Inf Retrieval J (2016) 19:230–255
DOI 10.1007/s10791-015-9276-9

The strange case of reproducibility versus representativeness in contextual suggestion test collections

Thaer Samar[1] · **Alejandro Bellogín**[3] · **Arjen P. de Vries**[1,2]

Received: 3 May 2015 / Accepted: 19 October 2015 / Published online: 28 December 2015
© The Author(s) 2015. This article is published with open access at Springerlink.com

Abstract The most common approach to measuring the effectiveness of Information Retrieval systems is by using test collections. The Contextual Suggestion (CS) TREC track provides an evaluation framework for systems that recommend items to users given their geographical context. The specific nature of this track allows the participating teams to identify candidate documents either from the Open Web or from the ClueWeb12 collection, a static version of the web. In the judging pool, the documents from the Open Web and ClueWeb12 collection are distinguished. Hence, each system submission should be based only on one resource, either Open Web (identified by URLs) or ClueWeb12 (identified by ids). To achieve reproducibility, ranking web pages from ClueWeb12 should be the preferred method for scientific evaluation of CS systems, but it has been found that the systems that build their suggestion algorithms on top of input taken from the Open Web achieve consistently a higher effectiveness. Because most of the systems take a rather similar approach to making CSs, this raises the question whether systems built by researchers on top of ClueWeb12 are still representative of those that would work directly on industry-strength web search engines. Do we need to sacrifice reproducibility for the sake of representativeness? We study the difference in effectiveness between Open Web systems and ClueWeb12 systems through analyzing the relevance assessments of documents identified from both the Open Web and ClueWeb12. Then, we identify documents that overlap between the relevance assessments of the Open Web and ClueWeb12, observing a dependency between relevance assessments and the source of the document

✉ Arjen P. de Vries
arjen@cwi.nl; a.p.devries@tudelft.nl

Thaer Samar
samar@cwi.nl

Alejandro Bellogín
alejandro.bellogin@uam.es

[1] Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

[2] Delft University of Technology, Delft, The Netherlands

[3] Universidad Autónoma de Madrid, Madrid, Spain

being taken from the Open Web or from ClueWeb12. After that, we identify documents from the relevance assessments of the Open Web which exist in the ClueWeb12 collection but do not exist in the ClueWeb12 relevance assessments. We use these documents to expand the ClueWeb12 relevance assessments. Our main findings are twofold. First, our empirical analysis of the relevance assessments of 2 years of CS track shows that Open Web documents receive better ratings than ClueWeb12 documents, especially if we look at the documents in the overlap. Second, our approach for selecting candidate documents from ClueWeb12 collection based on information obtained from the Open Web makes an improvement step towards partially bridging the gap in effectiveness between Open Web and ClueWeb12 systems, while at the same time we achieve reproducible results on well-known representative sample of the web.

Keywords Reproducibility · Contextual suggestion · Open vs archived web · Test collections evaluation · Filtering and recommendation · Web IR and social media search

1 Introduction

Recommender systems aim to help people find items of interest from a large pool of potentially interesting items. The users' preferences may change depending on their current context, such as the time of the day, the device they use, or their location. Hence, those recommendations or suggestions should be tailored to the context of the user. Typically, recommender systems suggest a list of items based on users' preferences. However, awareness of the importance of context as a third dimension beyond users and items has increased, for recommendation (Adomavicius and Tuzhilin 2011) and search (Melucci 2012) alike. The goal is to anticipate users' context without asking them, as stated in The Second Strategic Workshop on Information Retrieval (SWIRL 2012) (Allan et al. 2012): "Future information retrieval systems must anticipate user needs and respond with information appropriate to the current context without the user having to enter a query". This problem is known as contextual suggestion in Information Retrieval (IR) and *context-aware recommendation* in the Recommender Systems (RS) community.

The TREC Contextual Suggestion (CS) track introduced in 2012 provides a common evaluation framework for investigating this task (Dean-Hall et al. 2012). The aim of the CS task is to provide a list of ranked suggestions, given a location as the (current) user context and past preferences as the user profile. The public Open Web was the only source for collecting candidate documents in 2012. Using APIs based on the Open Web (either for search or recommendation) has the disadvantage that the end-to-end contextual suggestion process cannot be examined in all detail, and that reproducibility of results is at risk (Hawking et al. 2001, 1999). To address this problem, starting from 2013 participating teams were allowed to collect candidate documents either from Open Web or from the ClueWeb12 collection.

In the 2013 and 2014 editions of CS track, there were more submissions based on the Open Web compared to those based on the ClueWeb12 collection. However, to achieve reproducibility, ranking web pages from ClueWeb12 should be the preferred method for scientific evaluation of contextual suggestion systems. It has been found that the systems that build their suggestion algorithms on top of input taken from the Open Web achieve consistently a higher effectiveness than systems based on the ClueWeb12 collection. Most

of the existing works have relied on public tourist APIs to address the contextual suggestion problem. These tourist sites (such as Yelp and Foursquare) are specialized in providing tourist suggestions, hence those works are focused on re-ranking the resulting candidate suggestions based on user preferences. Gathering suggestions (potential venues) from the ClueWeb12 collection has indeed proven a challenging task. First, suggestions have to be selected from a very large collection. Second, these documents should be geographically relevant (the attraction should be located as close as possible to the target context), and they should be of interest for the user.

The finding that Open Web results achieve higher effectiveness raises the question whether research systems built on top of the ClueWeb12 collection are still representative of those that would work directly on industry-strength web search engines. In this paper, we focus on analyzing reproducibility and representativeness of the Open Web and ClueWeb12 systems. We study the gap in effectiveness between Open Web and ClueWeb12 systems through analyzing the relevance assessments of documents returned by them. After that, we identify documents that overlap between Open Web and ClueWeb12 results. We define two different sets of overlap: First, the overlap in the relevance assessments of documents returned by Open Web and ClueWeb12 systems, to investigate how these documents were judged according to the relevance assessments gathered when they were considered by Open Web or ClueWeb12 systems. The second type of overlap is defined by the documents in the relevance assessments of the Open Web systems which are in ClueWeb12 collection but not in the relevance assessments of ClueWeb12 systems. The purpose is to use the judgments of these documents (mapped from Open Web on ClueWeb12 collection) to expand the relevance assessments of ClueWeb12 systems resulting on having a new test collection. Figure 1 illustrates these different test collections, the details given in Sect. 3.3. Then, we focus on how many of the

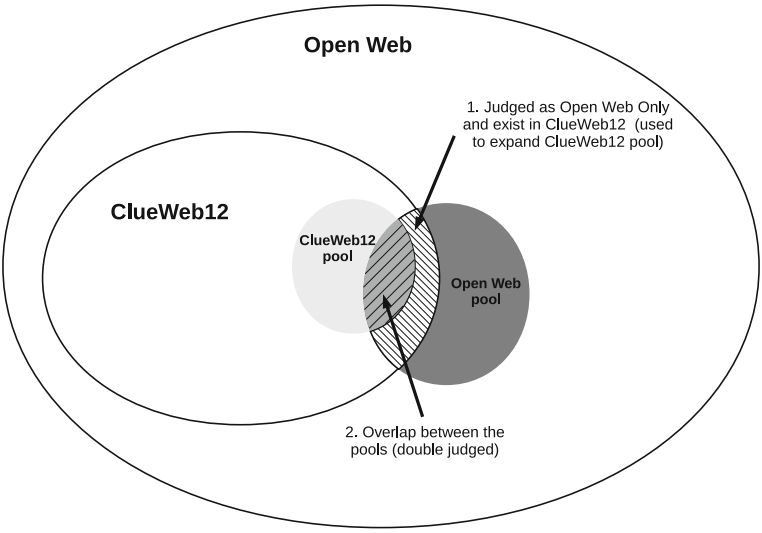

Fig. 1 Illustration of the relation between pools and the source of the documents. *Subset 1* represents the documents in the Open Web pool and were found in ClueWeb12 collection but do not exist in the ClueWeb12 pool (this subset is used to expand the ClueWeb12 pool). *Subset 2* represents the overlap between the Open Web pool and ClueWeb12 pool, documents in this subset were double judged (we use this subset to show the bias between Open Web and ClueWeb12 results)

documents returned by Open Web systems can be found in the ClueWeb12 collection, an analysis to assess the reproducibility point of view. Finally, we apply the knowledge about the tourist information available in the Open Web for selecting documents from ClueWeb12 to find a representative sample from the ClueWeb12 collection. Specifically, we address the following research questions:

RQ1 Do relevance assessments of Open Web URLs differ (significantly) from relevance assessments of ClueWeb12 documents?
RQ2 Can we identify an overlap between Open Web systems and ClueWeb12 systems in terms of documents suggested by both?, how are those documents in the overlap judged?
RQ3 How many of the documents returned by Open Web systems can be found in the ClueWeb12 collection as a whole?
RQ4 Can we identify a representative sample from the ClueWeb12 collection for the CS track by applying the tourist domain knowledge obtained from the Open Web?

