Children's roles in web search

Master's Thesis

Pieter Dekker

Submitted in partial fulfillment of the requirements for the degree of master of science in media and knowledge engineering

Thesis committee:
Prof. dr. ir. A. P. de Vries
Dr. ir. R. C. Hendriks
Dr. J. A. Redi
C. Eickhoff, Msc
H. Jochmann, Msc

Thesis supervisors:
Prof. dr. ir. A. P. de Vries
C. Eickhoff, Msc

July, 2011

TU Delft

Contents

Preface

This thesis highlights the research being done between July 2010 and July 2011. It contains the results of my graduation project of the Media and Knowledge Engineering master.

I would like take this chance to thank a number of people for helping me during this project. First I want to thank the 30 children from primary school ‘De Kroevendonk’ who were willing to participate in this study. Without their participation this project would not have been a success. I also want to thank their teachers for their flexibility during the time we carried out the experiment at school.

A special thanks for my supervisors Arjen P. de Vries and Carsten Eickhoff for their positive feedback during this project; especially in times when I was a bit skeptical about the results.

And last but not least I would like to thank my wife for supporting me this year especially the last days for reviewing several parts of this thesis.

1. Introduction

1.1 Problem description

In the last decade children in the age of 7 to 12 years old are getting more and more acquainted with computers. According to the EU Kids online report [1] 83% of the children in the age of 6 to 10 and 96% of the children in the age of 11 to 14 use the internet. The same publication shows that children are younger when they go online (2005 to 2008). Also within school education children are encouraged to use the computer. This is the case with for example assignments in which children need to search for information in order to write a paper or to create a slideshow presentation. For now we will focus on the case that school children have to look up information on the Internet. The teacher hands out an information search task to his class of 25 children. They all have their own computer and start searching for the answer of the assignment. The teachers' task is to help children who have trouble finding useful information. Some children have a lot of experience in searching the Internet and are very well capable to find the answer on their own. Others do not have that level of expertise and could look for a half an hour and do not find anything useful. The teachers' job is to identify children who lack experience and could use his guidance. But in order to identify which children have trouble finding the right information the teacher should spend some time with everyone. Because it could easily take a minute to identify any troubles. It is therefore not feasible for a teacher without any technological aid; it will take too much time to spend time with every child. In this study we will look at a possible solution for this particular problem.

1.2 Related work

Several studies have been published about Internet search behavior. We will present publications about search behavior which are related to this study. This section ends with related work which will be the starting point of this research.

1.2.1 Information seeking models

Several studies have been conducted to investigate the possibilities of creating models of children's information seeking behavior. Shenton and Dixon [2] reviewed models that have been developed based on results from research among children in the age of 4 to 18. The grounded model of information seeking via the Internet consists of 11 different actions or influences. Before the information seeking starts, you have already 5 different parts in the information seeking process that influence the search behavior like: the origin of need, the directness of use and the place of access. The information seeking itself is divided in 6 different blocks like the employed approach, the familiarity of the site and the result of interaction.

Another model is presented by Bilal [3]. It is a model of Arabic-speaking children's interaction with the International Children's Digital Library (ICDL). The ICDL is a web interface that introduces children to various cultures with books. This model is more than the previous mentioned model focused on the information seeking itself and has several modes of behavior and possible moves between these modes. Examples of these modes are:

- Browsing
  A child scans the list of book thumbnails and moves to the next page with thumbnails.
- Backtracking
  A child uses the back arrows of ICDL and the Back button of Internet Explorer.
- Navigating

A child amplifies and goes back to the original state of the page by using the plus- and minus sign of the ICDL interface.

1.2.2 Children's roles

The starting point of this research is the work of Druin and others. They started with a pilot study to understand how children, in the age of 7, 9 and 11 years old, search the Internet using keyword interfaces at home [4]. This study showed that children have several barriers to overcome when searching the Internet like spelling, typing, query formulation and deciphering results. In following qualitative home study among 83 children 7 distinctive search roles were identified [5]. We will explain this last study more in detail, because we will use these roles in our study as well.

83 children in the age of 7, 9 and 11 years old participated in the home study. The parents and children were interviewed. After the interview the children were asked to do several search activities on the computer. The children were allowed to use any web search engine available. But because Google is the most used search engine in the Netherlands we offered a button on the website which linked to it.

The following questions were asked:

- Open-ended questions (understanding the motivation of children to search)
- Task-oriented questions (making comparisons)
- Personal, targeted question (making comparisons)
- Multi-step question (exposing breakdowns)

After analysis several characteristics emerged as the framework for defining the seven distinctive search roles. Examples of such characteristics are use of spelling, search brakedowns and the use of natural language. The following search roles were defined. A more extensive explanation of these roles can be found in [5]:

- *Developing searcher*
  Tends to search with natural language and are able to complete simple queries, but have trouble with complex ones.
- *Domain-specific searcher*
  Limits searches to find specific content of personal interest. Returns continually to a small number of specific websites.
- *Power searcher*
  Has sophisticated search skills and is able to use keywords instead of natural language. Do not suffer a lot from breakdowns and is able to solve the more complex search questions.
- *Non-motivated searcher*
  Is not persistent when searching and is not motivated to find alternative solutions.
- *Distracted searcher*
  Has trouble staying on the current search task and is easily distracted by visual movement.
- *Visual searcher*
  Likes to search information in visual content like images and videos.
- *Rule-bound searcher*
  Searches for information through a set of rules and is not able to deviate from these rules.

Other related work will be introduced in the corresponding sections.

1.3 Contribution

The starting point of this research is the result of Druins' study with seven different search roles. Labeling every search task manually takes a lot of time. In light of the practical situation explained in the introduction we will look for a way to perform this labeling automatically. Possibly aiding programs can be created that use this method to help children searching the Internet. In this study we will try to find a way to automatically assign search roles to the search behavior of children performing a search task on the Internet.

PuppyIR

This research is part of the PuppyIR project. This project is co-funded by the European Union and has a lifespan of three years. It is carried out by eight different partners. These are the TU Delft, four other universities, a children hospital, a multi-disciplinary museum and an international information services company. Aim of this project is to help children search the Internet.

1.4 Outline

The outline of this thesis is as follows. The experimental setup is described in Chapter 2. Results of the experiment are used in Chapter 3 for analysis. The final evaluation is carried out in Chapter 4. Conclusions are drawn in Chapter 5 and this chapter also yields a section with future work recommendations.

2. Experimental setup

2.1 Participants

This section describes how participants are selected. It also gives their demographics and how we ensured their privacy and safety.

2.1.1 Participant selection

As stated in the problem description we decided to investigate the search behavior of children in a school environment. Primary school 'De Kroevendonk' in Roosendaal was prepared to let children participate in the experiment during school hours. Because of the age of the children the parents had to sign for their involvement. We handed out an informational letter to the children with attached an informed consent form [Appendix A]. This letter was handed out to children in the 6th, 7th and 8th grade (according to the Dutch school system) of the school. In every class a short introduction was given by the researcher to introduce him and to give some information about the experiment itself and research in common. After a few weeks the informed consent was handed out again to children who also wanted to participate, because some children probably forgot to give the form to their parents and probably some parents forgot to sign it. In total we received 30 forms from a pool of 100 children. Based on this number we did not have the luxury of making a selection, we needed every participant for this experiment.

2.1.2 Demographics

In order to give a complete picture of the participant pool we will present several demographics about the participants themselves and some demographics about the city they live in.

In total 30 children participated in the experiment from which 5 children contributed to the trial and 25 children to the final experiment. The trial group is too small to be representative and we do not want to blur the demographic numbers. Therefore we will only show the demographics of the final group.

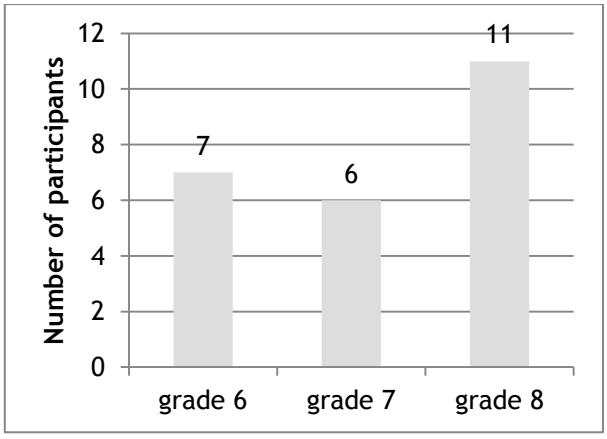

Figure 1 Participants per grade

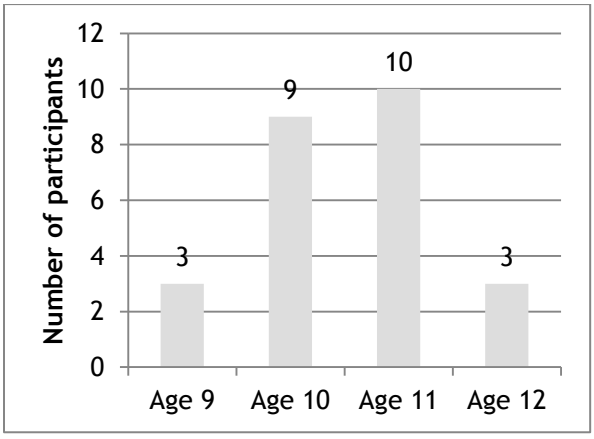

Figure 2 Number of participants per age

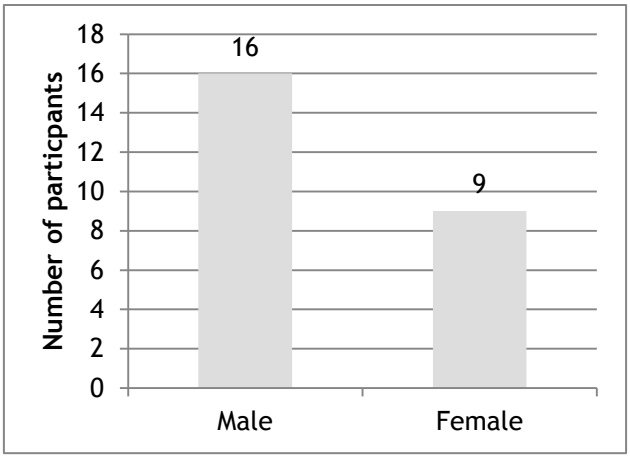

Figure 3 Participants per gender

2.1.3 Privacy and safety

The participants in this experiment are in the age from 8 to 12 years old. Therefore we had to be very careful in the things we said them and the things we let them do. Experiments with young participants are bound to Dutch and European law [6,7,8,9]. This research is part of the Puppy-IR project which has obligated guidelines [10] to ensure the safety and privacy of the children. By regulations of the Puppy-IR project every survey including people should be reviewed by the ethics board to ensure the privacy and safety of the participants.

2.1.3.1 Informed consent form

All participants are in the age of 12 or younger and therefore by European regulation the parents have to sign an informed consent form. The informed consent and any given information give parents and their children a clear view of the research. Both the parents and children should be able to understand what this research is about and what is expected from the children. The informed consent form with additional information [Appendix A] about the research was handed out to the children in order to give to their parents. We decided to give a short presentation in the grades 6 to 8 (according to Dutch school system) so we could adapt the information to the level of the children. After the presentation we gave the children some time to ask questions about the research.

2.1.3.2 Data handling

Personal information of the participants is stored in a database. The four tasks per participants are logged which consists of every action the participant performs at the computer during the experiment. Storage of this data is bound to several regulations. Also, several regulations tell with whom this information can and cannot be shared based on the informed consent. Several important aspects stand out:

- Only members of the Puppy-IR project are allowed to view the collected data.
- After finishing this particular research the collected data (screen video and HCIB logs) will be destroyed including any personal information (like name class, age and gender). Only the processed data will remain which cannot be used to trace back to the participants.

2.2 Trial experiment

This section describes the experimental setup. We have conducted a trial before the actual (final) experiment in order to be able to eliminate any flaws in its design.

2.2.1 Setup

The school did not have the possibility to give us the same (class)room during the weeks we conducted our research. Even in one day we could not rely on one free room. Therefore we decided to work with laptops to make our setup more flexible. Two laptops were used, one for the participant and one for the researcher to keep track of time and take notes. In order to make the laptop easier to use for the participants we used a separate mouse to avoid touch pad use. At school every child uses a regular computer with a separate mouse. Therefore, the touch pad could be less intuitive to use for the children and possibly influence their mouse usage during the experiment.

2.2.1.1 Protocol

The experiment is designed in such a way that the sessions between participants are comparable. We have created a protocol which we used to structure everything that is being said to the children. The full protocol can be found in Appendix B. We will discuss the protocol in short below and highlight several aspects:

1. Everything is set in place in an empty room.
2. With every teacher we made arrangements when to pick up children from their classroom activities. Randomly, we picked a name from the list and went to pick him or her up.
3. During the walk from the classroom to the room where we set up the equipment the researcher talked about what he or she was doing at that time. The researcher makes some small talk to comfort the participant.
4. Arriving at the room the researcher tells the participant behind which laptop he or she could sit. The researcher sits behind the other laptop.
5. The researcher explains the procedure to the participant.
6. The researcher asks if the participant has any questions and tries to clarify the things that are unclear.
7. If no questions are left, the participant starts the experiment by clicking on his or her name. The following questions were asked:
    a. Hoe vind je het om mee te doen aan dit onderzoek? (How do you like it to participate in this study?)
    b. Hoe vaak maak jij gebruik van een computer? (How often do you use a computer?)
    c. Hoe vaak maak jij gebruik van internet? (How often do you use the Internet?)
8. After these three questions the participant starts with the four search tasks which are explained in Section 2.2.1.2. The participant receives the first question on paper and starts searching.
9. The participant searches for the answers on the four questions.
10. When the participant is done the researcher tells him or her that one last question will be asked.
11. After completing the experiment the researcher asks what the participant thought about the questions. Whether they were hard or not. He also makes other small talk in order to not end the session too abruptly.

We used Google as default search engine behind the buttons on the website, but we never emphasized this to the participant. The questions about the participant experience with computers and the Internet we used to get a clear picture of the participants.

2.2.1.2 Search tasks

The main goal of the experiment is to have children search for answers. In the experiment of Druin children were given multiple different kinds of search tasks like open-ended questions, targeted questions and multi-step questions. Several studies found that the complexity or specificity of a search task influences the performance. Druin used this explicitly in the complex multi-step questions. This type of question is used to investigate the breakdowns for the more sophisticated searchers.

Studies like [11] and [12] give a clear separation of question types. We decided to also cover multiple types of question in order to see whether this would influence the search behavior of children. The referenced studies show a clear separation between two types of questions. It depends on the study how this separation exactly is made and how the questions are called. We chose the separation of closed (factual) and open questions. These questions are also very common in educational courses at a primary school. In addition we added a multi-step question like Druin has done [5]. By adding this question we hope to identify the power searcher who will have a better chance in succeeding than other searchers.

Search tasks:

1. Closed (or factual) questions
   These questions ask for a concrete answer. Example: What is glass made of?
2. Open questions
   These questions are of a more explorative character. The answers can deviate a little. Example: What can you find about the life of Queen Beatrix?
3. Multi-step questions
   Someone needs to find multiple pieces of information to get to the final answer. Example: What is the name of the wife of the prime minister?

We started with two closed questions, next an open question and finally a multi-step question. This specific order was chosen on purpose.

1. Hoeveel liter water drinkt een tijger per dag? (How many liters does a tiger drink each day?)
2. Hoeveel broers en zussen heeft koningin Beatrix (How many brothers and sisters does queen Beatrix have?)
3. Wat kun je vinden over de eerste auto die ooit gemaakt is? (What can you find about the first car ever built?)
4. Op welke dag is de minister-president van Nederland jarig in 2011? (On which day is the birthday of the prime-minister of the Netherlands in 2011?)

2.2.2 Data collection

We need information about the search behavior of children in order to build a classifier that could determine that behavior automatically. Examples of that information are the user input by keyboard or mouse. A more advanced way is to use an eyetracker to monitor where users are looking at on the screen [13].

We hope that the results of this research can be used to create an aiding program for teachers in classrooms. Therefore we need to keep in mind the possibilities of regular computer classrooms on primary schools. For example the eyetracker mentioned before can be useful in identifying search behavior, but is practically out of reach because of the costs. Another limitation is the small ICT budget of primary schools. A lot of schools get hand-down computers from companies. This means that their computers are not high standard and have limitations in memory and CPU power.

