Conversations with GUIs

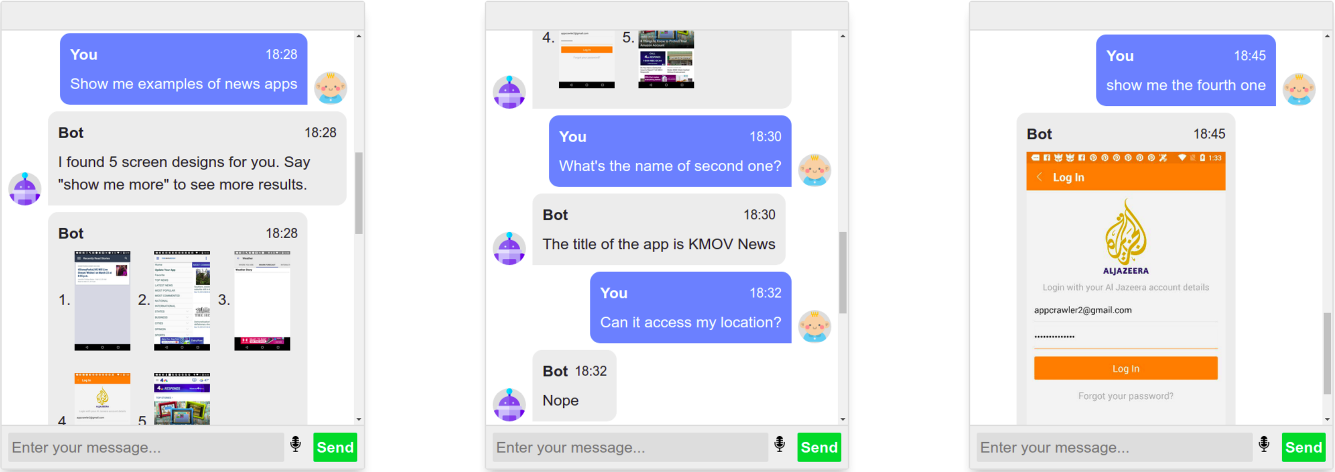
Kashyap Todi*
kashyap.todi@gmail.com
Aalto University
Finland

Luis A. Leiva*
name.surname@uni.lu
University of Luxembourg
Luxembourg

Daniel Buschek*
daniel.buschek@uni-bayreuth.de
Department of Computer Science
University of Bayreuth
Germany

Pin Tian
tianpin313@gmail.com
Aalto University
Finland

Antti Oulasvirta
antti.oulasvirta@aalto.fi
Aalto University
Finland



You 18:28
Show me examples of news apps

Bot 18:28
I found 5 screen designs for you. Say "show me more" to see more results.

Bot 18:28
1. 2. 3.
4 5

4. 5.

You 18:30
What's the name of second one?

Bot 18:30
The title of the app is KMOV News

You 18:32
Can it access my location?

Bot 18:32
Nope

You 18:45
show me the fourth one

Bot 18:45

Figure 1: We introduce the concept of *conversations with GUIs*: Users pose queries to retrieve information from an annotated dataset of GUIs, for example for design inspiration. The information can be textual or graphical, depending on the user's query and intent.

ABSTRACT

Annotated datasets of application GUIs contain a wealth of information that can be used for various purposes, from providing inspiration to designers and implementation details to developers to assisting end-users during daily use. However, users often struggle to formulate their needs in a way that computers can understand reliably. To address this, we study how people may interact with such GUI datasets using natural language. We elicit user needs in a survey ($N = 120$) with three target groups (designers, developers, end-users), providing insights into which capabilities would be useful and how users formulate queries. We contribute a labelled dataset of 1317 user queries, and demonstrate an application of a conversational assistant that interprets these queries and retrieves information from a large-scale GUI dataset. It can (1) suggest GUI screenshots for design ideation, (2) highlight details about particular GUI features for development, and (3) reveal further insights about applications. Our findings can inform design and implementation of intelligent systems to interact with GUI datasets intuitively.

CCS CONCEPTS

• **Human-centered computing → Natural language interfaces**; • **Computing methodologies → Information extraction**.

KEYWORDS

Conversational assistants; Chatbots; GUI; Dataset; NLP; NLU

ACM Reference Format:
Kashyap Todi, Luis A. Leiva, Daniel Buschek, Pin Tian, and Antti Oulasvirta. 2021. Conversations with GUIs. In *Designing Interactive Systems Conference 2021 (DIS '21), June 28-July 2, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3461778.3462124

[*] Authors contributed equally.

DIS '21, June 28-July 2, 2021, Virtual Event, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8476-6/21/06…$15.00
https://doi.org/10.1145/3461778.3462124

1 INTRODUCTION

Information seeking is frequently used as a problem-solving tool, especially during the early stages of design and development [17]. For this purpose, emerging datasets of graphical user interfaces (GUIs), such as *Rico* [6], can be considered very valuable assets. They contain a vast amount of useful information, such as technical details, designs aspects, and information about features and properties of applications. The amount of information, and the various ways in which they can be used, far exceeds typical image-based datasets (e.g. [7]). This information could be used for various purposes, such as to aid the development of new apps, support data-driven design [6, 14], and even provide end-users with usage hints, guidance, and details during daily use. However, querying such datasets is often non-trivial and may require programming expertise, for example using a JSON-based API [14].

An important challenge in every information retrieval system is the so-called "semantic gap" [27], or the difficulty of articulating information needs in a way that a computer can understand reliably [25]. In non-design focused domains, information access is increasingly addressed with conversational assistants. For example, in everyday tasks, chatbots can help users retrieve information from the web, and execute actions on behalf of the user, such as set an alarm or make an appointment [2, 23]. In a more specific domain, conversational assistants have also been used recently to retrieve information from text documents [28].

In this paper, we explore the novel combination of these two lines of research – GUI design datasets and conversational information access. Concretely, inspired by the promise of the "conversations with documents" paradigm [28], we develop the concept of *"conversations with GUIs"* and study it in the context of mobile app datasets. The conversational modality is not intended to replace visually oriented interactions but to support it. Our proposed technical concept can enable users to interact with datasets of annotated GUIs conversationally, without requiring any technical expertise about the dataset or programming knowledge to retrieve information. In particular, they could pose natural language queries such as *"show me some search bar designs"*, *"when was the app updated?"*, or *"which of my apps require permission to use the camera?"*, to find information that helps them in their tasks or provides useful points of reference.

Designing and developing such interactive support is challenging. GUI datasets contain an immense amount of information. Crucially, given this large number of possibilities while querying the dataset, it is firstly not clear which capabilities are essential for users, and for different user groups. It is important to better understand user needs with regards to the supported capabilities so as to prioritise functionalities supported during information retrieval, and to fine-tune assistance during interactive use. Secondly, to bridge the semantic gap, and to support the goal of conversational assistance without requiring technical knowledge, we need to identify how users would frame their intentions as queries during conversations with GUIs. To address these challenges and gain a better understanding of the design space of possible conversations with GUIs, and subsequently to inform the design of conversational assistance, we conducted an online survey and gathered insights from 120 participants. In particular, users from three groups — designers, developers, and end-users — first ranked various capabilities informed by research literature and by information available in typical GUI datasets. Second, through elicitation, we collected a total of 1317 queries entered by our participants when prompted with three information retrieval tasks. We enriched these queries by manually labelling them with regard to four aspects: (1) *query scopes*: whether a query referred to an individual GUI view, an app consisting of multiple screens, or the entire dataset; (2) *query purpose*: the user's intention behind the query, e.g. to filter based on a criteria, to request for suggestions; (3) *response format*: the expected format in which the retrieved information was expected, e.g.image, text, binary, numeric; and (4) *information features*: particular features that the query was referring to, e.g. accessibility, privacy, design.

