UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Conversations with Documents

An Exploration of Document-Centered Assistance

ter Hoeve, M.; Sim, R.; Nouri, E.; Fourney, A.; de Rijke, M.; White, R.W.

DOI
10.1145/3343413.3377971

Publication date
2020

Document Version
Author accepted manuscript

Published in
CHIIR '20

Link to publication

Citation for published version (APA):
ter Hoeve, M., Sim, R., Nouri, E., Fourney, A., de Rijke, M., & White, R. W. (2020). Conversations with Documents: An Exploration of Document-Centered Assistance. In *CHIIR '20: proceedings of the 2020 Conference on Human Information Interaction and Retrieval : March 14-18, 2020, Vancouver, BC, Canada* (pp. 43-52). The Association for Computing Machinery. https://doi.org/10.1145/3343413.3377971

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:07 Oct 2026

Conversations with Documents

An Exploration of Document-Centered Assistance

Maartje ter Hoeve[1,*] Robert Sim[2] Elnaz Nouri[2] Adam Fourney[2] Maarten de Rijke[1] Ryen W. White[2]
[1]University of Amsterdam, Amsterdam, Netherlands [2]Microsoft Research, Redmond, WA
m.a.terhoeve@uva.nl, rsim@microsoft.com, elnouri@microsoft.com
adamfo@microsoft.com, derijke@uva.nl, ryenw@microsoft.com

ABSTRACT

The role of conversational assistants has become more prevalent in helping people increase their productivity. Document-centered assistance, for example to help an individual quickly review a document, has seen less significant progress, even though it has the potential to tremendously increase a user's productivity. This type of document-centered assistance is the focus of this paper. Our contributions are three-fold: (1) We first present a survey to understand the space of document-centered assistance and the capabilities people expect in this scenario. (2) We investigate the types of queries that users will pose while seeking assistance with documents, and show that document-centered questions form the majority of these queries. (3) We present a set of initial machine learned models that show that (a) we can accurately detect document-centered questions, and (b) we can build reasonably accurate models for answering such questions. These positive results are encouraging, and suggest that even greater results may be attained with continued study of this interesting and novel problem space. Our findings have implications for the design of intelligent systems to support task completion via natural interactions with documents.

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in HCI**;
• **Information systems** → **Question answering**.

KEYWORDS

Document-centered assistance; Productivity; Digital assistants; Question answering

ACM Reference Format:
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 *2020 Conference on Human Information Interaction and Retrieval (CHIIR '20), March 14–18, 2020, Vancouver, BC, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3343413.3377971

*Work done while the first author was an intern at Microsoft Research AI.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CHIIR '20, March 14–18, 2020, Vancouver, BC, Canada
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-6892-6/20/03…$15.00
https://doi.org/10.1145/3343413.3377971

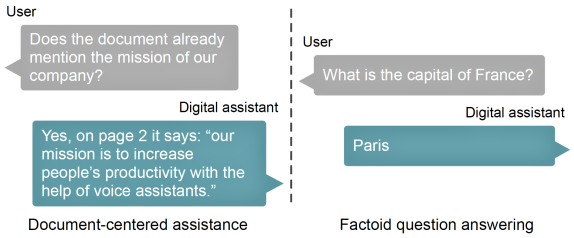


Figure 1: An example of document-centered assistance (left) vs. factoid question answering (right).

1 INTRODUCTION

Digital assistants are used extensively to help people increase their productivity [25]. A person can rely on their voice assistant, such as Amazon Alexa, Microsoft Cortana, or Google Assistant, to set an alarm while cooking, to play some music in the background, and to do a Web search on a recipe's ingredients. Conversational interaction is also playing an increasingly important role in helping people to increase their productivity for work-related tasks [33].

One area of interest that has not seen significant progress is document-centered assistance. Consider the following example: a person is driving to a crucial business meeting to prepare for a day with potential investors. The person is co-authoring a document about their company that will be provided to its investors, and it will be finalized in the upcoming business meeting. To be optimally prepared for the meeting, the individual wants to review what is already in the document. Since they are driving, they do not have direct access to the document, so they call their conversational assistant. The assistant has access to the document and can answer any query related to the document. The driver might pose queries such as "*does the document mention the mission of our company?*" or "*summarize what it says about our growth in the last two years.*" – queries that help them understand what is already outlined in the document and what they still have to add to finalize the document. At the same time, the driver is unlikely to ask factoid questions, such as "*who is the CEO of our company?*," given that they are already familiar with the organisation. In fact, previous work in the context of email and Web search has shown that people's information needs are different when they are a co-owner of a document than when they are not [1]. We hypothesize a similar difference in information needs in the context of document-assistance, motivated by the given example. This implies that document-centered assistance should critically differ from existing question answering (QA) systems, which are mostly trained to give short answers to factoid questions [e.g., 27, 29]. Figure 1 gives an example of this difference. Document-centered assistance would also differ from non goal-oriented "chit-chat" scenarios [e.g., 31, 37] – in our

document-centered scenario, people have very clear information needs.

In this paper, we investigate this space of document-centered assistance. This is an important task, since good document-centered assistance has the potential to significantly increase a person's productivity. We specifically focus on text consumption and document comprehension scenarios in a work context, and we seek to answer the following three research questions:

(**RQ1**) What kinds of conversational assistance would people like to receive in a document consumption scenario?
(**RQ2**) What kinds of queries might people use to receive this assistance when conversing with a document-aware assistant?
(**RQ3**) How well do initial baseline models do in a document-centered scenario?

With this work we contribute:

(**C1**) An understanding of assistant capabilities that are important to enable the document consumption scenario;
(**C2**) Insights into the types of questions people may ask in the context of document-centered assistance;
(**C3**) A detailed exploration of a human-annotated dataset with: (a) a collection of work-related documents, (b) questions a person might ask about the documents, given some limited context, (c) potential answers to the questions as represented by text spans in the document, (d) additional metadata indicating some properties of the questions (for instance, it is a yes/no closed question, or the question is unanswerable given the document);
(**C4**) Baseline experiments applied to the dataset exploring ways to handle document-centered questions.

Our research consists of three steps: (1) we perform a survey to answer RQ1 and RQ2 (Section 3); (2) we proceed with a data collection step, outlined in Section 4; and we answer RQ3 in Section 5.