The remainder of the paper is organized as follows: first we discuss related work (Sect. 2), followed by a description of the experimental setup (Sect. 3). After that we present an analysis to compare Open Web and ClueWeb12 relevance assessments (Sect. 4). Then we discuss how much of the Open Web systems can be reproduced from the ClueWeb12 collection, and we evaluate them on the ClueWeb12 test collection (Sect. 5). After that we discuss how to apply tourist domain knowledge available on the public Open Web to annotate documents from the ClueWeb12 collection. Finally, we discuss conclusions drawn from our findings (Sect. 7).

2 Related work

In the Recommender Systems area, recommendation algorithms for several types of content have been studied (movies, tourist attractions, news, friends, etc.). These types of algorithms are typically categorized according to the information they exploit: collaborative filtering (based on the preferences of like-minded users Resnick et al. 1994) and content-based filtering (based on similar items to those liked by the user Lops et al. 2011). In the Information Retrieval area, approaches to *contextual suggestion* usually follow a content-based recommendation approach. The majority of related work results from the corresponding TREC track, focusing on the specific problem of how to provide tourist attractions given a location as context, where many participants have relied on APIs of location-based services on the Open Web. Candidate suggestions based on location are then ranked based on their similarity with the known user interests. In this case, the key challenge is to model user interests.

Given the description of a set of examples (suggestions) judged by the user, existing studies exploit the descriptions of the suggestions to build her profile, usually represented as the textual information contained in the description of the suggestions. Sappelli et al. (2013) build two user profiles: a positive profile represents terms from those suggestions liked by the user before, whereas a negative profile is based on descriptions of suggestions disliked by the user. In Hubert et al. (2013); Yang and Fang 2012) both the descriptions and the categories of the suggestions are used to build the user profiles. In Yang and Fang (2013), the authors proposed an opinion-based approach to model user profiles by leveraging similar user opinions of suggestions on public tourist APIs. If the user rated a

suggestion as relevant, then the positive profile represents all positive reviews of that suggestion. The negative profile represents all negative reviews of the suggestion rated as irrelevant to the user. The aforementioned approaches consider different ranking features based on the similarity between candidate suggestions and positive and negative profiles. On the other hand, a learning to rank model exploiting 64 features using information obtained from Foursquare is presented by Deveaud et al. (2014). They used four groups of features: (a) city-dependent features which describe the context (city) such as total number of venues in the city and total number of likes, (b) category-dependent features that consist of the count of the 10 highest level categories obtained from Foursquare, (c) venue-dependent features which describe the popularity of the venue in the city, and (d) user-dependent features describing the similarity between user profiles and the suggestions. The most effective features were the venue-dependent features, that is, those indicating venue importance.

Besides recommendation, a critical part of our work is how to build test collections and create sub-collections from them. Because of this, we now introduce the topic and survey some of the most relevant works on that area. Creating a test collection is the most common approach for evaluating different Information Retrieval systems. Any test collection consists of a set of topics, a set of relevance assessments, and a set of retrievable documents. Since the beginning of IR evaluation by means of test collections, many researchers have looked at test collections from different angles. For example, what is the optimal number of topics to obtain reliable evaluations? In Voorhees and Buckley (2002) the authors find that to have a reliable order of the systems, at least 50 topics have to be used in the evaluation stage. More recently, in Dean-Hall and Clarke (2015) the authors use data from the CS track to give insights about the required number of assessors. The problem of analyzing the impact of different sub-collections (as a set of test collections) is also studied in the literature. In Scholer et al. (2011), the authors split TREC ad-hoc collections into two sub-collections and compared the effectiveness ranking of retrieval systems on each of them. They obtained a low correlation between the two rank runs, each run based on one of the two sub-collections. Later, in Sanderson et al. (2012) a more exhaustive analysis is presented. The authors studied the impact of different sub-collections on the retrieval effectiveness by analyzing the effect over many test collections divided using different splitting approaches. Their study was based on runs submitted to two different TREC tracks, the `ad hoc` track from 2002 to 2008 and the `terabyte` one from 2004 to 2008. The authors found that the effect of these sub-collections is substantial, even affecting the relative performance of retrieval systems. In Santos et al. (2011), the authors analyze the impact of the first-tier documents from ClueWeb09 collection in the effectiveness. The analysis was carried out on the TREC 2009 Web track, where participating teams were encouraged to submit runs based on Category A, and Category B. These categories were extracted from ClueWeb09 collection. Category A consists of 500 million English documents, Category B is a subset from Category A, it consists of 50 million documents of high quality seed documents and Wikipedia documents (they represent the first-tier documents). By analyzing the number of documents per subset and the relevance assessment, the authors found a bias towards Category B documents, in terms of assessed documents and those judged as relevant. In order to investigate this bias, they analyze the effect of first-tier documents on the effectiveness of runs based on Category A. First, they found that there is a high correlation between effectiveness and number of documents retrieved from the first-tier subset. Second, by removing all documents not from the first-tier subset, the effectiveness of almost all runs based on Category A was improved.

In the context of the CS track these questions arise again, since in this track participants share the same topics (profile, context) but they have to return a ranked list of documents for each topic, where these candidate documents can be selected from either the Open Web or ClueWeb12 collection. Considering the potential impact that different collections may have on the retrieval effectiveness, one of our main interests in the rest of the paper is to study the gap in effectiveness between Open Web systems and ClueWeb12 systems in order to achieve reproducible results on a representative sample of the Web from ClueWeb12 collection.

3 Experimental setup

3.1 Dataset

Our analyses are based on data collected from the TREC 2013 and 2014 Contextual Suggestion tracks (`CS 2013`, `CS 2014`). The CS track provides a set of profiles and a set of geographical contexts (cities in the United States) and the task is to provide a ranked list of suggestions (up to 50) for each topic (profile, context) pair. Each profile represents a single assessor past preferences for a given suggestion. Each user profile consists of two ratings per suggestion, on a 5-point scale; one rating for a suggestion's description as shown in the result list (i.e., a snippet), and another rating for its actual content (i.e., a web page). There are some differences between 2013 and 2014: First, the 50 target contexts used each year are not the same. Second, seeds cities from which the example suggestions were collected: in 2013 examples were collected from Philadelphia, PA, whereas in 2014 examples were collected from Chicago, IL and Santa Fe, NM. Third, the number of assessors also changed in these editions of the track. More details about the CS track can be found in the track's overview papers (Dean-Hall et al. 2013, 2014), for 2013 and 2014, respectively.

The evaluation is performed as follows. For each topic—(profile, context) pairs—the top-5 documents of every submission are judged by the actual users whose profile is given (resulting in three ratings: description, actual document content, and geographical relevance assessments) and by NIST assessors (an additional rating for the geographical relevance assessment). Judgments are graded: subjective judgments range from 0 (strongly uninterested) to 4 (strongly interested) whereas objective judgments go from 0 (not geographically appropriate) to 2 (geographically appropriate). In both cases, a value of -2 indicates that the document could not be assessed (for example, the URL did not load in the judge's Web browser interface).

Documents are identified by their URLs (if they are submitted by runs based on Open Web) or by their ClueWeb12 ids (if they are submitted by runs based on ClueWeb12). In our study, we use `ClueWeb12-qrels` to refer to relevance assessments of ClueWeb12 documents, and `OpenWeb-qrels` to refer to relevance assessments of Open Web URLs, both sets of assessments built from the three relevance assessments files provided by the organizers: desc-doc-qrels, geo-user-qrels, and geo-nist-qrels.

The following metrics are used to evaluate the performance of the participating teams: Precision at 5 (P@5), Mean Reciprocal Rank (MRR), and a modified Time-Biased Gain (TBG) (Dean-Hall et al. 2013). These metrics consider geographical and profile relevance (both in terms of document and description judgments), taking as thresholds a value of 1 and 3 (inclusive), respectively.