In our setup we used two programs to log the users' actions. The first program is the Usaproxy proxy server [14]. This proxy server logs every http request. It also injects javascript code which is responsible for recording mouse movements, keyboard and mouse input. This setup is easy to use and makes it possible to measure search behavior on a large scale [15]. We used the adapted version of UseProxy from [15]. It had one important drawback, because it could not detect delete and backspace buttons in Internet Explorer. Therefore in our experiment we decided to use Firefox as Internet browser. These delete actions could be estimated based on other actions so Internet Explorer would be an option. But we did not want to introduce any chances of having incorrect data.

The second program is a screen recorder. After the sessions we had to be able to review the session afterwards to identify the roles manually. We already looked at a program which has that capability, the Morea recorder [16]. This program is used in usability testing and has capabilities to record the screen and other inputs like the keyboard input. We have not used it to log the users' actions, because Morea uses its own recording format and cannot be read by other software. The screen recording is also stored in its own format, but we do not need it for further processing only for reviewing the sessions. Therefore, we decided to use Morea for the screen recordings.

2.2.3 Issues encountered

This trial experiment was conducted among 5 children. During these five sessions several issues came up which we will discuss next.

Questions

Only 2 of the participants could find the answer to the first question. It seemed to be difficult for them to find an answer with Google. This in itself is not a problem, because some questions just could be more difficult than others. But the downside of a tough question at the start is that participants can get discouraged about their abilities.

Time limit

During the trial, we did not have a strict time limit on the questions. We first looked at how much time the questions would take up on average. One of the sessions lasted about 45 minutes, which is too long for children to stay focused.

Screen recorder

We started with the Morea Recorder to record screen actions. Unfortunately at the end of the trial experiment the trial period of the Morea Recorder expired and we could not get a free renewal period. The Morea Recorder creates a format only readable by Morea which makes it more difficult to deal with the screen recording compared to a regular .avi file

Data collection
We investigated the collected UsaProxy log files after the trial sessions and unfortunately we noticed some flaws:

1. Not every web page request was registered by UsaProxy. Every call to the web server is recorded. Which means that not only the actual url request is logged, but also every image or advertisement. Usaproxy makes a distinction between these http requests and tries to identify the url requests which is the information we need. Unfortunately in some cases this specific request did not make it into the logs and was already filtered out.
2. Another problem arose in combination with the previous flaw. Beforehand we knew that the use of the back button could not be logged directly. This button is an important tool in browsing the Internet. We solved this by comparing sub-sequent http requests. For example when someone goes to example.com and from that site her or she goes to another web page and finally uses the back button to go back, it should be visible in the http request logs. Unfortunately, because of the instable http request logging this program is not suitable for our purposes.
3. The third problem was the Javascript embedded in the web pages. We noticed that sometimes the user already moved his mouse while the Javascript functions did not yet log these actions. In these cases it took a couple of seconds before the logging started. We knew beforehand that there is a white spot between web pages. But the combination of an Internet connection that was not that fast and some web pages that seem to make it last longer before the Javascript loaded this white spot became a problem.

Because of these problems we decided that UsaProxy was too unstable to use in the final experiment. We adapted the setup for the final experiment. The changes are discussed in the Section 2.3.

2.3 Final experiment
This section describes the final setup and how data is collected with our final participants group.

2.3.1 Setup
The final setup has not changed in ways that would be noticeable by the participants. In this section we will only discuss the parts that are updated since the trial experiment. These changes are based on the issues we found during the trial [see Section 2.2.3]

Search questions

The questions needed to be rephrased to be more comprehensible for the children. The first question was replaced, because most participants could not find the answer.

1. Wat eten walvissen? (What do whales eat?)
   This question completely replaced the question in the trial.
2. Hoeveel broers en zussen heeft koningin Beatrix (How many brothers and sisters does queen Beatrix have?)
   We did not change anything with this question.
3. Wat kun je vinden over de eerste auto die ooit gemaakt is? Schrijf hiervan enkele dingen op. (What can you find about the first car ever built? Write some things down about it)
   The question itself was not changed, but we added the sentence that the participants should write some things about that first car, because one of the participants during the trial wrote down: ‘veel’ (much) on his answer sheet.

4. Op welke dag van de week is de minister-president van Nederland jarig in 2011? (On which day of the week is the birthday of the prime-minister of the Netherlands in 2011?)
   We adapted this last question by adding the part 'dag van de week' (day of the week). Some participants gave the date as answer instead of the day of the week. By just looking for the date this 3-step question would change to a 2-step question.

Time limits
We decided to set time limits per question to limit the total amount of time per participant. On one hand this would prevent a participant from missing too much of the class activity at that moment. On the other hand we wanted to prevent that the participants would get bored or would become frustrated when the answer could not be found. We set the total of the search tasks at 30 minutes. For the first two questions the participants had 6 minutes, for the second one 8 minutes and for the last one 10 minutes.

2.3.2 Data collection

Unfortunately our data collection methods were not good enough to facilitate the final experiment. We started looking for a replacement of the Usaproxy proxy server. First other proxy servers were considered, but they all seem to have the same problem when it comes to logging the http request based on direct input of the user. We looked for other options besides proxy servers and came across a Firefox add-on, the HCI Browser [17]. HCI is short for Human Computer Interaction. This tool was developed by Rob Capra at the Interaction Design Lab in the School of Information and Library Science at the University of North Carolina at Chapel Hill. Instead of an in-web page Javascript logging tool like with Usaproxy this add-on makes it possible to log every action within the browser. We used the version from 7 july 2010.

The add-on had a built in toolbar which displays a questionnaire and could present tasks to the user. Not every aspect of this add-on was needed in our research and some aspects even were undesirable when it comes to children as users. The tool is published open-source under GNU General Public License. Therefore we were able to adapt it to our acquirements. We changed HCIB in two ways: the logging mechanism and the frontend toolbar.

HCIB toolbar
The toolbar showed a lot more information than needed. Before the trial we already built a website which was used to show questions to the children. We thought it would be better and more understandable for children to use this website instead of the HCIB toolbar, because they are more familiar with web pages than built in add-ons. Therefore we stripped the toolbar to just one line with a start and stop button, which could be used to manual start and stop the logging. By minimizing the size of the toolbar it would not distract too much from the actual content.

HCIB logging
The basic logging mechanism was more stable compared to UsaProxy and logged almost all the information we needed. We changed and added a few parts to make the HCIB more suitable for our experiment.

Screen recording
For recording the screens we chose to use the opensource CamStudio 2.0 [18] which records the screens to a regular .avi video file.

2.4 Collection results

After the final experiment we had data of 25 different children. Below an example of the data collected by our adapted version of the hci browser:

1292494008865 16-12-2010 11:06:48 Focus clientx=1366 clienty=605 http://www.google.nl/ http://www.google.nl/

1292494008929 16-12-2010 11:06:48 LoadCap clientx=1366 clienty=605 http://www.google.nl/ http://www.google.nl/

1292494009544 16-12-2010 11:06:49 MouseMove x=596 y=254 http://www.google.nl/

1292494010592 16-12-2010 11:06:50 MouseMove x=606 y=253 http://www.google.nl/

1292494010822 16-12-2010 11:06:50 LClick x=606 y=253 undefined http://www.google.nl/

1292494012166 16-12-2010 11:06:52 KeyPress key=D keycode=68 combi= http://www.google.nl/

1292494012503 16-12-2010 11:06:52 KeyPress key=E keycode=69 combi= http://www.google.nl/

1292494013273 16-12-2010 11:06:53 KeyPress key=Space keycode=32 combi= http://www.google.nl/

1292494013795 16-12-2010 11:06:53 KeyPress key=E keycode=69 combi= http://www.google.nl/

1292494014167 16-12-2010 11:06:54 KeyPress key=E keycode=69 combi= http://www.google.nl/

Unfortunately, we had problems with two of the 25 collected datasets.

First we had a participant who closed the Internet browser by accident. It seemed he forgot to use the home button to get back to the assignment website. After restarting Firefox and a click on the start button in the HCIB toolbar the participant went on with the assignment. Afterwards the log showed that after the restart the logging was stopped. It appeared that the logging was blocked by the previous sessions and could not continue with pressing on the start button. Without a HCIB log we were not able to use the results of this participant.

We discovered a second problem when trying to analyze the results. The starting time of the posted questions was not recorded correctly with one of the participants. These time stamps were recorded by basic php functions and logged in the mysql database. We could not figure out what the cause of this problem was. In all other cases the timestamps were recorded correctly. Fortunately, based on the video logs and the HCIB logs we could correct the timestamps manually.

In total we had 24 participants from which the results were usable. All together we have a database of 96 search tasks completed which we will refer to as tasks. In the next section we will take a closer look at these tasks and try to make a classifier based on the information we collected.

3. Data analysis

3.1 Labeling

After performing the experiment we had a database of 24 children and a total of 96 search tasks. Every task needed to be labeled based on the roles identified by Druin. The roles are explained shortly in [5] and based on the given information the tasks were labeled by viewing the screen recordings. To eliminat most of the bias of having just one person labeling the data we had two experts. These experts labeled the tasks independently from each other.

3.1.1 Labeling results

The results of this labeling can be found in Figure 4.

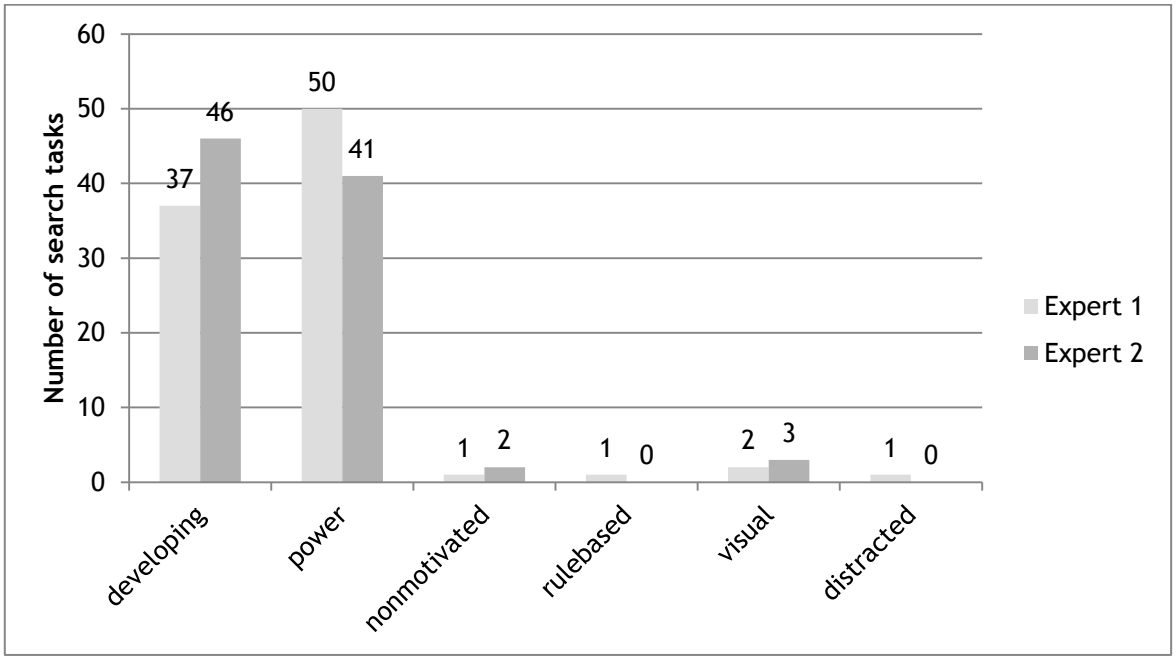

Figure 4 Number of labels per expert per role

As can be seen from this figure we have two dominant roles: the developing role and the power role. All other roles have been found only sporadically.

The two experts agreed on 82% of the tasks. This percentage of agreement in itself is not a good measure to explain the level of inter-rater agreement, because it does not take into account the accidental agreement, or otherwise called agreement by chance. Therefore, we used the Cohen's Kappa [19] statistic to account for this:

$$K = \frac{P(actual\ agreement) - P(expected\ agreement)}{1 - P(expected\ agreement)}$$

$$K = \frac{0{,}82 - 0{,}45}{1 - 0{,}45}$$

$$K = 0{,}67$$

We now had a more solid number to work with. Unfortunately the interpretation of Kappa is not straightforward. Table 1 is often used in order to make the interpretation of this statistic possible.

Kappa	Agreement
< 0	Less than chance agreements
0.01 – 0.20	Slight agreement
0.21 – 0.40	Fair agreement
0.41 – 0.60	Moderate agreement
0.61 – 0.80	Substantial agreement
0.81 – 0.99	Almost perfect agreement

Table 1 Kappa interpretation [20]

Based on Table 1 we can say that our Kappa result concludes that the agreement between the two experts is substantial.

As stated in Section 2.2.1.2 we gave the participants three different kinds of search tasks. So far we have seen the agreement between the experts globally. But it is interesting to see whether roles are easier to identify with a particular search task compared to other tasks. Therefore we calculated the interrater agreement per search task.

Search task	Variable	Number
1	p (proportion of agreement)	0,92
	p (expected agreement)	0,51
	K	**0,83**
2	p (proportion of agreement)	0,71
	p (expected agreement)	0,47
	K	**0,45**
3	p (proportion of agreement)	0,75
	p (expected agreement)	0,41
	K	**0,57**
4	p (proportion of agreement)	0,96
	p (expected agreement)	0,46
	K	**0,92**

Table 2 Interrater agreement per search task

Table 2 shows that the two experts have an almost perfect agreement (according to [20]) with search task 1 and 4, while having a moderate agreement on the other search tasks.

This agreement result is explainable. Search task one is relatively easy and almost every child succeeds in this task. Children with more experience should not have a lot trouble in retrieving the answer. On the other hand children with almost no experience could need more time to solve the task. Search task 4 is relatively hard and this makes a clear separation between children with a developing role and children with a power role. It is therefore highly likely that the experts often agree on the same search role. Search task 2 and 3 are not too hard, but also not too easy. The distinction between roles could therefore be not clear enough, hence the lower interrater agreement.

3.1.2 Golden standard

Although the overlap between the experts is relatively high we still have two labels for each task. One way to deal with this is leaving the labels as they are and try to incorporate this information in a multi-label classifier. But the amount of data is relatively low and therefore the results of a multi-label classifier will not be representative. Also looking at the educational goals of this research this option is not ideal. Therefore we chose to look for a golden standard. The two experts discussed the

tasks on which they had a different opinion and agreed to a final label. The resulting golden standard is made visible in Figure 5 and Figure 6.

