The Effect of Multi-Document Summarizations on User SERP Experience

Koyena Pal

Thesis Advisor: Dr. Carsten Eickhoff

Thesis Reader: Dr. Jeff Huang

A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science with honors

Department of Computer Science
Brown University
Providence, Rhode Island

April 2021

Contents

Acknowledgements

First and foremost, I would like to thank my advisor, Professor Carsten Eickhoff for his advice and support since junior year. This project would not exist without his expertise and guidance. I would also like to thank my second reader, Professor Jeff Huang, who has been incredibly helpful throughout my thesis journey, especially during my user study setup phase. Speaking of the user study, I would also like to thank Jerome Ramos for taking the time to explain and help me understand his past project, which has been extremely helpful for my thesis preparation that branches from his work.

I must also express my deep gratitude to my family back at home, who have been my unconditional supporters and well wishers for all my life. I would also like to extend my thanks to my friends and mentors in the Computer Science community for everything they have done to support me through this process and for making Brown a home away from home.

Lastly, Karen, Elizabeth, Abbie, and Cat – I could not have asked for a more amazing quarantine pod. Thank you for your support throughout, and especially, during the pandemic.

Abstract

A featured snippet is an element of commercial Search Engine Results Pages (SERP) that appears on the top as a small text box. It describes the single most relevant page and aims to directly answer a user's query. Past studies have shown that users are comfortable and satisfied with this setup to answer their queries effectively. However, this feature appears occasionally and does not always cover multiple concepts related to a query. Hence, in this paper, we present an investigation of user experience when different degrees of summary snippets are provided on SERPs. The degrees are based on the number of top query-relevant websites that are inputted to generate the summary snippets. Our user study demonstrates that the inclusion of summary snippets in SERPs leads to better search success and has promising results to increase transparency, user trust, and search efficiency for various types of queries.

1. Introduction

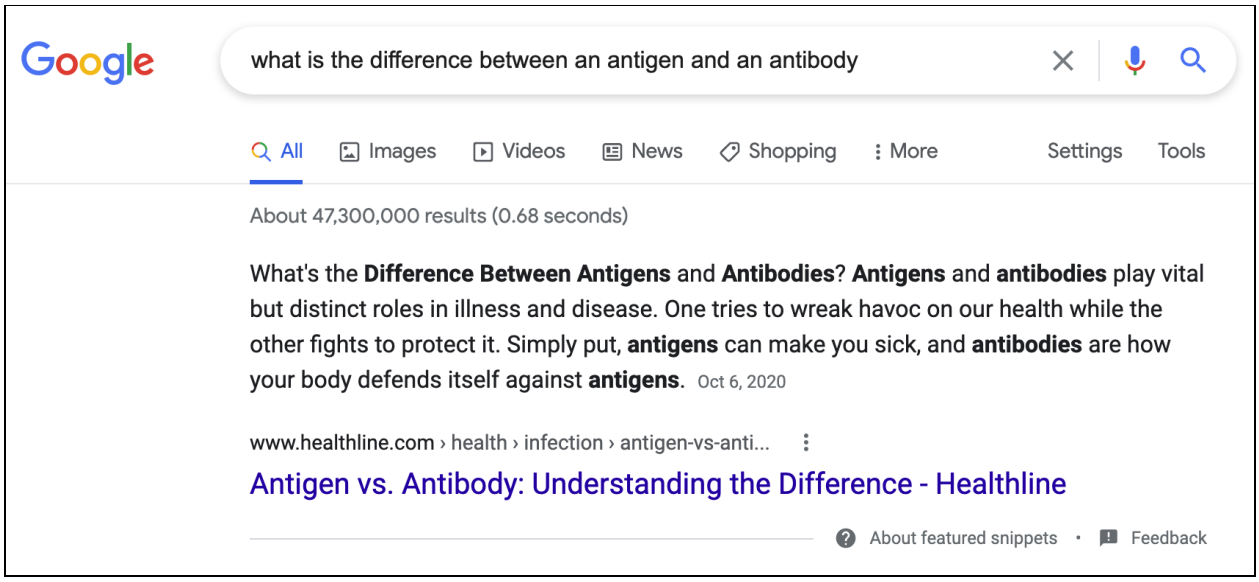

Figure 1. Example of a featured snippet

When it comes to finding essential information online, we often type our queries on a search engine and interact with the consequent Search Engine Results Page (SERP). The time it takes to gather and understand relevant information depends on two main aspects of a search engine – the retrieval model and the search interface.

The retrieval model focuses on accurately ranking documents in terms of relevance to the query. The search interface, on the other hand, prioritizes on presenting results of the retrieval model in an effective and user-friendly manner. Google, for instance, has improved its user SERP experience by introducing featured snippets [1], which are short text blocks that appear on the top of search results to quickly answer a user's query directly on the SERP. Figure 1 showcases an example where we query 'what is the difference between an antigen and an antibody' and the Google SERP responds with a featured snippet at the top that describes a possible answer, which most relates to the question. By doing so, this search engine was able to answer our query without having us look at the rest of the SERP or visiting any of the ranked documents.

However, such featured snippets do not appear for every query. For instance, if we type ‘round trip RI and NJ travel restrictions,’ a featured snippet answering this query does not show up.

In this paper, we create SERPs that contain summary snippets for those informational search queries that do not currently have featured snippets on a commercial Search engine. In addition, in a crowdsourced user study, we measure their effect on user SERP experience. The snippets are generated using data from the top *n* documents for a given query. We experimented with *n* values to also look into which *n* value would be the most suitable for generating summary snippets. Therefore, in this paper, we explore the following research questions:

1. How does the top *n* value affect the relevance and effectiveness of generated summarized snippets?
2. How do summarized snippets affect user search in terms of a) perceived transparency, b) user trust and c) search efficiency?

The remainder of this paper is structured as follows: Section 2 discusses related work on SERP and summarization algorithms. Section 3 introduces our summary snippet generation algorithm and proposed search interface. Section 4 describes the setup of our user study while Section 5 showcases the results gathered from that study. We continue discussing the implications of the results in Section 6. Finally, in Section 7, we conclude with an outlook of future directions of our research.

2. Related Works

We present two distinct types of work that contribute to the field of SERP and retrieval models. In Section 2.1, we review findings that evaluate current SERPs as well as new features added to the SERP. In Section 2.2, we present an overview of various summarization algorithms, which relate to this work in terms of how well we can generate a summary snippet by implementing them.

2.1 Search Engine Results Page

With respect to evaluating the SERP, Ramos et al. [2] explore the various ways that ranked results can be displayed – having the title of the website only, titles with simple explanation of the website, or titles with explanation and bars that explain how each query term contributes to the overall score of the document content. The study concludes that the inclusion of explanations, even simplistic ones, leads to better transparency, increased user trust, and better search efficiency. Many search engines like DuckDuckGo and Google Search employ this outlook by listing their search result links with text blocks that preview the content in those websites. Another study conducted by Jiang et al. [3] looks into the effects of SERP information in the academic search context. This information includes featured snippets, publication date, and citation count. Based on their user study, it was concluded that more users were satisfied with SERPs displaying snippet information than displaying other elements regardless of their familiarity with the search tasks' topics. This SERP display is used by many search engines such as DuckDuckGo and Google Search. To be consistent with most search engines, we also present the result links in our user study in the same manner.