Findings from our survey shine light on what type of information users might want to access conversationally, and how they would explicate such intents through textual or verbal queries. To demonstrate the benefits and potential applications to interactive systems, we implemented a prototypical conversational assistant that understands such queries and retrieves information from the *Enrico* mobile dataset [15]. We release our labelled dataset and our open-source implementation to support future work in this area. For instance, our survey findings can inform the capabilities implemented when designing chatbots or agents that assist users in discovering and retrieving information for specific tasks such as GUI design or development. Furthermore, researchers can use our dataset of user queries to train machine learning models that can accurately interpret and classify user intentions, and provide interactive assistance accordingly.

To summarise, the main contributions of this paper are:

(1) An exploration of the design space of user queries for conversations with GUIs, assessed in a large-scale online study (N=120).
(2) A dataset of labelled user queries, plus an annotated version of the *Enrico* dataset containing additional app metadata, as informed by our survey results.
(3) An application in a conversational assistant that is capable of answering such questions, asked via text or voice input.

2 RELATED WORK

Conversations with GUIs builds on emerging research in three areas: conversational user interfaces (CUIs); projects that report and utilize GUI datasets, for example, for data-driven design; and work on CUIs for information retrieval.

2.1 Conversational User Interfaces

CUIs cover an increasing range of applications and tasks. For example, they have found their way into people's homes and everyday life via smart speakers and smartphones, realised as speech-based agents [23]. Related, speech-based CUIs are increasingly used in the automotive domain, where they serve tasks such as navigation, information, and entertainment [4]. Moreover, CUIs also appear as chatbots in a variety of application domains, such as providing "live" customer service on websites or in mobile messaging apps. Other investigated application areas of CUIs include health and wellbeing [13] and focus and productivity [9]. Several lines of research at the intersection of interactive systems design and artificial intelligence further show the growing relevance of CUIs. For example, researchers investigate personalisation of conversational agents [4, 18, 30], using conversation style and content to learn about the user [29, 34], and facilitating (online) studies and evaluations as an alternative to questionnaires [12, 32].

In summary, these successful applications of CUIs motivated us to explore their use in a novel domain: interactively retrieving information on GUI designs. Moreover, we go beyond the related work’s prevalent focus on end-users as a target group of CUIs, by explicitly investigating the needs of designers and developers.

Recent work has developed CUIs for tasks related to visual design. For example, Scones [11] is a system that allows users to create sketches with multiple visual elements via natural language, such as “draw a pizza on the table”. Similar to our use case, this relates dialogue to design aspects, such as visual elements and layout information. However, in contrast to the related work, we 1) address GUIs instead of free-form sketches, and 2) utilise CUIs for exploring a large body of existing visual designs instead of composing a new design.

Finally, our work also differs from visual question answering [1], which addresses questions about a given image by generating a textual answer. In contrast, an “answer” in our use case might be text, numbers, a GUI, or a part of its design (e.g. colour, element) — and might relate to a single view, a whole app, or even several apps. This motivates our survey to assess which kinds of queries people with different backgrounds might be interested in posing in the context of GUIs and their use and design.

2.2 GUI Datasets as a Design Resource

As our assistant retrieves information from a dataset of annotated GUIs, here we discuss work on such datasets and data-driven design. Sahami Shirazi et al. [24] automatically analysed layout files of 400 Android apps. They presented descriptive statistics, for example, regarding the number of views, layout variants, and GUI elements overall and for different app categories. Kumar et al. [14] proposed “design mining” for websites, and scraped elements and their visual features (e.g. colours, location, size) from 100,000 rendered websites . They supported queries to this dataset via an API in JSON format. With the Rico dataset, Deka et al. [6] presented a large collection of mobile GUIs scraped from 9,700 Android apps across 27 categories. They also built a search model that could retrieve GUIs deemed to be visually similar to a given query GUI view.

Relevant to our work, Kumar et al. [14] and Deka et al. [6] highlighted opportunities for using their datasets for data-driven design applications, including design search. This motivates our work here: We focus on making the information contained in such large design datasets accessible to a broad range of user groups, including but not limited to designers. In particular, we explore the use of natural language queries via a conversational assistant, in contrast to JSON-based APIs or image-based similarity search. These are arguably rather technical approaches, and require a concrete starting point or knowledge about possible queries. In this regard, we envision CUIs as an additional, less technical approach, with a low barrier of entry. CUIs enable people to get various practically useful pieces of information from design datasets simply by asking natural language questions.

Practically, we use an enriched version of the Rico dataset, *Enrico* [15], in order to 1) provide example GUI screenshots in our survey, and to 2) implement our proof-of-concept assistant (see Section 6.2). Figure 4 shows example GUIs from the *Enrico* dataset. Looking ahead, our conversational approach could also serve as an interface to interact with GUI datasets enriched by recent work on computational (semantic) modelling of GUIs [5, 10, 16] or work on extracting additional information from GUI visuals [33].

2.3 CUIs for Information Retrieval

In classic information retrieval systems, it is the user’s responsibility to adapt their search needs using specific keywords and/or syntax [20], which leads to the aforementioned semantic gap (see section 1). Crucially, CUIs address this issue by letting the user formulate their queries using natural language.

Many conversational assistants in smart speakers or smartphones offer search and information retrieval, such as asking for the weather forecast or looking up a needed or entertaining piece of information on the web [2, 23]. Here, we are not interested in such general everyday queries but rather conversational queries to a domain-specific (design) dataset. In this context, the most closely related approach to our work is the recent project by ter Hoeve et al. [28], framed as “conversations with documents”. They explored the use of a conversational assistant to enable users to retrieve information from a text document via natural language queries. Concretely, their motivation focuses on reviewing information in text, with example queries such as “Does the document already mention the mission of our company?”.

Their successful application and user interest motivates us to explore a related approach for querying GUIs, which are more complex in nature. Similar to text documents with sentences, sections, chapters, and so on, GUIs also have global and local composite and hierarchical structures (e.g. multiple radio buttons in a group, sections in a view, views in an app), to which users might refer in their questions. There are also many clear differences between text documents and our focus on GUIs. For example, most GUI designs closely integrate and rely on textual and visual information in combination, while ter Hoeve et al. [28] mostly focused on questions about information in the text. Moreover, layout and other design-related questions are likely more relevant in the context of GUIs, compared to the focus on text content in the related work. Finally, the user groups and their main tasks and interests are different. In our survey, as described in the following sections, we explicitly address the needs of designers, developers, and end-users.

