
There may be differences between this version and the published version. You are advised to consult the publisher’s version if you wish to cite from it.

© The Authors 2018. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in the Proceedings of the 25th ACM International Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24-28 Oct 2016, pp. 731-740. ISBN 9781450340731 (doi:10.1145/2983323.2983805)

http://eprints.gla.ac.uk/183472/

Deposited on: 04 April 2019
Agents, Simulated Users and Humans

An Analysis of Performance and Behaviour

David Maxwell
School of Computing Science
University of Glasgow
Glasgow, Scotland
d.maxwell.1@research.gla.ac.uk

Leif Azzopardi
Department of Computer & Information Sciences
University of Strathclyde
Glasgow, Scotland
leif.azzopardi@strath.ac.uk

ABSTRACT

Most of the current models that are used to simulate users in Interactive Information Retrieval (IIR) lack realism and agency. Such models generally make decisions in a stochastic manner, without recourse to the actual information encountered or the underlying information need. In this paper, we develop a more sophisticated model of the user that includes their cognitive state within the simulation. The cognitive state maintains data about what the simulated user knows, has done and has seen, along with representations of what it considers attractive and relevant. Decisions to inspect or judge are then made based upon the simulated user’s current state, rather than stochastically. In the context of ad-hoc topic retrieval, we evaluate the quality of the simulated users and agents by comparing their behaviour and performance against 48 human subjects under the same conditions, topics, time constraints, costs and search engine. Our findings show that while naive configurations of simulated users and agents substantially outperform our human subjects, their search behaviour is notably different from actual searchers. However, more sophisticated search agents can be tuned to act more like actual searchers providing greater realism. This innovation advances the state of the art in simulation, from simulated users towards autonomous agents. It provides a much needed step forward enabling the creation of more realistic simulations, while also motivating the development of more advanced cognitive agents and tools to help support and augment human searchers. Future work will focus not only on the pragmatics of tuning and training such agents for topic retrieval, but will also look at developing agents for other tasks and contexts such as collaborative search and slow search.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval: Search Process; H.3.4 [Information Storage and Retrieval]: Systems and Software: Performance Evaluation

1. INTRODUCTION

Interactive Information Retrieval (IIR) is a complex, non-trivial process where a searcher undertakes a variety of different actions over a search session [29]. Motivated by an information need arising from a task, the searcher, in an Anomalous State of Knowledge (ASK) [11], aims to find relevant material that would help satisfy said information need, and thus complete their task. During the search process, besides the mechanical actions of querying, examining snippets and assessing documents for relevance (for example), the searcher’s cognitive state also changes as they interact with the information that they encounter [10, 12, 28, 30, 46]. Their initial cognitive state and background knowledge, along with any state changes that occur during the search task, all affect the searcher’s behaviour and subsequent interactions. Searchers learn more about the topic, identify new concepts and salient terms, firm up their understanding and notion of relevance, and perhaps may even change their notion of relevance as they read. For example, Eckhoff et al. [23] presented an analysis of query reformulations, showing that the majority of new search terms were acquired from snippets and documents examined. Other studies have shown that the attractiveness of a snippet affects whether a searcher is likely to click on them [25, 64], whereas their prior knowledge of the domain affects the queries issued, and their subsequent interactions [63]. In contrast, when considering the current state of the art in terms of user models used for IIR simulations, such models: (i) are highly abstracted; (ii) generally ignore the cognitive state of searchers (i.e. ignore the information encountered); and (iii) typically make decisions based upon the roll of a dice (i.e. stochastically).

The focus of this work is to incorporate a representation of cognitive state into user models in order to create simulated search agents, rather than employing stochastic, simulated users. As such, this work revisits the idea of building semi-autonomous search and autonomous agents that attracted much research attention twenty years ago, and is once again regaining popularity. Our goal here however is to develop more credible and realistic simulations of users and their behaviours, examining how well search agents (with cognitive state) behave and perform when compared to the existing simulated users1 and human searchers. The main contributions of this work are threefold: we (i) propose a new user model for simulation that maintains a representation of the user’s cognitive state; (ii) develop search agents that are autonomous in nature; and (iii) perform an empirical comparison between simulated users, search agents and humans.