Although featured snippets can assist in answering our queries, the current framework usually describes a page of the most relevant website, rather than a set of relevant websites [1]. Hence, when there are multiple concepts for a given topic, snippets can be limiting in terms of coverage. This observation is evident in Aqle et al.'s study [4] where the authors compared their search

interface, InteractSE, with Google Search's interface and found that visually impaired users had better satisfaction and completion time with InteractSE. This search interface groups search results based on discovered concepts and presents this grouping in a tree-structure, which provides an overview of these results and allows users to navigate through search results based on the concept they were searching for. In our study, we modify the framework of the featured snippets, such that it presents a summary based on some number of top relevant websites. By doing so, we attempt to answer the query by combining the content contained in multiple websites. As a result, we integrate the conceptual idea behind Aqle et al.'s study through our summary snippet generation, which is described in detail in Section 3.1.

2.2 Summarization Algorithms

Extensive research has been done on text summarization algorithms. We can generally characterize them according to the number of documents used, how the information was gathered, and the algorithm learning method. Summarization algorithms can either consume one ('single-document') or multiple ('multiple-document') documents during the summarization process. If these algorithms can only summarize documents from a certain field, we call them specific summarization algorithms. Otherwise, they are known as generic summarization algorithms. Since we want to create summary snippets for a variety of topics, we focused on generic summarization algorithms. Amongst generic summarization algorithms, we looked into multi-document algorithms as they can integrate key information from multiple text documents and build a brief report, which single-document does but for one text document. Hence, we focused our summarization algorithm search on multi-document generic ones. Amongst these algorithms, there are two types – extractive and abstractive.

Extractive summarization algorithms gather sentences from the text data that they are inputted with, which are then concatenated together to form a summary. There have been both supervised and unsupervised cases for this type of algorithm. Gupta et al. [5], for instance, create a supervised multilingual extractive summarizer that uses nine features to pick sentences that are most relevant to the input documents. On the other end of the spectrum, we have unsupervised algorithms like Uçkan et al.'s algorithm [6], KUSH, which is an extractive, generic

summarization that inputs unlabeled text data and outputs a summary with the help of graphed independent sets.

Abstractive summarization algorithms, on the other hand, create summaries by generating sentences based on their input data. Such an algorithm allows for us to have the ability to control the summary size in terms of words as well as sentences. The Pointer-Generator Network [7], for instance, is a self-supervised algorithm that is based on a neural sequence-to-sequence model and has a transformer architecture and attention mechanism to learn about the documents inputted and to generate a summary. In our study, we use an abstractive summarization algorithm named SummPip [8], which is an unsupervised algorithm that constructs sentence graphs by incorporating both linguistic knowledge and deep neural concepts, which are then used to generate a summary. We explain more about this algorithm in Section 3.1.

3. Methods

To create a SERP with summary snippets for a given query result, we first implement a mechanism that helps generate the summaries. In Section 3.1, we explain this step in more detail. After creating summaries, we build a SERP that has the summary text at the top of the page in a box with useful headers and footers to help users navigate through the page. The second subsection, Section 3.2, describes this interface specifically.

3.1 Summary Generation

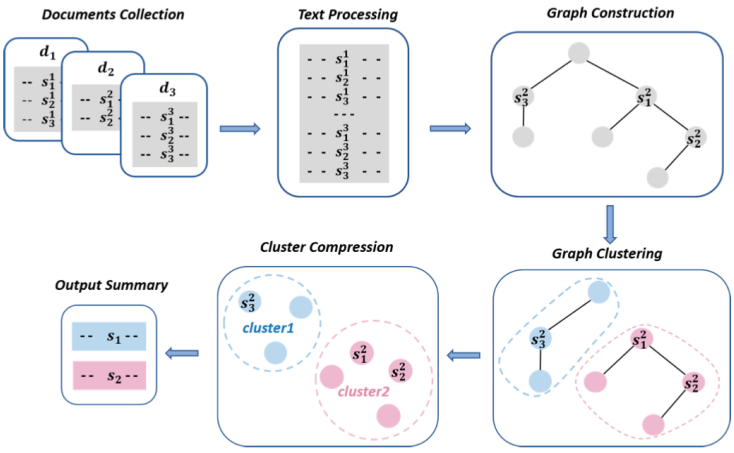

Figure 2. SummPip Framework [8]

The SummPip algorithm [8] is primarily implemented to generate the summary snippet itself. As shown in Figure 2, SummPip takes in original documents, D, which is a set of documents (in this case, webpages) denoted as d_1, d_2, d_3 and so on, and has sentences that are displayed in the s_q^p format, where p represents the document number and q represents the sentence number within that particular document. With these inputs, it first performs text processing. In this step, it mainly executes sentence splits and outputs a list of sentences, which are then used to construct sentence graphs. At this stage, the algorithm attempts to identify pairwise sentence connections that resemble the discourse structure of the documents. It then applies spectral clustering to

separate these sentences into clusters such that each cluster contains a set of semantic related sentences. Finally, the algorithm performs a multi-sentence compression, where it generates a single summary sentence for each cluster found in the previous step. It groups these sentences together to form the summary for the inputs provided.