3 STUDY METHOD

GUI datasets such as *Rico* [6] and *Enrico* [15] contain an immense amount of information about the applications and their interfaces. This ranges from low-level visual design details such as the type and style of elements on a particular screen or page, to higher-level aspects such as the purpose of a page or privacy-related permissions of the application. As such, GUI datasets are quite different from typical image-based datasets (e.g. [7]). GUIs contains a multiplicity of features that convey different types of information, and have different purposes.

To better understand users' information needs related to such GUI datasets, and how they might interact with them, we conducted an online survey with three potential groups of users: designers, developers, and end-users. We differentiated between these three particular groups due to the inherently different nature of tasks they might undertake while interacting with GUIs. In this section, we describe the survey methodology in detail. In the following two sections (section 4 and 5), we elaborate upon our research questions and report results.

3.1 Overview

We created an online survey where motivating scenarios and prompts were adapted towards each of the three user groups. The survey consisted of two parts. In the first part, we aimed to better understand what information types and features were desirable. To this end, participants rated perceived utility of various *pre-defined capabilities* for interacting with a sample GUI dataset. In the second part, we aimed to capture how participants might interact with such datasets by formulating queries. Here, sample screenshots from the dataset were displayed, and participants were asked to *freely pose queries* to a hypothetical conversational agent, or chatbot, that would help them during information seeking. Please note that participants had no previous knowledge about the kind of GUIs that could be found in our dataset, nor the final set of capabilities of the conversational assistant we were interested in developing.

3.2 Participants

We recruited our participants via Prolific.[1] To ensure high-quality responses, participants were required to have an approval rate of 95%, and could complete the study only once. We pre-screened participants with normal or corrected-to-normal vision. To recruit designers and developers, we additionally pre-screened participants who had been working in these respective industry sectors. While the pre-screening strategy attempted to achieve an equal distribution between the three user groups, participants could freely select the group they most closely identified with during the survey, resulting in an uneven distribution. Overall, 120 people (49 female, 70 male, 1 prefer not to say) between 18 and 53 years ($M = 26.1$, $SD = 7.3$) participated, out of which, 24 self-identified as UI/UX designers, 32 as developers, and 64 as end-users. The study took 20 minutes on average to complete. Participants were paid £2.5 (3.28 US dollars) upon completion, which corresponds to an hourly wage of £7.5/h ($9.7/h). Participation was under informed consent, and the study adhered to European privacy laws (GDPR).

3.3 Procedure

After an introductory briefing and informed consent, participants were asked to specify the target group (UI/UX designer, developer, end-user) they identified with the most. Textual descriptions of the motivating scenario and question prompts were adapted towards each of the three groups. For example, while designers were asked to consider their typical tasks of creating GUI designs while answering questions, end-users were asked to think about their daily usage and needs. To provide people with context, we showed a short video animation[2] demonstrating an example of an assistant (chatbot) being used to find information on GUIs, similar to Figure 1.

Next, participants completed the first part, where they provided ratings, on a five-point scale, for each pre-defined capability (presented in randomised order). In the second part of the survey, participants were encouraged to freely ask a hypothetical chatbot up to five queries for each of the three tasks (single GUI, single app, dataset). The order of these three tasks was retained across participants as it represented increasing levels of information content and complexity. Participants ended the survey by providing demographic information (gender, age).

4 RESEARCH QUESTION: IDENTIFYING INFORMATION NEEDS

Given the extensive amount of information available in annotated GUI datasets, they can offer a large number of capabilities and features for reference. However, not all capabilities are necessarily useful or desirable by users. Further, individual user groups (designers, developers, end-users) might find different sets of features more useful than others. To gain a better understanding of user needs during information retrieval tasks, we formulated our first research question:

RQ1: *What capabilities would people find useful while interacting with a GUI dataset? How do these differ between user groups?*

4.1 Materials and Method

We addressed RQ1 in the first part of our survey. Here, we followed the need-finding method of ter Hoeve et al. [28] and formulated 21 capabilities to cover a broad range of potentially useful functionalities for interacting with an annotated GUI dataset. We derived these by transferring capabilities from the related work on conversations with documents [28] to GUIs (e.g. "Find text in the document" would become "Find text in the GUI", and similarly for navigation, sharing, copy/paste, etc.). We added further capabilities based on typical metadata and other details available in GUI datasets [6, 15] such as application categories, privacy information, and GUI components. Finally, we added capabilities relating to information retrieval across many apps, as motivated by the related work on these datasets (e.g. showing similar designs). These covered a variety of information available within the dataset, such as design attributes, application metadata, and GUI descriptions. The full set of capabilities is listed in Table 1. Their presentation order was randomised between participants.

4.2 Results

To answer our first research question related to users' information needs, we aggregated participant ratings on perceived usefulness of each of the identified capabilities. Table 1 summarises the results by listing each capability, average ratings for each user group, and weighted average across all groups. In addition, it also ranks and highlights the top-five capabilities for each group.

[1] https://prolific.co

[2] Survey material is available in our data repository: https://osf.io/g25wh/

#	Capability	Designers 🖌 (N=24)	Developers <> (N=32)	End-users 👤 (N=64)	Weighted Average
1	Show GUIs of a particular application category (<> 4, 👤 2)	3.63	4.00^4	3.98^2	3.91
2	Show similar GUI designs (🖌 1, <> 2)	4.13^1	4.19^2	3.64	3.89
3	Show GUIs with some particular features (🖌 2, <> 3)	3.79^2	4.13^3	3.70	3.83
4	Show GUIs that serve a particular purpose (🖌 5, <> 1)	3.67^5	4.22^1	3.74	3.85
5	Enquire about privacy information of an app (👤 1)	3.63	3.41	4.00^1	3.77
6	Show GUIs filtered by rating or popularity (<> 5, 👤 5)	3.58	3.97^5	3.79^5	3.80
7	Show GUIs that have certain privacy features (👤 3)	3.58	3.47	3.95^3	3.75
8	Find some text in the GUI, if present (🖌 4)	3.67^4	3.69	3.67	3.67
9	Add a comment/bookmark to the GUI (or to some feature)	3.58	3.53	3.59	3.57
10	Enquire if an app contains a GUI with a particular purpose	3.50	3.66	3.73	3.66
11	Enquire about the popularity of an application	3.29	4.00	3.61	3.65
12	Highlight a component of the GUI (👤 4)	3.58	3.22	3.80^4	3.60
13	Copy the GUI, or a part of it	3.38	3.63	3.71	3.62
14	Enquire about the purpose of a particular GUI	3.54	3.34	3.59	3.51
15	Navigate to a certain feature or region of the GUI (🖌 3)	3.75^3	3.44	3.42	3.49
16	Enquire whether a particular GUI contains some feature	3.25	3.72	3.42	3.47
17	Share the GUI, or a collection of GUIs, with someone	3.50	3.31	3.41	3.40
18	Enquire how many times a feature is present in the GUI	3.50	2.88	3.47	3.32
19	Describe the GUI	3.29	2.84	3.44	3.25
20	Enquire about the developer information	3.04	2.63	2.85	2.83
21	Read out all the text in the GUI	2.92	2.47	3.23	2.96

Table 1: Full list of the pre-defined capabilities for the first part of the survey, and a summary of ratings results. The top-five capabilities for each user group are annotated, and highlight inter-group commonalities and differences.