3.2 URL normalization

A recurring pre-processing step to produce the various results reported in the paper concerns the normalization of URLs. We have normalized URLs consistently by removing their `www`, `http://`, `https://` prefixes, as well as their trailing "forwarding slash" character `/`, if any. In the special case of the URL referencing an `index.html` Web page, the `index.html` string is stripped from the URL before the other normalizations are applied.

3.3 Mapping `OpenWeb-qrels` to ClueWeb12

We identify documents that are included in `OpenWeb-qrels` and exist in ClueWeb12 collection (these documents are subsets 1 and 2 in Fig. 1). We achieve this by obtaining the URLs from the `OpenWeb-qrels`, then, we search for these URLs in the ClueWeb12 collection. To check the matching between qrels URLs and ClueWeb12 document URLs, both were normalized as described in Sect. 3.2. We shared this subset with the CS track community.[1] In Table 1 we summarize the statistics derived from the Open Web and ClueWeb12 relevance assessments in 2013 and 2014. We observe that the qrels do contain duplicates, that are not necessarily assessed the same. The differences can be explained by the CS track evaluation setup, where the top-5 suggestions per topic provided by each submitted run were judged individually (Dean-Hall et al. 2013, 2014).

We have separated these documents into two subsets: subsets 1 and 2 from Fig. 1. First, the subset 1 represents documents that were judged as Open Web documents and that have a matching ClueWeb12 document, however they do not exist in ClueWeb12 relevance assessments; we refer to this subset as (`OpenWeb-qrels-urls-in-ClueWeb12`). We consider these documents as additional judgments that can be used to expand the ClueWeb12 relevance assessments. The second subset consists of documents that overlap between Open Web and ClueWeb12 relevance assessments – that is, they were judged twice –, we refer to this subset as `ClueWeb12-qrels` (`qrels-overlap`).

3.4 Expanding `ClueWeb12-qrels`

We expand the ClueWeb12 relevance assessments by modifying the provided qrels files mentioned in Sect. 3.1. We achieve this by replacing in the qrels the URLs with their ClueWeb12 ids (if they exist) based on the subset identified in Sect. 3.3.

3.5 Mapping URLs from Open Web runs to the ClueWeb12 documents URLs

In this section, we describe how we map all URLs found by Open Web systems (in the submitted runs) to their ClueWeb12 ids. We need this mapping to evaluate Open Web systems on ClueWeb12 collection. In order to achieve this, we obtain the URLs from the Open Web runs. Then, we search for these URLs in ClueWeb12 collection by matching the normalized URLs against documents normalized URLs in ClueWeb12 collection. The result of this process is a mapping between URLs in the Open Web runs and their corresponding ClueWeb12 ids (`OpenWeb-runs-urls-in-ClueWeb12`). Table 2 presents a summary about the Open Web URLs and the number of URLs found in ClueWeb12

[1] https://sites.google.com/site/treccontext/trec-2014/open-web-to-clueweb12-mapping.

Table 1 Summary of judged documents form the Open Web and the ClueWeb12 collection

	2014			2013		
	total	Unique	In ClueWeb12	Total	Unique	In ClueWeb12
Open Web runs	35,697	8,442	1,892	28,849	10,349	2,894
ClueWeb12 runs	8909	2674	All	7329	3098	All

The `total` column shows the total number of judged documents, while the `unique` presents the number of unique documents

Table 2 URLs obtained from Open Web runs

	2014	2013
Total number of URls	15,339,209	35,949,067
Unique number of URLs	75,719	102,649
Found in ClueWeb12	10,014	26,248

collection. As we see in the table, for CS 2013 around 25.6 % of URLs have a matching document in ClueWeb12, while for CS 2014 only 13.2 % exist in ClueWeb12 collection.

4 Comparing Open Web and Closed Web relevance assessments

In this section we present an analysis to compare Open Web and ClueWeb12 relevance assessments. In Bellogín et al. (2014), we already showed that Open Web runs tend to receive better judgments than ClueWeb12 results, based on analyzing the CS 2013 results. We repeat here the same experiment in order to investigate whether such tendency is still present in the 2014 test collection. We first compare Open Web and ClueWeb12 in general (the distribution of relevance assessments of documents returned by Open Web systems versus those documents returned by ClueWeb12 systems). Next, we focus on the documents in the overlap of the relevance assessments between Open Web systems and ClueWeb12 systems.

4.1 Fair comparison of test collections

In this section, we study RQ1: Do relevance assessments of Open Web URLs differ (significantly) from relevance assessments of ClueWeb12 documents? We analyze the distribution of profile judgments of documents returned by Open Web and ClueWeb12 runs. In our analysis, we leave out the user, context, and system variables, and compare the judgments given to documents from the Open Web against those from ClueWeb12. In Fig. 2, we observe that the Open Web histogram is slightly skewed towards the positive, relevant judgments. Even though we are not interested in comparing the actual frequencies. This would not be fair, mainly because there were many more Open Web submissions than ClueWeb12 ones. Specifically, in TREC CS 2013, 27 runs submitted URLs from the Open Web, and only 7 runs used ClueWeb12 documents. However, it is still relevant to see the relative frequency of -2's or -1's (document could not load at assessing time), used in CS 2013 and CS 2014, respectively. 4's (strongly interested) in each dataset: this is an

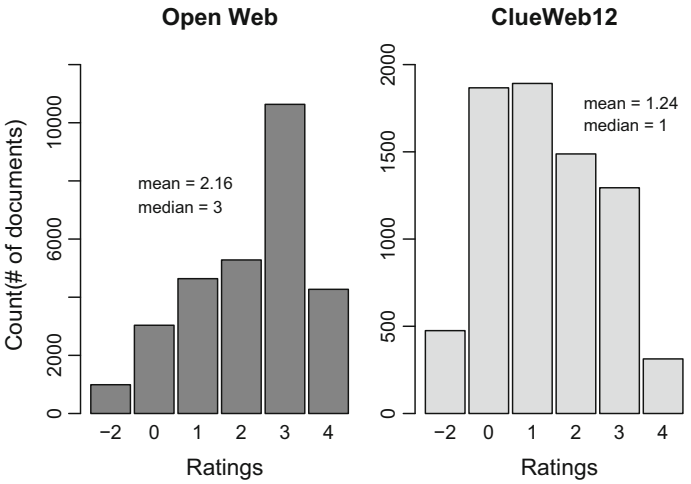

Fig. 2 Judgments (document relevance) histogram of documents from Open Web (*left*) and from ClueWeb12 (*right*) **CS 2013**

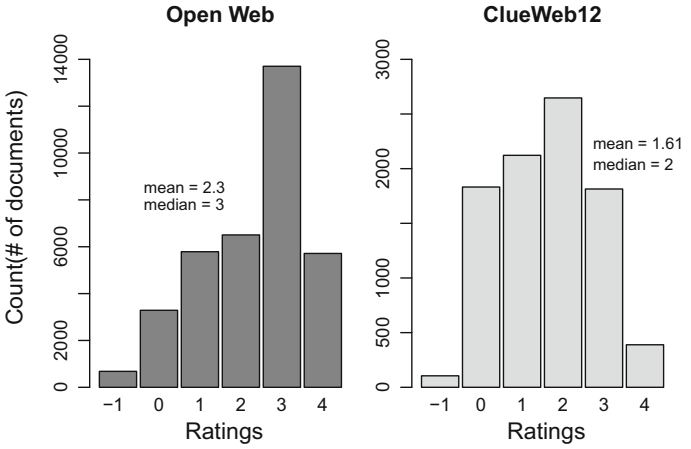

Fig. 3 Judgments (document relevance) histogram of documents from Open Web runs (*left*) and ClueWeb12 runs (*right*) **CS 2014**

important difference which will impact the performance of the systems using ClueWeb12 documents.

Figure 3 shows the same analyses based on 2014 test collection. In that year of the track, 25 runs submitted URLs from the Open Web, and only 6 runs used ClueWeb12 documents. We find that the judgments of documents from Open Web are skewed towards the positive (relevant) side, while judgments of documents from ClueWeb12 are—again—skewed towards the negative (not relevant) part of the rating scale, similar to the findings on the 2013 test collection.