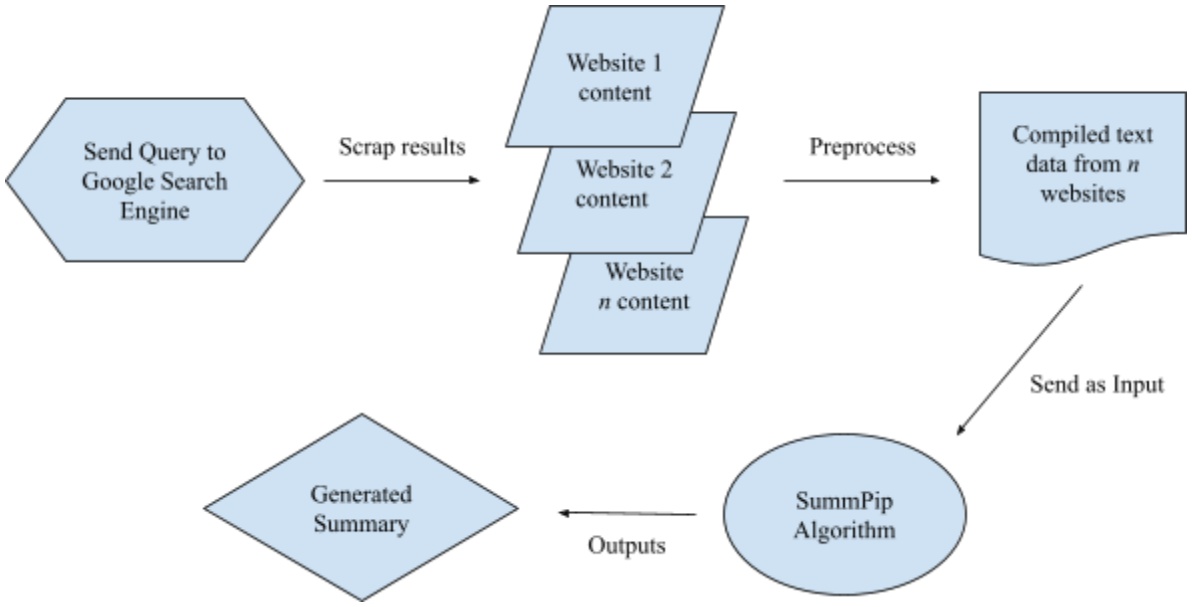

Figure 3. Pipeline for generating summary snippets

To employ the SummPip algorithm according to our needs, we performed the steps displayed in Figure 3. First, we typed the query in the Google Search engine. We then stored some number of top websites that Google recommends users to see. In our study, we define this 'some number' as *n* and in our experiment, we use *n* values of 1, 2, 5 or 7 to create summary snippets. We performed these steps in an automated fashion by creating a Python script that takes in two arguments – the top *n* value and the query. Based on these arguments, the script collects the top n website URLs from the Google Search engine for the given query. Since webpages tend to contain much noise data (navigation tabs, advertisements, redirect links, and more), we used the BeautifulSoup package to focus on extracting the main document content. This was done by gathering text present in a body tag of the webpage. Finally, we transferred all this content to a text file, which was then inputted into the SummPip algorithm.

Apart from the document content, the SummPip model also has other parameters in order to perform the steps in Figure 2. It takes in a file that contains a word-to-vector matrix, whose model is pre-trained using Gensim. Since we want to generalize this summarization model, we used the Multi-News dataset [9], which is a large-scale dataset for multi-document summarization and has news articles from a variety of fields such as health, tech, and politics.

Furthermore, as we want the summarized text to be a snippet (about 300 character text), we also focused on finding optimum values for other parameters of this model, namely `nb_clusters` and `nb_words`. The `nb_clusters` determines the number of sentences that the model at most produces in the output summary. Since summary snippets tend to be around three to five sentences, we chose the number five. The parameter, `nb_words`, on the other hand, controls the minimum length of each sentence in the output summary. We chose the number eight, because the snippets generated had an acceptable length with a satisfactory level of important information retained.

With all these parameters taken into consideration, we were able to run the SummPip model to create a summary snippet for a given query.

3.2 User Interface

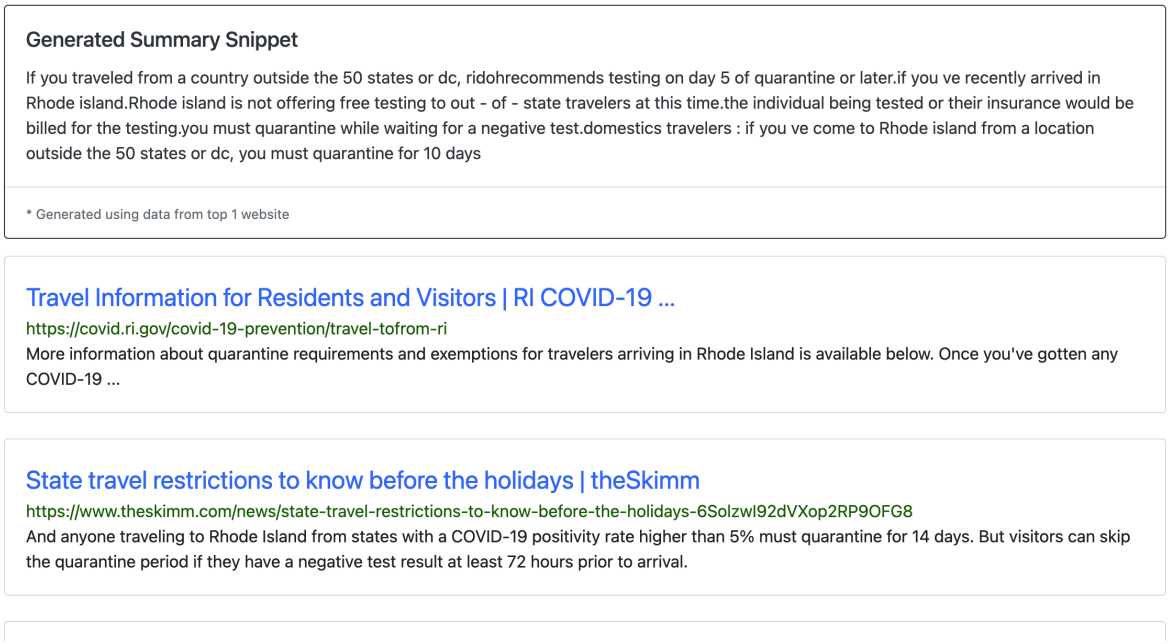

Figure 4. Search Interface (Results with Summary Snippets)

Figure 4 shows our user interface that is designed to resemble the Google Search engine and other modern search engines. Throughout our design process, we wanted to ensure that users are familiar with the interface and find it easy to use. By doing so, we reduce the possibility of measuring behavioral traces that are caused by user confusion. Hence, apart from the "Generated Summary Snippet" section, every other aspect including search bars and the ranked website results are identical to the Google Search interface.

Within the "Generated Summary Snippet" section, we have the summarized snippet as well as a footnote within the box, which indicates the data from which this snippet was generated. With this information, users can check if the generated text is relevant to the query or not. As a result, they could either visit the first *n* documents if the text is relevant or paginate to documents beyond the top n if it is not relevant.

4. Experiments

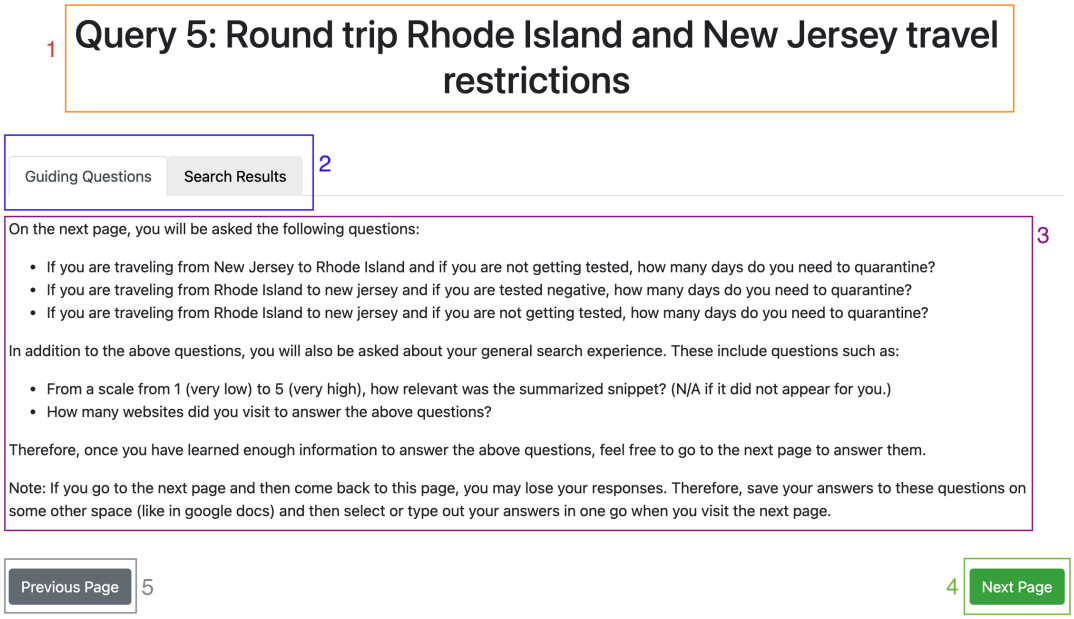

Figure 5. Search Interface used in experiments (Guiding Questions section)