In general, we observed that capabilities for finding a GUI, or filtering GUIs, that matched some criteria (feature, purpose, category, etc.) were rated highly by all groups (capabilities # 1–4, 6–7). In contrast, capabilities related to explaining the GUI, such as describing it (#19) or reading out the text (# 21), had lower ratings across groups. It should be noted that while our study was limited to participants with normal or corrected vision. Since such capabilities can support accessibility needs, they might be more desirable for specific groups of users outside the scope of this study. Between user groups, while enquiring about privacy aspects (#7) and highlighting components (#12) within a GUI were among the top-five capabilities for end-users, they were not as important for designers or developers. Conversely, while designers considered capabilities for navigating to certain regions (#15) and finding text within GUIs (#8) as important, these were not the top priorities for developers and end-users.

To study in more detail how information needs varied between user groups, we conducted further analysis of the results. The effect of user group on perceived usefulness for each capability was tested with repeated-measures ANOVA. Table 2 summarises all capabilities that revealed statistically significant differences between groups.

Post-hoc t-tests (Bonferroni-Holm corrected) with pooled SDs were conducted to make pairwise comparisons between groups for the above six capabilities. We observed statistically significant differences ($p < .05$) between end-users and developers for capabilities # 2, 5, 12, 18 and 21. While developers deemed #2 more useful compared to end-users, the other capabilities (# 5, 12, 18, and 21) were rated higher by end-users than developers. For capability #11, there was a statistically significant difference between developers and designers ($p < .05$), where designers found the capability more useful than developers. For all six above capabilities, we did not observe statistically significant differences between designers and end-users. The remaining 15 capabilities did not show statistically significant differences between groups.

#	Capability	$F(2, 117)$	η_p^2
2	Show similar GUI designs	4.56	0.07
5	Enquire about privacy information	3.34	0.05
11	Enquire about app popularity	3.23	0.05
12	Highlight a component of the GUI	3.29	0.05
18	How many times a feature is present in the GUI	3.84	0.06
21	Read out all the text in the GUI	3.83	0.06

Table 2: ANOVA tests for pre-defined capabilities. For the sake of brevity, only the statistically significant tests ($p < .05$ in all cases) are reported together with effect sizes.

4.3 Summary

Our results provide first insights into which capabilities are deemed useful, by different user groups, while querying a dataset of GUIs (RQ1). This knowledge can provide useful guidance for developing information retrieval features for querying GUI (design) data, conversational or not. A key insight here relates to the inter-group variations in information needs. As some capabilities are perceived as highly useful across all groups, this indicates that general-purpose conversational assistance can benefit users irrespective of their background, skills, and tasks. We also observe some particular differences between each of the groups, that often corresponds to particular tasks and needs. These findings can be applied to customise or select capabilities for interactive systems that are tailored to particular user groups. While we provide results for 21 capabilities, covering a wide range of use-cases, these are not exhaustive. Future work can expand upon this to study further capabilities. Further, while we cover the domain of mobile app datasets, addressing other domains remains an open question.

5 RESEARCH QUESTION: ELICITING USER QUERIES

Beyond understanding users' information needs, it would also be advantageous to capture how users might interact with systems that offered them capabilities of querying annotated GUI datasets. This can aid in developing conversational assistants or keyword-based search engines. To study this, we formulated the following research question:

RQ2: *How do people frame natural language queries while retrieving information from GUI datasets?*

5.1 Materials and Method

We addressed this question in the second part of our survey. Here people had to enter queries for a hypothetical conversational assistant (chatbot) that would retrieve information from a GUI dataset. We created three scenarios using GUI images from the *Enrico* dataset [15] as task stimuli. In the first scenario, people were asked to frame queries pertinent to a *single GUI*. An image of a single GUI was displayed, and details regarding the available information were specified textually. Per person, the GUI image was randomly assigned from five such images. In the second scenario, *application-level* queries were requested. Here, a participant was shown a set of five GUIs (screens) from one app. The app was randomly selected between participants from a pool of four apps. In the final scenario, participants elicited dataset-level queries. As stimulus, a subset of 25 GUIs from the entire dataset was displayed in a grid.

5.2 Results

We collected a total of 1473 elicitation queries from participants. We manually inspected all queries, and excluded all irrelevant entries. These were typically those not suited to the presented scenario and context of GUI datasets, and instead intended for general assistants like Siri or Alexa (e.g. *what is the current weather*, *send an email*, etc.). In total, we excluded 156 entries, resulting in a total of 1317 valid user queries.

To systematically understand the design space of queries users may ask, we manually labelled and coded queries according to three dimensions: *query scope*, *query purpose*, and *response format*. First, three authors independently generated codes by skimming responses for one scenario each, inductively creating a codebook. Next, through discussion and agreement, the codebooks were merged to create the final set of codes, grouped into meaningful dimensions. Finally, each author coded one task and cross-checked the coding of another task. Discrepancies resulting from this were resolved via discussion.

The final coding included the following dimensions and codes:

(1) **Query scope**: This specifies the search scope within the dataset. A query could pertain to a particular *GUI*, an *application*, or the entire *dataset*. In total, participants posed 413 *GUI*, 461 *app*, and 443 *dataset* scope queries.

(2) **Query purpose**: This indicates the user's purpose or intention while asking the chatbot a question, and can be one of the following:
   (a) *Filter*: Retrieve results that match some criteria (e.g. *show me all social apps*)
   (b) *Find*: Search for a particular GUI, or information within a GUI (e.g. *where is the search bar?*)
   (c) *Suggest*: Get recommendations or suggestions (e.g. *what colour palette should I use?*)
   (d) *Inform*: Get insights or details about a GUI, app, or dataset (e.g. *does this app use the camera?"*)
   (e) *Educate*: Ask for help or assistance (e.g. *how do I add a contact?*)
   (f) *Execute*: Perform an action on a GUI, app, or dataset (e.g. *Highlight the navigation menu*)

   In total, participants asked 368 *filter*, 42 *find*, 76 *suggest*, 716 *inform*, 49 *educate*, and 66 *execute* queries.

(3) **Response format**: When users pose queries, the format in which responses are expected can vary: either using *images*, short or long *text*, *binary* yes/no, or *numeric* values. From the queries posed by participants, 509 *images*, 236 *text*, 500 *binary*, and 72 *numeric* responses were identified.

Figure 2 shows this space of queries using the above dimensions, and provides a breakdown of the number of queries for each response format by the scope and purpose. Further, it also shows the number of queries by each user group for the different scopes and purposes. It can be observed that while some combinations of purpose and scope have a large number of entries (e.g. *app+inform*, *GUI+Inform*, *dataset+filter*), other parts of the space have only a few (e.g. *dataset+educate*, *app+filter*). Clear differences in response formats can also be observed here: while some categories require mostly image-based formats (e.g. *filter*), it can vary for others (e.g. *inform*). Finally, we can also observe some differences between user groups. For example, while end-users posed a large number of queries with the purpose to *inform*, developers had a larger number of *filter* queries. We can also observe that end-users focused more on *GUI*- or *app*-scoped queries, developers preferred *dataset* ones, while designers' queries were distributed across all three scopes.

