotlight Tasks Data Generation

Prompt 1: Screen Summarization Dataset Generation Prompt

Design a question-answer pair where a user asks the assistant to provide a summary of the screen. Keep in mind that the interaction should be between a mobile user and a visual AI assistant. You are given a screenshot of the mobile screen where the user is and a short summary describing the contents of the screen. Use the provided summary as a starting point to describe the screen for the user and keep it short and concise.
Format the output in the following form:
USER: "<question for providing summary>"
AI ASSISTANT: "<generated summary output>"
Provided Summary: "Discussion forum app for anime and K-dramas"

Prompt 2: Widget Captioning Dataset Generation Prompt

Design a question-answer pair where a user asks the assistant to provide a phrase that best describes the functionality of the interactive element [bbox]. Keep in mind that the interaction should be between a mobile user and a visual AI assistant. You are given a screenshot of the mobile screen where the user is with a bbox in red colour around the widget and its location in the form of [x1, y1, x2, y2] normalised to the size of the screen. You will also receive a sample caption that describes the widget enclosed in the bbox at the provided location. Use the provided caption as a starting point to describe the functionality of the widget for the user and keep it short and concise.
Format the output in the following form:
USER: "<question for describing the functionality of the widget at [bbox]>"
AI ASSISTANT: "<generated caption for the functionality of the widget>"
Provided inputs:
bbox location: [0, 320, 840, 189]
caption: "advertisement"

C.2 Ferret-UI-anyres Prompts

Prompt 1: Complex ScreenQA Base Prompt

Answer only the question asked in text and if the query is counting or arithmetic based only output the numerical value as the answer. Here is the question you shall answer: $\{user\_query\}$

Prompt 2: ScreenQA Short Base Prompt

If the question is about counting, only answer in a single number. Here is the question you shall answer: $\{user\_query\}$

Prompt 3: Spatial Polar Task Base Prompt

Please make sure that the answer is only Yes/No. Here is the question you shall answer: $\{user\_query\}$

C.3 Zero-Shot Inference Prompts

Prompt 1: OCR Task Base Prompt

You are an AI visual assistant designed to help users with questions regarding content on a UI screen.
You will be provided with an image of the UI screen with a red bounding box surrounding a region.
Please extract the text inside the bounding box region from the given image.
Only output the text and nothing else.
Here is the question you shall answer: $\{user\_query\}$

Prompt 2: Screen Summarization Task Base Prompt

You are an AI visual assistant designed to help users with questions regarding content on a UI screen.

You will be provided with an image of the UI screen. Your task is to very briefly summarize the mobile screen given. Only output the screen summary and nothing else.

Here is the question you shall answer: $\{user\_query\}$

Prompt 3: Widget Captioning Base Prompt

You are a digital assistant that needs to answer queries from the user about a mobile UI screen. You will be given a query from user and the screen as inputs. Carefully focus on the screen and the red bounding box given. Keep your answers short and concise.

Please only use the following format: ANSWER: <answer> User: $\{user\_query\}$

Prompt 4: ScreenQA Short Base Prompt

Answer user queries about the mobile screen. The answer should be as short as possible. User: $\{user\_query\}$

Prompt 5: Grounding Tasks Base Prompt

Please provide the bounding box coordinates of the region this sentence describes: <ref>$user\_query$</ref>

Prompt 6: ScreenQA Short Base Prompt

You are a digital assistant that needs to answer queries from the user about a mobile UI screen. You will be given a query from user and the screen as inpouts. Carefully focus on the screen and the red bounding box given. Your task is to predict the icon type of the icon given in the red bounding box.