1) *Pre-defined query showcased as a title to indicate the current query*
2) *Tabs to go back and forth between guiding questions and search results.*
3) *Guiding questions for the given query to aid users in searching for and understanding relevant conceptual or technical information.*
4) *Button to redirect to the related questionnaire after users finish searching.*
5) *Button to redirect users to the previous query section*

We have conducted our experiments on Amazon Mechanical Turk, which is a crowdsource user study platform. The users in this platform were faced with one of five distinct experimental conditions:

a) A search interface with no summary snippets;
b) A search interface with summary snippets generated using the first ranked website;
c) A search interface with summary snippets generated using the top two ranked websites;
d) A search interface with summary snippets generated using the top five ranked websites;
e) A search interface with summary snippets generated using the top seven ranked websites.

All five experimental conditions were served by the same information retrieval model. Each condition was presented to a total of 56 unique users who were asked to navigate through five different predefined queries with exploration tasks, which were afterwards validated through multiple-choice questions.

Based on the number of top *n* websites used, the SummPip algorithm can take between a couple of seconds to about twenty-five minutes. To ensure uniform user experience, we used pre-defined queries to pre-generate the search results. Furthermore, to learn how efficient users are at gathering and understanding relevant information using a given search interface, we chose task topics ranging from medical and tech to travel. This would allow us to also study the effects of prior domain expertise of varying levels with respect to interface and quiz interaction. Hence, users were tasked to answer one-to-three multiple choice questions for the following queries:

- *Query 1 : Lupus VS Rheumatoid Arthritis [medical]*
- *Query 2 : Foods and supplements to lower blood sugar [health]*
- *Query 3 : Marriott cyber-attack 2020 [technical news]*
- *Query 4 : Subnet Mask [technical]*
- *Query 5 : Round trip Rhode Island and New Jersey travel restrictions [travel]*

The multiple-choice questions asked for each query are posed as guiding questions for the users as shown in Figure 5. This would assist users to pursue a focused search to answer the related questions. Once satisfied with their research, they are redirected to a multiple-choice questionnaire.

Apart from the questions that are specific to the understanding of the query content, we also ask users the following about the corresponding query questions:

- *From a scale of 1 to 5, where 1 indicates very low and 5 indicates very high, how relevant was the summarized snippet in answering the above questions?*
- *How many websites did you visit to answer the above questions?*

- *From a scale from 1 to 5, where 1 indicates hardly familiar and 5 very familiar, how familiar were you with <query> at the start of the user study?*

With these, we learn about how much a user depended on the summarized snippet to answer the query questions.

For each included session, we recorded the time taken to read, understand and answer the queries. In addition, we stored the completed questionnaires along with the top *n* value that indicates which summarized snippet was showcased. We also have a short three-question exit survey that seeks to understand how the user felt about the transparency, simplicity, and trustworthiness of the search interface. Responses were collected on a five-point Likert scale, which ranges from 1 (“strongly disagree”) to 5 (“strongly agree”). The users were also given the option to write free-form feedback to provide their opinion on their user search experience. To filter out users that did not make serious attempts to answer the questionnaires, we enforced a 66% correctness threshold that users must meet in order to receive payouts.

5. Results

In this section, we compare the five experimental conditions in terms of perceived interface quality, search success, and task efficiency.

5.1 Perceived Interface Quality

To understand how users assessed the interface quality, we looked into aspects of transparency, simplicity, and trustworthiness. In addition, we also explored how relevant the summary snippets were with respect to the queries in those conditions that displayed summary snippets. The following subsections focus on results gathered from the exit surveys and from a question asked within each query questionnaire respectively.

5.1.1. Transparency, Simplicity, and Trustworthiness of Search Interfaces

The three exit survey questions we asked on a five-point Likert scale are:

A. Transparency in search results made it easier to find relevant documents.
B. Finding relevant results to the pre-defined search was intuitive.
C. The results from the search engine are an accurate representation of the truth.

We determined perceived transparency from the responses to A, perceived simplicity from the responses to B, and perceived trustworthiness from responses to C. In Table 1, we display the mean and median values gathered for all collected answers. We see that the mean values for the experimental condition including the summary snippets from top 5 websites scored highest in terms of simplicity and trustworthiness. In addition, snippets generated from top 7 websites scored the highest in terms of transparency. Hence, we can see that most of the conditions with summary snippets provided better perceived transparency, simplicity and trustworthiness than the one without summary snippets.

Experimental Condition	Perceived Transparency		Perceived Simplicity		Perceived Trustworthiness	
	Mean	**Median**	**Mean**	**Median**	**Mean**	**Median**
Top 0	4.06	4	4.06	4	3.93	4
Top 1	3.60	4	3.20	4	3.60	4
Top 2	4.00	4	3.69	4	3.77	4
Top 5	4.00	4	**4.27**	4	**4.36**	4
Top 7	**4.10**	4	4.0	4	4.10	4

Table 1: Mean and Median Values for Each Experimental Condition's Exit Responses. The highest value for each exit question has a green-colored background cell. The lowest value for each exit question has a pastel red-colored background cell.

However, based on the mean values, snippets generated from top 1 websites fared the worst in all aspects. To understand whether these differences are significant, we ran the Student's t-test [11] between the two groups, i.e., submissions with no summary snippets and submissions with summary snippets generated from top 1 website for perceived system quality in terms of transparency, simplicity, and trustworthiness respectively. Table 2 displays the results gathered from this test. Since all the p values > 0.05, the result indicates that the difference in the system quality perception is not statistically significant.

Perceived Interface Quality (Top 0 vs Top 1)	T-Statistic	P-value
Transparency	0.65	0.52
Simplicity	1.25	0.22
Trustworthiness	-0.10	0.92

Table 2. Student t-test results between answer distributions in Top 0 and Top 1 conditions

In terms of median values, all conditions scored the same. This observation is further reinforced by Kruskal-Wallis statistical significance test [10] between answer distribution of condition with no summary snippets (top 0) and ones with summary snippets (anything but top 0). The results (transparency p value = 0.37, simplicity p value = 0.29, trustworthiness p value = 0.77) indicate that the perceived system quality was not significantly different. Hence, together with the analysis of mean values, it indicates that search interfaces with summary snippets are perceived to be better, but not significantly enough than ones without summary snippets.