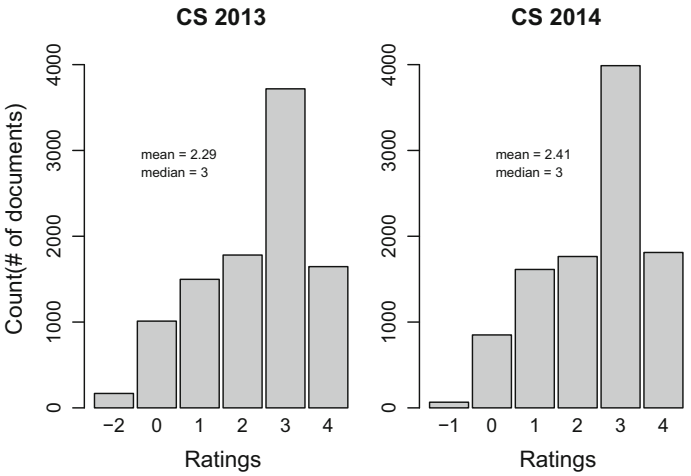

Fig. 4 Judgments histogram of documents from Open Web qrels which exist in ClueWeb12 collection for CS 2013 (*left*) and CS 2014 (*right*)

4.2 Difference in evaluation of identical documents from Open Web and ClueWeb12

In Sect. 3.3, we identified two subsets of overlap between Open Web and ClueWeb12 results: first, `OpenWeb-qrels-urls-in-ClueWeb12` that maps URLs from `OpenWeb-qrels` to ClueWeb12 collection, and `qrels-overlap` that contains documents that exist in both `OpenWeb-qrels` and `ClueWeb12-qrels`. Based on these datasets, we investigate RQ2: Can we identify an overlap between Open Web systems and ClueWeb12 systems in terms of documents suggested by both?, how are those documents in the overlap judged?

Figure 4 shows the distribution of relevance assessments of documents in `OpenWeb-qrels-urls-in-ClueWeb12` for both CS 2013 and CS 2014. We observe that the distribution of judgments of these documents have a similar behavior as the whole Open Web judged documents. More precisely, we observe that the distribution is skewed towards the positive ratings when we look at 3 and 4 ratings for 2013 and 2014 datasets.

Now we focus on the `qrels-overlap` subset which contains documents shared by both `OpenWeb-qrels` and `ClueWeb12-qrels`. Our aim here is to detect any bias towards any of the document collections (the Open Web vs. ClueWeb12) based on the available sample of the judgments. In principle, the relevance judgments should be the same for the two sources, since in each situation the same document was retrieved by different systems for exactly the same user and context, the only difference being how the document was identified (as a URL or as a ClueWeb12 id). Figures 5 and 6 show how documents in the `qrels-overlap` were judged as Open Web URLs and as ClueWeb12 documents in CS 2013 and CS 2014 test collections, respectively. We find that the documents in the overlap were judged differently. The judgments distributions of the documents shared by both `OpenWeb-qrels` and `ClueWeb12-qrels` suggest that there is a bias towards `OpenWeb-qrels` and this bias is consistent in 2013 and 2014 data. For CS 2013, part of the differences in judgments was attributed to a different rendering of the

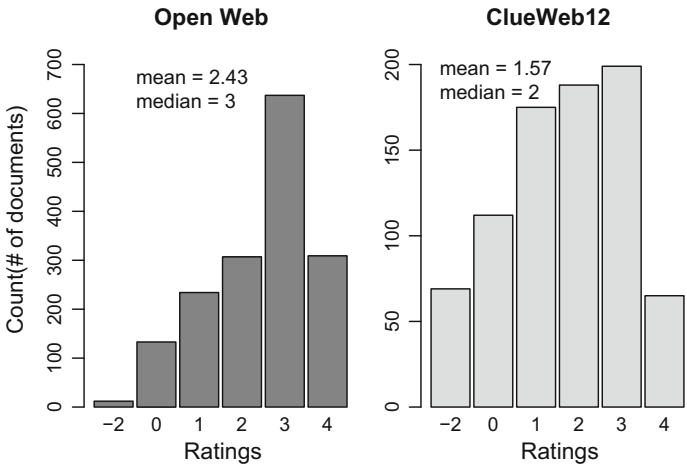

Fig. 5 Judgments histogram of documents that exist in both Open Web qrels and ClueWeb12 qrels. Figure on the (*left*) shows how these documents were judged as Open Web URLs, while the figure on the (*right*) shows how the same documents were judged as ClueWeb12 documents **CS 2013**

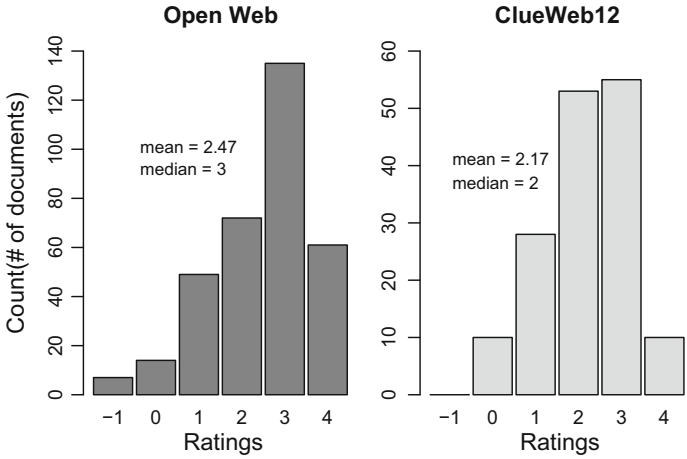

Fig. 6 Judgments histogram of documents that exist in both Open Web qrels and ClueWeb12 qrels. Figure on the (*left*) shows how these documents were judged as Open Web URLs, while the figure on the (*right*) shows how the same documents were judged as ClueWeb12 documents **CS 2014**

document for each source.[2] Assessors are influenced by several conditions, one of them is the visual aspect of the interface, but also the response time, the order of examination, the familiarity with the interface, etc. (Kelly 2009). Therefore, it is important that these details are kept as stable as possible when different datasets are evaluated at the same time. It is also interesting to note that the number of ClueWeb12 documents that could not load is

[2] Confirmed via email with the organisers for 2013 dataset.

Table 3 Performance of Open Web systems on Open Web data vs. their performance on ClueWeb12 data

	P@5			% ClueWeb12	MRR			TBG		
	Original	Replaced	%	In top-5	Original	Replaced	%	Original	Replaced	%
UDInfoCS1	0.5094	0.1444	−71.7	3.6	0.6320	0.2375	−62.4	2.4474	0.2273	−90.7
UDInfoCS2	0.4969	0.1379	−72.2	6.6	0.6300	0.2448	−61.1	2.4310	0.2993	−87.7
SimpleScore	0.4332	0.1063	−75.5	3.1	0.5871	0.1974	−66.4	1.8374	0.1970	−−89.3
ComplexScore	0.4152	0.1000	−75.9	3.5	0.5777	0.1500	−74.0	1.8226	0.1900	−89.6
DuTH_B	0.4090	0.1509	−63.1	24.9	0.5955	0.2999	−49.6	1.8508	0.4280	−76.9
1	0.3857	0.1688	−56.2	35.2	0.5588	0.3371	−39.7	1.5329	0.5450	−64.4
2	0.3731	0.1696	−54.5	32.6	0.5785	0.3144	−45.7	1.5843	0.5290	−66.6
udel_run_D	0.3659	0.1898	−48.1	39.8	0.5544	0.4182	−24.6	1.5243	0.7448	−51.1
isirun	0.3650	0.1568	−57.0	38.0	0.5165	0.2862	−44.6	1.6278	0.4265	−73.8
udel_run_SD	0.3354	0.1238	−63.1	25.4	0.5061	0.3131	−38.1	1.2882	0.4463	−65.4
york13cr2	0.3309	0.1198	−63.8	36.9	0.4637	0.2633	−43.2	1.3483	0.3762	−72.1
DuTH_A	0.3283	0.0991	−69.8	15.2	0.4836	0.2009	−58.5	1.3109	0.2287	−82.6
york13cr1	0.3274	0.1159	−64.6	36.9	0.4743	0.2667	−43.8	1.2970	0.3943	−69.6
UAmsTF30WU	0.3121	0.1182	−62.1	22.0	0.4803	0.2459	−48.8	1.1905	0.3626	−69.5
IRIT.OpenWeb	0.3112	0.1149	−63.1	25.0	0.4915	0.2492	−49.3	1.4638	0.4248	−71.0
CIRG_IRDISCOA	0.3013	0.1006	−66.6	23.0	0.4567	0.2010	−56.0	1.1681	0.2303	−80.3
CIRG_IRDISCOB	0.2906	0.1074	−63.0	24.3	0.4212	0.2042	−51.5	1.1183	0.2550	−77.2
uncsils_param	0.2780	No match	NaN	No match	0.4271	No match	NaN	1.3115	No match	NaN
uogTrCFP	0.2753	0.1000	−63.7	1.0	0.4327	0.3700	−14.5	1.3568	0.3784	−72.1
ming_1	0.2601	No match	NaN	No match	0.3816	No match	NaN	1.0495	No match	NaN
uncsils_base	0.2565	No match	NaN	No match	0.4136	No match	NaN	1.1374	No match	NaN
ming_2	0.2493	No match	NaN	No match	0.3473	No match	NaN	0.9673	No match	NaN
uogTrCFX	0.2332	0.0500	−78.6	0.8	0.4022	0.1562	−61.2	1.0894	0.1542	−85.8
run01	0.1650	0.1722	4.4	100.0	0.2994	0.3194	6.7	0.7359	0.7735	5.1