1More specifically, simulations that employ stochastic user models.
2. RELATED WORK

In the period ranging from the mid 1990s to early 2000s, there was significant interest in developing software agents that assist searchers in finding relevant information [22]. Most of the agents developed during this period were semi-autonomous, meaning that they would assist the searcher in a collaborative fashion. This was in contrast to agents that operated in a fully autonomous state, thus working independently of the searcher [43]. For example, one of the first agents proposed was Letizia [42]. Letizia would track the behaviour of a searcher interacting with a Web browser via a plugin and work in the background, predicting what other Web pages would be of interest to the searcher based upon his/her interactions. Other examples of similar agents (or bots) include CiteSeer [13]\(^2\), The Info Agent [21], Newsweeder [39], The Remembrance Agent [55], SoftBot [24], WebWatcher [2] and Webbound [40]. These agents would in essence either actively monitor the inputs and interactions of the searcher, or simply examine source(s) of prior knowledge. From these two approaches, the agents would then attempt to infer the searcher’s intent and information needs. Several of these agents would then issue queries to multiple sources (e.g. CiteSeer [13] would issue queries to multiple academic publication databases) in an attempt to try and identify and suggest additional relevant material.

Today, general purpose Web search engines amass large volumes of interaction data, and are thus able to infer intent and relevance reasonably well [33]. Many techniques pioneered by the agents listed above have been subsumed and incorporated into these search engines. We posit that this is why the development of such agents has not continued. However, with task completion engines [7] and Artificial Intelligence (AI) agents such as Apple’s Siri, Microsoft’s Cortana and Facebook’s M now being actively developed, we are entering into a new phase of Information Retrieval (IR) research and a resurgence of agents that assist users in completing higher order tasks [29]. In this work however, we take a different approach. We develop an autonomous simulated agent through the process of building a richer and more complex model of the searcher (and their behaviours) for the purposes of IIR simulation and evaluation.

2.1 Simulation in IR and IIR

Simulation has been used in many different ways in IR. For example, approaches include simulated work tasks and tracks [14, 61], simulated and synthetic data collections [3, 5, 34, 58], and simulated interaction [6, 16, 19, 41, 62]. However, simulation has been used largely for evaluation and exploration to (i) determine how well an IR system performs, or to (ii) determine how the performance changes under different conditions and behaviours [4, 8, 9].

There has also been a growing interest with the simulation of interaction, and in particular how to model searcher behaviour [6, 19]. This is for a variety of reasons. Firstly, simulation provides a cost effective means of evaluation which is reliable, repeatable and therefore reproducible. Secondly, most, if not all evaluation measures either implicitly or explicitly encode some form of user model (typically a stopping model) in order to make measurements. Simulation also provides a means in which to explore different searcher

\(^2\)CiteSeer [13] is the precursor to the modern day search engine CiteSeer\(^3\), available at http://citeseerx.ist.psu.edu.

\(^3\)The exception being when exploratory simulations are conducted to examine ‘what-if’ scenarios.
example, a user’s knowledge about the topic is not consid-
ered, and their decisions are made stochastically based upon
whether the document (snippet) is relevant or not [9, 49, 50].
This inherently limits the applicability and generalisability of
the models because relevance judgements are required.
Furthermore, numerous trials are required to form an aver-
age, which substantially increases the cost and time to run
such (Monte-Carlo style) simulations. An alternative frame-
work proposed by Carterette et al. [16] seeks to simulate the
interaction of users with the search engine and search result
lists for the purposes of evaluation (through a dynamic
test collection). While not explicitly stated, the model be-
hind the simulation follows a similar process as described
above. The simulation is instantiated using probabilistic
models that are constructed based upon training data from
actual searchers, and considers querying, stopping and how
long simulated users spend on documents (dwell times). To
determine whether a link is clicked or not, a classifier is then
trained to determine the attractiveness of a snippet, rather
than the purely stochastic approach used in previous studies
(e.g. [9, 49, 50]). In this paper, we draw upon these past
works, combining and extending these models and frame-
works, to develop more realistic simulated users, ones with
cognitive state and agency, i.e. search agents.

3. EXTENDED USER MODEL

The starting point for our new simulation model is the
CSM used by Maxwell et al. [50], which is in turn an amal-
gamation of previously proposed models [9, 16, 49, 59]. How-
ever, this extended model includes a number of additional
and augmented steps. As shown in Figure 2, the process is
as follows: the user first examines the topic, and then gen-
erates a series of candidate queries. The user then selects
a query to issue to the underlying search engine. When
the Search Engine Results Page (SERP) is returned, the
user examines a snippet and decides whether it is attractive
enough to click. If this is the case, the user clicks and as-

Figure 2: A model of the search process typically used in simulations. Here it has been augmented to show
that various interactions invoke changes in the user’s state - denoted by the State subprocess clouds.

bles cognitive state change with subflows to the grey clouds
labelled ‘State’ as illustrated in Figure 2. The model
representing the user’s cognitive state, which we shall refer to
as the User State Model (USM), is shown in Figure 3. The
USM consists of several parts/representations: prior back-
ground knowledge; the information need; lists of previous
interactions (current state interactions) and a series of mod-
els used for generating queries (query model); deciding on
the attractiveness of a snippet (attractiveness model); and
relevance of a document (relevance model).

