Historical Clicks for Product Search: GESIS at CLEF LL4IR 2015

Philipp Schaer[1] and Narges Tavakolpoursaleh[12] *

[1] GESIS – Leibniz Institute for the Social Sciences, 50669 Cologne, Germany
firstname.lastname@gesis.org
[2] University of Bonn, Computer Science / EIS, 53117 Bonn, Germany

Abstract. The Living Labs for Information Retrieval (LL4IR) lab was held for the first time at CLEF and GESIS participated in this pilot evaluation. We took part in the product search task and describe our system that is based on the Solr search engine and includes a re-reranking based on historical click data. This brief workshop note also includes some preliminary results, discussion and some lessons learned.

1 Introduction

In 2015 the Living Labs for Information Retrieval initiative (LL4IR) for the first time organized a lab at the CLEF conference series [5]. This lab can be seen as a pilot evaluation lab or as stated by the organizers: "a first round". GESIS took part in this pilot to get a first hand experience with the lab's API [1] and the rather new evaluation methodology. The main focus was not on winning the implicit competition every evaluation campaign is, but to learn more about the procedures and the systems used. Since this lab had no direct predecessor we could not learn from previous results and best practices. Previous systems that we used in other CLEF labs, namely the CHiC lab on cultural heritage [4], were from a totally different domain and could therefore not be directly applied to the use cases of LL4IR. So, the main objective of this pilot participation was to establish the retrieval environment and to surpass the obvious issues in the first place. After the initial three tasks were cut down to only two, we took part in the remaining task on product search using the REGIO JÁTÉK e-commerce site.

In the following paper we will present our approaches and preliminary results from their assessments.

2 Product Search with REGIO JÁTÉK

One of the remaining of formally three different tasks were the product search on the e-commerce site REGIO JÁTÉK. As previously noted in the Living Labs Challenge Report [2] this specific task introduces a range of different challenges,

* Authors are listed in alphabetical order.

issues, and possibilities. In the report some issues like "generally little amount of textual material associated with products (only name, description, and name of categories)" were noted. On the other hand additional information included in the available metadata were listed, among these were: (1) Historical click information for queries, (2) collection statistics, (3) product taxonomy, (4) product photos, (5) date/time when the product first became available, (6) historical click information for products, and (7) sales margins.

In our approach we decided to re-use the historical click data for products and a keyword-based relevance score derived from a Solr indexation of the available product metadata.

2.1 Ranking Approach

We used a Solr search server (version 5.0.0) to index all available metadata provided by the API for every document related to the given queries. For each document (or more precisely for each product) we additionally stored the corresponding query number. This way we were able to retrieve all available candidate documents and rank them according to the Solr score based on the query string. Additionally we added the historical click rates as a weighting factor into the final ranking if this was available at query time.

Some query related documents had no term which can be matched with the query string and therefore we were not able to retrieve every query related document on a mere query string-based search. We had to add the query number to the query itself as a boolean query part to fix this issue and used Solr's query syntax and the default QParserPlugin[3]. This syntax allows the use of boolean operators and of different boosting factors which were both used in the query formulation:

```
qid:query[id]^0.0001 OR (qid:query[id]^0.0001 AND query[str])
```

Using this query string we got a Solr-ranked list of documents for each query which were then re-ranked using the historical click rates as outlined in algorithm 1. Basically it's a linear combination of a boosted search on the document id (field name docid) and the vector space-based relevance score of the query string. This is a typical "the rich are getting richer" approach where formally successful products are more likely to be once again ranked high in the result list. The approach was inspired by a presentation by Andrzej Białecki [3].

2.2 Solr Configuration

As stated above we used a Solr installation. To keep the system simple, we used the original Solr configuration and imported the REGIO dump using the originally provided schema.xml and the configuration from table 1. We did not include any language specific configurations for stemmers or stop word lists, since the Hungarian Solr stemmer returned the same results as the generic stemmer. We used the following standard components for text_general fields:

[3] `https://wiki.apache.org/solr/SolrQuerySyntax`

Algorithm 1: Re-ranking algorithm merging the Solr ranking score and the historical click rates.