The icon type can only be of the types: $expand\_less$, redo, follow, bluetooth, add, notifications, avatar, edit, $arrow\_backward$, call, lock, font, microphone, menu, globe, $thumbs\_down$, $skip\_previous$, folder, playlist, filter, settings, close, emoji, delete, build, wallpaper, $thumbs\_up$, explore, swap, refresh, star, search, volume, sliders, time, photo, $zoom\_out$, weather, $date\_range$, send, videocam, more, info, $national\_flag$, bookmark, gift, power, reply, launch, email, dialpad, copy, group, $filter\_list$, home, repeat, warning, minus, cart, compare, music, $arrow\_downward$, $arrow\_forward$, visibility, $expand\_more$, book, shop, help, save, dashboard, share, favorite, facebook, pause, list, $location\_crosshair$, location, check, description, $skip\_next$, label, undo, fullscreen, $file\_download$, play, $attach\_file$, navigation, $network\_wifi$, flash, twitter, av_rewind, history, chat, $arrow\_upward$, layers, switcher, $av\_forward$, flight.

Please only answer from the given icon types.

Keep your answers short and concise.

Please only use the following format:

ANSWER: <answer>

User: $\{user\_query\}$

Prompt 7: Widget Recognition Base Prompt

You are a digital assistant that needs to answer queries from the user about a mobile UI screen.
You will be given a query from user and the screen as inputs.
Carefully focus on the screen and the red bounding box given.
Your task is to predict the widget type of the widget given in the red bounding box.
The widget type can only be of the types: Image, Web View, Text, Icon, List Item, Drawer, Video, On/Off Switch, Input, Checkbox, Slider, Background Image, Pager Indicator, Advertisement, Text Button, Card, Radio Button, Map View.
Please only answer from the given widget types and print nothing else other than the widget type. Keep your answers short and concise.
Please only use the following format:
ANSWER: <answer>
User: $\{user\_query\}$

D Instruction Templates

Task	Instruction Template
Screen Summarization	<image>{Question}
Widget Captioning	<image>{Question}
OCR	<image>Given the screenshot image, extract the text in bounding box: {Question}
Icon Recognition	<image>Given the screenshot image, output the icon type for element in bounding box: {Question}
Widget Recognition	<image>Given the screenshot image, output the widget type for element in bounding box: {Question}
Find Text	<image>Please give the bounding box coordinates for the question: {Question}
Find Icon	<image>Please give the bounding box coordinates for the question: {Question}
Find Widget	<image>Please give the bounding box coordinates for the question: {Question}
Spatial Polar	<image>Given the image and two bounding boxes, answer Yes/No for the question: {Question}
Complex ScreenQA	Given the image, answer the following question with no more than five words. {Question}
ScreenQA Short	Based on the image, respond to this question with a short answer: {Question}

Table 5: Instruction Templates Used for MoIT dataset

E TinyLLava-Gemma Results

Setting	Screen2 Words	Widget Capt.	OCR	Icon Recog.	Widget Recog.	Complex QA	Short QA	Spatial Polar
Zero-Shot	34.23	25.66	4.20	0	8.29	34.76	5.26	54.8
Finetuned TinyLLava	**78.9**	**117.32**	21.6	36.03	**41.58**	41.23	4.64	**92.68**

Table 6: TinyLLava-Gemma results for perception tasks before and after training on our MoIT dataset

