Exploring LSTMs for Simulating Search Sessions in Digital Libraries

Sebastian Günther, Paul Göttert, and Matthias Hagen

Martin-Luther-Universität Halle-Wittenberg, Halle (Saale), Germany
first-name.last-name@informatik.uni-halle.de

Abstract. We explore the application of long short-term memory models (LSTM) to simulate search behavior in a digital library. Like web search engines, also digital libraries update the retrieval backend or the user interface. However, with the typically rather small user base, evaluating the changes based on user behavior analysis is difficult. To improve this process, we analyze whether an LSTM-based model can generate realistic user behavior data. Trained on a cleaned version of the SUSS dataset (555,008 search sessions), the LSTM model uses the whole session history to predict the next interaction. Our preliminary experiments show that this approach can generate realistic sessions.

Keywords: Simulation · Search Behavior · User Modeling · LSTM

1 Introduction

Web search engines like Google are able to evaluate and improve their retrieval backend via A/B tests on millions of daily user sessions. Most digital libraries, however, have much less traffic—making reliable evaluations via A/B tests much more difficult. Thus, several previous studies suggested to simulate digital library sessions via Markov models or other "classic" machine learning-based approaches. In our study, for the first time, we explore recurrent neural networks (RNN) with a long short-term memory architecture (LSTM) for session simulation. We start by cleaning an existing digital library session log on which we then use Keras and Tensorflow to train and tune LSTM models.

Instead of creating individual simulation models for specific aspects like query reformulation, stopping behavior, or dwell time, we want to evaluate whether a combination of features can be used to directly simulate complex behavior. Our focus is on simulating realistic interaction sequences while abstracting from fine-grained details like, for instance, the exact strings of possibly submitted queries. Besides the LSTM-based simulation approach, we also present and analyze metrics for session similarity and the quality of whole session logs.[1] Our study highlights the importance of not "overfitting" the simulated sessions to be too similar to the original data, but to enable the creation of also somewhat different sessions when utilizing machine learning for simulation.

[1] Code and data: https://github.com/webis-de/tpdl22-lstm-session-simulation

2 Related Work

Search interactions usually fall into a few key steps (i.e., query formulation, snippet and document examination, etc.) that Maxwell et al. [5] captured in the Complex Searcher Model (CSM) and implemented in the SimIIR framework [4]. SimIIR's implementations for the individual steps can easily be combined but are rather static and have limited interactions with each other (e.g., result clicks do not influence query formulation). For more realistic simulations, past interactions can play an important role as demonstrated by Cheng et al. [2] who used session history in their LSTM-based LostNet model for reranking and query prediction. In our study, we will thus try LSTMs to simulate whole sessions of interactions.

However, simply predicting future interactions from historic data also is difficult. Kinley et al. [3] run a user study with 50 participants on factual, exploratory, or abstract search tasks. The logged sessions (queries, time per interactions, etc.) show that search behavior is heavily dependent on the task. The same searcher is likely to have a high variation in different tasks that a machine learning model without knowledge of the task types might miss.

Besides simulating realistic search behavior, analyzing optimal strategies can also be interesting. Baskaya et al. [1] studied the impact of user behavior factors on the retrieval effectiveness. They came to the conclusion that there is no single best strategy for every task but that queries of two to three words (likely dependent on the search platform) work well and that average user behavior is not necessarily more realistic than optimal behavior.

An important aspect of realistic or optimal search simulation also is the time—with reading as a major factor (snippets, documents). Weller et al. [7] analyzed reading time for different text characteristics (e.g., font type, topic, length). On logged data of 1,000 study participants, they found that a simplistic text length-based model works very well to predict reading time. We will use regression models to simulate interaction times.

3 Search Session Dataset and Data Preparation

For our simulation study, we use the Sowiport User Search Sessions dataset (SUSS).[2] Sowiport [6] was a digital library for the field of social science and provided access to 18 national and international literature databases (total of 7.2 million entries like books or websites). The digital library was operated until 2017 by GESIS, the German Centre of Gerontology, the German Central Institute for Social Issues, the Friedrich Ebert Foundation, the University and City Library of Cologne, the Berlin Social Science Center and the Bertelsmann Foundation. In our study, we use the first version of the SUSS dataset, which was collected over a 1-year period starting from April 2, 2014. The dataset contains 558,008 sessions with a total of 7,982,427 lines (a user interaction can have more than one line; on average, every session has seven interactions).

[2] https://data.gesis.org/sharing/#!Detail/10.7802/1380

As the Sowiport platform is not available anymore and since the dataset documentation is somewhat limited, we tried to infer some information regarding the meaning and creation process of the user session log. Based on this, we filter, correct, and transform entries from the original dataset to extract sessions that can be used for LSTM simulation model training and evaluation.

Filtering. The SUSS dataset covers usage of the entire Sowiport platform. As we are only interested in activities related to search sessions, we removed sessions with no search-related interactions (e.g., sessions where users only visit pages about the history of Sowiport). More closely inspecting the log, we also encountered irregularities within sessions that we were not able to explain or fix. We therefore created a set of rules to detect anomalous log entries. However, as a large portion of sessions includes at least one such entry, we did not remove the entire sessions—this would have substantially reduced the overall number of usable sessions—but we only removed the anomalous actions and adjusted the remaining session data accordingly.