It should be noted that in prior simulations, previous in-
teractions are typically recorded providing some ‘state’ in-
formation [44, 45, 50]. In a study by Maxwell et al. [50],
simulated users were aware of how many documents had
been previously seen, and how many were considered rele-
ant. This information was used to decide when to stop.
In our model, such actions, and judgements are also recorded.
However, we go further and explicitly model and record
the actions and information encountered during the search
process. This information along with existing background
knowledge is then used to inform decisions regarding attrac-
tiveness and relevance. Consequently, as the state of the user
changes (via interaction), the query, attractiveness and rel-


Team: Student
4. EXPERIMENTAL METHOD

The focus of our experiments is to explore the behaviour and performance of simulated users and search agents. These simulations are compared against human searchers because we are trying to replicate human performance and behaviour for evaluation purposes. In developing our simulations, we are careful to design and build simulated users and agents that are grounded, such that they are instantiated based upon actual data and past findings.

4.1 Corpus, Topics and System

For this study, we used the TREC AQUAINT test collection along with two topics from the 2005 Robust Track, № 347 (wildlife extinction) and № 435 (curbing population growth). The collection was indexed using the Whoosh IR toolkit, where stopwords were removed and Porter stemming applied. The retrieval model used was PL2 \( (c = 10.0) \) [1]. An IR simulation toolkit was developed to encode the CSM and USM, and is freely available for download so that experiments can be replicated.

4.2 Human Subjects

We recruited 48 undergraduate students at the University of Glasgow to undertake two ad-hoc topic retrieval search tasks, using a standard search interface. The details of this user study are described and reported by Maxwell and Azzopardi [47]. Subjects of the study were given a total of 20 minutes (1200 seconds) to complete each search task, using the aforementioned system and setup. For each search task, the TREC topic and title descriptions were given, with both providing details on what constituted as a relevant document. Subjects were asked to find as many relevant documents as possible, with the greatest level of accuracy. Second, and more intuitively, is that when searching, people often identify a high number of potentially relevant documents, but later only use a subset of those documents [15]. Third, since subjects were tasked with finding a set of relevant documents, a set-based measure was more appropriate (rather than say MAP or other ranked based measures). In addition to precision, recall and F1, we also measured - and report - the total number of TREC relevant documents identified, and the corresponding Cumulative Gain (CG) [32].

In terms of behavioural data, we recorded a variety of interactions. These included the queries issued, the snippets hovered over and clicked (considered attractive), and any documents assessed and marked relevant. The time taken to perform each of these different interactions was also recorded. From this data, we could ascertain the number of queries issued, the number of snippets examined, and the number of documents viewed and marked. When coupled with TREC relevance assessments, we could then determine how many TREC relevant documents were marked, and the associated probabilities (refer to Table 1) of clicking an attractive snippet, or marking a relevant document.

4.3 Simulated Subjects

Using the CSM and USM, we are able to generate a variety of different simulated users/agents of varying sophistication. To instantiate a simulated user/agent, we need to define the following components: (i) interaction times/costs - how long it takes to perform each action; a (ii) querying strategy - how queries are generated, ranked and selected; a (iii) continuation strategy - how many snippets are examined before moving onto the next query; a (iv) snippet attractiveness decision maker - to decide on whether the document should be examined or not; a (v) document relevance decision maker - to decide on whether the document should be marked or not; and any (vi) prior/background knowledge - what the simulated user/agent knows a priori.

**Interaction Times/Costs** To fairly compare our simulated users/agents against human subjects, we imposed the same time limit of 20 minutes and associated a cost in terms of time with each simulated action (refer to Table 2). To ground our simulations, the individual, per-subject interac-
Table 1: Mean interaction probabilities over the 48 human subjects. Individual interaction probabilities were used for stochastic simulated users.

<table>
<thead>
<tr>
<th>Probability</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(C</td>
<td>R_A)$ (Clicking an attractive snippet)</td>
</tr>
<tr>
<td>$P(C</td>
<td>N_A)$ (Clicking an unattractive snippet)</td>
</tr>
<tr>
<td>$P(M</td>
<td>R_A)$ (Marking a relevant document)</td>
</tr>
<tr>
<td>$P(M</td>
<td>N_A)$ (Marking a non-relevant document)</td>
</tr>
</tbody>
</table>

Table 2: Mean interaction times (in seconds), ± standard deviations (SD) over the 48 human subjects that were used in this study.