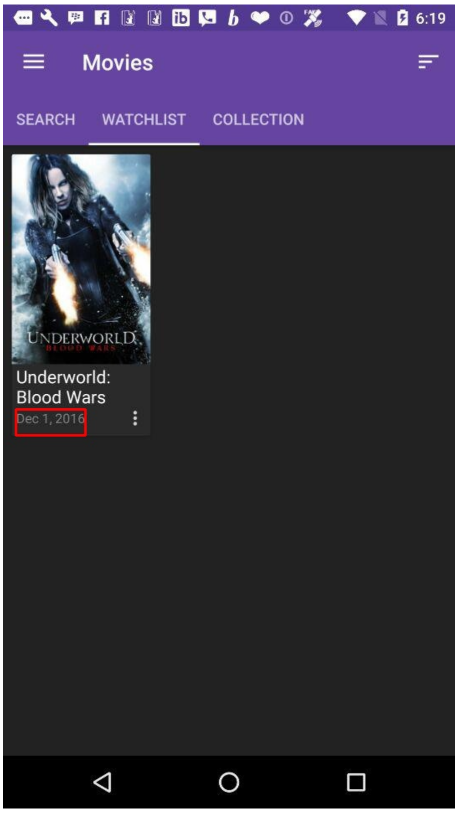
F MoIT Dataset Statistics

UI Task	Method	Count of QA pairs
Screen Summarization	InternVL2	10000
Widget Captioning	InternVL2	19475
Find Icon	Template	16796
Widget Classification	Template	19299
Icon Recognition	Template	16796
Find Widget	Template	19299
OCR	Template	18871
Spatial Polar	Template	19261

Table 7: MoIT dataset statistics

G Error Cases

OCR-Related Error



User Query

What does the text inside the red bounding box on the given screen say?

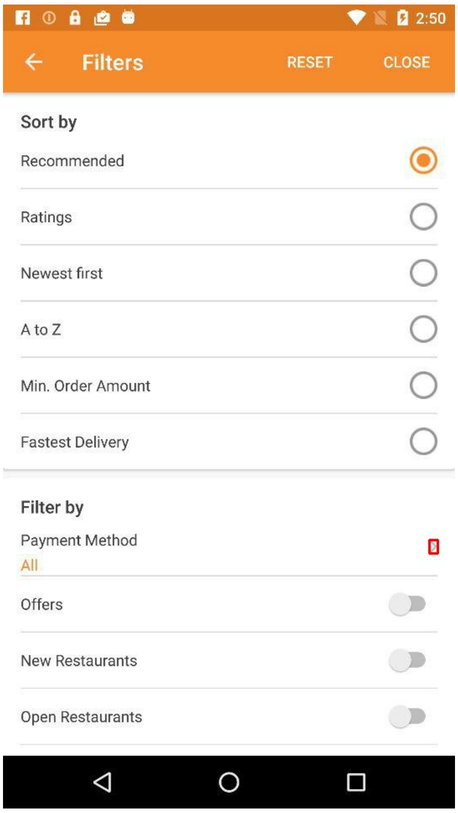
Model Output

Underworld: Blood Wars

Error Analysis

Here, the ground truth answer is "**Dec 1, 2016.**". However, the model misinterprets the adjacent text as the intended text within the bounding box and outputs "Underworld: Blood Wars" instead. We hypothesize that this occurs because the adjacent text is significantly larger, leading the model to prioritize it over the smaller text inside the bounding box.

Icon Recognition Error



User Query

Can you identify the icon type of the element enclosed by the red bounding box in the provided screen?

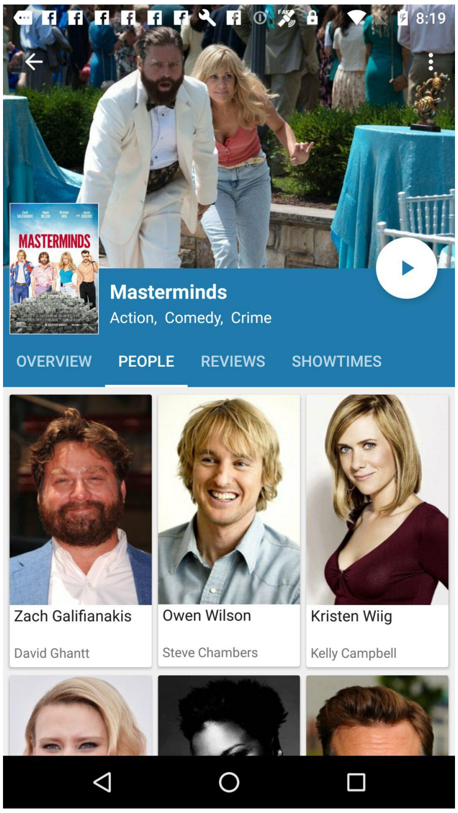
Model Output

play

Error Analysis

Here, the correct answer is "*arrow_forward*". Since the bounding box is tightly wrapped around the icon, the model mistakenly interprets it as part of the icon itself, leading it to incorrectly classify the icon as play.