2 RELATED WORK

This paper is related to two broad strands of research. In the first part of this section we look into voice controlled document narration and natural language interactions with productivity software, which is relevant to the first step of our research, the survey. Our initial modeling steps focus on single-turn conversations, and so we conclude this section with work on question answering.

2.1 Voice-Controlled Document Narration

Document-centric assistance in the context of text-consumption is related to prior work that explores adding voice interactions to screen readers. Screen readers are accessibility tools that narrate the contents of screens and documents to people who are blind, or who have low-vision. In this space, Ashok et al. [2] implemented CaptiSpeak – a voice-enabled screen reader that maps utterances to screen-reader commands and navigation modes (e.g., "read the next heading", "click the submit button"). More recently, Vtyurina et al. [34] developed VERSE, a system that adds screen reader-like capabilities into a more contemporary virtual assistant. VERSE leverages a general knowledge-base to answer factoid questions (e.g., "what is the capital of Washington"), but then differentiates itself by allowing users to navigate documents through voice (e.g., "open the article and read the section headings"). An evaluation with 12 people who are blind found that VERSE meaningfully extended the capabilities of virtual assistants, but that the QA and document navigation capabilities were too disjoint – participants expressed a strong interest in being able to ask questions about the retrieved documents. This strongly motivates the research presented in this paper.

2.2 Interactions with Productivity Software

There is an increasing interest in how people use different devices for their work-related tasks [e.g., 9, 14, 16, 35]. Martelaro et al. [24] show that in-car assistants can help users to be more productive while commuting, yet in easy, non-distracting traffic scenarios. While digital assistance in cars is a recent development [e.g., 23], natural-language interfaces have existed for much longer in more traditional work scenarios; for example the search box in products such as Microsoft Office and Adobe Photoshop. Bota et al. [4] research search behavior in productivity software, specifically in Microsoft Office, and characterize the most used search commands. Fourney and Dumais [11] investigate different types of queries users pose to a conversational assistant. Specifically they focus on *semi implicit system queries* and *fully implicit system queries*. They show that different types of queries can be reliably detected and that forms of query alteration can boost retrieval performance.

2.3 Question Answering

Question answering is the task of finding an answer to a question, given some context. A lot of progress has been made in the area, driven by the successful application of deep learning architectures and the increase of large scale datasets [e.g., 10, 15, 17, 19, 26–28, 32, 39, 40]. Although these datasets are all unique, they mostly contain factoid questions that can be answered by short answer spans of only a few words. In addition, none of them contain queries that reference the document directly as the subject of the query, a distinction that can cause existing QA models to yield irrelevant or confusing responses.

Considerable research has targeted neural QA [e.g., 3, 5, 7, 12, 18]. Recently, Devlin et al. [8] introduced BERT, or Bidirectional Encoder Representations from Transformers. BERT is a language representation model that is pretrained to learn deep bidirectional representations from text. A pretrained BERT model can be fine-tuned on a specific task by adding an additional output layer. BERT has made a tremendous impact in many NLP tasks, including QA. In this paper, we base the baseline models on BERT transformers.

Some QA work has focused specifically on the low resource setting that we are also interested in in this work. Various approaches have been applied to augment small datasets to achieve good performance on language tasks ([e.g. 6, 12, 20, 38]). In order to accommodate our low-resource scenario, the data we have collected is supplemented with publicly available QA datasets.

All of the work cited above plays a role in setting context for our scenario. With the possible exception of VERSE, none have specifically explored how people might want to receive conversation-based assistance with documents, and in particular documents that they have rich context about. In the next section, we explore what features and queries users are most likely to pose to their assistant when a document is the focus of the conversation.

3 STEP 1 – SURVEY

In the first step of our research, we aim to answer (**RQ1**) *What kinds of conversational assistance would people like to receive in a document consumption scenario?*, and (**RQ2**) *What kinds of queries might people use to receive this assistance when conversing with a document-aware assistant?* To do so, we conduct a survey to explore the space of queries that people might pose when communicating with a voice assistant about a document, while not having full access to this document. We focus on a consumption scenario while on the go (i.e., limited primarily to voice and some touch input/output). Specifically, participants in our survey are presented with the following scenario: "*You are on your way to a business meeting. To help you prepare, your manager has sent you an email with a document attached. The objective of the meeting is to finalize this document, so that it can be shared with the rest of the organization. Your manager's email also includes the introduction of the document. You have been able to read this introduction, so you have an idea what to expect. You have not read the full document yet, but you can assume the document is approximately 6 pages long. On your way to the business meeting you do not have time to access the document, but you do have your smartphone equipped with a voice assistant like Alexa, Google Assistant, or Cortana. The voice assistant can help you navigate and understand what is written in the document, so that you will arrive prepared at your meeting. The voice assistant can answer your questions via audio or by displaying information on your smartphone screen.*"

3.1 Survey Overview

Our survey consisted of two parts, corresponding to RQ1 and RQ2. In the first part, our primary goal was to explore three sub questions: (1) do users recognize the outlined scenario as relevant to their daily lives?, (2) would users find voice assistance in the outlined scenario helpful?, and (3) what range of features are important to users in a voice-first document consumption scenario? Having identified the range of functionalities that a document-centered conversation might cover, in part two of our survey we aimed to gain a better understanding of the types of questions users might ask. Therefore, we collected questions that are grounded in specific documents. To this end, participants were primed with the same scenario as in the first part. The scenario is simulated by presenting them with an email that mimicked the email they received from their manager while on the go. The email contained the document introduction as a means to give them context about a specific document, to ensure that participants were able to ask informed questions, yet did not have full knowledge about what is written in the document. Figure 2 shows an example of an email provided to participants.

3.2 Participants

Our task was performed by 23 participants in a judging environment comparable to Amazon Mechanical Turk.[1] Participants were all English speaking and U.S.-based. Participants were paid at an hourly rate, removing the incentive to rush responses. We did set a maximum time of ten minutes per document.