Data: runs of production system correspond to the queries to products of REGIO JAŤEḰ site
Result: runs of our experimental system according to the document's fields and click-through rate
for *query in queries* **do**
    run = get_doclist(query)
    ctr = get_ ctr(query)
    for *doc in run* **do**
        doc_ detail = get_ docDetail(doc)
        BuildSolrIndex (doc_detail,qid)
    end
    myQuery = $(docid_1\hat{}ctr_1$ OR $docid_2\hat{}ctr_2$ OR ... OR $docid_n\hat{}ctr_n)$ OR (qid^0.0001 AND query['str'])
    myRun = solr.search(myQuery)
    update_runs(key, myRun , feedbacks)
end

- StandardTokenizerFactory: A general purpose tokenizer, which divides a string into tokens with various types.
- StopFilterFactory: Words from the Solr included stopword lists are discarded.
- LowerCaseFilterFactory: All letters are indexed and queried as lowercase.

The detailed indexation configuration of the given fields is listed in table 1. We used a very limited set of fields for the first round and were basically only searching in the title and description field. We changed this in the second round where we included all the available metadata in the search.

3 Results

3.1 Official Run

The results of the campaign were documented by giving numbers on the (1) impressions per query, (2) wins, losses, and ties calculated against the production system, and (3) the calculated outcome (#wins/(#wins+#losses). As noted in the documentation a win "is defined as the experimental system having more clicks on results assigned to it by Team Draft Interleaving than clicks on results assigned to the production system". This means that any value below 0.5 can be seen as a performance worse than the production system. Due to the problem of unavailable items in the shop the expected outcome had to be corrected to 0.28 as unavailable items were not filtered out for participating systems (for more details check the workshop overview paper [5]).

Our system received 523 impressions in the two weeks test period. This makes roughly 37.4 impressions per day and 1.6 impressions per hour. Although we

Table 1. Solr field configuration for the available product metadata. Since different configurations for round #1 and #2 were used we also report in the usage of the fields for the two evaluation rounds.

field	type	multiValued	round #1	round #2
title	text_general	✓	✓	✓
category	text_general			✓
content	text_general			✓
creation_time	string			✓
docid	text_general		✓	✓
main_category	text_general			✓
brand	text_general			✓
product_name	text_general			✓
photos	string	✓		✓
short_description	text_general			✓
description	text_general		✓	✓
category_id	string			✓
main_category_id	string			✓
bonus_price	float			✓
available	string			✓
age_min	string			✓
age_max	string			✓
characters	string	✓		✓
queries	text_general	✓		✓
gender	string	✓		✓
arrived	string			✓
qid	string		✓	✓
characters	string			✓
site_id	string			✓

don't have any comparable numbers we interpret these impression rates to be quite low. If we compare to the other teams we received the lowest number of impressions while for example system UiS-Mira received 202 more impressions in the same time period (725 impressions which is 38% more impressions than we got). This is not quite in line with the principle of giving fair impression rates between the different teams. Another thing regarding the impressions is the fact that different queries had very different impression rates (see figure 1). While some of them got more than 50 impressions others (actually 5) were not shown to the users at all.

Our approach did not perform very well due to some obvious misconfiguration and open issues of our implementation. In fact we provided the least efficient ranking compared to the other three participants [5]. We could achieve an outcome of 0.2685 by getting 40 wins vs. 109 losses and 374 ties. On the other hand no other participant was able to beat the baseline with an outcome rate of 0.4691. The best performing system received an outcome rate of 0.3413

Table 2. Some basic statistics on the results.

	round #1	round #2
test queries	50	50
impressions total	523	816
impressions per day	37.4	58.1
queries with no impressions	5	2
queries with outcome > 0.5	4	13
queries with outcome > baseline	10	13

Table 3. Outcome, wins, losses and ties from round #1 and #2.

	outcome	wins	losses	ties
round #1	0.2685	40	109	374
round #2	0.4520	80	97	639

(system UiS-Mira) and was able to be better than the expected outcome of 0.28 but below the simple baseline provided by the organizers.