In addition to the above three dimensions, we also labelled *information features* for each query. These features can be used to identify the type of data or information within the GUI dataset required for providing responses. A query could pertain to multiple features. For example, *"Which apps ask for camera permissions"* is labelled as having both *privacy* and *sensor* features. Figure 3 describes a list of 13 features, and the number of querie referencing each of them.

Legend

Response Formats		User Groups
🖼 Image	Tт Text	🖌 Designers
✓ Binary	## Numeric	<> Developer
		👤 End-Users

Purpose / Scope	Filter	Find	Suggest	Inform	Educate	Execute	User Group
GUI	🖼 8 ✓ 1	🖼 23 Tт 1 ## 3	🖼 22 Tт 8 ✓ 15 ## 1	🖼 22 Tт 43 ✓ 206 ## 11	🖼 15 Tт 13	🖼 14 Tт 4 ✓ 3	🖌 92 <> 72 👤 249
App	🖼 3 ✓	🖼 7 Tт 1	🖼 10 Tт 97 ✓ 259 ## 43	🖼 10 Tт 97 ✓ 259 ## 43	🖼 3 Tт 16 ## 1	🖼 1 Tт 1	🖌 76 <> 83 👤 302
Dataset	🖼 323 Tт 30 ✓ 1 ## 2	🖼 7 Tт ✓ ##	🖼 2 Tт 5 ✓ 7 ## 11	🖼 2 Tт 5 ✓ 7 ## 11	Tт 1	🖼 42 Tт 1	🖌 100 <> 204 👤 139
User Group	🖌 76 <> 165 👤 127	🖌 9 <> 5 👤 28	🖌 55 <> 17 👤 4	🖌 84 <> 136 👤 496	🖌 19 <> 7 👤 23	🖌 25 <> 29 👤 12	🖌 268 <> 359 👤 690

Figure 2: The design space of queries posed by participants in our survey. Queries had varying *scopes* (GUI-, app-, or dataset-level) and served different *purposes* (6 levels), and responses (output) could be presented in different *formats* (4 levels).

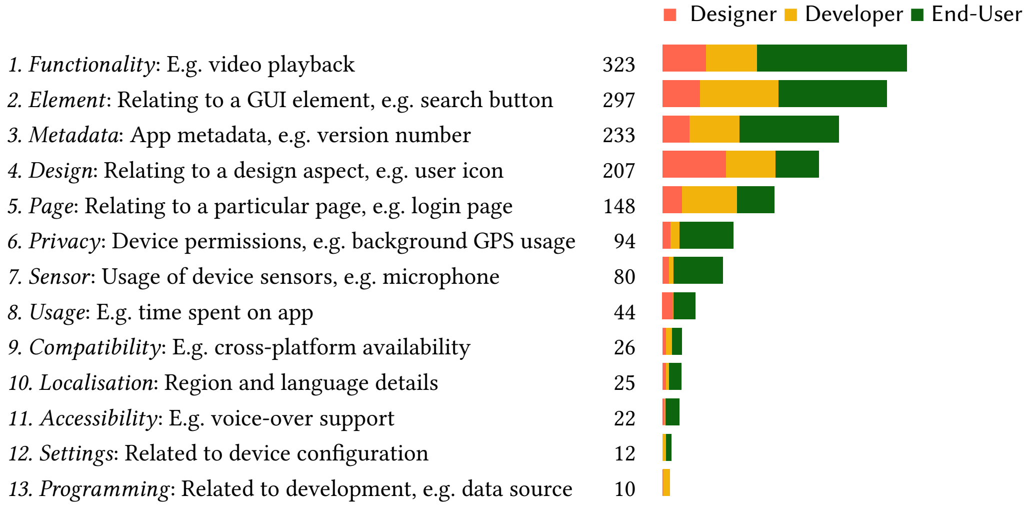


Figure 3: *Information features* that participants referred to in their queries. Multiple features could be included within a single query (e.g. “Does this app have access to my microphone” references both privacy and sensor).

5.3 Summary

Our study contributes a labelled dataset of 1317 queries that can be posed to conversational assistants while querying GUI datasets. Our design space breaks down elicited queries by scope, purpose, and response format to better understand how people frame intents during information seeking tasks in this context. Our findings can inform the implementation of conversational assistants by highlighting the different types of queries that a system could expect from the user, and formats in which it should present results. The labelled dataset is openly available in our data repository. Future work can expand upon our findings by covering additional user groups and contexts. For example, it would be beneficial to gain insights from users with visual or motor impairments regarding their information needs and how they would frame queries that specifically address these needs. Further, while our study is limited to queries posed in English, future works can follow our method to investigate other languages as well.

6 APPLICATION: AN INTERACTIVE CONVERSATIONAL ASSISTANT

We implemented a conversational assistant for interacting with GUIs, both to provide a demonstration of technical feasibility, and to manifest the results of the survey as an open-source prototype system to stimulate further work.

6.1 Example Usage Scenarios

In the following, we illustrate how our conversational assistant is used by designers, developers, and end-users for some typical information retrieval tasks when interacting with an annotated dataset of mobile app GUIs.

Designer: Joe is a mobile app designer looking for inspiration to design a new GUI for an app. He begins by asking the assistant for related designs: "**Show me login pages**". He gets a list of three login pages, displayed as images, but is not satisfied and asks "**Show me more**". The assistant keeps track of the conversation, and provides three more examples of login pages. Joe now wants to filter similar designs to the first one in this new list: "**Show me more like the first one**". And the assistant provides related design examples.

Developer: Jane is a mobile app developer who wants to know if apps similar to the one she is currently developing offer in-app purchases. She begins by telling the assistant what kind of app she wants to get information about: "**I'm creating a Fitness app**". She gets a list of three fitness apps that she can reference for further analysis. She now asks about the app listed in the second place: "**Does the second one have in-app purchases?**", and the assistant provides this information by consulting the dataset.

End-user: Jun is an end-user who wants to know more about an app they have just downloaded. They begin by asking details about the app developer: "**Who developed "4 Warn Weather"?**"[3] The assistant replies with the developer name and contact email. Jun can ask more questions such as "**Does it make use of GPS?**", and the assistant replies with corresponding insights.

Summary: As illustrated, through natural language commands and conversations, both Jane and Jun can get precise information about particular apps, and Joe can explore or exploit existing app designs. These interactive capabilities, without requiring technical expertise, have not been previously offered to such a wide range of user groups to the best of our knowledge.

6.2 GUI Dataset

For our implementation, we used the *Enrico* dataset [15], a curated and enriched subset of *Rico* [19]. The dataset includes 1460 mobile app GUIs categorised according to a design taxonomy of 20 GUI layout categories, such as news, login, settings, tutorial, etc. (Fig. 4). Each GUI comprises a screenshot with additional data such as annotations of semantic wireframes, GUI elements, visual and structural data, and interactive design properties.