<table>
<thead>
<tr>
<th>Time Required to...</th>
<th>Seconds ± SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>...issue a query</td>
<td>15.1 ± 4.1</td>
</tr>
<tr>
<td>...examine a SERP</td>
<td>1.1 ± 1.1</td>
</tr>
<tr>
<td>...examine an individual snippet</td>
<td>1.3 ± 0.4</td>
</tr>
<tr>
<td>...examine a document</td>
<td>21.45 ± 9.9</td>
</tr>
<tr>
<td>...mark a document as relevant</td>
<td>2.57 ± 1.7</td>
</tr>
</tbody>
</table>

egy. We used three previously examined strategies, the first of which is the standard baseline used in simulations and many evaluation measures. This is a fixed depth strategy, named SS1 by Maxwell et al. [49, 50]. Under SS1, the simulated user will stop examining snippets/documents after examining $x_1$ snippets regardless of their (ir)relevance. We also employ two other strategies which were based upon the frustration point [20] and disgust [38] rules. The first of these is based on the total number of non-relevant snippets observed, such that the simulated user will stop examining a ranked list after observing $x_2$ non-relevant snippets. The other strategy is based upon observing a contiguous number of non-relevant snippets, such that the simulated user will stop examining the list after observing $x_3$ non-relevant snippets in a row. These latter strategies are called SS2 and SS3 respectively in [49, 50]. Maxwell et al. compared actual search stopping data against the three aforementioned strategies, and showed that SS1 with $x_1 = 13$, SS2 with $x_2 = 9$, and SS3 with $x_3 = 5$ provided a good approximation of actual search behaviour over a similar set of search tasks, and on the same collection. Consequently, we employ these values to ground our simulations.

**Attractiveness/Relevance Decision Making** For simulations involving stochastic decision making regarding the attractiveness of snippets (i.e. the probability of clicking on a snippet) and the relevance of a document (i.e. the probability of marking a document relevant), we again used the interaction data from the human subjects, coupled with TREC relevance judgements, to determine interaction probabilities. Table 1 presents the average probabilities of clicking snippets and marking documents. In our experiments, we used individual interaction probabilities - not the average - for each of the simulated users to ensure a fair comparison against the actual searchers. For search agents, we employed a language modelling framework to make such decisions, the details of which are provided in Section 4.3.2.

**Prior Background Knowledge** We assume that simulated users have some idea of the distribution of terms. This knowledge is used when ranking/selecting queries, and in some decision making components. For this, we used data from TREC123 (AP88-89 and W87-92) so that it is independent of the corpus that the simulated users/agents were searching. For the search agents (detailed in Section 4.3.2), we also utilised word2vec [52] to build semantic representations used by decision making components, which were also derived from the AP88-89 and W87-92 collections.

Unless stated otherwise, these times, querying strategies and continuation strategies were used by the simulated users and agents. Next, we outline simulated users that do not consider a user’s state. These simulated users represent the current state of the art. We then discuss how we created and instantiated a series of search agents.

### 4.3.1 Naïve and Stochastic Simulated Users

For our baselines, we operationalised a series of simple and standard simulated users. The first two simulated users were based upon a ‘TREC style’ interaction model.

The TREC user closely mimics the traditional style of IR experimentation. This simulated user issues a single query, based upon the given topic title (TT) terms (e.g. wildlife extinction) for topic № 347. The user has no recourse to relevance, and blindly considers all documents as relevant. The continuation strategy is SS1, but with $x_1 = 1000$. How-

Note that our human subjects issued queries of 3.4 terms in length on average, so QS generates queries of comparable length.
ever, since the user only has 1200 seconds, the simulation ends before they reach that depth.

Similar to the TREC user defined above, the stochastic TREC user operates in much the same way. However, the attractiveness and relevance of snippets and documents respectively is based upon individual human subject interaction probabilities, with the averages presented in Table 1. The user still only issues a single query, but this setup introduces some variation into its actions.

Our final baseline simulated user is referred to as the stochastic user. This user represents the state of the art in simulation, and a number of researchers have employed similar variants in their work [8, 9, 16, 17, 49, 50, 60]. This simulated user bases decisions upon interaction probabilities for attractiveness and relevancy, and is therefore stochastic in nature. For these users, simulations were run 10 times with the mean values for measures reported over those runs.

### 4.3.2 Autonomous Search Agents

We developed and explored a variety of search agents based upon: (i) the amount of prior knowledge possessed; (ii) the attractiveness and relevance decision making components; and (iii) whether or not these models are updated over the course of the session. In this paper, we will use a language modelling framework to represent the user’s state and how they make decisions. While other representations of knowledge can be used, we leave this - and exploring other mechanisms for decision making - for future work.