5.1.2 Relevance of Search Interfaces with Summary Snippets

As stated in the Experiments section, we asked users to answer the following at the end of each query section questionnaire: *from a scale of 1 to 5, how relevant was the summarized snippet in answering the verification questions?*

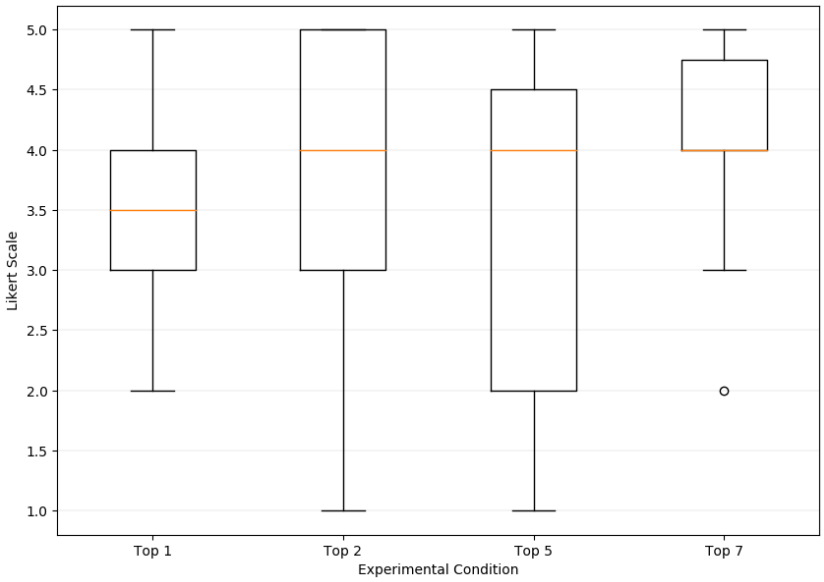

Figure 6. Box plot for Perceived Relevance of Summary Snippets in Query 1 Search Results

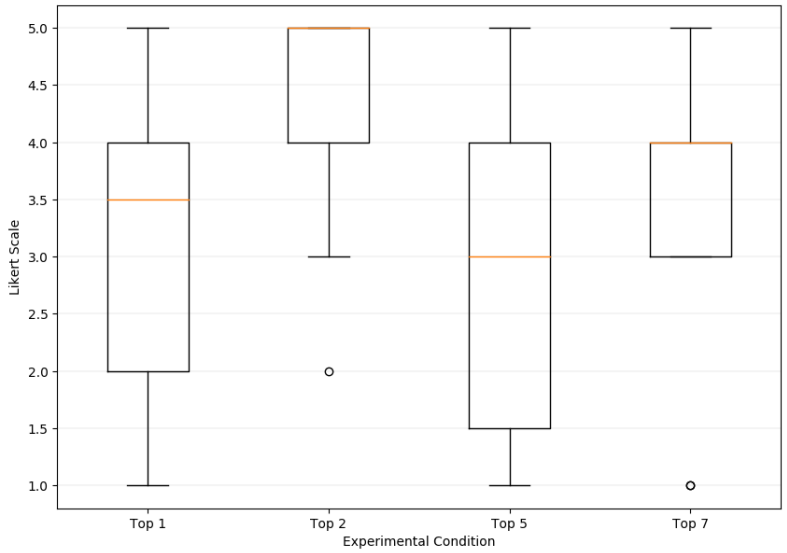

Figure 7. Box plot for Perceived Relevance of Summary Snippets in Query 2 Search Results

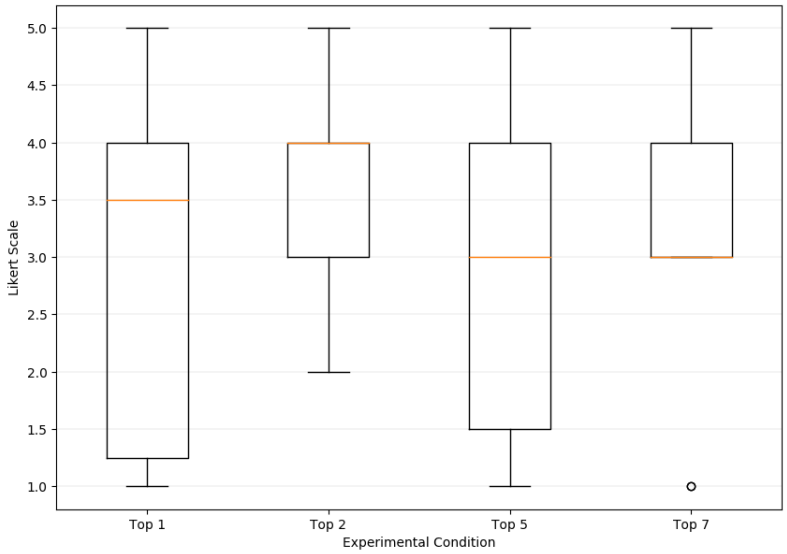

Figure 8. Box plot for Perceived Relevance of Summary Snippets in Query 3 Search Results

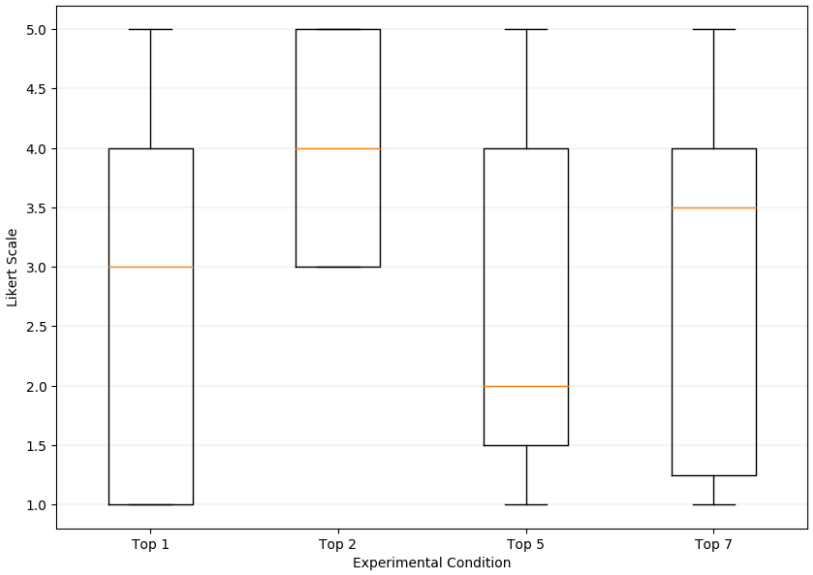

Figure 9. Box plot for Perceived Relevance of Summary Snippets in Query 4 Search Results

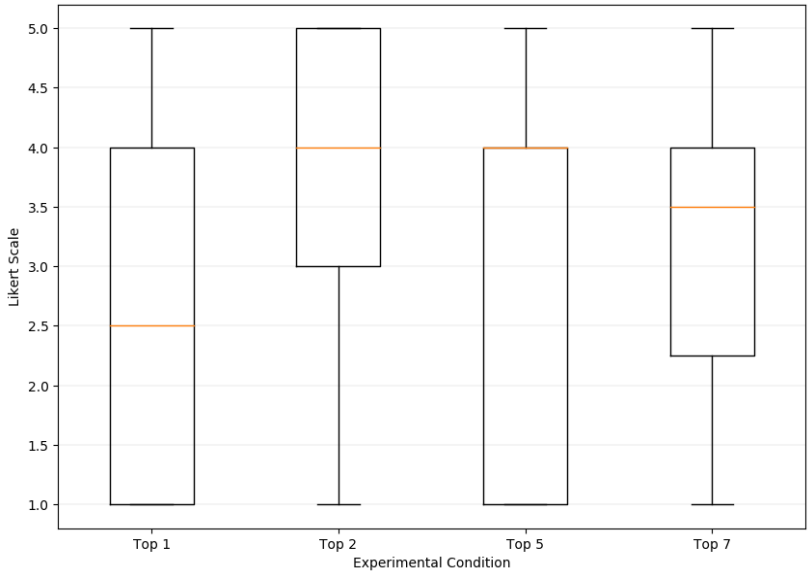

Figure 10. Box plot for Perceived Relevance of Summary Snippets in Query 5 Search Results