Instructions given to participants. Before the task, participants were provided with detailed guidelines of the task and trained to follow them. In these guidelines, we explicitly encouraged participants to ask questions that were document-centered, i.e., to closely keep the outlined scenario in mind when asking questions. The participants were instructed to avoid questions that might be posed about any document, and answered using more mechanical solutions (e.g., *who is the author?*, *how many pages?*), and steered towards a scenario where they imagined having some familiarity with the document subject. Although we acknowledge that these more general questions are highly relevant, we argue that we do not need many sample questions of this type to fully understand the space of potentially relevant mechanical questions. Note that in the first part of the survey we investigated what participants would find the most and least important features in the outlined scenario, and this gives them the opportunity to select more mechanical features. Participants were explicitly told to imagine their ideal voice assistant and to not limit themselves by any prior assumptions about the capabilities of currently existing voice assistants.

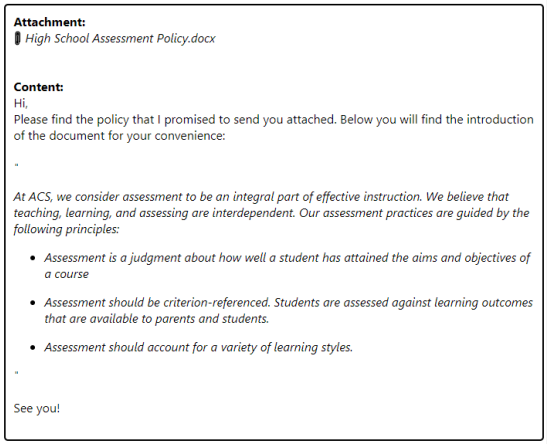


Figure 2: Sample e-mail used to inform participants.

Participant training. Participants performed two training rounds, after which we provided them with feedback on their constructed questions in part two. This way we aimed to ensure that participants understood the task and devise high quality responses.

3.3 Document Selection

We selected 20 documents from a larger data set of 615 documents in Microsoft Word format. These documents were retrieved from a broad crawl of the Web and meet the requirements that they are written in English and can be easily summarized. This last requirement, which was manually verified, ensures that we have a high quality dataset where noisy documents such as online forms are excluded. We selected the 20 documents from this set based on: (1) the document should contain a clear introduction; (2) the document should be between 3 and 10 pages long; and (3) the topic of the document should be understandable for non-experts on this topic and should not be offensive to anyone. Table 1 gives more details on the nature of the selected documents. In addition to these 20 documents, we chose another two documents with which to train the participants. Although slightly deviating from the co-ownership scenario, providing users with documents ourselves allowed us to collect data in a more structured way, which we can use for the remaining research questions at a later stage. In the second part

[1] https://www.mturk.com/

Table 1: Categories of selected documents (20 in total) and their frequency in the survey distributed to participants.

Document category	Document count
Report	3
Job application	3
Description of a service	3
General description	2
Guidelines	3
Policy	3
Informative / Factsheet	3

of the survey, the question collection round, each participant was asked to pose five questions about a given document. We required 20 judges per document. Since we have 20 documents we acquire 400 human intelligence tasks ("HITs"), resulting in 2000 questions.

3.4 Survey Results

In this section, we provide the precise formulation of our survey questions, as well as the participants' responses to these questions.

3.4.1 Part 1 – Survey Questions.

(1) *Do you recognize the outlined scenario (i.e., needing to quickly catch up on a document while on the go) or some variation of it as something you experience in your daily life?*

22 out of 23 participants indicated that they recognized the scenario.

(2) *Do you expect to find it helpful if a voice assistant helps you to quickly familiarize yourself with the document in the outlined scenario?*

22 out of 23 participants indicated that they would find this helpful.

(3) *From the list below, choose three capabilities that you would find* ***most useful*** *in a voice-powered AI assistant to help prepare you for the meeting.*

Participants could choose from the capabilities listed in Table 2. We randomized the order in which the features were presented, to avoid position biases. Note that the prompt specifically references the consumption scenario that participants are primed to consider. The results are given in Figure 3a. Please refer to Table 2 to match the abbreviation on the x-axis with the feature description.

(4) *From the list below, choose three capabilities that you would find* ***least useful*** *in a voice-powered AI assistant to help prepare you for the meeting.*

Again, participants could choose from the capabilities in Table 2 and again this list is randomized for each participant. Figure 3b shows the results for this question. Comparing the results in Figure 3a and Figure 3b shows that participants are very consistent in the capabilities they find most and least useful.

(5) *Can you think of any other features that you would like the voice assistant to be capable of? Please describe.*

We divided the participants' answers into "mechanical" features and "overview" features. A sample of the answers is presented below.

Mechanical features:

- "Voice recognition to unlock phone"
- "Automatic spelling and grammar check"
- "Remind me where I stopped when reading"
- "The ability to link another app, such as maps or notes to the document directly"
- "Bookmarking specific sections for future reference"

Table 2: Assistant capabilities suggested to participants and judged for their utility. Abbreviations were never shown to users and are only used to map plots in this paper to the corresponding capability.

Abbr.	Capability
cut	Cut content from the document using voice
dict	Dictate input to the document
find	Find specific text in the document using voice input
form	Change text formatting using voice
gener	Respond to general questions about the document content, using voice input and output
hilit	Highlight text using voice
ins	Insert new comments into the document using voice
navi	Navigate to a specific section in the document using voice input
paste	Paste content from the device clipboard using voice
read	Read out the document, or parts of it, using voice output
res	Respond to existing comments in the document using voice
rev	Revise a section of text using voice input
send	Send or share a section of text using voice input
sum	Summarize the document, or parts of it, using voice output

- "Another useful feature would be the ability to add highlighted text to multiple programs simultaneously such as email notes and any other app"
- "The Assistant should be able to turn tracked changes on and off and accept/reject changes and clean up a document and finalize"

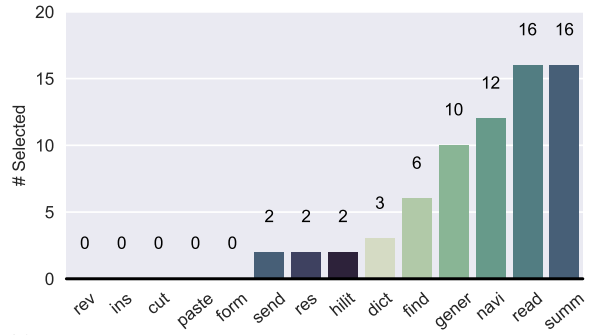
Overview features:

- "Give bullet points of main topics"
- "Give information about key points"
- "Just highlight key points, summarize document"
- "I would like for the voice assistant to be able to pick out the main points and read them out to me via voice output"
- "If the assistant was able to give a synopsis then ask 1 or 2 questions to be sure the user understands the info"

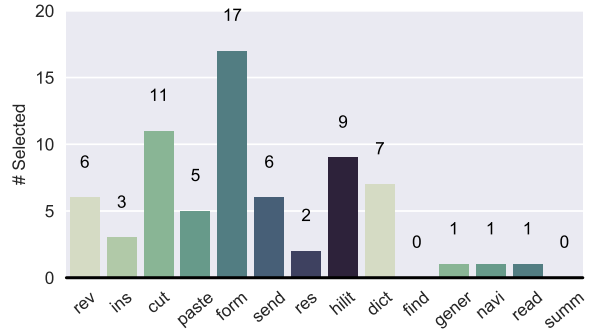
3.4.2 Part 2 – Collecting Questions. Here we present the results of the second part of the survey, in which participants were prompted to generate questions about a document. Recall that the participants were only shown the document introduction or preamble and did not have visibility into the full document text.

(6) *Please ask five questions to your voice assistant that would help you understand what is written in the document.*

We can divide participants' answers into a hierarchy of question categories. Note that the responses can be both questions and directives (e.g., "*go to Section X*"). Since the vast majority of the collected responses are questions, for brevity we refer to both of these response types as *questions*. Figure 4 shows the hierarchy. It was developed by sampling a set of participants' questions, which an expert studied and categorized. Three experts then reviewed all questions and categorized them according to the proposed taxonomy. By reviewing where the experts disagreed, some minor adjustments were made to the hierarchy to arrive at the final one shown here. Level 1 of the hierarchy corresponds to how the question can be best responded to, or what kind of system or model



(a) Responses to question 3 – most useful assistant capabilities.



(b) Responses to question 4 – least useful assistant capabilities.

Figure 3: Most and least useful assistant capabilities; names explained in Table 2. On the y-axis: the number of times this particular capability was selected by participants (max = 23).

would be suited best to handle the questions. Because document-centered questions are the main interest of our current research, we divide those into another set of categories, describing the intents of users on this level in more detail. This is level 2. We also subdivide the yes / no questions into the rest of the categories of level 2 and call this level 3. We do this because it is questionable whether a person would really be satisfied with a simple "*yes*" or "*no*" in response. We describe the question types in Table 3, and also provide verbatim examples sourced from the participants' responses. Figure 5a shows the distribution of question categorizations on level 1. Document-centered questions form the largest category of the questions. Recall that participants had to ask 5 questions per document; we investigated whether these questions differed in type. E.g., did participants ask mechanical questions first ("*bring me to Section 2.*") and then a document-centered question ("*what does it say there about X?*")? We did not find such a difference. We also investigated whether the type of document (Table 1) was an indication for the types of questions that were asked, but found no difference between document types. The user was a strong indication for the type of question that was asked, indicating varying interpretations of the outlined scenario. Some users ask only factoid questions, some users only ask document-centered questions and only a few ask a mixture of all question types.

The division of category labels for level 2 is shown in Figure 5b. As can be seen, the majority of questions are closed form yes / no questions. Figure 5c shows how these questions were categorized on level 3, yielding only 3 copy-editing questions, 2 overview questions, and 1 navigational question, rounding down to 0% in Figure 5c.

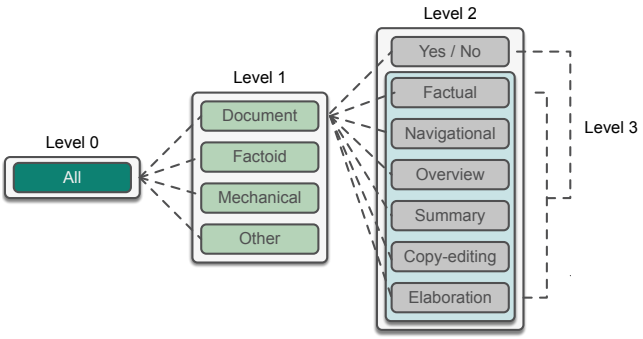


Figure 4: Question hierarchy.

3.5 Classifying Question Types

We trained a simple, yet effective logistic regression classifier to classify the question types. From Table 4 it becomes clear that we can accurately learn to classify different question types, especially at higher levels in the hierarchy. These labels are extremely helpful for a number of tasks: they are useful to decide what type of answer the user is expecting, or the type of model that should deliver a response. An accurate classification on the first level is important for this task: do we want to use a rule-based system, a factoid QA model, or a newly trained document-centered QA model? The results on the second level can be used to decide whether or not we face a yes / no question and therefore may have to start the answer with "yes" or "no." In a question generation setting, the labels can also be used to condition the question generation process.

3.6 Answering RQ1 and RQ2

The results of the survey allow us to answer our first two research questions. We have identified a range of capabilities that users would like to see in a document-centered assistance scenario, and we have identified a hierarchy of questions that users would ask. Document-centered questions are different from factoid QA questions and form an interesting new category of questions to research.

4 STEP 2 – DATA COLLECTION

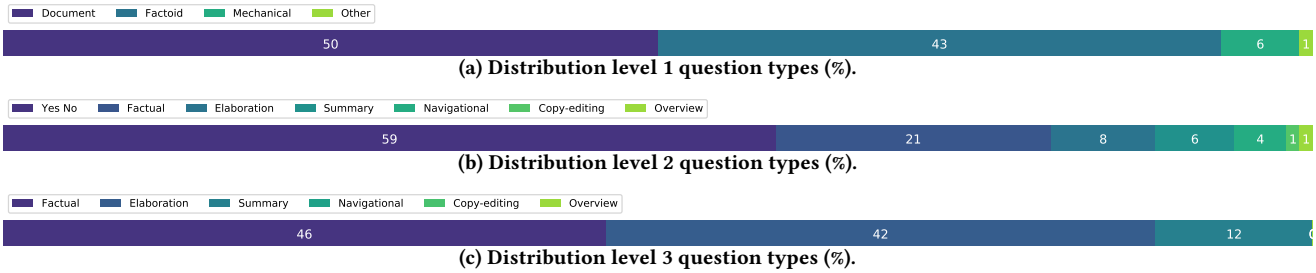
The first step of our work shows that users pose different types of questions to a digital assistant when seeking document-centered assistance than are typically present in modern QA datasets. To dive deeper, we first scale up our data collection to gather more questions and proposed answers to those questions. In this section, we describe our data collection process and the statistics of the collected data. We refer to the collected data as "DQA" dataset, short for Document Question Answering.