**Prior Knowledge** Aside from using the term distribution \( p(t|\theta_C) \) as background knowledge, we considered for some of our agents incorporating topical background knowledge built through word associations (word2vec) [52]. When an agent reads through the topic, it is primed to expect certain other terms or concepts. By using word2vec, we can to a certain extent simulate this in order to create language models that are less sparse (see below). The topical background knowledge was built using the resources by Zuccon et al. [65]. We used the ap8889 and wsj8792 vectors that were generated using skipgrams with 200 dimensions and a window of 5. These values were shown to perform well in [65]. For each term in the topic, we calculated the word to word associations using the cosine similarity function, and took the top 100 word associations and scores. The scores for each term were then summed up, and then normalised to produce a probability distribution \( p(t|\theta_B) \). For the topic wildlife extinction, the top ten words were for example: *murrelet, porpoise, species, habitat, migratory, birds, waterfowl, raptors, efforts and initiatives.*

**Attractiveness and Relevance Models** To decide if a snippet is attractive enough (*should this snippet be clicked?*), or a document is relevant enough to mark (*is this document relevant to the topic?*), we constructed language models for attractiveness \( p(t|\theta_A) \) and relevance \( p(t|\theta_R) \) respectively. The snippet or document observed by the agent was then scored using this model within the normalised log likelihood method [51], where the normalisation was based upon the length of the observed text:

\[
O(S|\theta_A) = \frac{1}{n(S)} \sum_t p(t|S) \log \frac{p(t|\theta_A)}{p(t|\theta_C)}
\]

where \( n(S) \) is the length of the snippet, \( p(t|S) \) the probability of the term in the snippet (maximum likelihood estimate), and \( p(t|\theta_A) \) is defined by:

\[
p(t|\theta_A) = \lambda \left[ \frac{w_T}{z} p(t|\theta_T) + \frac{w_I}{z} p(t|\theta_{AS}) + \frac{w_B}{z} p(t|\theta_B) \right] + (1 - \lambda_A) p(t|\theta_C)
\]

where \( p(t|\theta_T) \) is the probability of a term appearing in the topic, and \( p(t|\theta_{AS}) \) is the probability of a term in the set of snippets considered attractive. The weightings \( w_T, w_I, \) and \( w_B \) denote how much emphasis is placed upon the topic, interaction and background semantic knowledge respectively. \( z \) is a normalising constant. If \( O(S|\theta_A) > \mu_A \), then the snippet \( S \) was considered attractive. We computed the document relevance scores in a similar fashion, and again used a threshold \( O(D|\theta_R) > \mu_R \) to decide whether the document was considered relevant. For the purposes of this work, we explored a range of thresholds for \( \mu_A \) and \( \mu_R \). Intuitively, if the \( \mu \) values are negative, the agent would then be more liberal in accepting the snippet/document. If they are positive, then the agent will be more conservative. For the relevance decision making component, we follow the same process and compute \( O(D|\theta_R) \), using the following language model for document relevance:

\[
p(t|\theta_R) = \lambda \left[ \frac{w_T}{z} p(t|\theta_T) + \frac{w_I}{z} p(t|\theta_{RD}) + \frac{w_B}{z} p(t|\theta_B) \right] + (1 - \lambda_R) p(t|\theta_C)
\]

where \( p(t|\theta_{RD}) \) is the probability of term given the set of relevant documents marked relevant during the course of the session.

**Updating State** As suggested in the CSM, the USM is updated at various points during the search process if \( w_I > 0 \). For the Snippet Attractiveness Model \((p(t|\theta_A))\) and the Document Relevance Model \((p(t|\theta_R))\), the models are updated before a decision is made based upon snippets/documents that have been examined and considered attractive/relevant respectively. The language models \( p(t|\theta_{AS}) \) and \( p(t|\theta_{RD}) \) are calculated by counting the number of times each term appears in this observed text, and then normalised to obtain the maximum likelihood estimate.

**Agent Setup** Altogether, we can create an agent that updates its state and makes decisions based upon probabilistic language models. While there are many possible combinations of weightings, we consider the simplest in this work as a starting point (where \( w_T = 1, w_I = 1 \) and \( w_B = 1 \)). Given this agent configuration, we then explored different thresholds for snippet attractiveness and document relevance, as well as different \( \lambda \) smoothing values, where we first set the \( \lambda_A = \lambda_R = [0.1, \ldots, 0.9] \) and \( \mu_A = \mu_R = [-0.4, \ldots, 0.4] \) in 0.1 increments - providing a range of strategies from liberal to strict [57]. We then fixed \( \lambda_A = 0.8 \) and \( \lambda_R \) to each of the threshold settings, and explored the range of \( \lambda_R \) and \( \mu_R \) settings. This was repeated for each of the querying and continuation strategies detailed above.