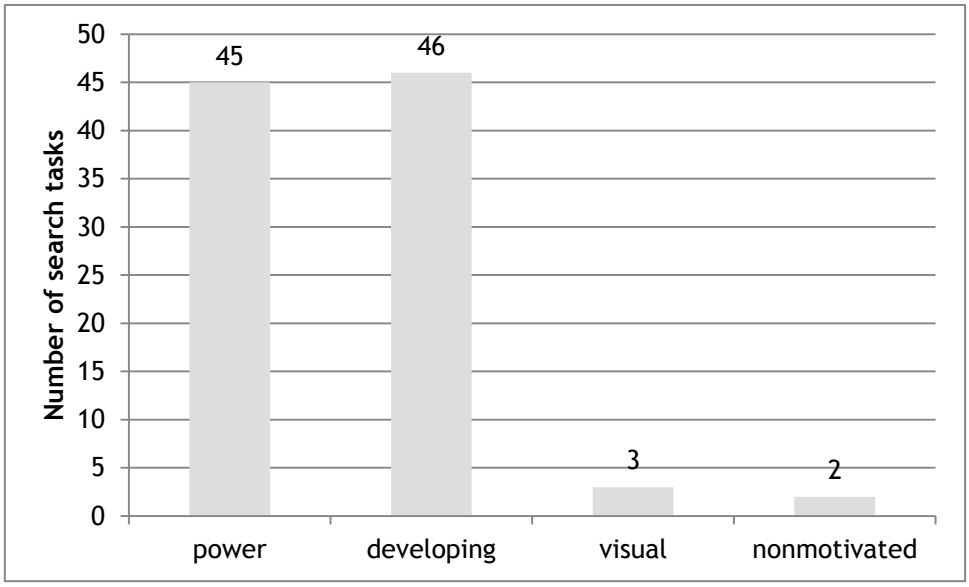

Figure 5 Number of search tasks per role

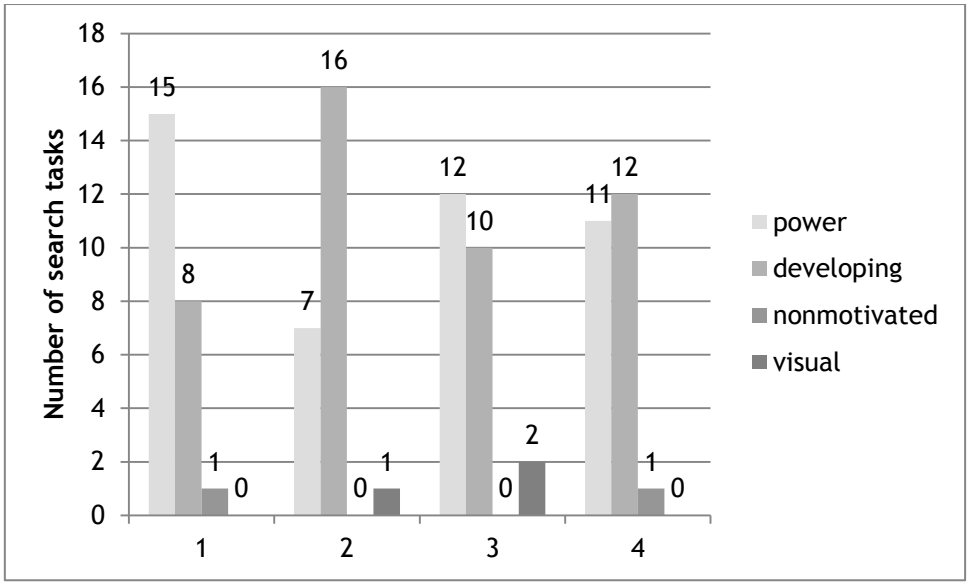

Figure 6 Role distribution per search task

3.1.3 Differences and similarities

Figure 7 shows the results of the work of Druin. At this point it is interesting to compare our result to this result. Comparing the results we see some differences and similarities:

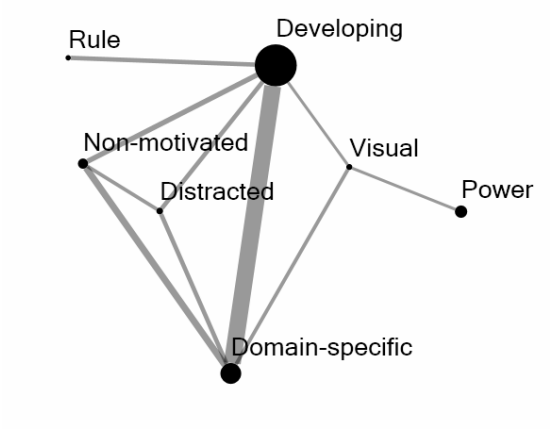

Figure 7 Search role distribution [5]

Differences

The one thing that immediately stands out is the distribution of the roles. Our results showed that the roles were primary distributed over the developing and the power role. Only a small portion is found among the other roles. This is quite a big difference if compared to the results of Druin's study. Also the developing role stands out, but the share of other roles is a lot bigger compared to the power role.

We could not prove the arguments for these differences irrefutably, but it could be explained with the following arguments:

- The study of Druin was performed in a home situation compared to our study in a school setting.
- The age of the participant pool in the work of Druin started at age 7 while our study started at age 9.

Similarities

Besides the differences we also saw similar trends. The developing role occurred more with younger children and the power role more with older children.

3.2 Additional information

During the experiment we have collected information about the experience of children and how much they did or did not like participating.

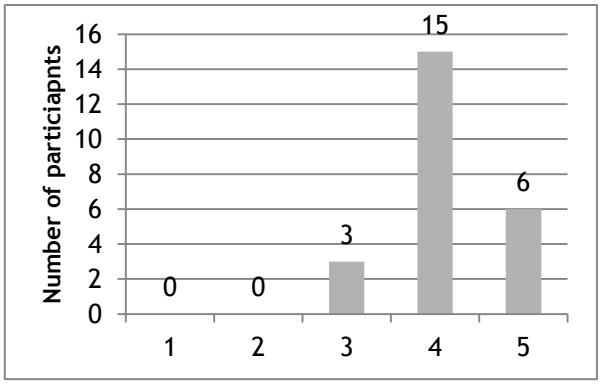

Figure 8 Question 1

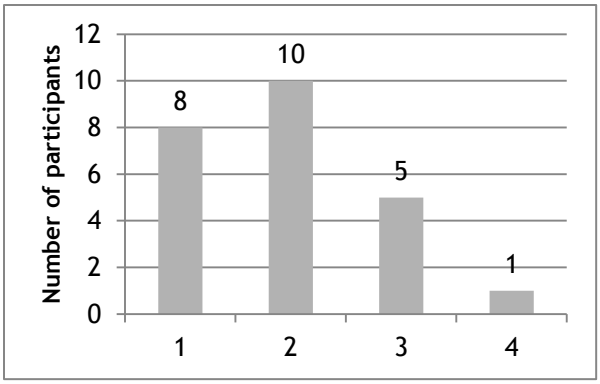

Figure 9 Question 2

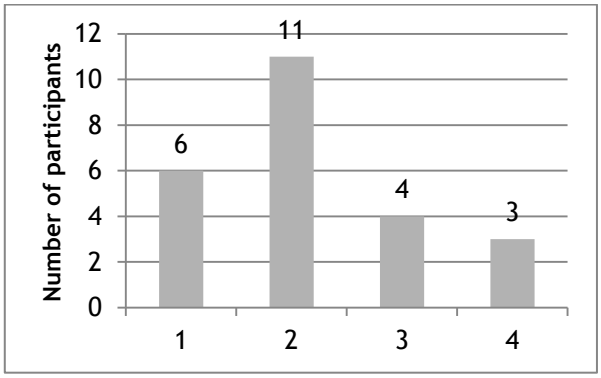

Figure 10 Question 3

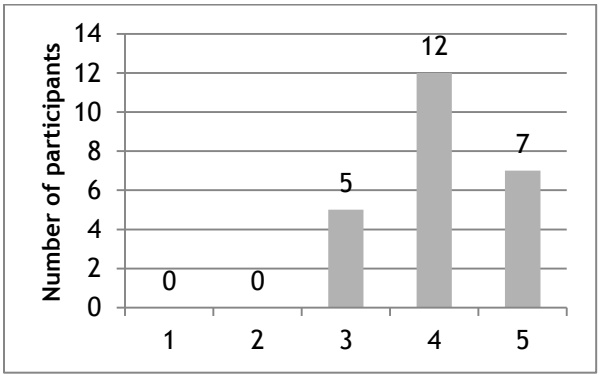

Figure 11 Question 4

Figures 9 and 10 give an indication of the childrens experience with computers and the Internet. As indicated in those figures the children have a reasonable amount of experience.

Figures 8 and 11 are showing the answers of children on the question how they feel about participating in this study. It seems that the children did like to participate.

More interesting is the correlation between search roles and computer and Internet experience.

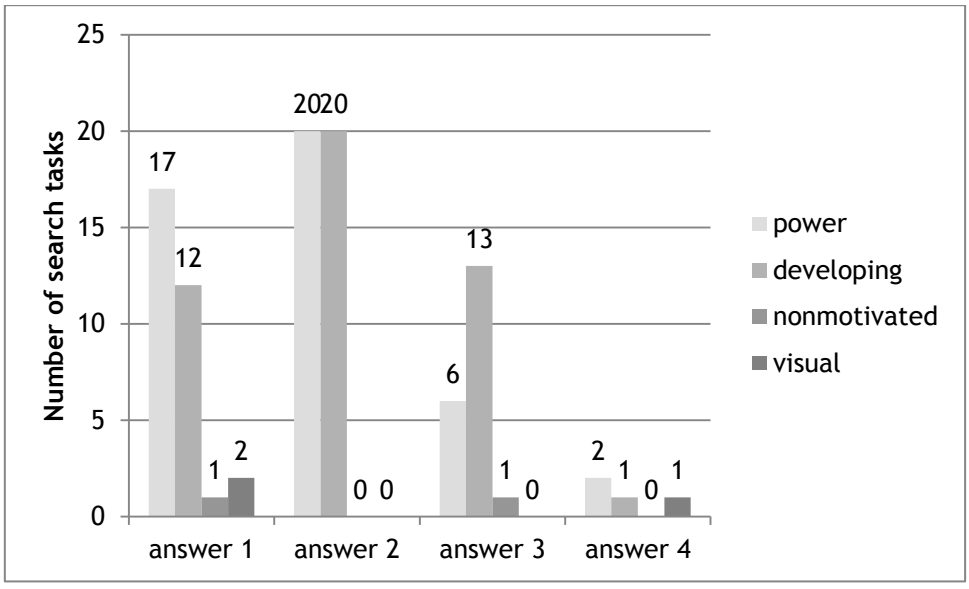

Figure 12 Relation between computer experience (question 2) and search roles

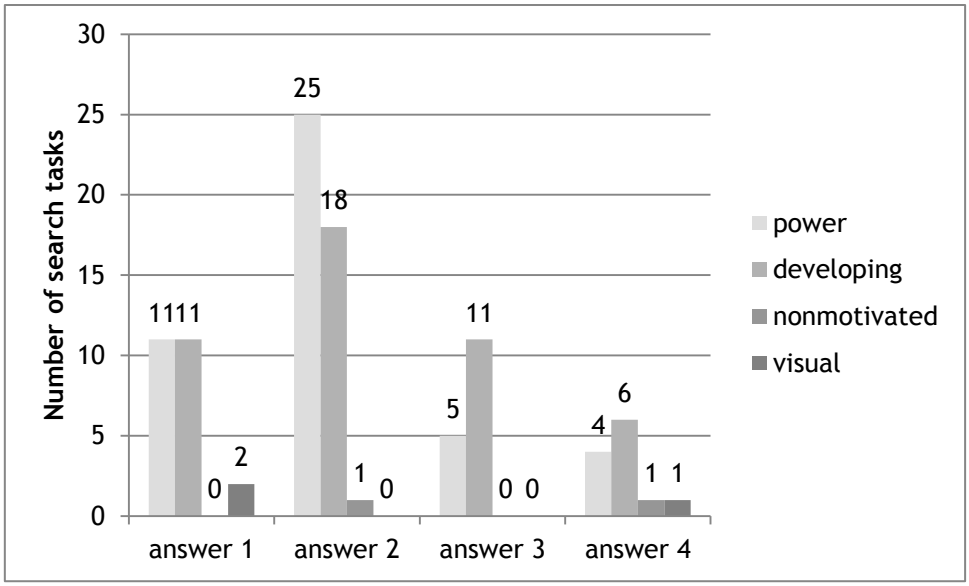

Figure 13 Relation between Internet experience (question 3) and search roles

The answers 1 to 4 in Figure 12 and 13 are the following: everyday (1), 2 or 3 times a week (2), once a week (3) and less than once a week (4). Every participant gave these answers once, but we have 4 labels (possibly different ones). The results are therefore blurred, but still give an indication of the relation between experience and search roles. The power searches are more experienced with a computer and the Internet compared to develop searchers.

3.3 Features

For every task in our data collection we have label describing the search role. The goal of this research is to build a model that can classify the tasks automatically. This model needs to describe the search behavior which is based on several features. Some of these features are based on literature and previous studies. Others are based on the possibilities we saw based on the collected information.

In this section we will give a description of all features we came up with. These features can be categorized in the following way:

1. *Task independent features*
   This set of features is independent of the task type. They will remain the same during the four search tasks. One can think of age, class and gender.
2. *Task dependent direct features*
   These features are task dependent and will probably change during a task. Examples of these features are mouse movements and the number of webpage visits.
3. *Task dependent indirect features*
   This set of features is also task dependent like the last one, but with the difference that these features cannot be taken directly from the data collection. More processing is needed in order to build this set. An example of such a feature is the number of newly added words in the Google search queries compared to the actual search question.

In total we identified 33 different features. This featureset is our final set, we already eliminated several features based on further analysis which is explained in Section 3.5. The number in brackets behind the feature name will be used later on as reference.

1. *Task independent features*
    a. Age (2)
       Age of the participant at the time he or she participated in the experiment. Druin already showed that for example older children are more likely to be power searchers than younger children. [21] Investigated the differences between children and adults as web users. Therefore we expect age to be a powerful feature.
    b. Class (1)
       The class the participant is in according to the Dutch school system. We could say that class is another form of age, because they have a strong correlation.

2. *Task dependent direct features*
    a. Total number of mouse movements (15)
       HCIB records every mouse movement. If the mouse is dragged the plugin records the movement at a steady interval. This feature counts every mouse movement. During the sessions we saw that some children were rather quick and used the mouse a lot, others took more time and did not use the mouse a lot while reading information. Mouse movement could therefore be a good indicator.
    b. Total number of web page visits (7)
       Every single web page that the participant visits sums up. This means that every page that a participant sees counts including all web pages that are visited when using the back button. We would expect children with a lot experience do not need to see a lot of web pages before finding the right information. The number of web page visits could possibly make a clear distinction between power and developing searchers.
    c. Total number of mouse clicks (10)
       Every left mouse click event that is in the data collection is counted and this sums up to a total of left mouse clicks during a task.
    d. Total number of search queries (6)
       Every single query that is put into a search engine like Google or Wikipedia is summed up. We expect experienced children to have a higher chance in finding the right queries sooner compared to children with less experience. Besides that we think that visual searchers are not likely to use a lot of queries.
    e. Typing speed (11)
       The type speed is calculated over a sequence of keyboard inputs without any interruptions with for example mouse moves. The average type speed is taken from all sequences and is set to the number of keyboard inputs per minute.
       Typing is one of the things children need to learn by practice. It is also one of the things children can get frustrated about [22]. Together with several other features it could give an indication of the experience the child has in web search. It could therefore help steering the classification to a certain role.
    f. Time spent on Google (12)
       The total time spend on Google search pages in seconds. Google is the number one tool to find answers in this study. We think that the time spend on Google reveals part of the the search behavior.
    g. Number of back button clicks (13)
       This measure takes the total back button clicks within a task. The back button is one of the mechanisms used in backtracking. It is one of the building blocks in the information seeking model in [3].
    h. Task time (3)
       The total time in seconds a participant is working on a task. The start time is recorded as soon as the participant sees the assignment on the website. The end time is recorded when the participant clicks to the next question on the website.

We expect experienced searchers to need significantly less time compared to other searchers.

i. Number of backspace keystrokes (30)
   The number of backspaces during a search task is summed up. Spelling is one of the possible breakdowns of children. Especially young children have trouble in spelling and therefore are not always capable typing in the right words [22]. The number of backspaces could give an indication of the misspelling.
j. Number of scrolls (31)
   This feature consists of the total number of recorded mouse scrolls. Scrolling is not a basic need when one uses the computer. During the sessions we saw that not every child used the mouse wheel.

3. *Task dependent indirect features*

One of the most important actions performed by the user is to define the search engine input. We defined several features that use the search engine input to retrieve information from. We will not look at the actual content of the queries themselves. The tasks only cover a very limited content domain and in combination with the relatively low number of participants we decided not to define features based on the actual content.

a. Average query distance (14)
   This distance is averaged over the sub-sequent queries based on the Jaccard distance. So first the distance between query 1 and query 2 is taken, next the distance over query 2 and 3 etc. and these distances are averaged.
   The Jaccard distance is based on the token overlap from which zero is a perfect match between two variables and 1 is a perfect mismatch.

Both the increase and decrease of the query input are methods to adjust the query in order to get the necessary search results. Park, Lee and Bae used this feature as part of the descriptive statistics of subsequent query input [23].

b. Number of query increases (18)
   From al subsequent queries (Search engine input) the number of times is counted that the query is increased in number of words. For example from ‘prime minister birthday’ to ‘prime minister netherlands birthday’.
c. Number of query decreases (17)
   From al subsequent queries (Search engine input) the number of times is counted that the query is decreased in number of words. For example from ‘prime minister netherlands birthday’ to ‘prime minister birthday’.

The experienced searcher will try to identify several keywords to use as query input [5]. This automatically means that we have less chance in finding stop words and question words in the query input of power searchers compared to the developing searchers. It also increases the likelihood of larger queries. The following 4 features are based on this argumentation.

d. Question words (19)
   This feature looks at every query input and checks whether a question word is used. Words like why, when, who etc. If one of these words is used, we set the feature on true otherwise on false.
e. Stop words (20)
   This feature counts the average usage of stop words used within queries.
f. Average query length (4)

For every search query the number of words is counted. The number of words is averaged to get the average query length.

g. Standard deviation query length (5)
For every search query the number of words is counted. The standard deviation of the number of words is calculated.