3.2 Unofficial 2nd Round

We also took part in the 2nd evaluation round and adapted some parameters of the system. As there was a misconfiguration in the Solr system of round #1 we only searched the titles and description of products. We fixed this bug so that for round #2 we correctly indexed all available metadata fields. Another issue from round #1 was that not all 50 test topics were correctly calculated. We only used the historical click data for 1 test topic and 13 training topics. The other topics were just the standard Solr ranking without any click history boosting. We fixed this issue for round #2 where now all 50 topics are correctly calculated according to the described boosting approach.

After we corrected these two points we observe a clear increase in the outcomes. The outcome increased to 0.4520 by getting 80 wins, 97 losses and 639 ties. Although the performace increase might be due to the fixes introduced by the organizers regarding unavailable items we could still see some positive effects: The performance of the other teams increased too but while we were the weakest team in round #1 we were now able to provide the second best system performance. We also outperformed the winning system from round #1. Nevertheless we (and no other system) was able to compete with the productive system.

Comparing the number of impressions we see a clear increase in queries that are above the 0.5 threshold and the baseline (13 queries each) and the impressions in total and per day are also increased. The issue of unbalanced impression rates stays the same for round #2 (see figure 1).

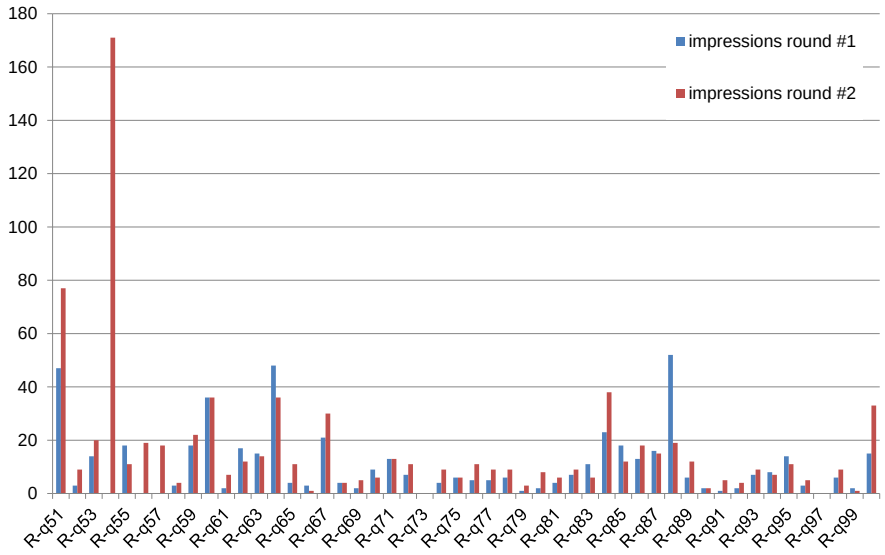

Fig. 1. Distribution of impressions per topic for the first and official CLEF round (blue) and the second unofficial test round (red).

4 Lessons Learned, Open Issues, and Future Work

The first prominent issue that arose when processing the data collection was the Hungarian content. Since we don't know Hungarian we were not able to directly read or understand the content or queries and therefore had used a language- and content-agnostic approach. Although the different fields were documented[4] the content was hidden behind the language barrier, except for obvious brand names like Barbie or He-Man.

It would have been really interesting and maybe useful to make use of further provided metadata like for example the classes of the two-level deep topical categorization system to which the products were assigned to. As we don't know more about this categorization system, expect for the ad-hoc translation[5] we could only add the category names to the index and leave it with that.

A typical problem with real world systems was also present in the available queries: Real world users tend to use rather short queries. For the 100 available query strings only 15 had more than one word and only 2 had more than 2 words

[4] `http://doc.living-labs.net/en/latest/usecase-regio.html#usecase-regio`

[5] `https://translate.google.com/translate?sl=auto&tl=en&u=http\%3A\%2F\%2Fwww.regiojatek.hu\%2Fkategoriak.html`

(R-q22: "bogyó és babóca" and R-q50: "my little pony"). The average word length per query was 1.17 and the average string length was 7.16 characters.