Correction. We identified a small subset of systematic irregularities cause by the logging process (e.g., missing duration for the last action of a session). In such cases, while we were unable to restore the original data, we "fixed" the sessions by extrapolating from respective interactions with time information. Although this technically changes the dataset, we still refer to this altered data as the original/real sessions in the remainder of our paper.

Transformation. The SUSS dataset stores all sessions in a table-like format with multiple rows per interaction. Instead, we group the data based on the interaction ID to have a single entry per interaction. We also split sessions at large time gaps between interactions (referred to as `part` in the log). This results in one object per session, for which features can be extracted to train models.

4 Model Training

We train LSTM models on 80% of the data using the open-source library Keras with the Tensorflow backend. Each input vector consists of at least two interaction steps from the training data. To evaluate the impact of features on prediction accuracy, we initially test two variants: one with five features (action length, action, subaction, origin action, response) and one with six additional features (searchterm type, searchterm length, searchterm complexity, sorted, page, informationtype). As the variant with eleven features is slightly better at predicting a session's next step, we continue using that model.

We use standard practices for feature encoding: normalizing continuous values after removing 5% outliers and one-hot encoding for categorical features. After some pilot studies, we choose two hidden layers for our models with 128 and 64 nodes, sigmoid as the activation function, cross-entropy as the loss function for our rather simple model, a learning rate of 0.001, and a batch size of 128.

Table 1. Basic characteristics of real sessions (test data) and simulated sessions.

Data	Interaction duration		Query length		Page number		Number of results	
	avg	**sd**	**avg**	**sd**	**avg**	**sd**	**avg**	**sd**
SUSS test data	46.95	354.18	12.52	8.02	1.24	0.59	11.71	5.28
LSTM-simulated	46.82	331.63	12.80	7.88	1.21	0.62	11.68	5.22

We also set class weights to boost interactions classes that are rare in our training dataset (e.g., search_person) and conclude training after 20 epochs, as the prediction accuracy only showed very small improvements with more epochs in the pilot experiments.

Some things the simulation has to "predict" are continuous values (e.g., interaction duration or query length) so that we use regression models for them. However, the SUSS data does not contain all the information needed to predict some values. An example is interaction time for reading interactions. Since the SUSS data does not contain the document content, reading time can only imperfectly guessed with some touch of randomness. Values like reading time or query length can only be simulated more realistically when more knowledge about the search intent, the result documents, or the shown snippets was available compared to what is contained in the SUSS data.

5 Experiments

Assessing simulated sessions is a difficult task, as there are no established agreements on which measures to choose. The tasks is further complicated by the multidimensional nature of the session data (i.e., multidimensional features and time durations). We therefore use three different approaches to assess the simulated sessions, with each approach covering different aspects.

5.1 Comparing Feature-based Metrics

We first compare some basic characteristics of real SUSS sessions from the test data to that of 1,000 LSTM-simulated sessions. The results in Table 1 show that, on average, the basic characteristics are very similar indicating that LSTM-based simulation seems to be promising.

5.2 Human Assessment

In our second assessment, we conduct a small pilot manual annotation to evaluate whether the sessions have a plausible "look and feel" from a human perspective. To examine the plausibility, we collected the sequence of interactions and duration, number of results, and the usage of pagination and sorting on individual timelines for 20 random real and 20 random simulated sessions. In a random order, each session was assessed as 'real' or 'simulated' with an optional reasoning by an expert familiar with the structure of actual SUSS data. In an exit questionnaire, our expert told us that their assessments were mostly based on the following three properties and possible issues of simulated sessions.

Interaction Sequence. Search sessions usually follow a cycle of submitting queries and examining results, comparable to the CSM [5]. This can be interleaved with changing parameters like sorting or pagination. Deviating from this cycle is an indicator for either a malformed session or could be attributed to a multi-browser-tab session.

Interaction Duration. Some interactions' durations can indicate abnormal behavior (e.g., assessing a document as relevant after zero seconds of reading). Again, there can also be legitimate reasons for such occurrences in real sessions like using multiple tabs, refreshing the page, or misclicks.

Parameters. A more technical detail to look at are the parameters of each interaction. While the parameter space is mostly plausible, there is still the possibility to create impossible combinations (e.g., viewing a results page with zero results and then examining one of "those" results). Such combinations are rare but could also be the result of using multiple tabs.

Note that the above properties exploited by the assessor and the possibly legitimate reasons for the issues to also occur in real sessions may have led to some wrong assessments. In our small pilot annotation, from the 20 real sessions, 16 were correctly identified as real, while 4 were falsely judged as simulated. From the simulated sessions, 8 were correctly identified as simulated, while 12 were convincing enough to be judged as real. So, also from this angle of looking at the simulated sessions, most of them seem plausible at least to the degree visible in their timelines.