To enable further capabilities, as identified by our survey, we extended the dataset with additional metadata from the Google Play Store, for all available apps. This includes app description, number of reviews, ratings, price, developer info, requested permissions, etc. Our extended dataset will be made openly available in our project repository.

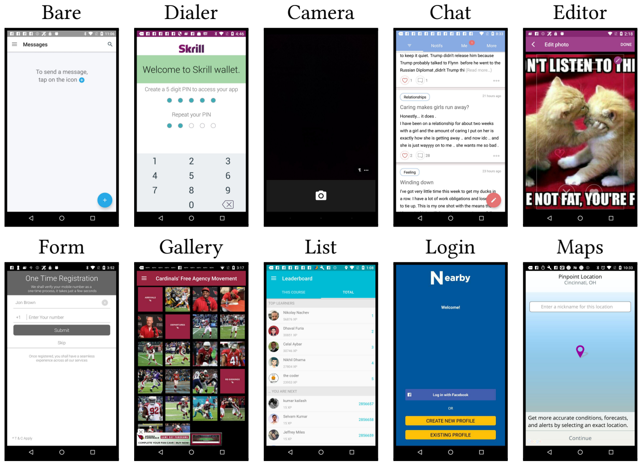


Figure 4: Our annotated version of the *Enrico* dataset contains mobile app GUIs with additional design and app information. This figure shows examples for different kinds of views, such as the login screen or a phone dialer.

6.3 Implementation

Our assistant is developed using a web-based architecture. The user (front-end) interacts with it in a browser, either using text or voice input. The back-end server processes queries, retrieves information, and returns it to the front-end browser.

6.3.1 *Front-end.*

Implemented using HTML, CSS, and JavaScript, our responsive UI (Fig. 1) resembles a messaging client, similar to other common web chatbots. Text input is supported across browsers. Voice-based input is currently only supported for webkit browsers. Google Chrome, for example, internally uses a server-based speech recognition engine.

6.3.2 *Back-end and Natural Language Understanding (NLU) Engine.*

A key module of our CUI is the natural language understanding (NLU) engine developed with the RASA framework.[4] RASA is a popular open-source machine learning framework to automate text- and voice-based assistants. The NLU engine comprises an NLU model and an NLU server. The NLU model is trained on our dataset of 1317 user queries to understand queries, and identify relevant entities. For our prototype, we limited our implementation to the top-five capabilities for each user group, informed by our survey.

An entity is a piece of user-provided information that complements a given user intent. For example, the intent `get_developer` requires the app name to provide a concrete response, so the app

[3] We added quotes here to clearly indicate the app name, but they are not required in practice.

[4] https://rasa.com/

name is an entity. Adding new intents and queries to the NLU engine is as simple as editing a configuration file and retraining the NLU model with representative examples.

The model pipeline comprises a spellchecker, a spaCy tokenizer and featurizer,[5] and a Support Vector Machine (SVM) classifier. The featurizer transforms the entered text in a GloVe [22] word embedding. The SVM uses a linear kernel ($\gamma = 0.1$) and is optimised via grid search using 5-fold cross-validation and F-measure (harmonic mean of Precision and Recall) as target metric. The CUI backend provides a REST API to connect the NLU engine to the frontend. The API is developed with Express, a popular framework for creating web applications in nodejs. All API communications are stateless (as per REST definition) and JSON-based. We further use *stories*, a mechanism provided by RASA to control conversation workflows. A story has a series of steps to achieve some task or goal, including fallback behaviours. This allows for more expressive and natural conversations, as the context of the conversation can be easily maintained; see Figure 1 for some examples.

Our prototype supports basic filtering and ranking capabilities, e.g. **"show me all the chat apps"**, **"now show me only those rated 4 or higher"**. Currently it does not support multiple intents per query, however this could be implemented in a future more comprehensive system with simple anchoring tokens such as the "and" keyword to split sub-queries.

6.4 Validation: NLU Engine

As training data for the NLU engine, we included 154 intent examples and 137 entity examples, which we manually extracted from our dataset of 1317 queries. To test the generalisation of the model prediction, a large testing dataset was generated and randomly sampled by Chatito[6], an online tool using a domain specific language (DSL) to generate datasets for NLU model validation. We used the built-in NLU test unit in RASA to evaluate the NLU model on 1452 sample queries as testing data. Our results indicated that the NLU model (semantic parser) has high performance (Table 3). For example, accuracy of intent prediction is 94.1%, and all other performance metrics (precision, recall, and F-measure) are within a similar range. This validates that our system can accurately recognise and classify user queries to provide conversational assistance.

	Accuracy	Precision	Recall	F-measure
macro avg	0.941	0.936	0.978	0.954
weighted avg	0.941	0.951	0.942	0.943

Table 3: Results for intent classification using our NLU Engine for conversational assistance, i.e. how likely an utterance will be translated to the right expression.

7 DISCUSSION AND FUTURE WORK

This paper introduces *conversations with GUIs* – interactions with annotated GUI datasets using a conversational assistant. In the following, we discuss some key insights and implications, and opportunities for future work.

7.1 Information needs for conversations with GUIs differ between designers, developers, and end-users

Our results reveal the different information needs of the potential user groups of conversational assistants in the context of GUIs. For example, this shows in the-top five features as rated by each group: *Designers* particularly valued queries for retrieving similar UIs, UIs with a certain feature/element (e.g. search bar), navigating to such features, finding text in the UI, and UIs for a particular purpose. *Developers* were also interested in queries on similar UIs, particular features, and purposes. They were further interested in asking about UIs based on metadata such as app ratings, popularity, and categories. Beyond the conversational context, these results also provide empirical evidence for interest of designers and developers in data-driven inspiration and comparative "quality checks", as motivated in related work [6, 14]. In contrast, *end-users* had rather different queries in mind. They were particularly interested in privacy information and features, highlighting certain UI elements, and finding GUIs of apps from a particular category and/or that are popular or rated highly by others. These differences are also noticeable in the elicited queries, where users posed questions to a hypothetical conversational assistant.

In summary, these insights motivate investigations of applications for each group in more detail. For example, for designers and developers, future work could study interactions and integration of such CUI features into design and development tools and workflows, informed by the top features emerging here. For end-users, a "privacy assistant" might be a particularly interesting CUI concept to explore further. This would relate the idea of "conversations with GUIs" to work in usable privacy and security, and awareness of related mobile risks and settings [21]. Finally, future work should also consider users of specific groups, such as those with visual or motor-impairments, to provide customised assistance.

7.2 Conversations with GUIs require varied types and formats of responses

Based on the elicited queries, conversations with GUIs lead to various expected response formats – images, text, yes/no replies, and numeric measures. Visual responses cover various levels, such as showing an entire GUI view, specific elements within a GUI (e.g. for design-related queries such as asking about "blue buttons"), or visually highlighting elements in a view (e.g. for *find* queries). The variety discovered here stands in contrast to the focus on textual content in the related work on conversations with documents [28]. For some questions (mostly of type *educate*), responses might even include an animation, video, or interactive guidance (e.g. when asking about possible interactions or navigation paths in a GUI). In conclusion, our results motivate supporting multimedia responses when creating assistants for conversations with GUIs.