In addition to the previous features we added a feature that measures the difference between the query input and the actual task. The posted questions are written in natural language. Therefore we expect the distance of the search queries compared to the task to be minimal with developing searchers and significantly larger with experienced users.

h. Task distance (21)

Developing searchers tend to use natural language [5]. In order to identify natural language we can use some of the previous described features like the total number of stop words and the number of question words. We can also apply part-of-speech tagging to identify the types of words. We defined the following three features based on verbs, nouns and adjectives:

i. Average number of verbs per search query (22)
The average number of verbs per search query.
j. Average number of nouns per search query (23)
The average number of nouns per search query.
k. Average number of adjectives per query (24)
The average number of adjectives per search query.

Several studies have looked at user authentication based on mouse movement among other user inputs [24,25]. Mouse movement was used to identify a user uniquely. It gives an indication of the behavior of a specific user and how he or she uses the mouse. We are not looking for specific users, but for mouse behavior with groups of users. But chances are that the characteristics can also be categorized. We defined the following 8 features based on this argumentation.

l. Average number of mouse movements (16)
The average number of mouse movements per second.
m. Average mouse move distance (25)
The distance between every two subsequent mouse moves is summed up to a total and divided by the number of distances to get an average.
n. Standard deviation mouse move distance (26)
The distance between every two subsequent mouse moves is calculated. Based on these distances the standard deviation is determined.
o. Average mouse move horizontal distance (27)
The horizontal distance between every two subsequent mouse moves is summed up to a total and divided by the number of distances to get an average.
p. Average mouse move vertical distance (28)
The vertical distance between every two subsequent mouse moves is summed up to a total and divided by the number of distances to get an average.
q. Average mouse move vertical/horizontal distance (29)
The ratio between feature number 28 and feature number 27.
r. Average subsequent move distance (32)
The distance between every two subsequent mouse moves is summed up to a total and divided by the number of distances to get an average.

s. Average mouse move speed (33)
   For every two subsequent mouse moves the distance and time is calculated. Based on these two numbers the mouse speed is calculated. The speed is summed up and averaged.
t. Average number of visits per webpage (8)
   The average number of visits per webpage.
u. Average display time per webpage (9)
   The average display time in seconds per webpage.

3.4 Setup analysis

Due to the low number of instances in our dataset we had to be very careful in how we apply feature selection and classification. The first three of the following steps are also explained in Figure 14.

1. The dataset is divided in a 10% test-set and a 90% training-set. The division is created by applying randomization and stratification. The test-set is only used for the final evaluation.
2. The created training-set is used to make 10 training- and test-set combinations by using the Weka 10-fold cross validation loop. So out the 90% training-set we get 10 separate training- and test-set combinations. Again randomization and stratification are applied.
3. On every of the 10 created training-sets feature selection is applied in a 10 fold cross validation loop. The result of these 10 feature selection rounds are the number of folds every feature is selected in. So for example for training-set number 4 we have a list like the following:

Feature	Number of folds this feature is selected in
1	9
2	5
…	
33	8

   Based on every of the 10 lists we created 11 subsets of features. First we created a subset with all features, next the subsets with only the features that are selected at least once until we got to the subset with only the features that were selected in every of the 10 folds. In total we got 110 different subsets.
4. Every one of the 110 sets is used to apply classification and from every classification result we recorded the percentage of correctly classified instances. We used several classifiers in this step.
5. Based on these 110 results we are able to see which classifier performs best in combination with the feature selection algorithm. The full training-set is used to apply the chosen feature selection algorithm. The corresponding classifier is trained with the 90% (full) training-set and tested against the original 10% test-set.

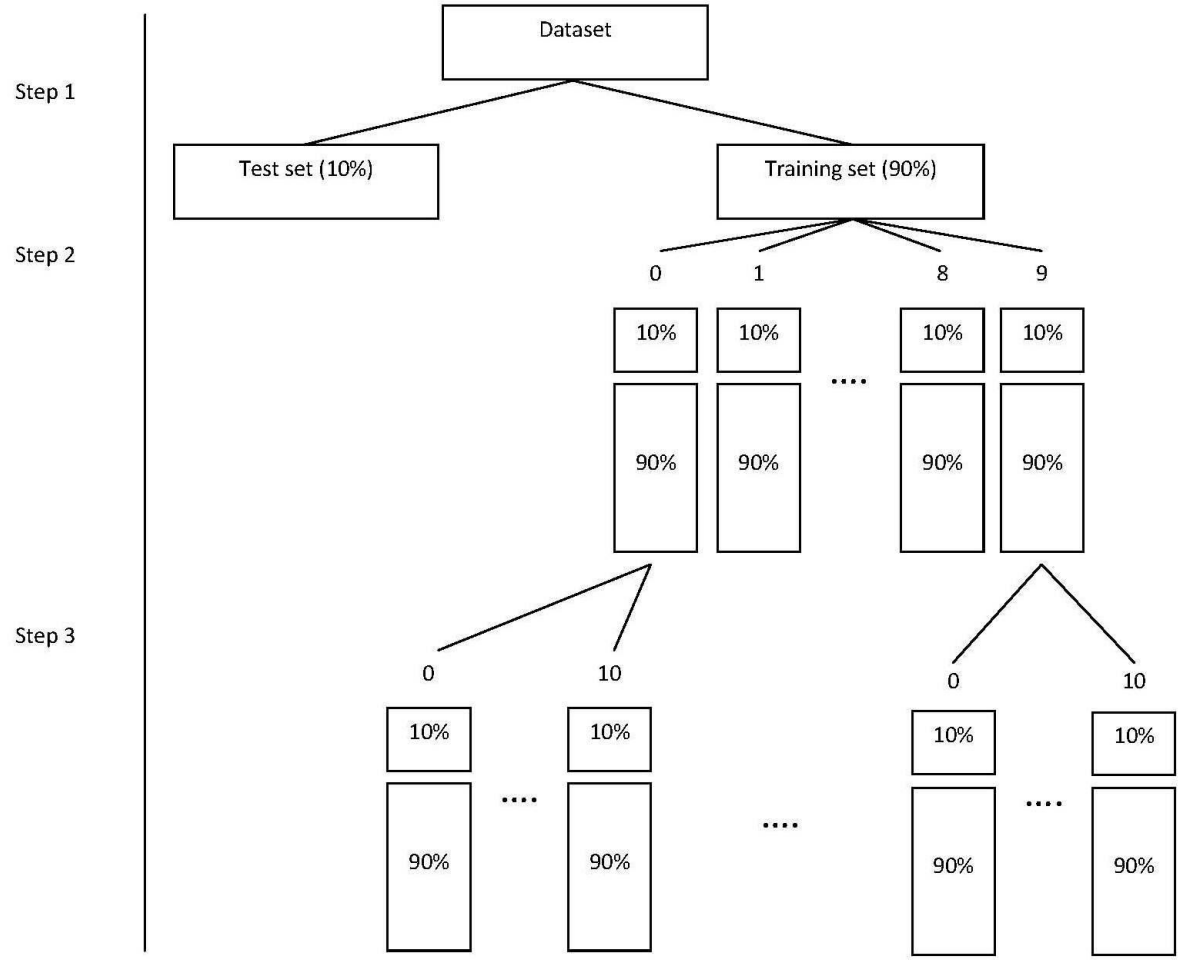

Figure 14 Setup analysis scheme

3.5 Feature selection

A part of almost every classification system is feature selection. Based on the raw data a lot of features are designed. Benefits of feature selection in common are:

- Reducing dimensionality (curse of dimensionality)
- Removing irrelevant or redundant features
- Reducing amount of data needed
- Improving predictive accuracy

In our case the number of data points is relatively low, so we do not have to apply feature selection to reduce computational costs. But we if we compare the number of datapoints to the number of features the problem arises that we are overfitting to the data. We almost get to the point that one feature is able to describe one datapoint. Only for that reason alone we already have to apply feature selection before continuing to the classification phase.

Before applying any feature selection algorithms we look at the features to see whether they have any predictive power. We eliminated two features:

- The ratio between the total time spend on Google images and the total task time. This feature was designed to help identify the visual searcher. Unfortunately due to the low number of visual searchers this feature was not distinctive.

- The number of loops on a page level. So every time a participant returns to a web page which he or she already visited before we counted it as a loop. We defined this feature based on the work of …. In our study it appeared that none of the participants visited the same web page twice or more. This feature was therefore removed from our featureset.

Weka

We use the WEKA (Waikato Environment for Knowledge Analysis) [26] to perform the feature selection or our dataset. Weka is a collection of machine learning algorithms which can be used in classification, clustering and feature selection. Besides a GUI Weka offers an API which makes it possible to incorporate the possibilities of Weka in our own Java program.

With feature selection we try to find a subset of the complete feature space that describes the data best. In order to do an exhaustive search we would require $2^{number\ of\ features}$ tests in order to obtain the best result. Although our feature space is not too big, it is already unpractical to do an exhaustive search. Therefore we need to resort to suboptimal algorithms like genetic or greedy algorithms. Feature selection consists of two parts. At first we need a method to search the space effectively and still get an optimal result. Besides searching the feature space the selected features should be evaluated to test the performance of a particular subset of features.

In order to reduce the number of comparisons we chose to use one evaluation method for our feature selection and two different search methods. We tried several evaluation methods and finally chose the Logistic classifier which yielded the best results. For search methods we chose the widely used greedy stepwise (backward search) and the more advanced genetic search algorithm.

Cross validation is applied on the training set with the before mentioned algorithms. The resulting tables can be found in Appendix C.

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	10	10	10	10	10	10	10	10	10	10	10,0
2	10	9	10	9	10	8	9	9	8	10	9,2
3	10	10	10	9	10	8	8	10	7	8	9,0
4	9	10	9	10	9	9	10	9	10	9	9,4
5	10	9	10	10	9	10	9	9	10	10	9,6
6	10	8	9	8	10	10	9	10	8	9	9,1
7	9	9	10	7	9	7	8	9	9	9	8,6
8	10	10	10	10	10	9	10	10	10	10	9,9
9	7	9	9	10	8	7	9	10	10	7	8,6
10	8	7	8	10	8	7	6	9	7	7	7,7
11	8	9	10	6	10	10	10	7	8	8	8,6
12	8	9	10	7	8	8	9	7	8	7	8,1
13	10	10	10	6	6	9	10	10	10	10	9,1
17	10	10	2	8	9	10	9	8	8	10	8,4
19	3	9	8	7	9	8	7	3	9	10	7,3

Table 3 Partial feature selection results, greedy stepwise (BF)

Table 3 gives us a partial view on the feature selection results based on the greedy stepwise back ward search algorithm.

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	10	10	10	10	10	9	10	9	10	9	9,7
2	3	1	3	5	3	4	2	3	1	3	2,8
6	9	5	9	9	9	9	10	3	7	9	7,9
23	9	8	9	6	9	9	9	9	10	9	8,7
30	8	6	3	5	8	9	10	8	7	8	7,2

Table 4 Partial feature selection results, genetic search

Comparing the two tables we can see a big difference between the two feature selection algorithms. The greedy stepwise algorithm takes the first 15 features almost anytime and disregards the features 26 to 33 while the genetic search picks features from the whole set and only has a few features selected almost every time. Interesting difference is feature number 2. This feature corresponds to the grade the participant is in. The genetic search algorithm does pick this feature significantly less compared to greedy stepwise. Possibly because of the correlation between feature 1 (age) and feature 2 the genetic search algorithm decides the information gain is not significant enough.

The results of these two feature selection rounds will be used to create 11 subsets from every of the 10 sets. First we create subsets including every feature, next subsets with only the feature that are at least selected once, until we have only the subsets with features that are selected in every round. The 110 subsets will be used in classification in order to identify which threshold of feature to use and which search algorithm to use.

3.6 Classification

In the previous section we have created a total of 110 training- and test-set combinations for feature selection with greedy stepwise and for feature selection with genetic search. This in itself already holds cross validation and therefore we do not have to apply any additional tricks to prevent overtraining. Several well known classifiers are available within Weka. We chose to use the following classifiers:

- ZeroR (baseline): dominant class selection
- Logistic Regression [27]
- Naive Bayes [28]
- J48 (decision tree) [29]
- MLP (Multi Layer Perceptron)

Table 5 lists the classification results of the logistic classifier based on the results of the greedy stepwise feature selection including the average and median per threshold. Other results can be found in Appendix D.

Threshold\Set	0	1	2	3	4	5	6	7	8	9	AVG	Med
0	67%	56%	67%	67%	56%	33%	75%	50%	63%	63%	60%	63%
1	56%	67%	56%	44%	44%	67%	50%	50%	63%	63%	56%	56%
2	67%	44%	56%	44%	44%	67%	50%	50%	63%	63%	55%	53%
3	67%	44%	56%	44%	44%	67%	50%	50%	75%	63%	56%	53%
4	67%	44%	56%	56%	67%	67%	50%	50%	63%	63%	58%	59%
5	67%	56%	56%	56%	67%	56%	63%	50%	75%	63%	61%	59%
6	67%	56%	56%	56%	67%	67%	63%	75%	75%	63%	64%	65%
7	67%	56%	67%	56%	67%	67%	63%	75%	75%	75%	67%	67%
8	44%	56%	67%	67%	67%	78%	38%	75%	75%	63%	63%	67%
9	67%	67%	67%	56%	56%	78%	38%	75%	50%	63%	61%	65%
10	67%	78%	78%	56%	78%	67%	63%	75%	50%	75%	69%	71%

Table 5 Classification results logistic classifier based on greedy stepwise feature selection

The two graphs below give an overview of the total result set of all used classifiers in which the results are averaged per classifier. Figure 15 displays the results with the subsets selected by the greedy stepwise algorithm; Figure 16 displays the genetic search results. The corresponding averaged percentages can be found in Appendix D.

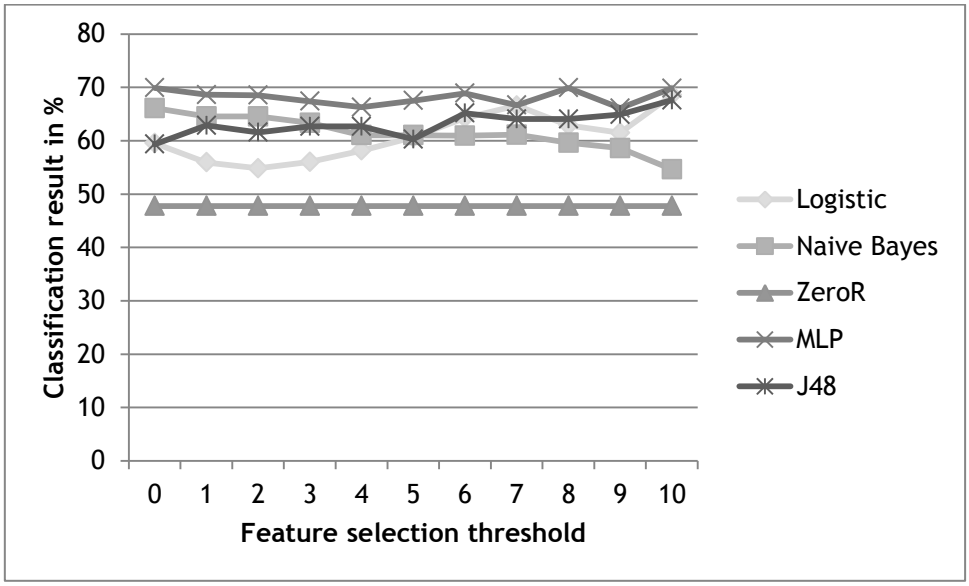

Figure 15 Classification results based on greedy stepwise (BS) feature selection

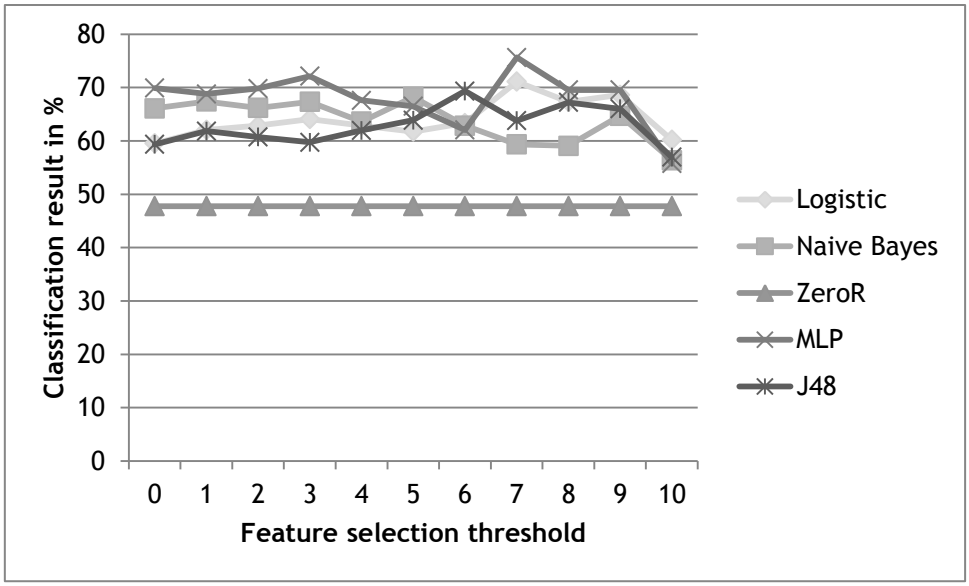

Figure 16 Classification results based on genetic search feature selection

The (Weka) ZeroR classifier is used as baseline in this experiment. This algorithm always chooses the most common class. Based on the numbers we can at least say that any of the classifiers give a better result than the baseline. But we need a statistical test to see whether the difference is significant. We conducted a paired t-test to confirm this. The t-test is used with $\propto = 0.05$.