### 4.4 Comparing Behaviour and Performance

To determine whether the performance or behaviour of the simulated users/agents was similar to the performance or behaviour of our actual users, we employed the non-parametric, two-sample Kolmogorov-Smirnov test. If there was no significant difference between the different simulated users/agents and actual users, then the performance and behavioural measures were drawn from the same distribution. A dagger (†) in our results therefore denotes that there was no significant difference at \( \alpha = 0.05 \).
To ensure that our comparison between search agents and simulated users was fair, we used two-fold cross validation to train our simulated agents, and reported the behaviour and performance of the test sets. To select the parameters $\lambda_A$, $\lambda_R$, $\mu_A$, and $\mu_R$, we found the combination that resulted in the best performance and behaviour measures used, given the parameter values noted above. This was so we could compare search agents - that for example issued a similar number of queries and clicked on a similar number of snippets - to the real-world subjects, and discuss the differences.

5. RESULTS

Tables in this section report: the number of queries issued ($#Q$); the number of snippets examined ($#Snip$); and the number of documents viewed ($#Doc$), as well as the number marked ($#M$). We also include in the results tables a series of performance measures, including: the number of documents marked that were TREC relevant ($#R$); the CG; precision ($P$); and the $F_1$ score ($F_1$).

Table 3 presents a summary of the behaviours of the 48 real-world subjects (over both topics). Also included in the table is the time spent per query and per document (in seconds, $T/Q$ and $T/Doc$ respectively). We report the best, second best, median and worst searchers (in terms of their $F_1$ scores), along with the mean over all 48 searchers. This shows that within our user population, there is quite a large variation in terms of behaviour and performance. Of note is that the speed of processing has a major impact upon a user’s final performance. We can see from the best and second best searchers that they spent less time on documents and as a result examined substantial more that the average searcher. This translated into $F_1$ and CG scores of $F_1 = 0.25$, $CG = 42.0$ and $F_1 = 0.16$, $CG = 27.0$ respectively. This is in stark contrast to the other searchers, who on average obtained $F_1$ scores of $0.07 - 0.08$ and CG scores of $10.0 - 12.6$. Behavioural and performance values for the TREC. TREC’s stochastic and stochastic users are presented in Table 4. Values reported are averaged over the two topics examined, examining the mean over the 48 simulated searchers per experimental configuration.

The TREC user obtains $CG = 30.0$, and $F_1 = 0.32$ for the title only query ($TT$) which was significantly more than the actual searchers over both measures. Using the other querying strategies, the TREC user also obtains significantly higher CG and $F_1$ scores. We can see that $QS3^\ast$ produces a better initial query, resulting in the highest CG of 35.4, $F_1 = 0.36$ and $P = 0.41$, while $QS3$ results performs slightly worse at $CG = 18.0$, $F_1 = 0.22$ and $P = 0.19$. These results show that a very simple search strategy can result in very high performance. Indeed, under querying strategies $TT$ and $QS3^\ast$, every human subject is outperformed in terms of their $F_1$ scores (the best real-world subject obtaining $F_1 = 0.25$, compared to $TT$ at $F_1 = 0.32$ and $QS3^\ast$ at $F_1 = 0.36$), with $QS3$ outperforming all but one of our human subjects. In essence, starting (or sticking) with a good query leads to excellent performance. Of course, the search behaviour exhibited by the TREC simulated users is quite different from the behaviours of actual searchers. None of the behavioural or performance measures were similar (as illustrated by a complete lack of daggers $\dag$ in Table 4).

Turning our attention to the stochastic TREC user, (denoted sTREC in Table 4), we see that because the simulated user does not examine every document encountered, then they have more time to examine snippets, and thus examine more snippets during their search sessions. Coupled with the interaction probabilities, this translated into somewhat lower performance: $CG = 5.6-11.6$, precision (0.14-0.30) and $F_1$ scores (0.11-0.18). Interestingly, introducing stochastic behaviour leads to performance which is more in line with the worst searcher, and at best close to the median searcher. Again, all the behavioural and performance measures of the stochastic TREC users were significantly different to the real-world subjects, except in terms of the number TREC relevant documents ($#R$) and the CG for two of the querying strategies ($QS3^\ast$ and $TT$).