Reasoning-Based Error



User Query

How many more actors are in the movie's cast than actresses?

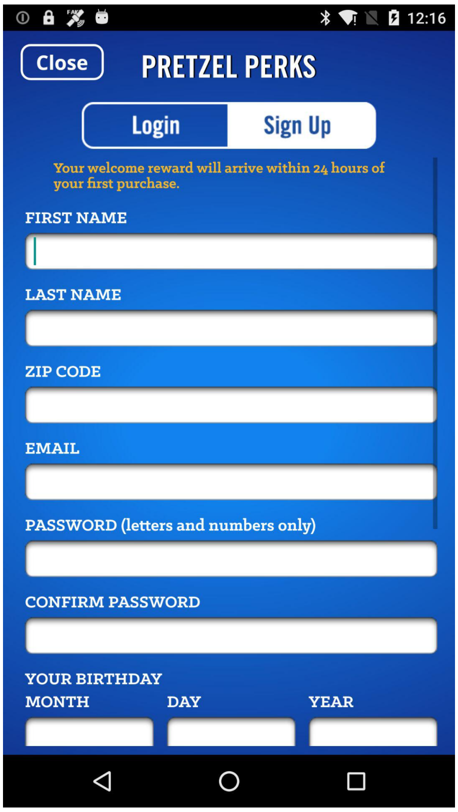
Model Output

2

Error Analysis

Here, the ground truth answer is **1**. However, the model misidentifies the actress count, likely due to incomplete cast images on the screen, resulting in an incorrect output of 2.

Reasoning-Based Error



Your welcome reward will arrive within 24 hours of your first purchase.

FIRST NAME

LAST NAME

ZIP CODE

EMAIL

PASSWORD (letters and numbers only)

CONFIRM PASSWORD

YOUR BIRTHDAY

MONTH DAY YEAR

User Query

Which option is selected: Login or Sign Up?

Model Output

Login

Error Analysis

Since the "Sign Up" button is highlighted, the correct answer is **Sign Up**. But, the model misidentifies the active button as Login which is not highlighted, likely due to varying UI styles across applications.

H Stage I vs Stage II Comparison for Perception Tasks

	Screen2 Words	Widget Capt.	OCR	Icon Recog.	Widget Recog.	Find Text	Find Icon	Find Widget	Spatial Polar
After Stage I	123.15	133.41	90.53	83.33	79.67	89.25	75.96	79.16	98.85
After Stage II	123.2	151.25	90.63	87.15	81.33	89.14	75.75	83.51	98.81

Table 8: Stage I vs Stage II performance comparison on the various perception tasks.

I Performance on the ScreenSpot Benchmark

LVLMs	Model Size	Training Data Size	Mobile (Text)	Mobile (Icon/Widget)
MoUI	1B	150K	53.43%	33.56%
MoUI	4B	150K	71.85%	47.65%
SeeClick	9.6B	1M	78.0%	52.0%