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$2{,}21 \cdot 10^{-6}$	$6{,}29 \cdot 10^{-8}$	$4{,}53 \cdot 10^{-13}$	$8{,}61 \cdot 10^{-10}$

Table 6 Paired t-test on classification results based on greedy stepwise feature selection

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$3{,}62 \cdot 10^{-8}$	$9{,}82 \cdot 10^{-8}$	$1{,}68 \cdot 10^{-7}$	$9{,}63 \cdot 10^{-8}$

Table 7 Paired t-test on classification results based on genetic search feature selection

The results of the statistical test indicate that the classifiers do perform significantly better than the baseline on both feature selection with both genetic search greedy stepwise backward search. The best combination of feature selection settings and classifier can be found in Appendix D. The genetic search feature selection algorithm at threshold 7 in combination with the MLP classifier gives the best performance at 76%.

3.7 Success prediction

So far in our research we examined the use of search behavior roles with the study of Druin as our starting point. We have obtained results with automatic classification. At this point we would want to look back at the original problem description. In that description a classifier based on success prediction would be more suitable. In this section we will look at the possibilities of prediction success with our selected features.

3.7.1 Success labels

We labeled every search task whether the participants succeeded in finding the answer or not. Questions one, two and four are rather straightforward to label. The answers to these questions are

just right or wrong. The answer to the third questions is less trivial. We looked at the answer and labeled it as success if it said something about the first car ever built like how it looked, what time it was invented or who created it. After labeling the complete set, the distribution looks as follows:

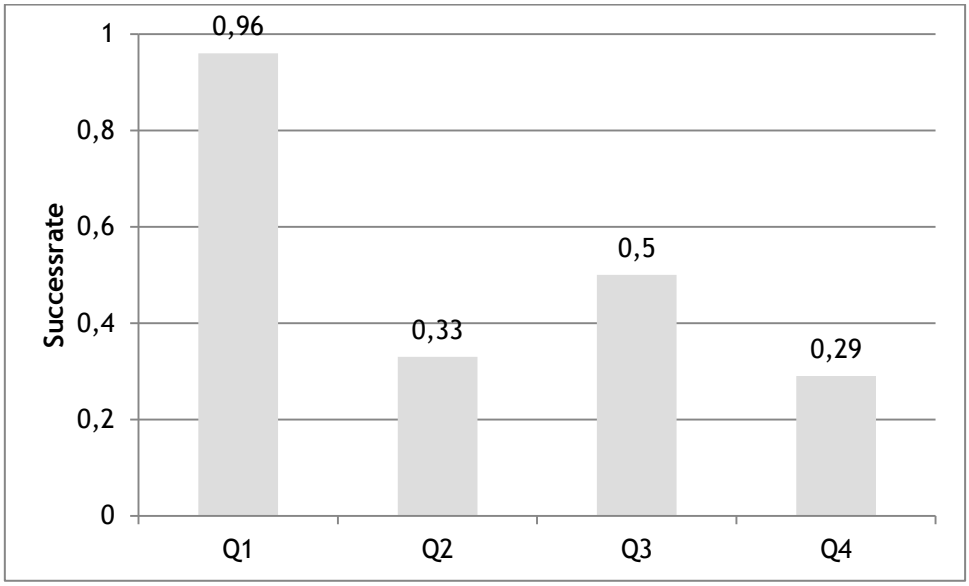

Figure 17 Successrate per question

This chart already shows two interesting things:

- Participants did not have trouble in finding the answer to question 1. Only one the participants could not find the answer. Possilby seems to be easy for all participants, only one of them could not find the answer.
- Question number 1 and 2 are of the same type, but participants had more trouble finding the correct answer to question 2. The information needed for question 2 is not directly given in the search result snippets given by Google. An indication of the answer to question 1 is already in these snippets. This could explain the difference in success rates.

In order to give more insight in the distribution of the data we also created a figure to compare the participants per age and per task:

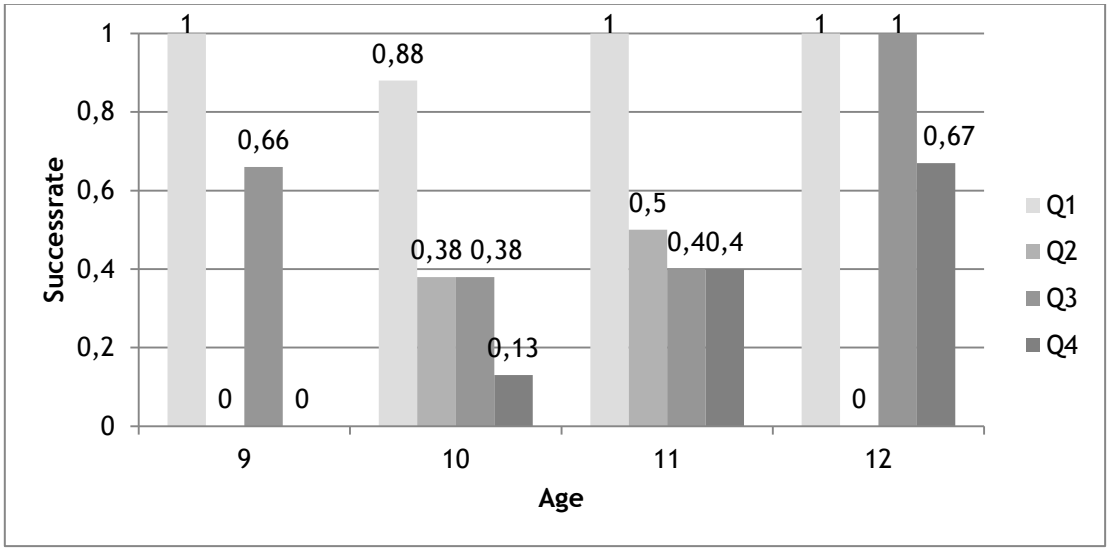

Figure 18 Successrate per age per question

Figure 18 is blurred, because the participants are not equally distributed per age. But still it gives us additional information about our participants:

- Older children were more likely to solve question 4.
- Based on success rates there does not seem to be much difference between 10 and 11 year old children.

3.7.2 Feature selection

We applied feature selection on the same featurset as we did with the search roles. Table 8 and 9 show the corresponding (partial) results.

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	9	7	8	10	9	10	10	8	10	8	8,9
2	10	10	10	10	10	10	10	10	10	10	10,0
3	9	7	10	10	9	7	10	9	8	10	8,9
4	8	10	10	9	7	8	8	9	9	8	8,6
5	10	9	7	10	8	10	10	9	7	10	9,0
6	9	9	7	10	10	10	9	10	8	8	9,0
7	8	10	6	9	8	7	8	7	8	9	8,0
8	9	9	8	9	9	10	8	7	9	9	8,7
9	8	10	6	10	9	8	8	9	8	8	8,4
10	7	6	6	6	10	8	8	8	7	10	7,6
11	9	8	5	7	7	8	8	9	7	8	7,6
12	10	9	9	8	10	7	9	9	9	9	8,9
13	9	10	6	10	10	10	10	9	8	8	9,0
14	10	10	10	9	10	9	10	9	10	10	9,7
15	6	10	10	10	9	10	8	10	7	8	8,8
16	10	9	10	10	10	9	10	10	10	9	9,7
17	10	9	10	10	10	10	10	10	10	10	9,9
18	8	10	6	10	7	9	9	10	10	10	8,9
19	6	10	9	8	7	8	8	9	7	7	7,9
20	10	10	9	9	10	9	9	9	8	9	9,2
21	9	6	8	10	10	9	8	10	8	10	8,8
23	10	10	10	10	9	7	9	8	10	10	9,3
25	10	7	10	9	8	7	10	8	10	9	8,8

Table 8 Partial feature selection results, greedy stepwise (BF)

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	5	5	7	7	8	8	9	4	3	6	6,2
2	7	9	5	4	6	8	5	8	7	5	6,4
14	9	6	6	6	9	10	10	6	7	7	7,6
20	7	10	6	7	3	9	8	8	7	7	7,2
23	9	10	9	10	9	6	8	10	9	8	8,8
25	6	8	6	7	6	8	7	8	7	7	7
26	9	7	9	9	7	8	7	8	10	8	8,2
30	7	8	7	8	8	9	6	8	10	9	8
32	0	7	10	8	7	6	10	9	9	8	7,4

Table 9 Feature selection results, genetic search

Comparing Table 8 and Table 9 we see that again the genetic search algorithm selects fewer features. Another interesting difference is feature number 1 (age). The genetic search algorithm selects this feature significantly less while in the search role feature selection this feature is selected almost everytime with both algorithms. Possibly success is not so strongly correlated with age compared to the search roles.

We applied the same steps as in Section 3.5. So we used the feature selection results to create 11 subsets of features.

3.7.3 Classification

We also used the same set of classifiers. The results can be found in the Figure 19 and Figure 20.

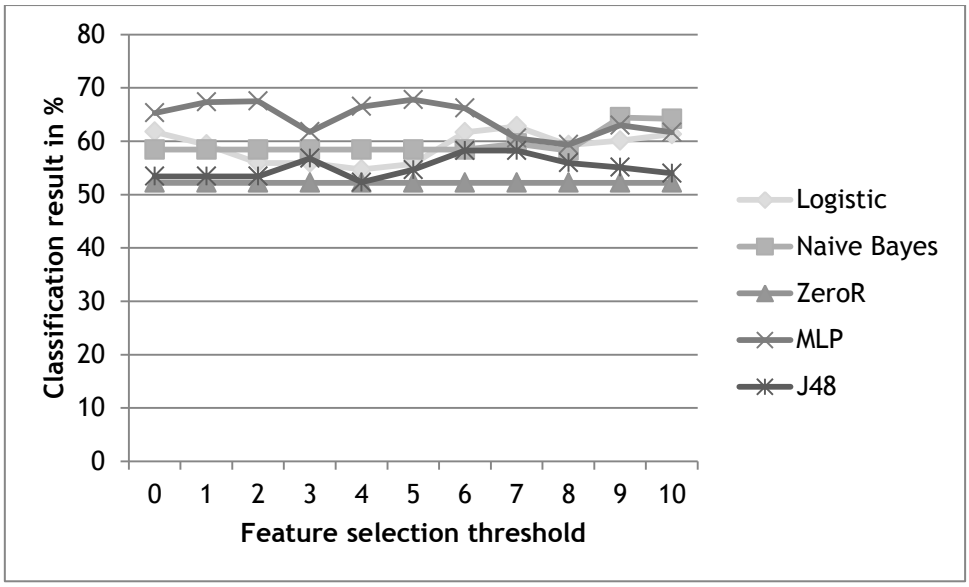

Figure 19 Classification results based on greedy stepwise (BS) feature selection

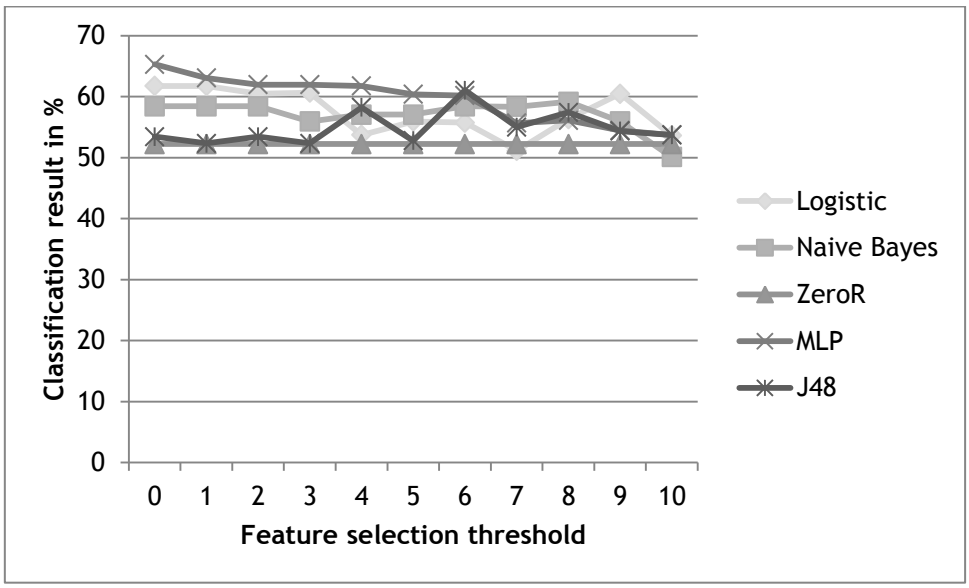

Figure 20 Classification results based on genetic search feature selection

The ZeroR classifier is again our baseline. This baseline is a bit higher compared to the baseline in predicting the roles. Also the classification results are lower compared to the results in role prediction. We conducted the same paired t-test with the same alpha level to investigate significance. The results are listed in Table 10 and 11.

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$1{,}58 \cdot 10^{-5}$	$1{,}18 \cdot 10^{-6}$	$1{,}37 \cdot 10^{-7}$	$9{,}15 \cdot 10^{-4}$

Table 10 Paired t-test on classification results based on greedy stepwise feature selection

The results from Table 10 indicate that all of the four classifiers outperform the baseline classifier significantly.

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$9{,}98 \cdot 10^{-4}$	$9{,}26 \cdot 10^{-5}$	$9{,}67 \cdot 10^{-5}$	$9{,}82 \cdot 10^{-3}$

Table 11 Paired t-test on classification results based on genetic search feature selection

Table 11 gives the classification results after the genetic search feature selection. Also In this case only the J48 classifier does not give a significant improvement over the baseline.

3.8 Combination of search roles and success

So far we have looked at predicting search roles and predicting success. A next step is to look at the combination of these two labels. First we will look at the distribution two labels combined. After that we will study whether knowing the search roles can influence the prediction of success.

Question	Developing role	Power role
Q1	100%	100%
Q2	25%	50%
Q3	23%	70%
Q4	8%	55%

Table 12 Success per search role per task

We made a chart which displays the success rate per role. From this point forward we will focus on the two main identified roles (developing and power). The other roles occur very rarely and because of their low numbers we cannot say anything useful about it, therefore these data points are left out.

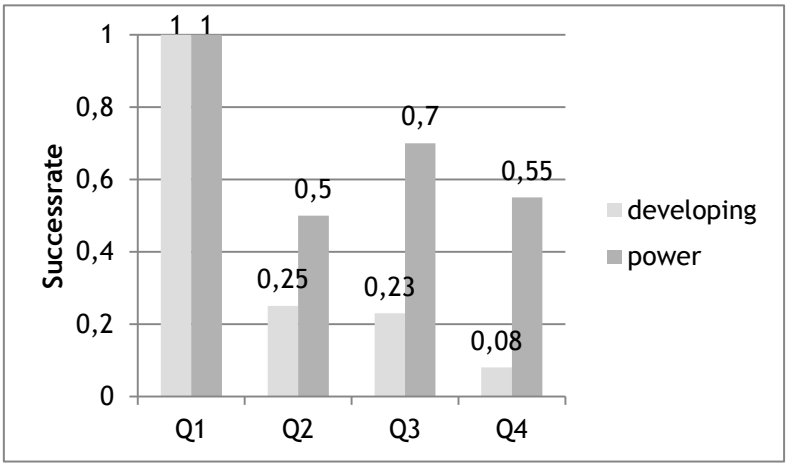

Figure 21 Successrate per question per role

Figure 21 indicates a significant difference between the successrate of a developing searcher and a power searcher. To confirm this we applied the paired t-test.

At this point it would be interesting to see whether the search roles could improve the prediction of success. In order to test this we took our initial set of features and added the role label to it. We applied the same two feature selection algorithms. The results of the feature selection step can be found in Appendix C. The feature selection results for the search role feature are shown in Table 13 and Table 14.