Figures 6-10 plot answer distributions over this for all four search interfaces. Each figure represents this for five different queries. Each boxplot within each figure represents a distinct SERP. There are a total of 20 such SERPs as there are 4 experimental conditions and 5 queries. In all of these plots, the yellow lines within the ‘boxes’ represent the median value for each experimental condition. The bottom border of the ‘boxes’ represents the lower quartile value

while the upper border represents the upper quartile value of the answers gathered. Based on these plots, we can see that 18 of 20 SERPs have median values of 3.0 and higher. Furthermore, all of these SERP combinations have third quartile values as 4.0 and higher. These are promising results as it implies that majority users found the SERPs with summary snippets to be relevant. These favorable results are consistent with the mean values shown in Table 3. We can see 13 out of 20 summary snippets have mean values of 3.0 and higher.

Amongst the mean values, the top 2 experimental has the highest overall mean value while the top 5 one has the least overall mean. To understand if this difference was significant, we ran the Student's t-test [11] between the answer distributions of those two conditions, i.e. top 2 and top 5. The calculated t-statistic of each t-test is 1.03 and the p value is 0.31. Since the p value > 0.05, the result indicates that the difference in the relevance perception between summary snippets generated from top 2 and from top 5 is not statistically significant.

Query	Top 1	Top 2	Top 5	Top 7
1	3.50	3.69	3.27	**4.00**
2	3.10	**4.38**	**2.91**	3.40
3	**2.90**	**3.69**	2.91	3.10
4	2.80	**4.00**	**2.72**	3.00
5	**2.70**	**3.46**	2.80	3.30
Overall	3.00	**3.85**	**2.93**	3.36

Table 3: Mean Values for Perceived Relevance for Each Query and Experimental Condition. The highest value for each query has a green-colored background cell. The lowest value for each query has a pastel red-colored background cell.

5.2 Search Success

To gauge the five experimental conditions' effectiveness on understanding search results, we evaluated each user's submission and calculated mean correctness per query as well as overall mean correctness.

Query	Top 0	Top 1	Top 2	Top 5	Top 7
1	77.8	73.3	**69.2**	**87.9**	86.7
2	**93.3**	**80.0**	84.6	90.9	90.0
3	64.4	60.0	**46.2**	**69.7**	63.3
4	**73.3**	**100.0**	96.2	95.5	90.0
5	73.3	**76.7**	71.8	75.8	**66.7**
Overall	73.9	75.8	**69.9**	**81.8**	76.7

Table 4: Overall and Specific (query-wise) Mean Correctness for Each Experimental Condition. The highest value for each query has a green-colored background cell. The lowest value for each query has a pastel red-colored background cell.

Based on the results displayed in Table 4, the highest overall mean correctness is 81.8% which is achieved by the search interface with summary snippets generated from top 5 websites. This value is about 7.9% higher than the overall mean correctness achieved by the control condition, i.e., the search interface with no summary snippets.

To understand if summary snippets have an effect on answering the verification questions correctly, we also conducted an independent samples t-test. We used the standard Student's t-test [11], because Levene's test for homogeneity of variances [12] indicated equal variances between the two groups, i.e., submissions with no summary snippets and submissions with summary

snippets (test statistic: 0.013, p value: 0.91). The calculated t-statistic of this t-test is -2.17 while the p value is 0.0404. Since the p value < 0.05, the result indicates that the presence of summary snippets significantly affects the user's ability to answer the verification questions correctly. Since the highest overall mean correctness is obtained by submissions that have summary snippets, the significance test implies that the summary snippet significance is favorable in terms of increasing user search success.

5.3 Task Efficiency

We gathered two types of data to understand how efficient users were in terms of completing tasks. One is related to how long a user takes to complete tasks per query. The other one is related to the number of websites a user visits to complete tasks per query. The following subsections analyze these data separately.

5.3.1 Time Taken

We recorded the time taken for users to complete each query section as well as the overall study to understand the effect of the experimental conditions on efficiently completing tasks.

Based on the results displayed in Table 5, the fastest overall mean time taken is 742.81 seconds which is about 90.66 seconds faster than the overall mean time taken achieved by the control environment, i.e, a search interface with no summary snippet (top 0).

We conducted an independent samples t-test just like in Section 5.2 to understand if summary snippets have an effect on answering the verification questions more efficiently. We used the standard Student's t-test [11] because Levene's test for homogeneity of variances [12] indicated equal variances between the two groups, i.e., submissions with no summary snippets and submissions with summary snippets (test statistic: 3.84, p value: 0.055 (which is greater than 0.050). The calculated t-statistic of this t-test is -0.68 while the p value is 0.50. Since the p value > 0.05, the result indicates that the presence of summary snippets does not significantly affect the user's ability to answer the verification questions efficiently. This implies that the condition

which got the fastest responses is not significantly, but just mildly better than the control condition.

Query	Top 0	Top 1	Top 2	Top 5	Top 7
1	**210.60**	299.90	253.77	221.64	**314.20**
2	**86.60**	130.80	154.31	135.18	**207.10**
3	235.87	253.00	223.31	**127.27**	**270.10**
4	**76.73**	**86.70**	77.15	86.55	85.00
5	223.67	**286.40**	216.39	**172.18**	173.90
Overall	833.47	**1056.8**	924.92	**742.81**	1050.3

Table 5: Overall and Specific (query-wise) Mean Time Taken (in seconds) for Each Experimental Condition. The fastest value for each query has a green-colored background cell. The slowest value for each query has a pastel red-colored background cell.

5.3.2 Number of Websites Visited

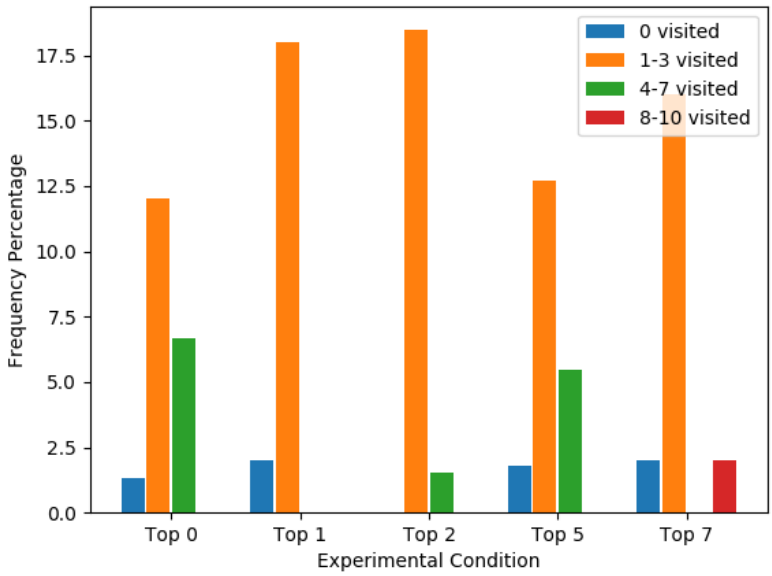

Figure 11. Number of websites visited in Each Experimental Condition for Query 1. The number of websites are displayed as groups: 0, 1-3 websites visited, 4-7 websites visited, and 8-10 websites visited.