Table 3 continued

	P@5			% ClueWeb12	MRR			TBG		
	Original	Replaced	%	In top-5	Original	Replaced	%	Original	Replaced	%
BaselineB	0.1417	n/a	n/a	100.0	0.2452	n/a	n/a	0.4797	n/a	n/a
BaselineA	0.1372	0.0841	−38.7	50.7	0.2316	0.1450	−37.4	0.5234	0.3001	−42.7
BOW_V17	0.1022	n/a	n/a	100.0	0.1877	n/a	n/a	0.3389	n/a	n/a
BOW_V18	0.1004	n/a	n/a	100.0	0.1971	n/a	n/a	0.3514	n/a	n/a
IRIT.ClueWeb	0.0798	n/a	n/a	100.0	0.1346	n/a	n/a	0.3279	n/a	n/a
RUN1	0.0628	n/a	n/a	100.0	0.1265	n/a	n/a	0.2069	n/a	n/a
csui02	0.0565	No match	NaN	No match	0.1200	No match	NaN	0.1785	No match	NaN
csui01	0.0565	No match	NaN	No match	0.1016	No match	NaN	0.1765	No match	NaN
RUN2	0.0565	n/a	n/a	100.0	0.1223	n/a	n/a	0.2020	n/a	n/a
IBCosTop1	0.0448	n/a	n/a	100.0	0.0569	n/a	n/a	0.1029	n/a	n/a

Under each metric we present three values: original, replaced, and the relative improvement in effectiveness. The column named `original` presents the performance of submitted runs using the original qrels as provided by the organizers, whereas the column `replaced` shows the performance of modified runs (replacing URLs with their match ClueWeb12 id and removing URLs with no match) using the expanded qrels. The % of ClueWeb12 documents in top-5 column presents the percentage of ClueWeb12 documents in the top-5 after replacing the URLs with their match ClueWeb12 ids while preserving the ranks. The ClueWeb12 systems (underlined) are included to show how they perform in comparison with Open Web systems evaluated on ClueWeb12 data. For ClueWeb12 systems no replacement has been applied, denoted by *n/a* under `replaced` and % of improvement **CS 2013 systems**

Table 4 Performance of Open Web systems on Open Web data versus their performance on ClueWeb12 data

	P@5			% ClueWeb12	MRR			TBG		
	Original	Replaced	%	In top-5	Original	Replaced	%	Original	Replaced	%
UDInfoCS2014_2	0.5585	0.2275	−59.3	22.0	0.7482	0.5506	−26.4	2.7021	0.8604	−68.2
RAMARUN2	0.5017	no match	NaN	no match	0.6846	no match	NaN	2.3718	no match	NaN
BJUTa	0.5010	0.1781	−64.5	28.3	0.6677	0.3290	−50.7	2.2209	0.4752	−78.6
BJUTb	0.4983	0.1805	−63.8	29.5	0.6626	0.3319	−49.9	2.1949	0.4955	−77.4
uogTrBunSumF	0.4943	0.0769	−84.4	0.9	0.6704	0.1628	−75.7	2.1526	0.1690	−92.1
RUN1	0.4930	no match	NaN	no match	0.6646	no match	NaN	2.2866	no match	NaN
webis_1	0.4823	0.1768	−63.3	25.8	0.6479	0.3600	−44.4	2.1700	0.6195	−71.5
simpleScoreImp	0.4602	0.1283	−72.1	4.2	0.6408	0.2632	−58.9	1.9795	0.2595	−86.9
webis_2	0.4569	0.1768	−61.3	25.8	0.5980	0.3600	−39.8	2.1008	0.6195	−70.5
simpleScore	0.4538	0.1147	−74.7	5.4	0.6394	0.2368	−63.0	1.9804	0.2477	−87.5
run_FDwD	0.4348	0.1581	−63.6	30.6	0.5916	0.3390	−42.7	1.7684	0.5429	−69.3
waterlooB	0.4308	0.0932	−78.4	11.0	0.6244	0.2263	−63.8	1.8379	0.2686	−85.4
waterlooA	0.4167	0.0951	−77.2	12.0	0.6021	0.2280	−62.1	1.7364	0.2587	−85.1
UDInfoCS2014_1	0.4080	0.1278	−68.7	17.7	0.5559	0.2629	−52.7	1.6435	0.3185	−80.6
dixlticmu	0.3980	0.1735	−56.4	29.0	0.5366	0.3210	−40.2	1.5110	0.5240	−65.3
uogTrCsLtrF	0.3906	0.0667	−82.9	0.9	0.5185	0.0903	−82.6	1.9164	0.1285	−93.3
run_DwD	0.3177	0.1177	−63.0	25.8	0.3766	0.1718	−−54.4	0.9684	0.1721	−82.2
tueNet	0.2261	0.0258	−88.6	2.6	0.3820	0.0452	−88.2	0.9224	0.0825	−91.1
choqrun	0.2254	0.1145	−49.2	33.2	0.3412	0.2223	−34.8	0.7372	0.3314	−55.0
tueRforest	0.2227	0.0258	−88.4	2.6	0.3604	0.0452	−87.5	0.9293	0.0825	−91.1
cat	0.2087	0.0954	−54.3	46.4	0.3496	0.1807	−48.3	0.6120	0.2544	−58.4
BUPT_PRIS_01	0.1452	0.1000	−31.1	16.2	0.4475	0.2982	−33.4	0.7453	0.3564	−52.2
CWI_CW12.MapWeb	0.1445	n/a	n/a	100.0	0.2307	n/a	n/a	0.6078	n/a	n/a
BUPT_PRIS_02	0.1425	0.0966	−32.2	17.4	0.3467	0.2080	−40.0	0.6601	0.2479	−62.4

Table 4 continued

	P@5			% ClueWeb12	MRR			TBG		
	Original	Replaced	%	In top-5	Original	Replaced	%	Original	Replaced	%
gw1	0.1024	0.0386	−62.3	24.4	0.1694	0.0800	−52.8	0.3646	0.1150	−68.5
Model1	0.0903	n/a	n/a	100.0	0.1979	n/a	n/a	0.3411	n/a	n/a
lda	0.0843	0.0457	−45.8	30.4	0.1564	0.0928	−40.7	0.2461	0.1159	−52.9
Model0	0.0582	n/a	n/a	100.0	0.1023	n/a	n/a	0.1994	n/a	n/a
runA	0.0482	n/a	n/a	100.0	0.0856	n/a	n/a	0.1647	n/a	n/a
CWI_CW12_Full	0.0468	n/a	n/a	100.0	0.0767	n/a	n/a	0.1256	n/a	n/a
runB	0.0254	n/a	n/a	100.0	0.0552	n/a	n/a	0.0614	n/a	n/a

Notation as in Table 3 **CS 2014 systems**

higher in CS 2013 (−2) compared to CS 2014 (−1), probably due to the efforts of the organizers in the latter edition of running a fairer evaluation (Dean-Hall and Clarke 2015).