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
34	2	0	0	1	3	0	1	0	0	3	1,0

Table 13 Feature selection result of the search role feature based on greedy stepwise backward search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
34	10	10	10	10	9	10	8	10	10	10	9,7

Table 14 Feature selection result of the search role feature based on genetic search

Table 13 shows us that the search role feature is almost never selected. Based on these results we can conclude that this feature will not improve the classification results. The results in Table 14 however indicate the opposite. Therefore we continued the classification round only with the results of the genetic search feature selection.

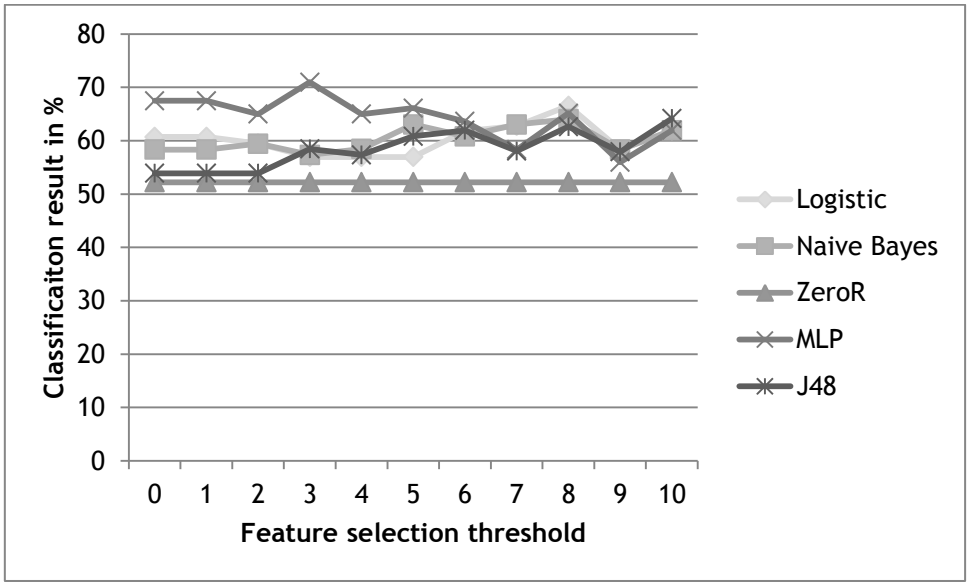

Figure 22 Classification results success prediction including search roles based on genetic search feature selection

We confirmed the visible improvement of the 4 classifiers over the baseline with the paired t-test.

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$4{,}24 \cdot 10^{-6}$	$5{,}26 \cdot 10^{-7}$	$2{,}79 \cdot 10^{-6}$	$2{,}01 \cdot 10^{-4}$

Table 15 Paired t-test on classification results based on genetic search feature selection including search roles

As shown in Table 15 the 4 classifier all perform significantly better than the baseline classifier. But the more interesting question is whether the classifiers can outperform themselves by including the search roles as feature. First we created 4 figures (Figure 24 to 27) to show the difference.

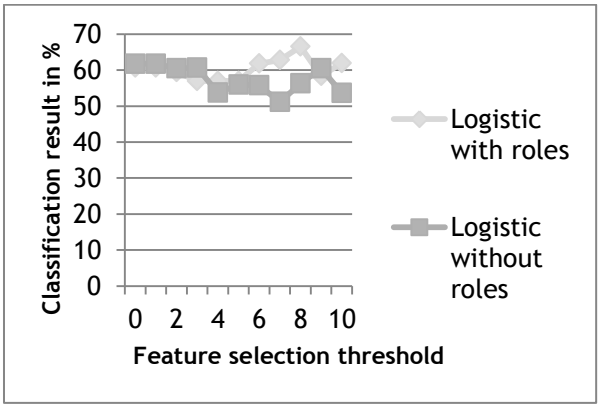

Figure 23

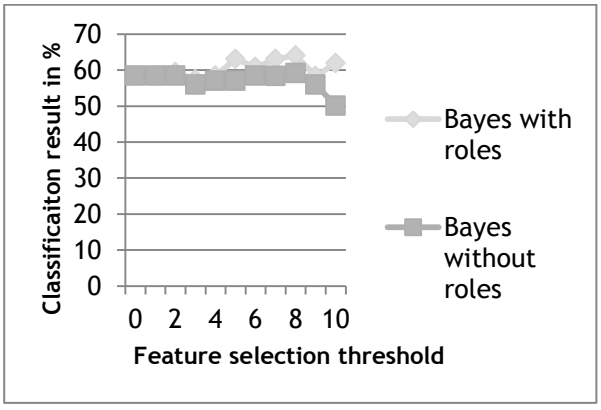

Figure 24

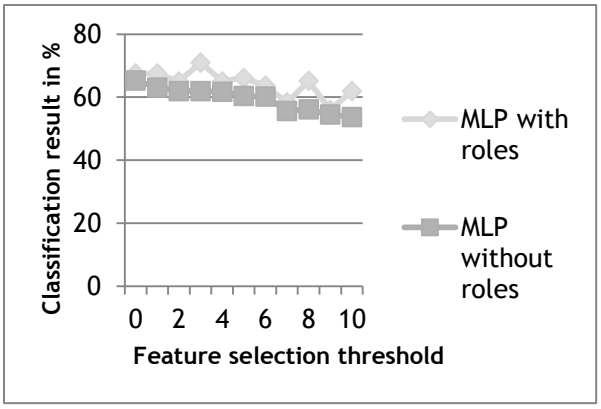

Figure 25

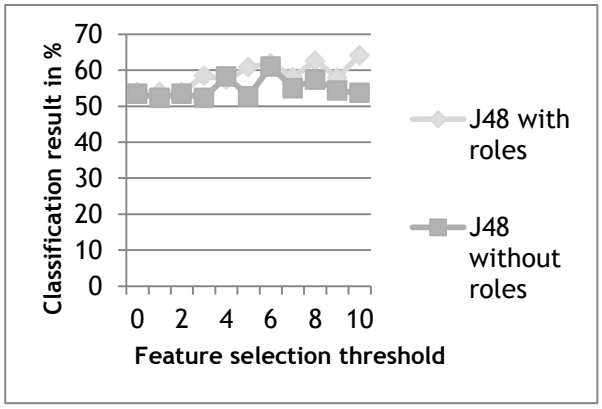

Figure 26

We applied the same paired t-test to compare the classification results of the 4 classifiers (based on genetic search feature selection) with and without the search roles as feature.

Classifier	Logistic	Naïve Bayes	MLP	J48
$P(T \leq t)$	$1{,}08 \cdot 10^{-1}$	$1{,}12 \cdot 10^{-2}$	$2{,}04 \cdot 10^{-4}$	$8{,}11 \cdot 10^{-3}$

Table 16 Paired t-test classification results based on genetic feature selection based on featureset with and without the search roles as feature

Except for the Logistic classifier the results show a significant improvement when including the search roles as feature [Table 16].

4. Evaluation

In the previous chapter we have learned which feature selection method to use, which threshold for features to use and which classifier performs best. Based on this information we use the test-set and training-set created in step 1 of our analysis setup. Feature selection is applied to the training-set in a 10-fold cross validation loop. The classification is tested against the (unseen) test-set.

4.1 Search role prediction

Based on the results from Chapter 3 we took the following algorithms to perform a final evaluation:

- Search algorithm: Genetic search
- Feature threshold: 7 (all features selected 7 times or more in a 10-fold CV)
- Classifier: MLP

The full training-set is used to perform a 10-fold cross validation feature selection with the above mentioned algorithm.

The following features are selected based on the given threshold: 1, 4, 6, 14, 19, 23, 24, 26, 27, 30, 33 (total of 11 features)

Classification is performed with the MLP classifier. The number of correctly classified instances is 50%. The corresponding confusion matrix can be found in Table 17.

Developing	Power	Nonmotivated	Visual	<- classified as
2	2	0	1	Developing
2	3	0	0	Power
0	0	0	0	Nonmotivated
0	0	0	0	Visual

Table 17 Confusion matrix search role prediction

Table 12 is compared to the following baseline in Table 18 which is retrieved by using ZeroR as classifier:

Developing	Power	Nonmotivated	Visual	<- classified as
5	0	0	0	Developing
5	0	0	0	Power
0	0	0	0	Nonmotivated
0	0	0	0	Visual

Table 18 Confusion matrix baseline role prediction

This baseline leads to a classification result of 50% correctly classified instances. This means that we cannot outperform the baseline classifier in our final classification. We only have 10 instances to test with, which means for every wrongly classified instance we lose 10% in performance. In combination with the results from Chapter 3 we expect a significant improvement when more data is available.

In Chapter 3 we have applied statistics to investigate the statistical significance of the classifiers. In this case we have only 1 classification result to compare with. It is therefore not sensible to apply a statistical test in this case. But we can look at how confident the classifier is about its decisions [Table 19].

Number	Actual label	Predicted label	P(developing)	P(power)	P(nonmotivated)	P(visual)
1	Developing	Developing	0.973	0.000	0.022	0.005
2	Developing	Visual	0.383	0.043	0.034	0.540
3	Developing	Developing	0.940	0.000	0.010	0.050
4	Developing	Power	0.010	0.988	0.001	0.001
5	Developing	Power	0.024	0.734	0.200	0.042
6	Power	Power	0.001	0.916	0.080	0.003
7	Power	Power	0.000	0.966	0.004	0.029
8	Power	Developing	0.650	0.004	0.005	0.341
9	Power	Power	0.000	0.973	0.020	0.007
10	power	developing	0.842	0.001	0.153	0.005

Table 19 Confidence scores on the predicted search role labels

The perfect result would be a very high chance on the correct label and a low chance on the other labels. For example with number 1 in Table 19. Unfortunately we also see a high confidence on number 4 while it is the wrong label. We can see in Table 19 that the classifier is highly confident on the correctly labeled instances. This is not always the case with a wrongly labeled instance like with number 8. So we possibly have room for improvement in further research.

4.2 Success prediction

Compared to the search role classification results the absolute differences in percentages between the baseline and the classifiers are smaller. But almost all combinations of classifiers and feature selection methods do give a significant improvement. For our final evaluation we again took the best performing combination:

- Search algorithm: Greedy stepwise (backward search)
- Feature threshold: 5 (all features selected 1 times or more in a 10-fold CV)
- Classifier: MLP

The full training-set is used to perform a 10-fold cross validation feature selection with the above mentioned algorithm.

The following features are selected based on the given threshold: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 22, 24, 25, 26, 27, 28, 29, 30 (total of 28 features)

Classification is performed with the MLP classifier. The percentage of correctly classified instances is 60% with the following confusion matrix:

No success	Success	<- classified as
4	1	No success
3	2	Success

Table 20 Confusion matrix success prediction

Table 20 is compared to the following baseline [Table 21] which is retrieved by using ZeroR as classifier:

No success	Success	<- classified as
0	5	No success
0	5	Success

Table 21 Confusion matrix baseline success prediction

This baseline leads to a classification result of 50% correctly classified instances. It means that we can outperform our baseline. Based on this single result we cannot apply statistics to confirm significance as explained in Section 4.1. But comparing this result to the findings in Chapter 3 we are confident that this result is very likely to be significant.

4.3 Search roles as feature

In chapter 3 we have also looked at the possibility of using search roles as feature in success prediction. The results were significantly better compared to the situation were the search roles were not included as feature.

Also in the case we choose the best performing combination which gave a classification result of 71%:

- Search algorithm: Genetic search
- Feature threshold: 3 (all features selected 1 times or more in a 10-fold CV)
- Classifier: MLP

The full training-set is used to perform a 10-fold cross validation feature selection with the above mentioned algorithm.

The following features are selected based on the given threshold: 1, 2, 3, 4, 5, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34 (total of 30 features)

Classification is performed with the MLP classifier. The number of correctly classified instances is 50% with the following confusion matrix:

No success	Success	<- classified as
2	3	No success
2	3	Success

Table 22 Confusion matrix success prediction including search roles as feature

Table 23 is compared to the following baseline [Table 24] which is retrieved by using ZeroR as classifier:

No success	Success	<- classified as
0	5	No success
0	5	Success

Table 23 Confusino matrix baseline success prediction

This baseline leads to a classification result of 50% correctly classified instances. This means that in this case we also cannot outperform the baseline classifier in our final classification. Using the same arguments as in Secion 4.1 we expect this result to be better when more data is available.

5. Conclusion

5.1 Conclusions

5.1.1 Role prediction

We started this research based on the results of Druin [5] in social science. Based on the results of Chapter 3 we can conclude that the chosen classifiers all perform significantly better than the baseline. Comparing the two feature selection algorithms and four classifiers we can achieve a classification result on the training data of at least 70% with genetic search and the MLP classifier. Unfortunately we could not provide a significant improvement over the baseline in our final evaluation.

5.1.2 Success prediction

Besides role prediction we also looked at success prediction. The results of Chapter 3 show a significantly better classification compared to the corresponding baseline. We achieved a classification result of at least 65%. We suspected these results to be better than the search role prediction. But it seems that predicting success with the given featureset is more difficult. Our final evaluation shows a 60% classification result compared to the 50% baseline.

To improve success prediction we added the search roles as feature. The MLP, Naïve Bayes and the J48 classifiers showed a significantly better classification result compared to the case without this feature. In the final evaluation we could not require an improvement; both the baseline and the MLP classifier retrieved a classification result of 50%.

5.1.3 Educational implications

The use of search roles is not that obvious. Roles like the rule-bound searcher and the visual searcher seemed to be based on a strategy while the developing and power searcher seemed to be based on experience. This difference between types of roles makes it more difficult to apply in a practical situation. It does not mean that it cannot be used at all. For example we can imagine courses that help children, specifically based on their search role, browse the Internet. We also think it can be used in a kind of auto-help function that aids children browsing the Internet. This is more elaborated in Section 5.2.3.

The introduction already states a possible application for the use of success. We think success prediction could easily be implemented in a school setting and could help the teacher to focus on children who have trouble searching the Internet. But looking at the classification results more research is needed before such a system would beneficial in practice.

5.2 Future work

The introduction of this thesis mentioned a classroom situation in which a teacher gives the class a web search assignment. This study is a first step in creating a computer program which helps teachers in identifying students who have trouble finding the right answer. Further research is needed before such a program would be applicable in practice. We will present several directions in which more research would be necessary or could lead to other interesting results.

5.2.1 Success and role prediction

We started this research with 7 distinctive roles identified in [5]. We also looked at the possibility of using the defined featureset in success prediction. Although we have shown that we can significantly improve classification over our baseline more research is needed to improve the results even more. The foucus in that research should be on success prediction, because this kind of classification can be more easily implemented in practice and would definitely save the teacher time in identifying children who have trouble browsing the web.

5.2.2 Intermediate classification

In this study we have focused mainly on classification on the complete dataset. In practice such a system could be used to identify the search roles. Based on the search roles a teacher could for example give specific guidelines to the students. Intermediate classification is not necessary in such a situation. But if we look at the success classification we do need intermediate classification. It is not useful to have a system that can tell you afterwards whether you succeeded or not. In practice we would like to have a system that could give an indication of your progress after two minutes for example. In that case a teacher would have the option to step in before the time limit is reached. Therefore we recommend further research in intermediate classification on several time limits for example at 2,4 and 6 minutes.

5.2.3 Auto-help

During the sessions we saw that several children already had trouble starting the search. For example with a relatively long question these children started to type every word of it at a very low speed. Other children for example looked at several results pages per query. This kind of search behavior is often not successful. Besides a teacher that helps children based on the systems success classification we could use a system that detects certain behavior and could give specific feedback. For example if the system detects that a child is looking at several results pages for one query it could suggest changing the query. Such a feature would save the teacher additional time.

6. References

Below a complete list of references which are used in this report.

[1] EU Kids Online: Final Report, Livingstone, S., Haddon, L., EC Safer Internet Plus Programme Deliverable D6.5, ISBN 978-0-85328-355-3 2009, page 5: table 1.

[2] Shenton, A. K., Dixon P., Models of young people's information seeking, J. Librarianship and Inf. Sci., 35, 1 (2003), 5-22.

[3] Bilal, D., Sarangthem, S., Bachir, I., Toward a Model of Children's Information Seeking Behavior in Using Digital Libraries.

[4] How Children Search the Internet with Keyword Interfaces, Druin, A., Foss, E., Hatley, L., Golub, E., Guha, M. L., Fails, J., Hutchinson, H. Proceedings of Interaction Design and Children (IDC 2009), Cuomo, Italy, 89-96.

[5] Children's Roles Using Keyword Search Interfaces at Home, Druin, A., Foss, E., Hutchinson, H., Golub, E., Hatley, L. CHI 2010.