5.3 Session Novelty

In our third assessment of the simulated sessions, we focus on the question of how "novel" the contained interaction sequences are since simulating search sessions usually has two somewhat conflicting goals. To be realistic, the sessions should be similar to real logged data in various aspects but at the same time they should also not be exact replicas (i.e., simply sampling from the existing logs is not asked for). Similar to Google's reported daily 15% of queries never seen before,[3] also simulated sessions should probably contain "new" sequences of interactions. Any machine learning-based simulation should not just output memorized sessions from the training data.

To estimate session similarity, we tried several features sets to represent sessions in a vector space and multiple metrics to compare the vectors. In our preliminary experiments, the best combination was interaction types and session duration as features and 'almost exact matches' as the similarity metric

[3] https://blog.google/products/search/our-latest-quality-improvements-search/

Table 2. Ratio of novel sessions in the SUSS test data and in the LSTM-based simulation in two scenarios: 80% or 90% of the SUSS data used for training.

	80% training	**90% training**
SUSS test sessions	5.46%	5.18%
Simulated sessions	5.91%	5.74%

(i.e., looking for sessions with the same interactions, in the same order, that take about the same amount of time).

In a respective experiment, we compare the "novelty" of simulated sessions to that of real SUSS sessions in two scenarios. In the first scenario, we train the LSTM simulation on the first 80% of the SUSS sessions and let the remaining 20% be simulated while in the second scenario the training uses 90% of the SUSS sessions and the remaining 10% then are simulated. In both scenarios, the ratio of simulated sessions with no almost exact match in the training sessions (i.e, the amount of "novel" sessions) is compared to the ratio of novel sessions in the test data (the real SUSS sessions not used for training). The results in Table 2 show that in both scenarios the novelty ratios of real and simulated sessions are in the same range. The LSTM simulation thus is promising from a novelty view.

6 Conclusion and Future Work

We have shown some preliminary results on using LSTM models to simulate search sessions in a digital library. For our study, we filtered and transformed the SUSS dataset to extract suitable search sessions. The interaction histories were compiled into short time series datasets, that we used in the training process. Using varying length and details for the input time series data, we trained multiple models for the prediction task. In a preliminary experimental analysis, we assessed the simulated sessions with respect to basic statistical characteristics, with respect to their plausibility in a human identification of real and simulated sessions, and with respect to their novelty compared to the training data. Our results indicate that LSTM-based session simulation is very promising from all three assessment angles.

Since our preliminary results are based on rather small-scale experiments so far, we want to generalize them in future work by comparing LSTM-based session simulation to other approaches like Markov modeling or the rather simpler approaches implemented in the SimIIR simulation framework. In case that the promising results we saw in our current study then still hold, we plan to integrate LSTM-based session simulation in SimIIR.

Acknowledgements This work has been partially supported by the DFG (German Research Foundation) through the project 408022022 "SINIR – Simulating INteractive Information Retrieval".

Bibliography

[1] Baskaya, F., Keskustalo, H., Järvelin, K.: Modeling behavioral factors in interactive information retrieval. In: He, Q., Iyengar, A., Nejdl, W., Pei, J., Rastogi, R. (eds.) Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM 2013), San Francisco, CA, USA, October 27 – November 1, 2013, pp. 2297–2302, ACM (2013)

[2] Cheng, Q., Ren, Z., Lin, Y., Ren, P., Chen, Z., Liu, X., de Rijke, M.: Long short-term session search: Joint personalized reranking and next query prediction. In: Proceedings of The Web Conference 2021 (WWW 2021), Virtual Event / Ljubljana, Slovenia, April 19-23, 2021, pp. 239–248, ACM/IW3C2 (2021)

[3] Kinley, K., Tjondronegoro, D., Partridge, H., Edwards, S.L.: Relationship between the nature of the search task types and query reformulation behaviour. In: Trotman, A., Cunningham, S.J., Sitbon, L. (eds.) The Seventeenth Australasian Document Computing Symposium (ADCS 2012), Dunedin, New Zealand, December 5–6, 2012, pp. 39–46, ACM (2012)

[4] Maxwell, D., Azzopardi, L.: Simulating interactive information retrieval: SimIIR: A framework for the simulation of interaction. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2016), Pisa, Italy, July 17–21, 2016, pp. 1141–1144, ACM (2016)

[5] Maxwell, D., Azzopardi, L., Järvelin, K., Keskustalo, H.: Searching and stopping: An analysis of stopping rules and strategies. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM 2015), Melbourne, VIC, Australia, October 19 – 23, 2015, pp. 313–322, ACM (2015)

[6] Mayr, P.: Sowiport User Search Sessions data set (SUSS) (Version: 1.0.0) (2016)

[7] Weller, O., Hildebrandt, J., Reznik, I., Challis, C., Tass, E.S., Snell, Q., Seppi, K.D.: You don't have time to read this: An exploration of document reading time prediction. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J.R. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020), Online, July 5–10, 2020, pp. 1789–1794, Association for Computational Linguistics (2020)