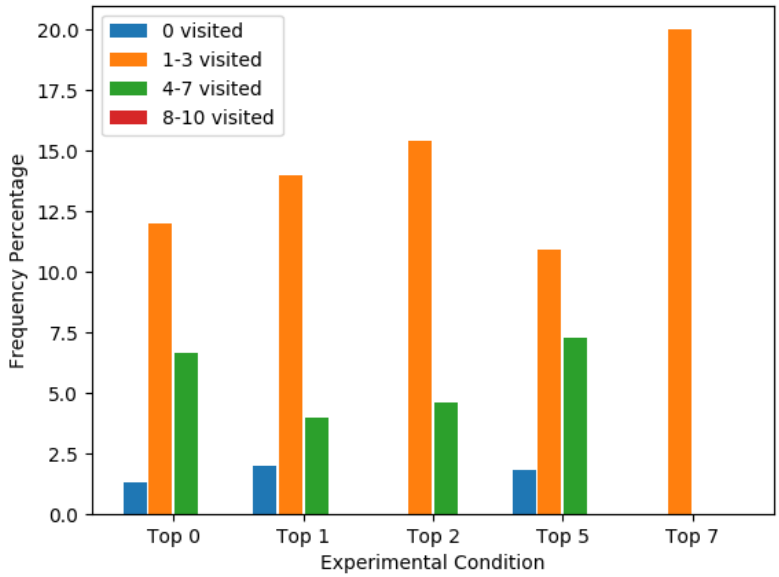

Figure 12. Number of websites visited in Each Experimental Condition for Query 2. The number of websites are displayed as groups: 0, 1-3 websites visited, 4-7 websites visited, and 8-10 websites visited.

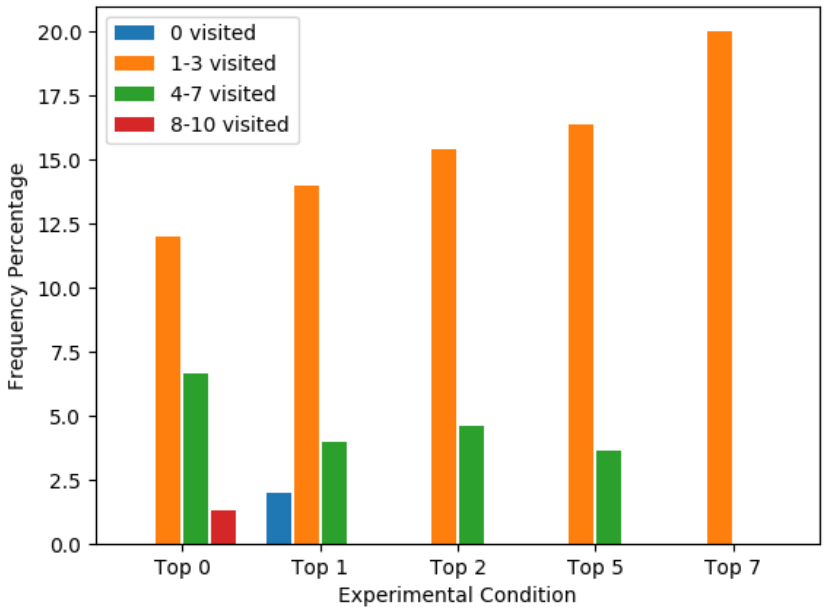

Figure 13. Number of websites visited in Each Experimental Condition for Query 3. The number of websites are displayed as groups: 0, 1-3 websites visited, 4-7 websites visited, and 8-10 websites visited.

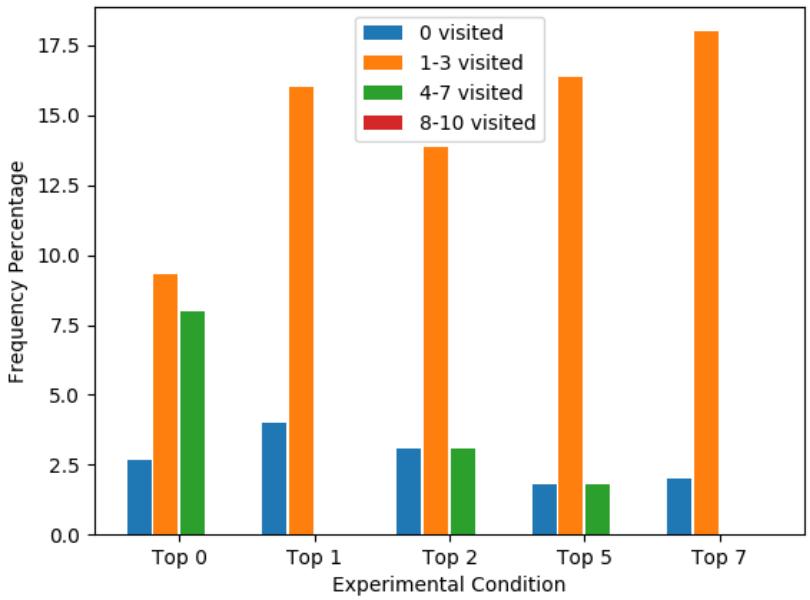

Figure 14. Number of websites visited in Each Experimental Condition for Query 4. The number of websites are displayed as groups: 0, 1-3 websites visited, 4-7 websites visited, and 8-10 websites visited.

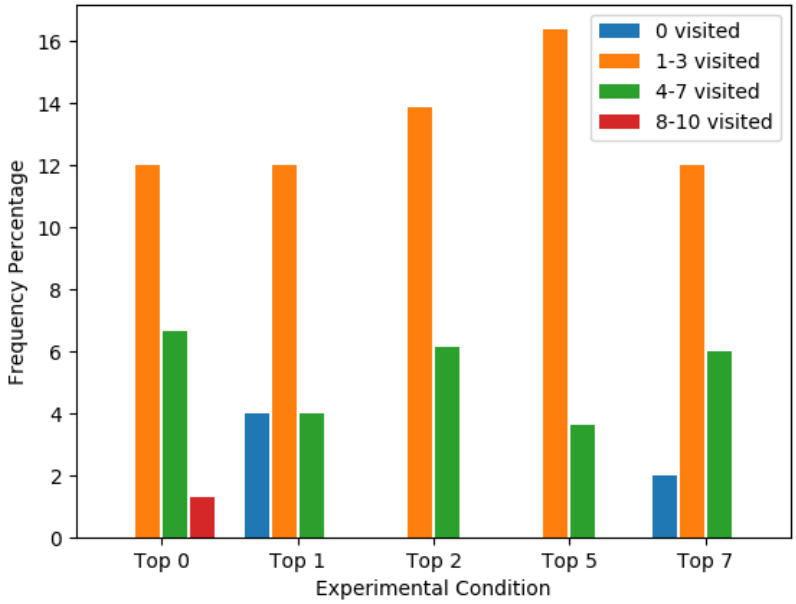

Figure 15. Number of websites visited in Each Experimental Condition for Query 5. The number of websites are displayed as groups: 0, 1-3 websites visited, 4-7 websites visited, and 8-10 websites visited.