Next, we examined the stochastic user. Recall that in this setup of simulation, users will issue a number of queries - not just one - and it represents the type of simulation undertaken in many recent studies. Table 4 also presents the behaviour and performance of these simulated users for the two querying strategies and the three continuation strategies, averaged over all runs and all simulated users. Exhibited behaviour of the stochastic user broadly resembles that of human searchers. First, precision is in the region of 0.17-0.28, and the CG ranges from 8.7 – 12.6. This impacts upon how well they searched ($F_1 = 0.13 – 0.18$). While the stochastic users do behave more like human searchers, they do perform markedly better than the median and mean searchers in terms of the $F_1$ scores, comparing $F_1 = 0.07$ and $F_1 = 0.08$ against $F_1 = 0.13-0.18$. Depending upon the configuration, stochastic users tend to examine the same number of snippets, find the same number of TREC relevant documents, and achieve similar levels of CG when compared to real-world subjects. This shows that these stochastic users are more reflective of actual searchers.
is because as the smoothing threshold is increased, the score and an increase in the number of snippets examined. This responds with an increase in the number of queries issued, users. As the smoothing threshold is increased, we see that query, and so tend to be more akin to the TREC simulated agents. Very liberal agents tend to only issue one query, and then considers the subsequent document as relevant. As

\[
\lambda_A = \lambda_R \quad \text{and} \quad \mu_A = \mu_R
\]

are averaged over the two topics undertaken.

Since our search agents can be configured in a number of different ways, we first explore the influence of the different parameters on their performance and behaviour. To show how the decision threshold interacts with the smoothing parameter, we have plotted in Figure 4 the mean number of queries, the mean number of snippets and documents examined, and the mean \( F_1 \) score. Plots are averaged over the two topics undertaken.

![Figure 4: Plots, exploring the \( \lambda \) and \( \mu \) space for QS3' and SS3. Left: \( \lambda_A = \lambda_R \) and \( \mu_A = \mu_R \); right: \( \lambda_A = 0.8 \) and \( \mu_A = 0.2 \). Top to bottom: the mean number of queries, the mean number of snippets and documents examined, and the mean \( F_1 \) score.](image)

Table 4: Results of our baseline runs, including the TREC, Stochastic TREC (sTREC) and Stochastic users. TT’ denotes where the TREC title query was issued. Results are averaged over the two topics run.

<table>
<thead>
<tr>
<th>Ag.</th>
<th>SS</th>
<th>#Q</th>
<th>#S</th>
<th>#D</th>
<th>#M</th>
<th>#R</th>
<th>CG</th>
<th>P</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>1</td>
<td>10.7</td>
<td>133.9</td>
<td>49.5</td>
<td>19.5</td>
<td>5.6</td>
<td>9.0</td>
<td>0.17</td>
<td>0.12</td>
</tr>
<tr>
<td>S</td>
<td>1</td>
<td>10.2</td>
<td>125.1</td>
<td>47.0</td>
<td>46.4</td>
<td>13.8</td>
<td>21.9</td>
<td>0.32</td>
<td>0.27</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>15.2</td>
<td>191.2</td>
<td>36.2</td>
<td>35.5</td>
<td>10.1</td>
<td>14.8</td>
<td>0.31</td>
<td>0.23</td>
</tr>
<tr>
<td>M</td>
<td>2</td>
<td>7.5</td>
<td>137.8</td>
<td>52.0</td>
<td>19.2</td>
<td>4.6</td>
<td>8.2</td>
<td>0.12</td>
<td>0.11</td>
</tr>
<tr>
<td>ALL</td>
<td>1</td>
<td>10.5</td>
<td>130.8</td>
<td>48.9</td>
<td>27.8</td>
<td>6.6</td>
<td>11.0</td>
<td>0.20</td>
<td>0.16</td>
</tr>
</tbody>
</table>

Table 5: Results of comparison runs for each measure with the parameter configurations of \( \lambda_A \), \( \lambda_R \), \( \mu_A \) and \( \mu_R \) yielding the lowest MSE over the given continuation strategy (SS) for the given measures (rows). Also included is the lowest MSE based on combining all behavioural measures (ALL).

<table>
<thead>
<tr>
<th>Ag.</th>
<th>SS</th>
<th>#Q</th>
<th>#S</th>
<th>#D</th>
<th>#M</th>
<th>#R</th>
<th>CG</th>
<th>P</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>1</td>
<td>10.7</td>
<td>133.9</td>
<td>49.5</td>
<td>19.5</td>
<td>5.6</td>
<td>9.0</td>
<td>0.17</td>
<td>0.12</td>
</tr>
<tr>
<td>S</td>
<td>1</td>
<td>10.2</td>
<td>125.1</td>
<td>47.0</td>
<td>46.4</td>
<td>13.8</td>
<td>21.9</td>
<td>0.32</td>
<td>0.27</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>15.2</td>
<td>191.2</td>
<td>36.2</td>
<td>35.5</td>
<td>10.1</td>
<td>14.8</td>
<td>0.31</td>
<td>0.23</td>
</tr>
<tr>
<td>M</td>
<td>2</td>
<td>7.5</td>
<td>137.8</td>
<td>52.0</td>
<td>19.2</td>
<td>4.6</td>
<td>8.2</td>
<td>0.12</td>
<td>0.11</td>
</tr>
<tr>
<td>ALL</td>
<td>1</td>
<td>10.5</td>
<td>130.8</td>
<td>48.9</td>
<td>27.8</td>
<td>6.6</td>
<td>11.0</td>
<td>0.20</td>
<td>0.16</td>
</tr>
</tbody>
</table>