[6] Dutch law, Wet van 26 februari 1998, houdende regelen inzake medisch-wetenschappelijk onderzoek met mensen (Wet medisch-wetenschappelijk onderzoek met mensen), http://wetten.overheid.nl/BWBR0009408.

[7] Dutch law, Wet van 6 juli 2000, houdende regels inzake de bescherming van persoonsgegevens (Wet bescherming persoonsgegevens), http://wetten.overheid.nl/BWBR0011468.

[8] European directive, Directive on privacy and electronic communications, 12-07-2002, http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2002:201:0037:0047:EN:PDF.

[9] European directive, 25-11-2009, http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2009:337:0011:0036:EN:PDF.

[10] Ethics Manual, Landoni, M., Kruisinga, F., 08-10-2010.

[11] Information seeking strategies of novices using a full-text electronic encyclopedia, Marchionini, G. Journal of the American Society for Information Science, 40(1), 54–66, 1989.

[12] Users, tasks and the Web: Their impact on the information-seeking behavior, Kim, K., S., Proceedings of the 21st national online meeting, USA, pp.189–198, 2000.

[13] Eye-tracking analysis of user behavior in WWW search, Granka, L., A., Joachims, T., Gay, G., Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 478 – 479, 2004.

[14] Knowing the User's Every Move – User Activity Tracking for Website Usability Evaluation and Implicit Interaction, Atterer, R., Wnuk, M., Schmidt, A., Proceedings of the 15th international world wide web conference, 2006.

[15] large scale with usaproxy.

[16] Morea Recorder, http://www.techsmith.com.

[17] HCI Browser: A Tool for Studying Web Search Behavior, Capra, R., Workshop on Understanding the User, SIGIR 2009.

[18] CamStudio 2.0, http://camstudio.org.

[19] Cohen J. A coefficient of agreement for nominal scales. Educational and Psychological Measurement 1960;20:37-46.

[20] Landis JR, Koch GG. The measurement of observer agreement for categorical data. Biometrics 1977;33:159-74.

[21] Differences and similarities in information seeking: children and adults as Web users, Bilal, D., Kirby, J., Information Processing and Management 38 (2002) 649 – 670.

[22] Hutchinson, H., Druin, A., Bederson, B.B., Reuter, K., Rose, A., Weeks, A.C. How do I find blue books about dogs? The errors and frustrations of young digital library users. Proceedings of the 11th International Conference on Human-Computer Interaction (HCII), Las Vegas, NV (2005).

[23] Park, M., Bae, J., Lee, S., End user searching: a Web log analysis of NAVAR, a Korean web search engine.Library & Information Science Research, 27(2).

[24] Pusara, M., Brodley, C.E., User re-authentication via mouse movements, Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security, October 29-29, 2004, Washington DC, USA

[25] A. Weiss, A. Ramapanicker, P. Shah, S. Noble and L.Immohr, "Mouse Movements Biometric Identification: A Feasibility Study", Proc. Student/Faculty Research Day, CSIS, Pace University, White Plains, NY, May 2007, pp.1-8.

[26] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I. H., The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1, 2009

[27] le Cessie, S. and van Houwelingen, J.C. (1992). Ridge Estimators in Logistic Regression. Applied Statistics, Vol. 41, No. 1, pp. 191-201.

[28] George H. John and Pat Langley (1995). Estimating Continuous Distributions in Bayesian Classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338-345. Morgan Kaufmann, San Mateo.

[29] I. H. Witten and E. Frank. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann, 1999.

Appendix A

Informed consent

www.puppyir.eu

Geachte ouders/verzorgers,

U ontvangt deze brief omdat uw zoon of dochter in groep 6, 7 of 8 van basisschool De Kroevendonk zit.

Ondergetekende, Pieter Dekker, doet wetenschappelijk (afstudeer-)onderzoek naar het zoekgedrag van kinderen op internet, voor de opleiding technische informatica aan de technische universiteit in Delft. Dit onderzoek vindt plaats in het kader van PuppyIR, een onderzoeksproject gefinancierd door de Europese Unie, met als doel om digitale informatie *beter* beschikbaar te maken voor kinderen, op een kindvriendelijke manier (bijvoorbeeld door het ontwikkelen van zoekfilters, of meer op kinderen gerichte zoekmachines). Als de nieuwsgierigheid is gewekt, kunt u meer informatie vinden in de bijlage of op www.puppyir.eu (in het Engels).

Mijn eigen onderzoek richt zich in het bijzonder op de vraag hoe kinderen nu eigenlijk zoeken op internet. Ik zou graag met een aantal leerlingen uit groep 6, 7 en 8 van De Kroevendonk een experiment uitvoeren, waarin ik hen vraag enkele zoekopdrachten uit te voeren op internet.

Voordat ik hiermee aan de slag kan, heb ik vanzelfsprekend eerst uw toestemming nodig. Mocht u geen bezwaar hebben tegen deelname van uw zoon of dochter aan mijn onderzoek, dan hoop ik dat u zo vriendelijk wilt zijn het bijgevoegde formulier in te vullen en te ondertekenen. U kunt het formulier afgeven bij de leerkracht van uw zoon of dochter.

Vanwege praktische beperkingen in de beschikbare tijd voor het uitvoeren van deze studie is het overigens niet mogelijk om alle kinderen te laten deelnemen; de kinderen die daadwerkelijk meedoen aan de studie worden willekeurig gekozen. Het kan dus voorkomen dat na uw eventuele toestemming, uw zoon of dochter toch niet hoeft deel te nemen aan het experiment.

Ik hoop dat ik u hiermee voldoende geïnformeerd heb. Mocht u nog vragen hebben, dan kunt u natuurlijk altijd contact met mij opnemen via het onderstaande e-mail adres.

Met vriendelijke groet,
Pieter Dekker

p.dekker@student.tudelft.nl
Student master technische informatica
Technische Universiteit Delft

PuppyIR

Onderzoek naar zoekgedrag op internet bij kinderen ten behoeve van de ontwikkeling van een zoeksysteem.

INFORMED CONSENT

www.puppyir.eu

PuppyIR
PuppyIR is een consortium van 8 verschillende instellingen waaronder universiteiten en een museum uit 4 Europese landen. Dit consortium heeft als doel kinderen op een kindvriendelijke manier toegang te geven tot informatie zoals deze bijvoorbeeld op het internet te vinden is. Het kan hierbij gaan om het ontwikkelen van internetfilters of een aangepaste zoekmachine specifiek voor kinderen.

Onderzoekskader
De geplande studie wordt uitgevoerd op basisschool De Kroevendonk te Roosendaal. Dit onderzoek voldoet aan de Nederlandse en Europese regelgeving, alsmede de ethische principes van onderzoek zoals beschreven in de ethische handleiding die is opgesteld voor PuppyIR onderzoek.

Doel van het onderzoek
Het onderzoek heeft als doel het zoekgedrag op internet te categoriseren. De resultaten kunnen worden gebruikt voor het ontwikkelen van een programma om kinderen te assisteren bij het zoeken op internet. We onderzoeken dit bij kinderen uit groep 6, 7 en 8.

Wie kan er meedoen?
Alle kinderen uit groep 6 t/m 8 van basisschool De Kroevendonk.

Wat wordt er van uw kind verwacht?
Uw kind zal achter de computer enkele vragen beantwoorden. Hij of zij zal de antwoorden op deze vragen zoeken op internet, met behulp van Google. Dit zal in totaal ongeveer een half uur gaan duren.

Wat doen we met de vragenlijsten/testen/andere vastgelegde gegevens?
Vragenlijsten
Vragenlijsten worden gebruikt om informatie in te zamelen over wat uw kind van de opdrachten vindt en wat zijn of haar ervaring is met computers en het zoeken naar informatie op internet.

Vragenlijsten worden vernietigd zodra de resultaten zijn verwerkt.

Logfiles
Logfiles worden gebruikt om alle handelingen die uw kind verricht op de computer vast te leggen en te analyseren. Dit geeft inzicht in de manier waarop kinderen zoeken naar informatie. Na afronding van het onderzoek zullen de logfiles worden vernietigd.

Persoonlijke gegevens
Persoonlijke gegevens worden strikt vertrouwelijk behandeld. Persoonlijke gegevens waarmee een kind kan worden geïdentificeerd worden gescheiden bewaard van gegevens die uit het onderzoek naar voren zijn gekomen. Persoonlijke gegevens worden nooit aan anderen gegeven en alleen gebruikt in het kader van dit onderzoek. Na afronding van het onderzoek zullen de persoonlijke gegevens worden vernietigd.

Wat gebeurt er als u wilt stoppen met het onderzoek?
Indien u de deelname van uw zoon/ dochter tijdens het onderzoek wilt stoppen kunt u dat op ieder moment doen, zonder daarvoor een reden aan te geven. U bent niet verplicht om uw zoon of dochter het onderzoek helemaal te laten afmaken, we hopen echter wel dat dat uw uitganspunt is. Indien u besloten heeft om uw zoon/ dochter het onderzoek niet te laten afmaken, verzoeken wij u om uw besluit zo spoedig mogelijk te melden aan Pieter Dekker via p.dekker@student.tudelft.nl

PuppyIR Informed Consent

Naam studie: Puppy IR: An open source environment to construct information services for children.

Naam deelnemer: ……
Nummer: ……
Datum (dd/mm/jjjj): ……

Zijn de ouders getrouwd en officieel voogd van de deelnemer? Ja/nee*

Ja → beide ouders moeten dit formulier ondertekenen
Nee → de persoon die de voogdij over het kind heeft moet tekenen. In het geval van co-ouderschap moeten beide ouders tekenen.

Wie heeft de voogdij over de deelnemer: Vader/ moeder/ anders*

Wanneer anders: Wie heeft de voogdij over de deelnemer? …….

Hierbij verklaren wij/ verklaar ik dat:

- Dhr P. Dekker mij/ons een kopie heeft gegeven van de informatiebrochure en mij/ons de opzet en het doel van het onderzoek volledig heeft uitgelegd.
- Dhr P. Dekker mij/ons de gelegenheid heeft gegeven om vragen te stellen over het onderzoek. Hij/zij heeft mij/ons uitgelegd dat ik/wij vrij ben/zijn om mijn/ons kind op elk moment uit het onderzoek terug te trekken.
- Ik/wij alles dat is uitgelegd begrepen heb/hebben en de informatiebrochure heb/ hebben gelezen.
- Ik/wij erin toestemmen om mijn/ons kind deel te laten nemen aan dit onderzoek.
- Ik/wij wel/niet* toestemming geef /geven om mij/ons in de toekomst na het einde van deze studie nogmaals te benaderen.

Vader van de deelnemer/ voogd

Plaats Datum (dd/mm/jjjj) Handtekening

Moeder van de deelnemer/ voogd

Plaats Datum (dd/mm/jjjj) Handtekening

Onderzoeker

Plaats Datum (dd/mm/jjjj) Handtekening

Appendix B

Experiment protocol

This appendix contains the experiment protocol used in this study.

1. At first I contacted the director of the primary school and explained how my research would look like and what its goal is. I got approval from the director to start contacting the teachers in order to make appointments.

2. I printed copies from the letter and the informed consent so the teachers could hand it out in their class.

3. In every class I presented myself and talked about my study, the university, about research in common and about the research I would be conducting. I explained that I wanted to do research in how people search the internet. I explained that probably 'Pietje' would search in a different way than 'Jantje'. I explained that I wanted to have a number of children doing 4 search assignments on the internet. I explained that these assignments are totally voluntary and that if you don't want to you don't have to. I also explained that because of their age their parents should give consent to join the experiment otherwise I couldn't ask them to join. I explained that I would pick one child at a time from class and let him or her sit behind a computer.

4. For every child I picked out of class I followed the following protocol:

4.1 Ik ga naar de klas en vraag aan de docent of ik iemand mee kan nemen voor het experiment.

4.2 Het kind (in het vervolg: hij) gaat mee naar de ruimte. Onder het lopen maak ik een klein praatje (bijv. over wat hij zojuist in de klas heeft gedaan). In de ruimte waar we naartoe lopen heb ik twee laptops heb neergezet op een tafel. 1 voor mijzelf (de onderzoeker) de andere voor de deelnemer.

4.3 Ik vraag: 'Zou je achter deze laptop willen gaan zitten?'

4.4 Ik zeg: 'Ik doe onderzoek naar hoe kinderen zoeken op internet om ze te helpen bij het vinden van informatie.'

4.4 Ik leg uit wat we gaan doen:

4.4.1 Ik zeg: 'Je mag zometeen op de website die je nu voor je ziet jouw naam uitzoeken en dan op start klikken. Je krijgt dan een aantal meerkeuze vragen te zien waarbij je het antwoord kunt aanklikken. Daarna krijg je achter elkaar vier vragen te zien. Bij elke vraag krijg je van mij een blad met de vraag en en stukje om het antwoord op het te schrijven. Het maakt voor mij niet uit of het antwoord goed of fout is, of dat je het antwoord nu wel of niet kunt vinden. Probeer het gewoon te doen zoals jij denkt dat goed is. Als je het antwoord hebt opgeschreven of het antwoord niet kunt vinden kun je op de 'home'-button (of het huisje) (ik wijs het aan op het scherm) klikken, je gaat dan terug naar de website met de vraag en je kunt dan doorklikken naar de volgende vraag. Na het maken van deze vier vragen staat er op de website nog 1 vraag die je mag beantwoorden en daarna ben je klaar.'

4.5 Ik zeg: ‘Heb je nog vragen?’ Als hij nog vragen heeft probeer ik het nog wat duidelijker of gemakkelijker uit te leggen

4.6 Hij zoekt zijn naam op in de lijst met namen en klikt op ‘volgende’.

4.7 Hij vult het antwoord op achtereenvolgens drie vragen in en klikt op ‘volgende’.

4.8 Hij ziet vraag 1 en klikt op op ‘ga naar Google’.

4.9 Voor alle vier de vragen geldt het volgende:

4.9.1 Als hij de vraag niet snapt probeer ik de vraag iets makkelijker en duidelijker uit te leggen.

4.9.2 Als de tijd voor de vraag bijna op is zeg ik: ‘Je tijd voor deze vraag is bijna op.’. Als de tijd op is zeg ik: ‘De tijd is op voor de vraag, je mag op de home-button (of het huisje) klikken om door te gaan naar de volgende vraag.’

4.9.3 Als hij vraagt of het antwoord dat hij zover heeft opgeschreven voldoende is, geef ik aan dat hij zijn moet opschrijven wat hij denkt dat goed is en als hij denkt dat het voldoende is is het voor mij ook voldoende.

4.9.4 Als hij aangeeft dat hij het antwoord niet kan vinden, geef ik aan dat dit geen probleem is en dat hij door kan gaan naar de volgende vraag.

4.10 Als de vierde vraag klaar is komt hij op een pagina met de vraag hoe hij het experiment vond. Hij vult de vraag in en klikt op ‘volgende’

4.11 Hij ziet een bedank pagina en ik bedank hem zelf ook nog: ‘Dit was het, bedankt dat je mee wilde doen, het ging goed!’. Daarnaast maak ik nog een praatje voordat ik hem weer naar de klas stuur.

Any other questions that popped up during the sessions we tried to answer in such a way that we did not compromise this protocol and still let them feel safe.