5 Reproducibility of Open Web systems

In this section, we investigate RQ3: How many of the documents returned by Open Web systems can be found in the ClueWeb12 collection as a whole? The goal of this analysis is to show how many of the results obtained by Open Web systems can be reproduced based on ClueWeb12 collection. In Sect. 3.5, we presented the number of URLs found by Open Web systems and have a matching documents in ClueWeb12 collection. Precisely in Table 2, we showed that for CS 2013 26,248 out of 102,649 URLs have a matching with ClueWeb12 documents (25.6 %), while for CS 2014 10,014 out of the 75,719 URLs (13.2 %) have ClueWeb12 documents match. In this section, we evaluate Open Web systems on ClueWeb12 data. Analyzing the impact of ClueWeb12 documents on the effectiveness of Open Web systems requires the following. First, we need to modify the Open Web runs using the `OpenWeb-runs-urls-in-ClueWeb12` dataset which has the mapping between Open Web URLs to ClueWeb12 ids. Second—for evaluation completeness—we use the expanded `ClueWeb12-qrels` which was generated based on the `OpenWeb-qrels` URLs found in the ClueWeb12 collection (`OpenWeb-qrels-urls-in-ClueWeb12` subset described in Sect. 3.4).

While modifying the Open Web runs, if the suggested URL has a matching in ClueWeb12, we replace the URL with its corresponding ClueWeb12 id. If the URL has no match, then we skip the line containing that URL. We hence change the ranking after skipping those URLs. We present the effectiveness of original Open Web runs and the effectiveness of modified runs (replacing URLs with ClueWeb12 ids), and we show the percentage of relative improvement in effectiveness of Open Web systems (on Open Web data vs ClueWeb12). Nonetheless, replacing the URLs with their matching ClueWeb12 ids and pushing up their ranks by removing the URLs which have no ClueWeb12 match will overestimate the performance and not show the corresponding impact on performance of those ClueWeb12 documents if the ranking was preserved. To give an insight about the importance of ClueWeb12 documents compared to the Open Web URLs that have no ClueWeb12 match, we also include the percentage of ClueWeb12 documents occurring in the top-5. To achieve this, when modifying the Open Web run, we replace the URLs with their match ClueWeb12 ids, and keep the URLs as they are if they do not have a match. Then, for each topic, we compute the percentage of ClueWeb12 documents in the top-5. The score for each run is the mean across all topics.

For CS 2013 systems (see Table 3) and for CS 2014 systems (see Table 4), we report the effectiveness of Open Web systems using their original run files as submitted to the track based on the original qrels (column named original). We report their effectiveness using the modified run files based on the expanded qrels as described above. Finally, we report the percentage of ClueWeb12 documents in the top-5 as described above (how many ClueWeb12 documents remain in the top-5 while preserving the URLs with no match).

In both tables, we observe the following: First, for some Open Web systems we were not able to reproduce their results based on ClueWeb12 data, mainly because some systems have no matching at all with ClueWeb12 collection. For systems that rely on the Yelp API to obtain candidate documents, we could not find any document whose host is Yelp in

ClueWeb12 collection, this is due to very strict indexing rules.[3] Second, we observe that the performance of Open Web systems decreases. However, this reduction in performance varies between systems, suggesting that pushing ClueWeb12 documents up in the submitted rankings by removing URLs with no ClueWeb12 id match has a different effect on each Open Web system. Third, some of top performing Open Web systems are performing very well when constrained to the ClueWeb12 collection. For example, in the CS 2014 edition, UDInfoCS2014_2, BJUTa, and BJUTb systems even perform better than ClueWeb12 systems (underlined systems in the table). Fourth, in terms of how representative ClueWeb12 documents in the top-5, the percentage of ClueWeb12 documents in the top-5 ranges from 1 to 46 % (19 % the mean across all Open Web systems, median = 22 %) for CS 2014 systems. For CS 2013, it ranges from 1 to 51 % (22 % the mean across all Open Web systems, median = 25 %).

6 Selection method for identifying a representative sample of the Open Web from ClueWeb12

In this section we study RQ4: Can we identify a representative sample from the ClueWeb12 collection for the CS track by applying the tourist domain knowledge obtained from the Open Web? We use the tourist domain knowledge available on the Open Web to annotate documents in ClueWeb12 collection. The aim is not only to obtain reproducible results based on ClueWeb12 collection, but also to obtain a representative sample of the Open Web.

6.1 Selection methods of candidate documents from ClueWeb12

We formulate the problem of candidate selection from ClueWeb12 as follows. We have a set of contexts (locations) $\mathtt{C}$—which correspond to US cities—provided by the organizers of the `CS` track. For each context $c \in \mathtt{C}$, we generate a set of suggestions S_c from the ClueWeb12 collection, which are expected to be located in that context.

We define four filters for selecting documents from ClueWeb12 collection, each of them will generate a sub-collection. The first filter is a straightforward filter based on the content of the document. The remaining three filters use knowledge derived from the Open Web about sites existing in ClueWeb12 that provide touristic information. We will show empirically that the additional information acquired from tourist APIs provides the evidence needed to generate high quality contextual suggestions. While our results still depend upon information that is external to the collection, we only need to annotate ClueWeb12 with the tourist domain knowledge identified to achieve reproducible research results. We describe the filters in more detail in the following sections.

6.1.1 Geographically filtered sub-collection

Our main hypothesis in this approach is that a good suggestion (a venue) will contain its location correctly mentioned in its textual content. Therefore, we implemented a content-based geographical filter `geo_filter` that selects documents mentioning a specific context with the format `(City, ST)`, ignoring those mentioning the city with different states or those matching multiple contexts. With this selection method we aim to ensure that the

[3] See http://yelp.com/robots.txt.

specific target context is mentioned in the filtered documents (hence, being geographically relevant documents). We will still miss relevant documents, for example due to misspellings or because they mention more than one city at the same web page.

In the simplest instantiation of our model, the probability of any document in ClueWeb12 to be included in the **GeographicFiltered** sub-collection is assigned to 0 or 1 depending on whether it passes the geo_filter:

$$P(s) = \begin{cases} 1, & \text{if(s) passesgeo\_filter} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Approximately 9 million documents (8,883,068) from the ClueWeb12 collection pass the geo_filter. The resulting set of candidates forms the first sub-collection, referred to as **GeographicFiltered**.

6.1.2 Domain-oriented filter

The first type of domain knowledge depends on a list of `hosts` that are well-known to provide tourist information, and are publicly available (and have been crawled during the construction of ClueWeb12). We manually selected the set of hosts $\mathcal{H} := \{$`yelp, xpedia, tripadvisor, wikitravel, zagat, orbitz,` and `travel.yahoo`$\}$, some of these host APIs were used by the Open Web systems. We consider these hosts as a domain filter to select suggestions from ClueWeb12 collection. Again, the probability of a document in ClueWeb12 to be a candidate suggestion is either 0 or 1 depending only on its host. We define the probability $P(s)$ as:

$$P(s) = \begin{cases} 1, & \text{if host(s)} \in \mathcal{H} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

We refer to the set of documents that pass the domain filter defined in Eq. (2) as `TouristSites`.

We assume pages about tourist information also have links to other interesting related pages, acknowledging the fact that pages on the same topic are connected to each other (Davison 2000). In order to maximize the extracted number of documents from the tourist domain we also consider the outlinks of documents from touristic sites. For each suggestion $s \in$ `TouristSites`, we extract its outlinks `outlinks(s)` and combine all of them together in a set $\mathcal{O}$; including links between documents from two different hosts (`external links`) as well as links between pages from the same host (`internal links`). Notice that some of the outlinks may also be part of the `TouristSites` set, in particular whenever they satisfy Eq. (2). Next, we extract any document from ClueWeb12 whose normalized URL matches one of the outlinks in $\mathcal{O}$. The probability of document s to be selected in this case is defined as:

$$P(s) = \begin{cases} 1, & \text{if URL(s)} \in \mathcal{O} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

The set of candidate suggestions that pass this filter is called `TouristSitesOutlinks`.

Table 5 Number of documents passing each filter. Documents that pass the first three filters represent the **TouristFiltered** sub-collection, whereas the **GeographicFiltered** sub-collection is composed by those documents passing the geo_filter

Part	Number of documents
`TouristSites`	175,260
`TouristSitesOutlinks`	46,801
`Attractions`	102,313
TouristFiltered sub-collection	324,374
GeographicFiltered	8,775,711

6.1.3 Attraction-oriented filter

The previously described selection method relies on a manually selected list of sites to generate the set of candidate suggestions. We will now consider a different type of domain knowledge, by leveraging the information available on the Foursquare API.[4] For each context c, we obtain a set of URLs by querying Foursquare API. If the document's URL is not returned by Foursquare (we are not interested in the page describing that venue `inside` Foursquare, but its corresponding webpage), we use the combination of document name and context to issue a query to the Google search API e.g., "`Gannon University Erie, PA`" for name `Gannon University` and context `Erie, PA`. Extracting the hosts of the URLs obtained results in a set of 1, 454 unique hosts. We then select all web pages in ClueWeb12 from these hosts as the candidate suggestions, with its probability defined in the same way as in Eq. 2.