4.1 Question Collection

For question collection, we randomly selected another 36 documents (recall Section 3.3) using the same selection criteria. We asked the same set of participants as in Step 1, now acting as crowd workers, to generate questions for these documents. This time we omitted the survey questions about the scenario and capabilities; we asked them to pose five questions about the document. Since we only presented the workers with the document introduction, it is likely that workers will also ask questions that cannot be answered from the document, more closely resembling a real life situation.

Table 3: Question type descriptions and examples.

Level	Question type	Examples
L1	**Document:** These are document-centered questions. That is, the question's phrasing explicitly or implicitly references the document. When asking such a question, a user is not looking for encyclopedic knowledge, yet rather for assistance that can help them to author the document. These types of questions are not present in existing QA datasets.	Does the document have specifications to the type of activity and sector improvement that will be offered?
	Factoid: Fact-oriented question that co-owners of a document are unlikely to ask. Answers are often only a few words long. Existing QA datasets cover these types of questions very well.	What is the date of the festival?
	Mechanical: Questions that can be answered with simple rule-based systems.	Highlight "Capability workers"
	Other: Questions that fall outside the above categories.	Read the email to me.
L2	**Yes / No:** Closed form (can be answered with 'yes' or 'no').	Does the document state who is teaching the course?
	Factual: Questions that can be answered by returning a short statement or span extracted from the document.	Where does the document state study was done?
	Navigational: Referring to position(s) in the document.	Go to policies and priorities in the doc.
	Overview: Questions that refer to the aim of the document.	What is the overall focus of the article?
	Summary: Questions that ask for a summary of the document or of a particular part of the document.	Find and summarize coaching principles in the document.
	Copy-editing: Questions when editing a document. They require a good understanding of the document to answer.	Highlight text related to application of epidemiologic principles in the document
	Elaboration: Questions that require complex reasoning and often involve a longer response.	Please detail the process to get access to grant funds prior to confirmation.



(a) Distribution level 1 question types (%).

(b) Distribution level 2 question types (%).

(c) Distribution level 3 question types (%).

Figure 5: Distribution of question types per hierarchical level. (Best viewed in color.)

Table 4: Question type classification results. Mean accuracy and variance after 5-fold cross validation.

Level 1	Level 2	Level 3
0.92 ($\pm 8.6e^{-5}$)	0.90 ($\pm 1.3e^{-4}$)	0.67 ($\pm 1.0e^{-3}$)

4.2 Answer Collection

Once we collected the questions, we asked the same pool of crowd workers to select answers for these questions. We presented workers with the full document and asked them to read it carefully. Then we asked them to answer five questions about the document. These questions were always a set of five questions that were asked by one of the crowd workers in the question collection round (not necessarily the same as the worker who is answering the questions). The questions were kept together and were presented in the same order as they were asked, due to the potential conversational nature of the questions. Note that this is only applicable to a few instances in the data, allowing us to train a single-turn QA model later. Each set of questions is answered by three crowd workers. An overview of the presented task is included in the supplementary material.

For each question, we display the following options after a click on the question: (1) This question or directive does not make sense; (2) The document does not contain the answer to this question; and (3) Please indicate the question type: (a) This is a yes / no question, (b) This is not a yes / no question. If a worker selects that the question is a yes / no question, we ask them to indicate whether the answer is "yes" or "no" and to select parts of the document with supporting evidence. If no supporting evidence could be found in the document (e.g., because the question was *"does the document contain information about topic X?"* and the answer was "no") we

Table 5: Answer and question types.

	Number	% (of total)
Annotated documents	56	–
Valid questions (= annotation tasks)	16375	100.00
Invalid questions (discarded)	425	–
Open questions	9442	57.66
Yes/no questions	6933	42.34
No answer	6543	39.96
No evidence	1748	25.21

asked workers to tick the box that supporting evidence cannot be highlighted. An example of the task including the expansion that is shown if a worker selects that the question is a yes / no question is given in the supplementary material. If the worker has not clicked any of the above mentioned options, it means the question is valid, open-ended, and answerable. For these questions, we asked workers to select the minimal spans of text necessary to answer the question. Workers could select up to three spans in the document; each span was at most 700 characters in length. Since some documents can be challenging to understand, we included a checkbox to indicate that the questions were difficult to answer or the document was hard to understand. Figure 6 shows an example of the highlighting tool. Text highlighted in the document (right-hand pane), is populated as a selected span in the left-hand pane (blue box).

We again performed 2 training rounds with the crowd workers, in which we ensured workers fully understand the task. During the data collection phase an expert spot-checked answer quality.

4.3 Dataset Statistics

Table 5 describes the distribution of annotations about the questions that were collected from the crowd workers. Recall that each question was judged and answered by 3 workers. Here we present the raw numbers.

During the question generation phase workers were not shown the full document, whereas the workers have access to the full text while selecting answers. This disparity is reflected in the statistic that 40% of questions were considered unanswerable from the text. This ensures that our dataset is suitable for training a system that can identify unanswerable questions.

Table 6 gives an overview of the number of spans and the lengths of spans that were selected by crowd workers. The average span length is substantially larger than the average span length of only a few words in most existing QA datasets. This supports our claim that the current document-centered scenario requires different types of data to train on. Table 7 describes the distribution of annotation responses, in particular the fraction of questions where workers were in full agreement about the impossibility of answering a question from the text (52%) (random full agreement would be 25%), as well as ROUGE-scores describing the mean self-similarity of selected spans across judges who responded to the same question. Hence, participants agreed well with each other.

5 STEP 3 – BASELINE MODELING

We present baseline models for passage retrieval and answer selection on our dataset. Our aim is to answer (**RQ3**) *How well do initial baseline models do in a document-centered scenario?*

Table 6: Span statistics. Span length in tokens.

	Statistic
Total spans	11702
Average number of spans per question (all)	0.715
Average number of spans per question with answer	1.45
Average span length per question (all)	26.69
Average span length per question with answer	37.35

Table 7: Agreement statistics.