Appendix C

Table C.1 Feature selection results based on greedy stepwise backward search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	10	10	10	10	10	10	10	10	10	10	10,0
2	10	9	10	9	10	8	9	9	8	10	9,2
3	10	10	10	9	10	8	8	10	7	8	9,0
4	9	10	9	10	9	9	10	9	10	9	9,4
5	10	9	10	10	9	10	9	9	10	10	9,6
6	10	8	9	8	10	10	9	10	8	9	9,1
7	9	9	10	7	9	7	8	9	9	9	8,6
8	10	10	10	10	10	9	10	10	10	10	9,9
9	7	9	9	10	8	7	9	10	10	7	8,6
10	8	7	8	10	8	7	6	9	7	7	7,7
11	8	9	10	6	10	10	10	7	8	8	8,6
12	8	9	10	7	8	8	9	7	8	7	8,1
13	10	10	10	6	6	9	10	10	10	10	9,1
14	4	5	7	3	5	6	5	5	5	7	5,2
15	6	6	6	9	5	7	7	9	7	7	6,9
16	5	5	6	9	6	9	5	8	3	6	6,2
17	10	10	2	8	9	10	9	8	8	10	8,4
18	9	4	2	0	8	8	8	10	6	8	6,3
19	3	9	8	7	9	8	7	3	9	10	7,3
20	4	4	0	0	7	4	2	1	2	4	2,8
21	1	4	0	0	3	5	4	2	6	3	2,8
22	1	0	0	0	3	2	2	1	4	1	1,4
23	0	2	0	0	3	5	4	5	4	3	2,6
24	7	8	7	0	8	8	7	0	6	6	5,7
25	0	1	0	0	0	0	1	0	1	1	0,4
26	0	0	0	0	0	0	0	0	0	0	0,0
27	0	0	0	0	0	0	0	0	0	0	0,0
28	0	0	0	0	0	0	0	0	0	0	0,0
29	0	0	0	0	0	0	0	0	0	0	0,0
30	0	0	0	0	0	0	0	0	0	0	0,0
31	0	0	0	0	0	0	0	0	0	0	0,0
32	0	0	0	0	0	0	0	0	0	0	0,0
33	0	0	0	0	0	0	0	0	0	0	0,0

Table C.2 Feature selection results based on genetic search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	10	10	10	10	10	9	10	9	10	9	9,7
2	3	1	3	5	3	4	2	3	1	3	2,8
3	6	4	1	3	4	0	6	8	3	5	4,0
4	5	1	5	3	4	3	6	2	5	6	4,0
5	4	9	5	2	4	3	7	7	8	7	5,6
6	9	5	9	9	9	9	10	3	7	9	7,9
7	5	2	3	1	3	6	3	2	2	3	3,0
8	0	3	5	1	0	0	2	5	4	1	2,1
9	0	0	0	1	2	0	0	0	0	1	0,4
10	6	5	5	2	5	2	4	5	4	3	4,1
11	2	3	4	6	5	5	3	4	2	4	3,8
12	4	4	5	4	6	5	5	7	5	5	5,0
13	4	7	6	4	4	3	3	8	6	3	4,8
14	3	3	6	4	8	2	8	2	5	7	4,8
15	2	4	4	2	1	2	1	2	4	2	2,4
16	2	4	4	5	5	2	4	7	4	7	4,4
17	4	7	1	5	2	1	1	2	3	1	2,7
18	3	4	0	2	3	5	2	2	4	2	2,7
19	4	6	6	3	4	3	9	8	6	5	5,4
20	1	1	3	0	1	2	1	3	4	2	1,8
21	2	6	2	4	5	1	2	3	5	4	3,4
22	0	1	0	3	1	2	1	0	0	1	0,9
23	9	8	9	6	9	9	9	9	10	9	8,7
24	7	3	7	2	7	9	8	2	5	4	5,4
25	6	6	6	5	2	6	2	5	6	3	4,7
26	5	7	6	2	7	5	9	3	6	6	5,6
27	5	5	2	5	6	4	7	4	5	3	4,6
28	2	2	7	7	5	6	3	7	6	6	5,1
29	3	1	1	3	3	4	1	2	0	3	2,1
30	8	6	3	5	8	9	10	8	7	8	7,2
31	4	3	5	1	7	2	5	4	7	7	4,5
32	5	5	4	3	5	3	3	5	7	7	4,7
33	6	6	6	6	6	5	4	2	4	1	4,6

Table C.3 Feature selection results for success prediction based on greedy stepwise backward search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	9	7	8	10	9	10	10	8	10	8	8,9
2	10	10	10	10	10	10	10	10	10	10	10,0
3	9	7	10	10	9	7	10	9	8	10	8,9
4	8	10	10	9	7	8	8	9	9	8	8,6
5	10	9	7	10	8	10	10	9	7	10	9,0
6	9	9	7	10	10	10	9	10	8	8	9,0
7	8	10	6	9	8	7	8	7	8	9	8,0
8	9	9	8	9	9	10	8	7	9	9	8,7
9	8	10	6	10	9	8	8	9	8	8	8,4
10	7	6	6	6	10	8	8	8	7	10	7,6
11	9	8	5	7	7	8	8	9	7	8	7,6
12	10	9	9	8	10	7	9	9	9	9	8,9
13	9	10	6	10	10	10	10	9	8	8	9,0
14	10	10	10	9	10	9	10	9	10	10	9,7
15	6	10	10	10	9	10	8	10	7	8	8,8
16	10	9	10	10	10	9	10	10	10	9	9,7
17	10	9	10	10	10	10	10	10	10	10	9,9
18	8	10	6	10	7	9	9	10	10	10	8,9
19	6	10	9	8	7	8	8	9	7	7	7,9
20	10	10	9	9	10	9	9	9	8	9	9,2
21	9	6	8	10	10	9	8	10	8	10	8,8
22	9	6	4	4	6	5	6	7	6	8	6,1
23	10	10	10	10	9	7	9	8	10	10	9,3
24	7	2	2	8	6	2	5	5	8	6	5,1
25	10	7	10	9	8	7	10	8	10	9	8,8
26	6	6	0	6	6	5	7	4	5	8	5,3
27	7	7	0	5	6	2	5	7	3	7	4,9
28	7	5	8	6	9	4	10	6	5	5	6,5
29	4	1	0	5	2	1	0	2	3	6	2,4
30	3	0	0	4	5	1	1	3	3	3	2,3
31	3	0	0	1	4	1	1	1	0	4	1,5
32	1	0	0	1	2	0	1	1	1	2	0,9
33	1	0	0	2	3	1	3	1	0	2	1,3

Table C.4 Feature selection results for success prediction based on genetic search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	5	5	7	7	8	8	9	4	3	6	6,2
2	7	9	5	4	6	8	5	8	7	5	6,4
3	5	6	5	5	5	4	7	6	3	5	5,1
4	6	7	9	7	4	5	7	6	6	5	6,2
5	2	6	7	7	6	4	8	6	5	7	5,8
6	8	5	7	6	7	7	7	7	4	5	6,3
7	4	5	4	7	7	5	5	3	4	7	5,1
8	5	4	4	3	5	7	8	4	1	6	4,7
9	3	8	6	0	6	4	5	4	2	5	4,3
10	2	6	5	8	6	5	4	3	4	5	4,8
11	7	5	5	4	3	8	3	3	3	7	4,8
12	9	6	4	5	6	8	8	9	6	7	6,8
13	3	8	4	3	4	5	5	3	4	3	4,2
14	9	6	6	6	9	10	10	6	7	7	7,6
15	4	7	7	5	6	5	4	6	7	7	5,8
16	8	6	9	3	6	8	4	5	6	6	6,1
17	7	2	3	8	5	10	8	3	7	6	5,9
18	3	6	4	6	8	5	5	2	3	5	4,7
19	7	7	7	7	7	5	3	9	8	6	6,6
20	7	10	6	7	3	9	8	8	7	7	7,2
21	4	5	2	5	8	6	4	4	3	4	4,5
22	5	5	8	2	4	6	1	2	4	5	4,2
23	9	10	9	10	9	6	8	10	9	8	8,8
24	4	4	6	6	7	5	4	4	5	4	4,9
25	6	8	6	7	6	8	7	8	7	7	7
26	9	7	9	9	7	8	7	8	10	8	8,2
27	5	9	9	6	5	6	5	5	5	4	5,9
28	6	4	3	5	5	9	7	8	7	6	6
29	1	7	5	4	6	2	3	4	4	4	4
30	7	8	7	8	8	9	6	8	10	9	8
31	7	4	5	7	4	8	6	8	5	6	6
32	0	7	10	8	7	6	10	9	9	8	7,4
33	4	6	3	6	6	4	5	2	6	2	4,4

Table C.5 Feature selection results for success prediction including the search roles as feature based on greedy stepwise backward search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	9	7	8	10	9	10	10	8	10	8	8,9
2	10	10	10	10	10	10	10	10	10	10	10,0
3	10	7	10	10	9	7	10	9	8	10	9,0
4	8	10	10	9	9	8	8	9	9	9	8,9
5	9	9	7	10	9	10	10	9	7	10	9,0
6	9	9	7	10	10	10	9	10	8	9	9,1
7	9	10	6	9	9	7	8	7	8	9	8,2
8	9	9	8	9	9	10	8	7	9	10	8,8
9	9	10	6	10	10	8	8	9	8	8	8,6
10	7	6	6	7	9	8	7	8	7	10	7,5
11	9	8	5	6	10	8	8	9	7	8	7,8
12	10	9	9	8	10	7	9	9	9	10	9,0
13	9	10	6	9	9	10	10	9	8	8	8,8
14	10	10	10	9	10	9	10	9	10	10	9,7
15	6	10	10	9	9	10	8	10	7	8	8,7
16	10	9	10	10	10	9	10	10	10	10	9,8
17	10	9	10	10	10	10	10	10	10	10	9,9
18	8	10	6	10	9	9	9	10	10	10	9,1
19	6	10	9	8	9	8	8	9	7	6	8,0
20	10	10	9	9	10	9	9	9	8	9	9,2
21	9	6	8	9	10	9	9	10	8	10	8,8
22	9	6	4	4	7	5	5	7	6	8	6,1
23	10	10	10	9	9	7	9	8	10	10	9,2
24	7	2	2	8	8	2	5	5	8	6	5,3
25	10	7	10	9	9	7	10	8	10	10	9,0
26	6	6	0	5	7	5	7	4	5	8	5,3
27	8	7	0	5	7	2	5	7	3	7	5,1
28	7	5	8	7	7	4	10	6	5	5	6,4
29	4	1	0	4	2	1	0	2	3	5	2,2
30	2	0	0	3	3	1	1	3	3	1	1,7
31	4	0	0	1	3	1	1	1	0	3	1,4
32	0	0	0	0	0	0	0	1	1	0	0,2
33	1	0	0	1	2	1	3	1	0	0	0,9
34	2	0	0	1	3	0	1	0	0	3	1,0

Table C.6 Feature selection results for success prediction including the search roles as feature based on genetic search

Feature\Set	0	1	2	3	4	5	6	7	8	9	AVG
1	8	7	10	4	6	6	4	6	3	8	6,2
2	8	7	6	9	9	10	9	8	9	4	7,9
3	2	7	2	4	4	3	4	5	3	6	4,0
4	5	7	8	6	5	2	6	4	5	5	5,3
5	5	8	0	2	3	5	4	8	6	6	4,7
6	3	1	0	4	2	3	1	4	4	4	2,6
7	9	6	6	7	6	6	6	7	6	7	6,6
8	6	4	1	7	7	6	3	6	5	9	5,4
9	9	8	9	8	6	8	9	7	7	8	7,9
10	6	8	0	7	6	6	2	7	7	4	5,3
11	2	4	9	7	8	3	6	3	4	4	5,0
12	7	3	7	4	6	4	6	8	8	6	5,9
13	4	5	4	7	9	8	6	8	7	9	6,7
14	7	6	3	6	7	7	8	7	9	5	6,5
15	8	10	9	8	8	8	6	9	9	8	8,3
16	10	9	10	8	8	10	6	9	8	8	8,6
17	6	7	5	6	8	7	9	7	5	3	6,3
18	4	4	4	4	3	4	4	7	4	4	4,2
19	8	8	10	7	8	10	9	9	7	6	8,2
20	10	10	6	9	8	8	7	8	7	9	8,2
21	6	6	2	8	6	5	7	6	4	5	5,5
22	5	3	10	6	2	4	5	6	5	4	5,0
23	2	9	3	10	8	6	8	7	7	7	6,7
24	6	5	7	7	4	9	6	6	7	5	6,2
25	6	7	1	5	6	5	8	7	2	5	5,2
26	7	9	10	5	9	7	8	9	7	9	8,0
27	5	4	8	4	3	3	1	4	3	4	3,9
28	6	5	7	7	5	5	7	8	7	6	6,3
29	9	8	8	7	9	10	9	4	5	5	7,4
30	4	5	2	2	3	1	2	4	2	6	3,1
31	1	3	4	3	2	5	4	5	4	1	3,2
32	4	8	5	9	9	7	9	8	8	9	7,6
33	9	9	4	7	4	9	8	10	7	6	7,3
34	10	10	10	10	9	10	8	10	10	10	9,7

Appendix D

Table D.1 Search role classification results based on greedy stepwise.

Threshold	Logistic	Naive Bayes	ZeroR	MLP	J48
0	59,54444%	66,11111%	47,77778%	69,91111%	59,35556%
1	55,93333%	64,54444%	47,77778%	68,62222%	62,87778%
2	54,82222%	64,54444%	47,77778%	68,52222%	61,57778%
3	56,02222%	63,34444%	47,77778%	67,41111%	62,68889%
4	58,15556%	61,03333%	47,77778%	66,30000%	62,68889%
5	60,65556%	61,03333%	47,77778%	67,51111%	60,27778%
6	64,26667%	60,93333%	47,77778%	68,90000%	65,18889%
7	66,57778%	61,12222%	47,77778%	66,67778%	64,07778%
8	62,87778%	59,63333%	47,77778%	69,91111%	64,07778%
9	61,48889%	58,62222%	47,77778%	66,11111%	64,91111%
10	68,52222%	54,64444%	47,77778%	69,82222%	67,60000%

Table D.2 Search role classification results based on genetic search

Threshold	Logistic	Naive Bayes	ZeroR	MLP	J48
0	59,54444%	66,11111%	47,77778%	69,91111%	59,35556%
1	62,04444%	67,41111%	47,77778%	68,80000%	61,85556%
2	62,87778%	66,21111%	47,77778%	69,82222%	60,74444%
3	64,08889%	67,32222%	47,77778%	72,13333%	59,73333%
4	62,87778%	63,61111%	47,77778%	67,60000%	61,95556%
5	61,76667%	68,24444%	47,77778%	66,48889%	63,90000%
6	63,34444%	62,87778%	47,77778%	62,04444%	69,35556%
7	71,12222%	59,35556%	47,77778%	75,65556%	63,80000%
8	67,32222%	59,08889%	47,77778%	69,54444%	67,22222%
9	68,62222%	64,73333%	47,77778%	69,54444%	66,02222%
10	60,18889%	56,38889%	47,77778%	55,74444%	56,94444%

Table D.3 Success classification results based on greedy stepwise

Threshold	Logistic	Naive Bayes	ZeroR	MLP	J48
0	61,76667%	58,43333%	52,22222%	65,28889%	53,43333%
1	59,35556%	58,43333%	52,22222%	67,32222%	53,43333%
2	55,93333%	58,43333%	52,22222%	67,51111%	53,43333%
3	55,93333%	58,43333%	52,22222%	61,76667%	56,76667%
4	54,72222%	58,43333%	52,22222%	66,48889%	52,32222%
5	55,74444%	58,43333%	52,22222%	67,78889%	54,63333%
6	61,67778%	58,43333%	52,22222%	66,21111%	58,24444%
7	62,78889%	59,54444%	52,22222%	60,56667%	58,24444%
8	59,26667%	58,24444%	52,22222%	59,35556%	55,93333%
9	60,10000%	64,45556%	52,22222%	62,97778%	55,10000%
10	61,30000%	64,17778%	52,22222%	61,67778%	53,98889%

Table D.4 Success classification results based on genetic search

Threshold	Logistic	Naive Bayes	ZeroR	MLP	J48
0	61,76667%	58,43333%	52,22222%	65,28889%	53,43333%
1	61,76667%	58,43333%	52,22222%	63,06667%	52,32222%
2	60,46667%	58,43333%	52,22222%	61,95556%	53,43333%
3	60,65556%	55,93333%	52,22222%	61,95556%	52,32222%
4	53,71111%	57,04444%	52,22222%	61,76667%	58,24444%
5	55,93333%	57,04444%	52,22222%	60,37778%	52,78889%
6	55,74444%	58,43333%	52,22222%	60,18889%	61,03333%
7	51,12222%	58,34444%	52,22222%	55,65556%	55,01111%
8	56,31111%	59,16667%	52,22222%	56,11111%	57,42222%
9	60,46667%	55,93333%	52,22222%	54,45556%	54,35556%
10	53,62222%	50,10000%	52,22222%	53,71111%	53,71111%

Table D.5 Success classification results based on genetic search including the search roles as feature

Threshold	Logistic	Naive Bayes	ZeroR	MLP	J48
0	60,69444%	58,33333%	52,22222%	67,50000%	53,88889%
1	60,69444%	58,33333%	52,22222%	67,50000%	53,88889%
2	59,44444%	59,44444%	52,22222%	65,00000%	53,88889%
3	56,94444%	57,36111%	52,22222%	70,97222%	58,47222%
4	56,94444%	58,47222%	52,22222%	65,00000%	57,36111%
5	56,94444%	63,05556%	52,22222%	66,11111%	60,83333%
6	61,80556%	60,83333%	52,22222%	63,61111%	61,94444%
7	62,77778%	63,05556%	52,22222%	58,33333%	58,05556%
8	66,52778%	64,02778%	52,22222%	65,13889%	62,63889%
9	58,33333%	58,33333%	52,22222%	55,97222%	57,91667%
10	61,94444%	61,94444%	52,22222%	61,94444%	64,16667%