A related consideration is the locus or anchor of a response. For example, a textual response could be presented as a chat, as in our prototype (Figure 1-centre), or anchored on a GUI image (e.g. annotation bubble). Overall, our insights and question set motivate and support further exploration of such design dimensions for CUIs.

[5] https://spacy.io/

[6] https://github.com/rodrigopivi/Chatito

7.3 Users' expectations of different query purposes imply varied CUI roles

Across all three user groups, the elicited user queries revealed six fundamental purposes that people are interested in during conversations with GUIs. Concretely, these are *filter*, *find*, *suggest*, *inform*, *educate* and *execute*.

Practically, these underline the rich possibilities that users see in such a conversational assistant. Conceptually, they also indicate potentially varying interpretations of such a CUI concept, namely regarding viewing it as a tool vs agent-like interface [8, 26]. On the one hand, some *suggest* queries clearly implied a personification of the CUI (i.e. asking for an opinion or implying that the CUI has background knowledge e.g. on a project or design). Similarly, some *educate*, *inform* and *execute* queries put the CUI into a personal assistant role. On the other hand, other queries indicate a user interpretation of the CUI concept that seems more akin to a tool – in particular, *filter* and *find* queries.

As highlighted in a recent survey [31], "intelligence" in UIs has been related to both tools and agents. In this light, we position our prototype implementation in this paper as a rather neutral agent style – a chatbot with a neutral presentation and without aiming for a human-like representation or personality. However, future work could investigate the idea of framing the CUI, for example for designers, as a more characteristic designer "persona" (or a set of designer assistants with different personas). This might particularly support *suggest* queries in that the personas could then give "their personal opinions/suggestions" on a GUI, and might make for an interesting point of comparison to other presentations for other queries, target groups, data-driven design concepts, and so on.

7.4 Implications for intelligent assistance, tool integration, and interaction design

Regarding *technical requirements*, our results motivate capabilities to first classify query scopes and purposes. Our prototype shows the feasibility of this and provides a starting point for future improvements. Response formats can then be chosen adequately. Our prototype also detects a variety of intents, as informed by our survey. Technically detecting the user group might also be helpful if a future assistant is deployed to cater to multiple groups in practice.

We further expect our collected questions to be useful for *training machine learning models* in this context. These models could be integrated into a production-level design tool such as Sketch or Figma and empower designers with an intelligent partner that can understand their needs as natural language queries. Our annotated dataset provides several types of information relating to mobile app GUIs. Creation of new datasets, or augmentation of existing ones, can increase the scope of capabilities for intelligent assistants such as the one we have implemented here.

We also see further interesting avenues regarding *UI and interaction design*. For example, for some questions, multiple response formats might be beneficial (e.g. colour could be presented visually or as RGB values). Similarly, yes/no binary questions about a feature could be answered as such, or by showing the feature in the GUI (e.g. "Does this app contain a login function?"). Such ambiguity could be handled with a *details-on-demand* interaction. For example, the assistant might respond with a compact text, embedding a link or hover effect that additionally shows a (visual) response. It could also learn from the user's subsequent interactions (e.g. opening the detail view) to inform future default response types.

More generally, UI and interaction techniques also depend on the targeted user group. For example, designers might use *speech input as a side-channel* to ask queries for inspiration or information while working on a design with tablet, mouse or keyboard. Thinking beyond our prototype here, such an assistant might allow users to relate queries to their current screen context (e.g. ask "Find UIs like *this*"), akin to the famous "put that there" [3]. Indeed, some user questions elicited in our survey already imply that users expect such contextual awareness (e.g. "Is there another version of this component?"), motivating this as an interesting concrete direction for future work.

8 CONCLUSION

This paper introduces *conversations with GUIs*, a novel concept to bridge the "semantic gap" for information retrieval in GUI design datasets. Such an approach can enable people to gain useful insights from such datasets via natural language, for example for design inspiration. Concretely, our online survey ($N = 120$) with designers, developers, and end-users provides the first and most important step towards developing conversational interactions with GUIs by capturing and understanding fundamental user needs. It also reveals vital similarities and differences between three user groups: designers, developers, and end-users. Crucially, designers' conversational needs here include finding similar GUIs as well as particular GUI features and text therein, plus finding GUIs for a certain purpose, and navigating to parts of a GUI design. In contrast, for example, end-users particularly valued privacy-related capabilities. Moreover, our elicited 1317 user queries reveal users' key assumptions and expectations, regarding design scope (GUI, app, dataset), purpose (filter, find, suggest, inform, educate, execute), response formats (image, text, binary, numeric), and thirteen distinct information features (e.g. design, usage, sensor, metadata). Building on these insights, our implementation of a first assistant application recognises such intents using natural language processing and retrieves relevant information from a dataset of mobile app GUIs, thus practically demonstrating our vision of *conversations with GUIs*. By releasing our labelled query dataset and assistant implementation to the community, we hope to stimulate further research on conversational tools for designers and in design contexts.

OPEN SCIENCE

We support further research efforts by releasing our survey material, labelled dataset of queries, and chatbot implementation, on our project page: https://userinterfaces.aalto.fi/hey_gui.

ACKNOWLEDGMENTS

This work was funded by the Department of Communications and Networking (Comnet), Finnish Center for Artificial Intelligence (FCAI), Academy of Finland (grant numbers 291556, 310947), and the Bavarian State Ministry of Science and the Arts and coordinated by the Bavarian Research Institute for Digital Transformation (bidt).

REFERENCES

[1] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2017. VQA: Visual Question Answering. *Int. J. Comput. Vision* (May 2017). https://doi.org/10.1007/s11263-016-0966-6

[2] Tawfiq Ammari, Jofish Kaye, Janice Y. Tsai, and Frank Bentley. 2019. Music, Search, and IoT: How People (Really) Use Voice Assistants. *ACM Trans. Comput.-Hum. Interact.* 26, 3, Article 17 (April 2019). https://doi.org/10.1145/3311956

[3] Richard A. Bolt. 1980. "Put-That-There": Voice and Gesture at the Graphics Interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques* (Seattle, USA) *(SIGGRAPH '80)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/800250.807503

[4] Michael Braun, Anja Mainz, Ronee Chadowitz, Bastian Pfleging, and Florian Alt. 2019. At Your Service: Designing Voice Assistant Personalities to Improve Automotive User Interfaces. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland) *(CHI '19)*. Association for Computing Machinery, New York, USA. https://doi.org/10.1145/3290605.3300270

[5] Sara Bunian, Kai Li, Chaima Jemmali, Casper Harteveld, Yun Fu, and Magy Seif El-Nasr. 2021. VINS: Visual Search for Mobile User Interface Design. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3411764.3445762

[6] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. 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* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3126594.3126651

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2009.5206848

[8] Umer Farooq, Jonathan Grudin, Ben Shneiderman, Pattie Maes, and Xiangshi Ren. 2017. Human Computer Integration <i>versus</i> Powerful Tools. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3027063.3051137

[9] Ted Grover, Kael Rowan, Jina Suh, Daniel McDuff, and Mary Czerwinski. 2020. Design and Evaluation of Intelligent Agent Prototypes for Assistance with Focus and Productivity at Work. In *Proceedings of the 25th International Conference on Intelligent User Interfaces* (Cagliari, Italy) *(IUI '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3377325.3377507