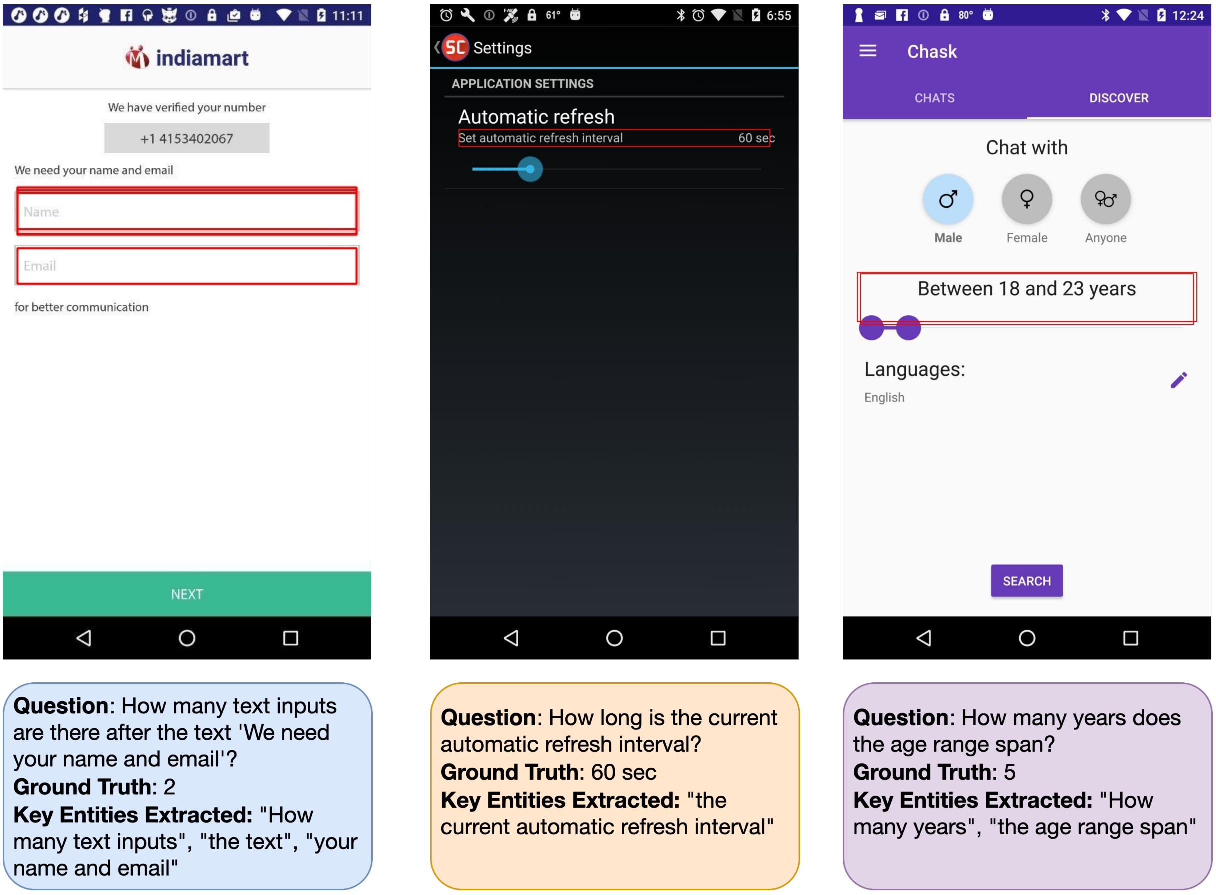
Table 9: Comparison of LVLMs across model size, training data, and mobile performance

J Comparison with other Mobile UI datasets

Data	**Size**	**Task Coverage**			**Dataset Generation Method**	
		Referring	**Grounding**	**Spatial**	**Perception**	**Reasoning**
ScreenAI	>400M	✓	✗	✗	Model generated	Model generated
FerretUI	250K	✓	✓	✗	Model generated	Model generated
MoUI (ours)	150K	✓	✓	✓	Template based generation	Human written

Table 10: Comparison of Mobile UI datasets

K Annotated Bounding Boxes after Stage I Training



Question: How many text inputs are there after the text 'We need your name and email'?
Ground Truth: 2
Key Entities Extracted: "How many text inputs", "the text", "your name and email"

Question: How long is the current automatic refresh interval?
Ground Truth: 60 sec
Key Entities Extracted: "the current automatic refresh interval"

Question: How many years does the age range span?
Ground Truth: 5
Key Entities Extracted: "How many years", "the age range span"

Figure 5: Examples of bounding box annotations of the extracted key entities from the user query by the perception aware model after the Stage I training.