	Metric	
Impossible full agreement (%)	52.09	
Impossible partial agreement (%)	47.91	
Rouge-1 F-score avg (questions with span)	52.44	(±8.79)
Rouge-2 F-score avg (questions with span)	44.92	(±11.14)
Rouge-L F-score avg (questions with span)	46.89	(±9.54)

5.1 Data Preprocessing

We use exactly the same format as the popular SQuAD2.0 [27] dataset for our preprocessing output. We keep all questions and answers for a random sample of 25% of the documents as a separate hold-out set. Recall that we have collected 3 answers per question, as we had 3 workers answer each question. We discarded all invalid questions and we ensured that the remaining labels (such as "yes / no questions") were consistent as follows. First we looked at workers' answers for whether the question was a yes / no question and computed the majority vote. We kept the answers of the workers who agreed with the majority vote and discarded the rest (if any). The majority vote has been shown to be a strong indication for the true label [21]. In case of a tie, we chose to treat this question as a yes / no question as it provided us with most information about the question, which is beneficial for training. If the question is now labeled as a yes / no question we continue to the answer (i.e., "yes" or "no"). Again we computed the majority vote and only kept the answers from workers who agreed with the majority vote. In case of a tie we chose "yes" as the answer, as this results in the richest label for the question. Then we followed the same procedure for the "no-evidence" checkbox, choosing to include spans in the event of a tie. Lastly, if the question was not labeled as a yes / no question, we applied the same majority vote and tie-breaking strategy for whether the document contains the answer. Using this approach, we kept approximately half of the collected question-answer pairs, but ensured that no model is trained on contradictory answers. This improved model performance. During training, we used the collected question-answer pairs as individual training examples, i.e., if we have 2 answers for a question given by 2 workers, we added them separately to our training set. This way we increased the number of training samples. At this stage, we also chose to add all selected spans for an answer separately to our training set. We leave multiple span selection for future work. During evaluation we treated all selected answers for a question as valid answers.

5.2 Passage Ranking

In this section, we describe our approach for initial passage ranking experiments on our new DQA dataset. We explore three baseline methods: random selection, BM25-based ranking, and selecting the first passage in the document.

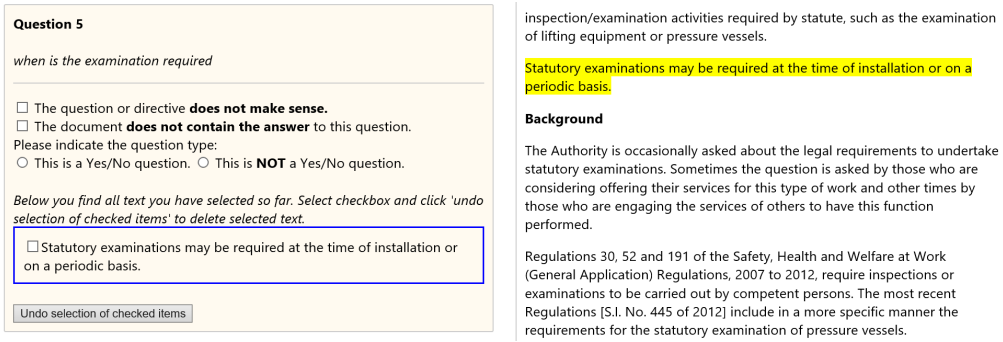


Figure 6: Question answering data collection selected text. (Best viewed in color.)

5.2.1 Passage Construction. During data collection, crowd workers selected answers to questions, yet they did not select the paragraphs or passages that include these answers. Therefore we constructed passages for all questions with answers as follows. We discard questions without answers in this experiment. We split each document in the dataset into sentences. We adopted a sliding window approach, moving our window one sentence at the time, constructing passages of size *window size*. We set the window size to 5. We also divided the selected answers into sentence chunks (or smaller, if only parts of sentences were selected). For each answer, we scored each passage by the number of chunks it contains. That is, a passage received a point for each chunk that is also in the answer.

5.2.2 Baseline Passage Ranking 1 – Random. For this baseline we retrieve a random passage. For each retrieved passage we compute the ROUGE-1 F-score, ROUGE-2 F-score, and ROUGE-L F-score (based on retrieved passage and ground truth) [22] and the *Precision@1*. Recall that we scored paragraphs based on the number of overlapping chunks with the selected answer. Therefore some paragraphs contain only part of the answer, and some contain the full answer. To account for this difference we computed a so-called *hard* and *soft Precision@1*. For the hard version, we assigned binary labels to retrieved passages; 1 if the retrieved passage contains (part of) the answer, 0 if it does not. For the soft version, we scored each retrieved passage as follows: we took the number of overlapping chunks of the retrieved passage and the answer and divided this by the maximum number of overlapping chunks. Since annotators may select answers from different passages, we optimistically took the best passage score per question, i.e., we returned a valid match if the selected passage matched any annotator response.

5.2.3 Baseline Passage Ranking 2 – First passage. For this baseline, we select the document's first passage as an answer to each question. We compute the same metrics as in Baseline 1. The purpose of this baseline is to establish to what extent answers to questions are biased by their presence in the preamble of the document, which was shown to study participants at question generation time.

5.2.4 Baseline Passage Ranking 3 – BM25. For this baseline, we retrieve the best matching passage with BM25 [30] and compute the same metrics as in Baselines 1 and 2.

5.3 Results for Passage Ranking

In Table 8 the results for the passage ranking experiments are shown. Analysis of variance (ANOVA), $F(2, 54) > 8.9$, $p < 0.0002$ yields significant differences between the three approaches. A post-hoc

Table 8: Results for passage ranking.

Model	P@1 soft	P@1 hard	Rouge-1 F-score	Rouge-2 F-score	Rouge-L F-score
Random	0.29	0.31	0.26	0.13	0.19
First	0.55	0.56	0.32	0.20	0.23
BM25	0.32	0.34	0.29	0.16	0.22

Tukey test $p < 0.05$ shows that first passage selection significantly outperforms Random for all measures, and BM25 for all measures except ROUGE-L. BM25 significantly outperforms Random only for ROUGE-L. We hypothesize that the performance of first passage selection can have a number of causes: (1) because workers have been shown the introduction of the document, many questions can be tailored towards information located in the introduction, (2) workers have read the document from beginning to end, which may have biased them towards selecting from the first part of the document and not from the later parts once they found the answer.