Table 4: Results of our baseline runs, including the TREC, Stochastic TREC (sTREC) and Stochastic users. TT’ denotes where the TREC title query was issued. Results are averaged over the two topics run.
cates when there was no significant difference between the agents and actual searchers. From Table 5, we can see that in terms of behaviours, the closest fits were when the stopping strategy was SS1 (except for the number marked when it was SS2). In terms of behaviour, the agents that were closest to actual searchers were when they were configured based upon the number of marked documents and with SS2, i.e. Agent M. In this case, they posed a similar number of queries, examined a similar number of snippets, and marked a similar number of documents as relevant - but examined more documents in total. However, the precision and CG of the agents were lower than the actual searchers. However, we can see that other configurations based upon behaviour can lead to substantially higher performance (i.e. Agents S and D in Table 5), and exhibit similar querying and snippet behaviour. Indeed, these agents outperformed every one of their real counterparts in terms of CG, precision and F1 scores. Agent ALL provides a good fit in terms of the number of queries and snippets, but still outperforms the actual searchers - but not to the same extent as Agents S and D.

These results are very encouraging as they suggest that we can configure the search agents based upon behaviours - without recourse to relevance judgements - and obtain interaction data that is similar to actual searchers. Here, we have only examined a subset of the possible space, and it is very likely that in order to better replicate actual searchers, the continuation strategy, smoothing parameters and decision thresholds would all have to be tailored to each individual.

6. DISCUSSION AND FUTURE WORK

In this paper, we have proposed an extension to the model commonly used to describe the search process for the purposes of simulation, and further augmented it with a representation of the user's cognitive state. The addition of the USM enables the creation of autonomous search agents rather than naive or stochastic simulated users. The addition of these components has required the combination and integration of a wide variety of different technologies and innovations within IR. This in turn has increased the complexity of developing simulated agents, but arguably creates simulated users which are more credible and realistic.

While it was not possible to fully explore the full range of configurations in this paper, we examined how the behaviour and performance of an agent changes, and how it could be configured to exhibit behaviours similar to human searchers. Further work is required to automatically select and tune parameters for the decision making components and continuation strategies, as well as balancing the interplay between the different components. Now that the models and infrastructure exist, it is possible to proceed in this direction. Nonetheless, our findings are promising and show that it is possible to create more realistic simulations and simulate interaction without recourse to relevance. We also observed that naive simulated TREC-style users behaved quite differently to human searchers, but obtained very high performance. Meanwhile, the simulated stochastic users - representing the previous state of the art - consistently performed poorly when compared to human searchers, but exhibited similar behaviours. With the development of autonomous search agents, a range of behaviours and performances can be configured to emulate humans, providing a more suitable and flexible alternative for the simulation and evaluation of IR systems.

To conclude, this work has advanced the state of the art - moving from stochastic to simultaneous simulated users to autonomous search agents. This was achieved by extending the CSM and introducing the USM to provide cognitive state and enable agency. These innovations open up a number of new research avenues that require further exploration, including: (i) how we best represent and update the searcher's cognitive state; (ii) what kind of framework should be used to make decisions (in this paper, we have used a probabilistic language modelling framework, but many others could be used); (iii) how we can best estimate or explore the range parameters to create different kinds of agents (i.e. agents like humans, and agents that perform or behave differently but more effectively, etc.); (iv) what we can learn from search agents about search systems (i.e. in terms of evaluation); (v) what areas we can apply and use such search agents (i.e. collaborative search, slow search, exploratory search, etc.); (vi) how searchers can effectively interact with such search agents; and (vii) how we can compare and evaluate the search performance and behaviours of agents and humans. Of course, other challenges exist, including the development of such agents for different tasks (novelty and diverse search tasks, etc.), contexts (work, leisure, casual, etc.) and environments (web, academic, enterprise, etc.).

Acknowledgments We wish to thank the anonymous reviewers for their insightful feedback. The lead author is funded by the UK Government through the EPSRC, grant number 1367507.

References