[10] Zecheng He, Srinivas Sunkara, Xiaoxue Zang, Ying Xu, Lijuan Liu, Nevan Wichers, Gabriel Schubiner, Ruby Lee, Jindong Chen, and Blaise Aguera y Arcas. 2020. ActionBert: Leveraging User Actions for Semantic Understanding of User Interfaces. In *AAAI-21*. https://arxiv.org/abs/2012.12350

[11] Forrest Huang, Eldon Schoop, David Ha, and John Canny. 2020. Scones: Towards Conversational Authoring of Sketches. In *Proceedings of the 25th International Conference on Intelligent User Interfaces* (Cagliari, Italy) *(IUI '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3377325.3377485

[12] Soomin Kim, Joonhwan Lee, and Gahgene Gweon. 2019. Comparing Data from Chatbot and Web Surveys: Effects of Platform and Conversational Style on Survey Response Quality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3290605.3300316

[13] A. Baki Kocaballi, Juan C. Quiroz, Liliana Laranjo, Dana Rezazadegan, Rafal Kocielnik, Leigh Clark, Q. Vera Liao, Sun Young Park, Robert J. Moore, and Adam Miner. 2020. Conversational Agents for Health and Wellbeing. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3334480.3375154

[14] Ranjitha Kumar, Arvind Satyanarayan, Cesar Torres, Maxine Lim, Salman Ahmad, Scott R. Klemmer, and Jerry O. Talton. 2013. Webzeitgeist: Design Mining the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/2470654.2466420

[15] Luis A. Leiva, Asutosh Hota, and Antti Oulasvirta. 2020. Enrico: A Dataset for Topic Modeling of Mobile UI Designs. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services* (Oldenburg, Germany) *(MobileHCI '20)*. Association for Computing Machinery, New York, NY, USA, Article 9. https://doi.org/10.1145/3406324.3410710

[16] Toby Jia-Jun Li, Lindsay Popowski, Tom M. Mitchell, and Brad A. Myers. 2021. Screen2Vec: Semantic Embedding of GUI Screens and GUI Components. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3411764.3445049

[17] Yu-Tzu Lin and Morten Hertzum. 2020. How Do Designers Make User-Experience Design Decisions?. In *Design, User Experience, and Usability. Interaction Design*. Springer International Publishing, Cham.

[18] Gesa Alena Linnemann and Regina Jucks. 2018. 'Can I Trust the Spoken Dialogue System Because It Uses the Same Words as I Do?'—Influence of Lexically Aligned Spoken Dialogue Systems on Trustworthiness and User Satisfaction. *Interacting with Computers* 30, 3 (03 2018). https://doi.org/10.1093/iwc/iwy005 arXiv:https://academic.oup.com/iwc/article-pdf/30/3/173/24805335/iwy005.pdf

[19] Thomas F. Liu, Mark Craft, Jason Situ, Ersin Yumer, Radomir Mech, and Ranjitha Kumar. 2018. Learning Design Semantics for Mobile Apps. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3242587.3242650

[20] Ying-Hsang Liu, Alexandre Arnold, Gérard Dupont, Catherine Kobus, and François Lancelot. 2020. Evaluation of conversational agents for aerospace domain. In *Proc. CIRCLE*.

[21] Maria Muszynska, Denise Michels, and Emanuel von Zezschwitz. 2018. Not On My Phone: Exploring Users' Conception of Related Permissions. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI EA '18)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3170427.3188625

[22] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar. https://doi.org/10.3115/v1/D14-1162

[23] Martin Porcheron, Joel E. Fischer, Stuart Reeves, and Sarah Sharples. 2018. Voice Interfaces in Everyday Life. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, USA. https://doi.org/10.1145/3173574.3174214

[24] Alireza Sahami Shirazi, Niels Henze, Albrecht Schmidt, Robin Goldberg, Benjamin Schmidt, and Hansjörg Schmauder. 2013. Insights into Layout Patterns of Mobile User Interfaces by an Automatic Analysis of Android Apps. In *Proceedings of the 5th ACM SIGCHI Symposium on Engineering Interactive Computing Systems* (London, United Kingdom) *(EICS '13)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/2494603.2480308

[25] Franco M. Segarra, Luis A. Leiva, and Roberto Paredes. 2011. A Relevant Image Search Engine with Late Fusion: Mixing the Roles of Textual and Visual Descriptors. In *Proceedings of the 16th International Conference on Intelligent User Interfaces* (Palo Alto, CA, USA) *(IUI '11)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/1943403.1943496

[26] Ben Shneiderman and Pattie Maes. 1997. Direct Manipulation vs. Interface Agents. *Interactions* 4, 6 (Nov. 1997). https://doi.org/10.1145/267505.267514

[27] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 12 (2000). https://doi.org/10.1109/34.895972

[28] Maartje ter Hoeve, Robert Sim, Elnaz Nouri, Adam Fourney, Maarten de Rijke, and Ryen W. White. 2020. Conversations with Documents: An Exploration of Document-Centered Assistance. In *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval* (Vancouver BC, Canada) *(CHIIR '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3343413.3377971

[29] Sarah Theres Völkel, Renate Haeuslschmid, Anna Werner, Heinrich Hussmann, and Andreas Butz. 2020. How to Trick AI: Users' Strategies for Protecting Themselves from Automatic Personality Assessment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3313831.3376877

[30] Sarah Theres Völkel, Penelope Kempf, and Heinrich Hussmann. 2020. Personalised Chats with Voice Assistants: The User Perspective. In *Proceedings of the 2nd Conference on Conversational User Interfaces* (Bilbao, Spain) *(CUI '20)*. Association for Computing Machinery, New York, NY, USA, Article 53. https://doi.org/10.1145/3405755.3406156

[31] Sarah Theres Völkel, Christina Schneegass, Malin Eiband, and Daniel Buschek. 2020. What is "Intelligent" in Intelligent User Interfaces? A Meta-Analysis of 25 Years of IUI. In *Proceedings of the 25th International Conference on Intelligent User Interfaces* (Cagliari, Italy) *(IUI '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3377325.3377500

[32] Thiemo Wambsganss, Rainer Winkler, Matthias Söllner, and Jan Marco Leimeister. 2020. A Conversational Agent to Improve Response Quality in Course Evaluations. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3334480.3382805

[33] Xiaoyi Zhang, Lilian de Greef, Amanda Swearngin, Samuel White, Kyle Murray, Lisa Yu, Qi Shan, Jeffrey Nichols, Jason Wu, Chris Fleizach, Aaron Everitt, and Jeffrey P. Bigham. 2021. Screen Recognition: Creating Accessibility Metadata for Mobile Applications from Pixels. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3411764.3445186

[34] Michelle X. Zhou, Gloria Mark, Jingyi Li, and Huahai Yang. 2019. Trusting Virtual Agents: The Effect of Personality. *ACM Trans. Interact. Intell. Syst.* 9, 2–3, Article 10 (March 2019). https://doi.org/10.1145/3232077