5.4 Answer Selection

In this section, we discuss how state-of-the-art models for answer selection perform on the DQA data and DQA enhanced with data from the SQuAD2.0 dataset [27]. We select this dataset for two reasons: first, it is a standard dataset for benchmarking Question Answering tasks and, second, like DQA, it contains questions marked as unanswerable, making it closely compatible with our collected data. All baselines were evaluated using the DQA hold-out set.

5.4.1 Passage Construction. For the answer selection experiments, we selected the passages for each answer using the same windowing method as in the passage ranking experiments. The only difference is that we now only considered passages that contain the full answer. For unanswerable questions, we selected the best matching paragraph with BM25. Even though our previous experiments showed that the answer is often in the first paragraph, we chose BM25 as a less biased and more informed selection procedure.

5.4.2 Baseline Answer Selection 1 – Fine-tuned BERT on SQuAD2.0. For QA, BERT is fine-tuned as follows. A question and a passage are fed to a pre-trained BERT language model. They are separated with a separator token. The final output layer is trained to select the start and end index of the answer, from the input passage. If no answer is detected in the passage, 0 is selected as index for both start and end. For the current baseline we fine-tuned HuggingFace's implementation of BERT Large [36] on 8 Titan XP GPUs, using SQuAD2.0. First, we ensured we got similar scores as reported in

the repository for the SQuAD2.0 tasks. Then, we evaluated the model on the DQA hold-out set. We included this baseline to test how a pretrained and fine-tuned BERT model on a very popular QA dataset performed on our DQA dataset without any adaption.

5.4.3 Baseline Answer Selection 2 – Fine-tuning on SQuAD2.0 with query rewriting. For this baseline, we used the same fine-tuned BERT model as for Baseline 1, yet this time we performed some simple query rewriting on the hold-out set to make our questions more comparable to those the model is fine-tuned on. For query rewriting, we computed the most common n-grams in our document train set. We manually inspected those n-grams and chose to delete the following document and conversational related patterns from our questions, expressed as Python regular expressions:

- `'^does( the)? document (\S)+ (you)? '`
- `'^does it (\S)+ '`
- `'^what does( the)? document (\S)+ (you)? ')`
- `'according to( the)? document(\s,\s|,\s|\s)')`
- `'in( the)? document '`
- `'^assistant, '`

5.4.4 Baseline Answer Selection 3 – Fine-tuning on DQA. For this experiment, we fine-tuned BERT Large using the DQA dataset, again used the same fine-tuning implementation as used previously. We evaluated on the DQA hold-out dataset.

5.4.5 Baseline Answer Selection 4 – Fine-tuning on DQA with query rewriting. This experiment resembles Baseline 3, but used the same query rewriting as in Baseline 2 to the train and the hold-out set.

5.4.6 Baseline Answer Selection 5 - Fine-tuning on SQuAD2.0 & DQA. This baseline resembles Baseline 1, but now we added our data to the existing SQuAD2.0 data set while fine-tuning the BERT Large model. We did this since our DQA dataset is not very large. We expected an improvement in performance when we enhanced our data with more data points. We shuffled the training input randomly. We evaluated on the DQA hold-out set.

5.4.7 Baseline Answer Selection 6 - Fine-tuning on SQuAD2.0 & DQA with query rewriting. This baseline resembles Baseline 5, but we performed the same query rewriting to DQA part of the train set and to the DQA hold-out set as in Baselines 2 and 4.

5.5 Results for Answer Selection

Table 9 shows the results for the answer selection experiments. Fine-tuning BERT on SQuAD2.0 and the DQA data significantly outperforms the other baselines. These results look promising but reveal an interesting new problem to work on as the scores are significantly lower than we are used to from the typical QA task leader boards such as the SQuAD2.0 challenge. It is interesting to see that query rewriting is not beneficial. We assume that our approach may have been too simplistic. We would like to experiment with different types of query rewriting in future work (e.g., [13, 41]).

5.6 Answering RQ3

We have shown that the initial baseline models perform reasonably well on our new document-centered domain. For the answer selection task, it is beneficial to add data from the Wikipedia domain (SQuAD2.0) during training. This improves the results, but also shows that document-centered assistance is a very different novel domain. While our initial experimental results are promising, there is still plenty of opportunity to improve the models in future work, for example by increasing the dataset size. We also expect improvements if we would train BERT on data similar to the DQA data. As BERT has been trained on the Wikipedia domain – the same domain as the SQuAD2.0 data – BERT could 'memorize' certain parts of the data during training, which could give an advantage when fine-tuning on the SQuAD2.0 data. DQA does not have this advantage. In some specific scenarios, using the meta-structure of the document might help to improve results. However, we consider not relying on this structure as the preferred option since this allows us to generalize quickly over a wide variety of documents.

Table 9: Results answer selection. All models fine-tuned BERT Large and were evaluated on the DQA hold-out set. AS means Answer Selection. AS 5 significantly outperforms the other baselines (Wilcoxon Signed-rank, $p < 0.001$).

Baseline	Training source	F1	EM
AS 1	SQuAD2.0	27.24	13.21
AS 2	SQuAD2.0 with Eval Query rewriting	26.79	13.09
AS 3	DQA	38.84	18.93
AS 4	DQA with Query rewriting	36.73	17.83
AS 5	SQuAD2.0 + DQA	**41.02****	**20.30****
AS 6	SQuAD2.0 + DQA with Query rewriting	37.28	18.52

6 CONCLUSIONS AND FUTURE WORK

In this paper, we explored the novel domain of document-centered digital assistance. We focused on a consumption scenario, in which individuals are a (co-)owner of a document. Through a survey, we identified a set of primary capabilities people expect from a digital assistant in a document-centered scenario, as well as a large set of questions that gave us insight into the types of queries that people might pose about a document when they have an approximate or good idea what the document is about. Our explorations shed light on the hierarchy of questions that might be posed, and demonstrate that the types of questions people ask in a document-centered scenario are different from the factoid questions in conventional QA datasets. We show that state-of-the-art QA models can be fine-tuned to perform with reasonable accuracy on the new DQA data. Yet, it has proven to be an unsolved task, which makes this a fertile area for future work. This research opens a new direction for digital assistance. Avenues for future work include deeper explorations of query rewriting to better tailor document-centered questions to conventional QA systems, and also exploring ways to scale up the data to a much larger and broader range of documents.