The set of documents that pass the host filter is referred to by `Attractions`.

Together, the three subsets of candidate suggestions `TouristSites`, `TouristSitesOutlinks` and `Attractions` form our second ClueWeb12 sub-collection that we refer to as **TouristFiltered**.

$$\textbf{TouristFiltered} := \texttt{TouristSites} \cup \texttt{TouristSitesOutlinks} \cup \texttt{Attractions}$$

Table 5 shows the number of documents found by each filter.

6.2 Impact of domain knowledge filters

Our contribution to the CS track included the following two runs: a first one based on the **GeographicFiltered** sub-collection, and a second one based on the **TouristFiltered** sub-collection. We have found that the run based on **TouristFiltered** sub-collection is significantly better than the one based on **GeographicFiltered** sub-collection in every evaluation metric (see Table 6). However, a more discriminative analysis should be done to properly estimate the impact of the tourist domain knowledge filters used to generate the **TouristFiltered** sub-collection, for this, we shall evaluate the performance of the different sub-collections generated by each of the domain knowledge filters.

Recall that assessments are made considering geographical and profile relevance independently from each other. The latter one is further assessed as relevant based on the document or on the description provided by the method. Considering this information, we

[4] https://developer.foursquare.com/docs/venues/search.

Table 6 Performance of the run based on **GeographicFiltered** sub-collection and the run based on **TouristFiltered** sub-collection

Metric	**GeographicFiltered**	**TouristFiltered**
P@5	0.0431	0.1374
MRR	0.0763	0.2305
TBG	0.1234	0.5953

Table 7 Effect of each part of the **TouristFiltered** sub-collection on performance

Metrics	`TouristSites`	∪ `TouristSitesOutlinks`		∪ `Attractions`		`Attractions`
P@5_all	0.0392	0.0518	32.1 %	0.1374	165.3 %	0.1057
P@5_desc-doc	0.0758	0.1004	32.5 %	0.2222	121.3 %	0.1562
P@5_desc	0.0917	0.1200	30.9 %	0.2788	132.3 %	0.1973
P@5_doc	0.1008	0.1310	30.0 %	0.2949	125.1 %	0.2101
P@5_geo	0.2067	0.2659	28.6 %	0.4808	80.8 %	0.4667
MRR_all	0.1378	0.1715	24.5 %	0.2305	34.4 %	0.1834
MRR_desc-doc	0.2213	0.2738	23.7 %	0.3630	32.6 %	0.2860
MRR_desc	0.2616	0.3133	19.8 %	0.4395	40.3 %	0.3674
MRR_doc	0.2817	0.3463	22.9 %	0.4718	36.2 %	0.3776
MRR_geo	0.5342	0.5865	9.8 %	0.6627	13.0 %	0.6132
TBG	0.2180	0.2705	24.1 %	0.5953	120.1 %	0.5138
TBG_doc	0.2305	0.2860	24.1 %	0.6379	123.0 %	0.5503

Union symbol (∪) represents adding suggestions from the sub-collection or filter presented in the previous column. The percentage shows the relative improvement in effectiveness due to filter

recomputed the evaluation metrics for each topic while taking into account the geographical relevance provided by the assessors, as well as the description and document judgments, both separately and combined (that is, a document that is relevant both based on the description and when the assessor visited its URL). We present in Table 7 the contribution to the relevance dimensions of each of the **TouristFiltered** sub-collection subsets, where each subset was selected based on different domain knowledge filter. The run based on **TouristFiltered** sub-collection contains documents from the three subsets. We modified the run based on **TouristFiltered** sub-collection by start computing effectiveness based only on suggestions from `TouristSites` subset (second column), then we add to them suggestions from `TouristSitesOutlinks`, and finally suggestions from `Attractions`. The main conclusion from this table is that the larger improvement in performance happens after adding the candidates from `Attractions` subset. It is interesting to note that the performance of this part alone (last column) is comparable to that of the whole sub-collection.

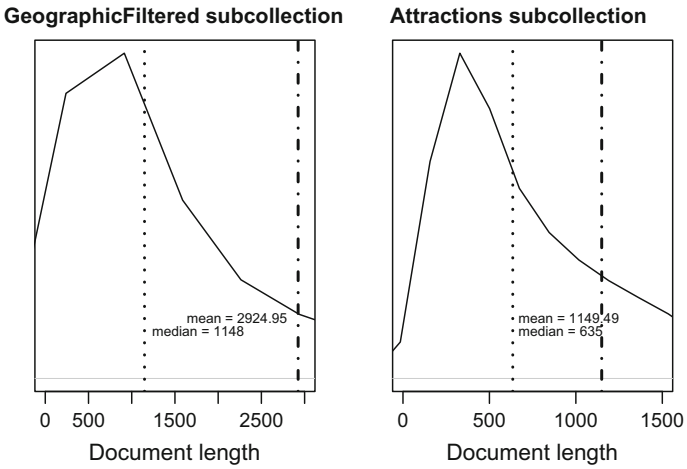

Fig. 7 Distribution of the document length in words for the **GeographicFiltered** (*left*) and `Attractions` (*right*) sub-collections. Note the different range in the X axis

6.3 Discussion

Because systems based on Open Web can still be competitive when the candidate documents are constrained to the ClueWeb12 collection, we have shown that there exist documents in ClueWeb12 that are relevant for the Contextual Suggestion task we address in this paper. However, the candidate selection process is challenging, and the use of external, manually curated tourist services make this task easier, by promoting those relevant documents at the cost of reducing the reproducibility of the whole process.

In this section we aim to understand the candidate selection process and to provide recommendations in order to improve it. With this goal in mind, we study the **GeographicFiltered** and `Attractions` sub-collections by comparing the actual documents that pass the corresponding filters, so that we can analyze these sub-collections from the user perspective (what will the user receive?) instead of from the system perspective (what is the performance of the system?), as we have presented previously in the paper.

A first aspect we consider is the document length (in terms of words included in the processed HTML code), which gives an insight about how much information is contained (and shown to the user) in each sub-collection. We observe from Fig. 7 that documents from the **GeographicFiltered** sub-collection are much larger than those from `Attractions`: their average length is twice as large as those from the other filter. This may suggest that relevant documents in the tourist domain should be short or, at least, they should not present too much information to the user. If this was true, it would be more interesting to retrieve—in the contextual suggestion scenario—home pages such as the main page of a museum or a restaurant, instead of their corresponding `Contact` or `How to access` sub-pages. Because of this, in the future we aim to take information about the URL depth into account when selecting the candidates, since it has been observed in (Kraaij et al. 2002) that the probability of being a home page is inversely related to its URL depth.

Related to the aforementioned aspect, we now want to check manually the content of some pages from each sub-collection. For this analysis we aggregate the judgments

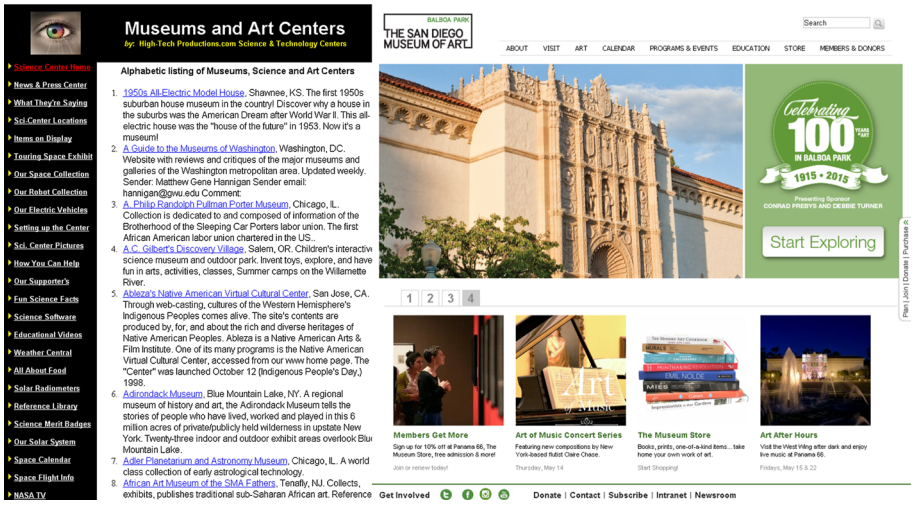

Fig. 8 Screenshots of a document retrieved by the **GeographicFiltered** sub-collection (*left*) and by the `Attractions` sub-collection (*right*). The document in the left (clueweb12-0202wb-00-19744) was rated in average with a value of 1.9, whereas the one in the right (clueweb12-0200tw-67-19011) with a 3

received to the documents submitted in each sub-collection, and then focus on documents with very bad or very good ratings in any of them. Specifically, we have found two candidate documents (presented in Fig. 8) that clearly illustrate the main difference between these two sub-collections, and further corroborates the previous assumption: the **GeographicFiltered** subcollection requires pages where the target city and state are present, which in turn favors pages containing listings of places located in that city, resulting in documents not very informative for an average tourist. On the other hand, the `Attractions` sub-collection tend to retrieve the home page of significant tourist places.