At the end of each query questionnaire, we asked users the following question: '*How many websites did you visit to answer the above questions?*' The users would choose one of the multiple choices: 0, 1-3, 4-7, 8-10. Figures 11 to 15 display the answer distribution for this question for each query in every experimental condition.

In Figure 11, we can see that less number of users visited more than 4 websites for experimental conditions with summary snippets generated from top 1 and top 2 compared to the control condition (top 0). As for the rest of the conditions, the number of websites visited is around the same as the number of websites visited in the control condition. In Figure 12, we can see that fewer percentage of users visited more than 4 websites for top 1, top 2, and top 7 conditions than for top 0. Once again, for the other conditions, they have similar frequencies with the control condition. In Figures 13, 14, and 15, all conditions with summary snippets have fewer frequency of websites visited compared to the condition without summary snippets. Hence, from all of

these plots, we can see that for all queries, the majority of the conditions with summary snippets have fewer frequencies of visiting search result websites than the control condition.

To see if this change is significant, we ran the Kruskal-Wallis statistical significance test [10] between the control condition (i.e., without summary snippets) and non-control conditions (i.e., with summary snippets) for each query. This change was significant for two of the five queries, namely Query 3 [Marriott cyber-attack 2020] and Query 4 [Subnet mask] as shown in Table 6. The results indicate that this change was significant for technical-related queries. For the other types of queries, the difference in the number of websites visited is mild.

Query	Average difference between number of websites visited in Top 0 and Non-top 0 conditions Per User	Top 0 vs. Non-Top 0
1	0.69	0.125
2	0.41	0.441
3	**1.12**	**0.0396**
4	**1.17**	**0.0367**
5	0.89	0.152

Table 6: Average difference of websites visited per user for a query and its statistical significance value. Significant p values ($p < 0.05$) are highlighted in bold and placed in green-colored cells.

6. Discussion

For all of the metrics considered in the Results section, the experimental conditions that contain summary snippets perform better than the control condition, i.e., a SERP with no summary snippet. In particular, the search success metric for SERPs with summary snippets is significantly better overall. Amongst the SERPs with summary snippets, the SERP with summary snippet generated from top 5 websites has the highest mean overall search success and fastest mean overall time taken to complete the tasks. It is interesting to note, however, that users consider the summary snippets generated from top 2 websites more relevant than ones generated from top 5 websites in all the 5 queries tested. However, in terms of both time taken and search success, this experimental condition did worse than the control condition and the SERP with summary snippets generated from top 5 websites.

As the Levene's test [12] indicates equal variances, we attribute the cause of such differences to our small-scaled study. Hence, by performing a larger-scaled study, we will be able to reduce the stark subjectivity difference of perceived relevance. If the difference still persists then, we would have a bigger dataset and other metrics useful to understand the differences better. One such criterion is the familiarity question we ask users at each query questionnaire. Since we performed a small-scaled study, we were unable to incorporate this question into our result analysis. When we have a larger sample size, we can use this metric to see how relevant the individuals, who are pretty familiar with the query, found the summary snippets to be. In addition, we can use this criterion to understand how individuals who are unfamiliar with the task topics fare in these experimental conditions.

7. Conclusion and Future Works

This paper describes the effect of having a Search Engine Results Page (SERP) with summary snippets that are generated from some number of top relevant websites. In a crowdsourced user study, we showed that SERPs with summary snippets lead to significantly better search success and task efficiency. While this setup did not significantly increase users' perceptions of system quality, the results are promising for future research in this direction. Some ideas of such research include creating a dynamic user study where custom queries can be created without being time-consuming. In order to achieve this, faster unsupervised generic summarization algorithms should also be created. Lastly, future works can include creating metrics to measure the effectiveness of various summary snippets generated from different summarization algorithms. We can then compare the results to find a more optimum way to build SERPs with summary snippets that are coherent in grammar and content.

References

[1] Google. How Google's featured snippets work. Retrieved April 3, 2021 from https://support.google.com/websearch/answer/9351707?hl=en

[2] Jerome Ramos and Carsten Eickhoff. 2020. Search Result Explanations Improve Efficiency and Trust. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (*SIGIR '20*). Association for Computing Machinery, New York, NY, USA, 1597–1600. DOI:https://doi.org/10.1145/3397271.3401279

[3] S. Jiang and P. Liamruk, "Effects of SERP Information on Academic Search Behaviours," 2020 - 5th International Conference on Information Technology (InCIT), Chonburi, Thailand, 2020, pp. 33-38, doi: 10.1109/InCIT50588.2020.9310951.

[4] Aqle, A., Al-Thani, D. & Jaoua, A. Can search result summaries enhance the web search efficiency and experiences of the visually impaired users?. *Univ Access Inf Soc* (2020). https://doi.org/10.1007/s10209-020-00777-w

[5] Gupta V. (2013) Hybrid Algorithm for Multilingual Summarization of Hindi and Punjabi Documents. In: Prasath R., Kathirvalavakumar T. (eds) Mining Intelligence and Knowledge Exploration. Lecture Notes in Computer Science, vol 8284. Springer, Cham. https://doi.org/10.1007/978-3-319-03844-5_70

[6] Uçkan, Taner & Karci, Ali. (2020). Extractive multi-document text summarization based on graph independent sets. Egyptian Informatics Journal. 21. 10.1016/j.eij.2019.12.002.

[7] See, Abigail & Liu, Peter & Manning, Christopher. (2017). Get To The Point: Summarization with Pointer-Generator Networks. 1073-1083. 10.18653/v1/P17-1099.

[8] Zhao, Jinming & Liu, Ming & Gao, Longxiang & Jin, Yuan & Du, Lan & Zhao, He & Zhang, He & Haffari, Gholamreza. (2020). SummPip: Unsupervised Multi-Document Summarization with Sentence Graph Compression. 1949-1952. 10.1145/3397271.3401327.

[9] Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. Multi-news: A large-scale multi- document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy, July 2019. Association for Computational Linguistics.

[10] William H Kruskal and W Allen Wallis. 1952. Use of ranks in one-criterion variance analysis. Journal of the American statistical Association 47, 260 (1952), 583–621.

[11] Haynes W. (2013) Student's t-Test. In: Dubitzky W., Wolkenhauer O., Cho KH., Yokota H. (eds) Encyclopedia of Systems Biology. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-9863-7_1184

[12] Levene, Howard (1960). "Robust tests for equality of variances". In Ingram Olkin; Harold Hotelling; et al. (eds.). *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*. Stanford University Press. pp. 278–292.