ACKNOWLEDGMENTS

We thank Julia Kiseleva for her helpful feedback. This research was supported by Ahold Delhaize, the Association of Universities in the Netherlands (VSNU), the Innovation Center for Artificial Intelligence (ICAI), and the Nationale Politie. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Qingyao Ai, Susan T Dumais, Nick Craswell, and Dan Liebling. 2017. Characterizing email search using large-scale behavioral logs and surveys. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 1511–1520.

[2] Vikas Ashok, Yevgen Borodin, Yury Puzis, and IV Ramakrishnan. 2015. Capti-speak: a speech-enabled web screen reader. In *Proceedings of the 12th Web for All Conference*. ACM, 22.

[3] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075* (2015).

[4] Horatiu Bota, Adam Fourney, Susan T Dumais, Tomasz L Religa, and Robert Rounthwaite. 2018. Characterizing Search Behavior in Productivity Software. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval*. ACM, 160–169.

[5] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051* (2017).

[6] Jeanne E Daniel, Willie Brink, Ryan Eloff, and Charles Copley. 2019. Towards Automating Healthcare Question Answering in a Noisy Multilingual Low-Resource Setting. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 948–953.

[7] Mostafa Dehghani, Hosein Azarbonyad, Jaap Kamps, and Maarten de Rijke. 2019. Learning to Transform, Combine, and Reason in Open-Domain Question Answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. ACM, New York, NY, USA, 681–689. https://doi.org/10.1145/3289600.3291012

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[9] Linda Di Geronimo, Maria Husmann, and Moira C Norrie. 2016. Surveying personal device ecosystems with cross-device applications in mind. In *Proceedings of the 5th ACM International Symposium on Pervasive Displays*. ACM, 220–227.

[10] Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179* (2017).

[11] Adam Fourney and Susan T Dumais. 2016. Automatic identification and contextual reformulation of implicit system-related queries. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 761–764.

[12] Wee Chung Gan and Hwee Tou Ng. 2019. Improving the Robustness of Question Answering Systems to Question Paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 6065–6075.

[13] Mihajlo Grbovic, Nemanja Djuric, Vladan Radosavljevic, Fabrizio Silvestri, and Narayan Bhamidipati. 2015. Context-and content-aware embeddings for query rewriting in sponsored search. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. ACM, 383–392.

[14] Tero Jokela, Jarno Ojala, and Thomas Olsson. 2015. A diary study on combining multiple information devices in everyday activities and tasks. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 3903–3912.

[15] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551* (2017).

[16] Amy K Karlson, Shamsi T Iqbal, Brian Meyers, Gonzalo Ramos, Kathy Lee, and John C Tang. 2010. Mobile taskflow in context: a screenshot study of smartphone usage. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009–2018.

[17] Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics* 6 (2018), 317–328.

[18] Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. 2019. RankQA: Neural Question Answering with Answer Re-Ranking. *arXiv preprint arXiv:1906.03008* (2019).

[19] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics* 7 (2019), 453–466.

[20] Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised Question Answering by Cloze Translation. *arXiv preprint arXiv:1906.04980* (2019).

[21] Yuan Li. 2019. *Probabilistic models for aggregating crowdsourced annotations*. Ph.D. Dissertation.

[22] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*. Association for Computational Linguistics, Barcelona, Spain, 74–81.

[23] Victor Ei-Wen Lo and Paul A Green. 2013. Development and evaluation of automotive speech interfaces: useful information from the human factors and the related literature. *International Journal of Vehicular Technology* 2013 (2013).

[24] Nikolas Martelaro, Jaime Teevan, and Shamsi T Iqbal. 2019. An Exploration of Speech-Based Productivity Support in the Car. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 264.

[25] Microsoft. 2019. Voice report. From answers to action: customer adoption of voice technology and digital assistants. https://advertiseonbing-blob.azureedge.net/blob/bingads/media/insight/whitepapers/2019/04%20apr/voice-report/bingads_2019_voicereport.pdf. Accessed: 2019-12-04.

[26] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human-Generated MAchine Reading COmprehension Dataset. (2016).

[27] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. *arXiv preprint arXiv:1806.03822* (2018).

[28] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250* (2016).

[29] Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics* 7 (2019), 249–266.

[30] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

[31] Chinnadhurai Sankar and Sujith Ravi. 2018. Modeling non-goal oriented dialog with discrete attributes. In *NeurIPS Workshop on Conversational AI:"Today's Practice and Tomorrow's Potential*.

[32] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830* (2016).

[33] Peter Tsai. 2018. Data snapshot: AI Chatbots and Intelligent Assistants in the Workplace. https://community.spiceworks.com/blog/2964-data-snapshot-ai-chatbots-and-intelligent-assistants-in-the-workplace. Accessed: 2019-10-04.

[34] Alexandra Vtyurina, Adam Fourney, Meredith Ringel Morris, Leah Findlater, and Ryen White. 2019. VERSE: Bridging Screen Readers and Voice Assistants for Enhanced Eyes-Free Web Search. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility. ACM*.

[35] Alex C Williams, Harmanpreet Kaur, Shamsi Iqbal, Ryen W White, Jaime Teevan, and Adam Fourney. 2019. Mercury: Empowering Programmers' Mobile Work Practices with Microproductivity. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology. ACM Press, New Orleans, Louisana, USA. https://doi. org/10.1145/3332165.3347932*.

[36] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv* abs/1910.03771 (2019).

[37] Rui Yan and Dongyan Zhao. 2018. Coupled context modeling for deep chit-chat: towards conversations between human and computer. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2574–2583.

[38] Wei Yang, Yuqing Xie, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. Data Augmentation for BERT Fine-Tuning in Open-Domain Question Answering. *arXiv preprint arXiv:1904.06652* (2019).

[39] Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 2013–2018.

[40] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600* (2018).

[41] Wei Vivian Zhang, Xiaofei He, Benjamin Rey, and Rosie Jones. 2007. Query rewriting using active learning for sponsored search. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 853–854.