Another factor that we did not think of, although it was clearly stated in the documentation and in the key concepts[6], was the fact that no feedback data was available during the test phase. As this came to our mind way too late we were only able to include historical click data for some queries. Therefore the validity of our results from round #1 is weak as their are to few queries to really judge on the influence of the historical click data vs. live click data. We were not able to include new feedback data into our rankings after the official upload and the beginning of the test phase. All the uploaded rankings were "final" and only depend on historical clicks. While this is of course due to the experimental setup it is not truly a "living" component in the living lab environment (metaphorically spoken). On top of that not every document received clicks and therefore some documents are missing any hint of being relevant at all.

Last but not least we had to struggle with speed issues of the LL4IR platform itself. As mentioned in the workshop report of 2014 there are known issues on "scaling up with the number of participants and sites talking to the API simultaneously" [2]. Although they state that these bottlenecks have been identified and that they have started addressing these it still takes some time to correspond with the API. To get a feeling for the lack of the systems performance: The extraction of the final evaluation outcomes took roughly 45 minutes to extract 100 short JSON snippets not longer than in listing 1.1. Same is true for the generation of the list of available queries, result list and other data sets. The development team should thing about caching these kind of data.

5 Conclusion

To sum it up, we succeeded in our primary objective of our participation which was to learn about the LL4IR API and the evaluation methodology. We could clearly improve our results from round #1 to round #2 and learned a lot during the campaign. We fixed some obvious configuration issues in our Solr system and were therefore desperately looking forward to the start of the second phase of the evaluation that started on 15 June 2015. As it turned out, these issues could be solved and the performance of the retrieval could be clearly improved. Although we are not able to simply compare round #1 and #2 due to the misconfiguration we can see the positive effects of the including of historical click data to boost on popular products.

Acknowledgements

Besides the previously (maybe unfair) mentioned complains about the system's and API's performance we are really impressed by it's functionality and stability. The online documentation and the support by the organizers were clear, direct and always helpful. Thank you.

[6] `http://doc.living-labs.net/en/latest/guide-participant.html#key`

Listing 1.1. Sample output of the outcome documentation


```
{
    "outcomes": [
        {
            "impressions": 36,
            "losses": 11,
            "outcome": "0.15384615384615385",
            "qid": "R-q60",
            "site_id": "R",
            "test_period": {
                "end": "Sat, 16 May 2015 00:00:00 -0000",
                "name": "CLEF LL4IR Round #1",
                "start": "Fri, 01 May 2015 00:00:00 -0000"
            },
            "ties": 23,
            "type": "test",
            "wins": 2
        }
    ]
}
```


References

1. Balog, K., Kelly, L., Schuth, A.: Head first: Living labs for ad-hoc search evaluation. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. pp. 1815–1818. CIKM '14, ACM (2014), http://www.anneschuth.nl/wp-content/uploads/2014/08/cikm2014-lleval.pdf
2. Balog, K., Kelly, L., Schuth, A.: Living labs for ir challenge workshop (2014), http://living-labs.net/wp-content/uploads/2014/05/LLC_report.pdf
3. Białecki, A.: Implementing click-through relevance ranking in solr and lucidworks enterprise (2011), http://de.slideshare.net/LucidImagination/bialecki-andrzej-clickthroughrelevanceranking insolrlucidworksenterprise
4. Schaer, P., Hienert, D., Sawitzki, F., Wira-Alam, A., Lüke, T.: Dealing with sparse document and topic representations: Lab report for chic 2012. In: CLEF 2012 Labs and Workshop, Notebook Papers: CLEF/CHiC Workshop-Notes (2012)
5. Schuth, A., Balog, K., Kelly, L.: Overview of the living labs for information retrieval evaluation (ll4ir) clef lab 2015. In: CLEF 2015 - 6th Conference and Labs of the Evaluation Forum. Lecture Notes in Computer Science (LNCS), Springer (September 2015)