Finally, we have run an automatic classifier on the documents of each sub-collection to gain some insights about whether the content of the pages are actually different. We have used decision trees J48 (Quinlan 1993) as implemented in the Weka library[5]) and tried with different combinations of parameters (stemming, stopwords, confidence value for pruning, number of words to consider, etc.). For the sake of presentation, we have used a very restrictive setting, so that a limited number of leafs are generated. In Fig. 9 we show the branch of the decision tree where *states* appears at least once in the documents; hence, we find that states is the most discriminative term in this situation, and in decreasing order the terms: *internist* and *america*. The classifier represented in this way was trained using a vector representation using the TF-IDF values of the terms in each document, considering only top-20 words with the highest frequency and discarding stopwords and numbers. Additionally, the classifier was parameterized with a confidence threshold of 0.5 and a minimum number of instances per leaf of 500. We conclude from the figure that the `Attractions` sub-collection uses a different vocabulary than the **GeographicFiltered** sub-collection, where terms such as *md*, *st*, or *america* tend to appear with much less frequency. In the future we want to exploit this information to improve the candidate selection process and the corresponding filters.

[5] http://www.cs.waikato.ac.nz/ml/weka/.

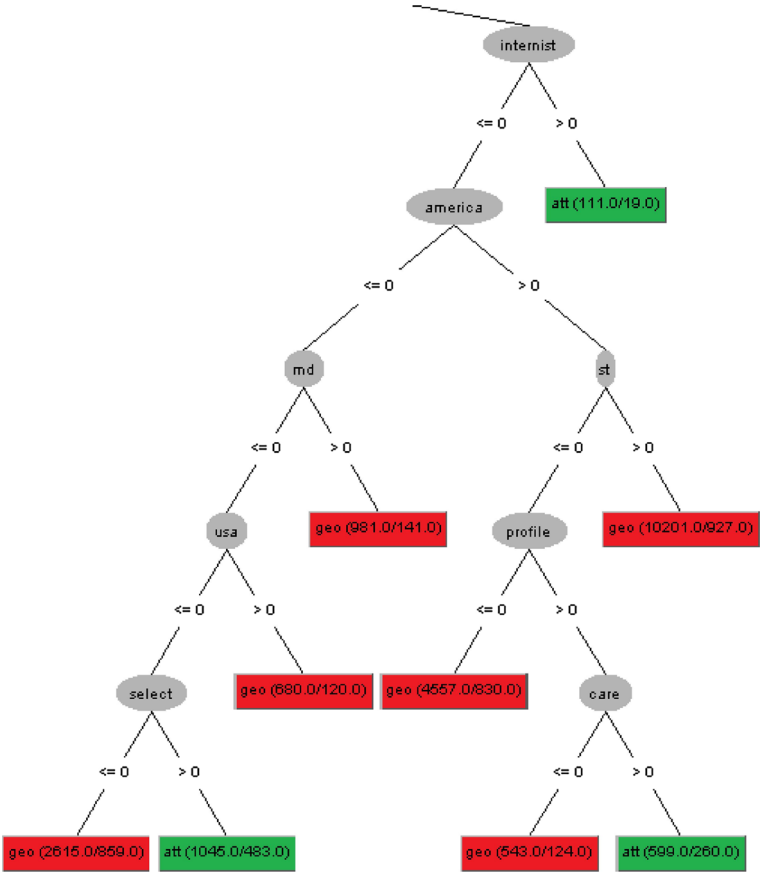

Fig. 9 Visualization of a branch of the J48 decision tree trained using documents from the `Attractions` (att) and **GeographicFiltered** (geo) sub-collections. This branch corresponds to the case where the term *states* appears at least once. In every leaf, the label of the classified instances appears together with the total number of instances reaching that leaf (*first number*) and the number of missclassified instances (hence, the lower this number the better)

7 Conclusions

In this paper we have analyzed and discussed the balance between reproducibility and representativeness when building test collections. We have focused our analysis on the Contextual Suggestion TREC track, where in 2013 and 2014 it was possible to submit runs based on Open Web or based on ClueWeb12, a static version of the web. In both editions of the track, there were more runs based on Open Web compared to those based on ClueWeb12 collection, which seems to go against any reproducibility criteria we may expect from such a competition. The main reason, as we have shown in this paper, for that behavior is that systems based on Open Web perform better than systems based on ClueWeb12 collection in terms of returning more relevant documents.

We have studied such difference in effectiveness from various perspectives. First, the analysis of relevance assessments of 2 years of the Contextual Suggestion track shows that

documents returned by Open Web systems receive better ratings than documents returned by ClueWeb12 systems. More specifically, we have found differences in judgment when looking at identical documents that were returned by both Open Web and ClueWeb12 systems. Second, based on an expanded version of the relevance assessments—considering documents in the overlap of Open Web and ClueWeb12 systems—and on generating ClueWeb12-based runs from Open Web runs, we have investigated the representativeness of ClueWeb12 collection. Although the performance of Open Web systems decreases, we find a representative sample of ClueWeb12 collection in Open Web runs. Third, we proposed an approach for selecting candidate documents from ClueWeb12 collection using the information available on the Open Web. Our results are promising, and evidence that there is still room for improvement by using different and more information available on the Open Web.

For future work, we plan to collect candidate documents from different crawls of the web besides ClueWeb12 collection, such as the `Common Crawl`.[6] Both crawls will complement each other and help to find more representative samples of the web; in this way we could evaluate them by participating in future editions of the Contextual Suggestion track. Another aspect we would like to explore in the future is that of improving the candidate selection filters. We have learnt some features that seem to be frequent in the sub-collection generated from the Open Web. We would like to incorporate that information into our geographical filters, so that better candidate documents are found, and—in principle—lead to better contextual recommendations.

Acknowledgments This research was supported by the Netherlands Organization for Scientific Research (WebART Project, NWO CATCH #640.005.001). Part of this work was supported by the Spanish Ministry of Science and Innovation (TIN2013-47090-C3-2). This work was carried out on the Dutch national e-infrastructure with the support of SURF Foundation.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In F. Ricci, L. Rokach, B. Shapira, F. Ricci, L. Rokach, B. Shapira, P. B. Kantor, & P. B. Kantor (Eds.), *Recommender systems handbook* (pp. 217–253). Boston, MA: Springer. doi:10.1007/978-0-387-85820-3_7. (**chapter 6**).

Allan, J., Croft, W. B., Moffat, A., & Sanderson, M. (2012). Frontiers, challenges, and opportunities for information retrieval: Report from SWIRL 2012 the second strategic workshop on information retrieval in lorne. *SIGIR Forum*, *46*(1), 2–32. doi:10.1145/2215676.2215678.

Bellogín, A., Samar, T., de Vries, A. P., & Said, A. (2014). Challenges on combining open web and dataset evaluation results: The case of the contextual suggestion track. In M. de Rijke, T. Kenter, A. P. de Vries, C. Zhai, F. de Jong, K. Radinsky, & K. Hofmann (Eds.), *Advances in information retrieval-36th European conference on IR research, proceedings, Springer, Lecture Notes in Computer Science* (Vol. 8416, pp. 430–436). Amsterdam, The Netherlands: ECIR 2014. doi:10.1007/978-3-319-06028-6_37.

Davison, B. D. (2000). Topical locality in the web. In *Proceedings of the 23rd annual international conference on research and development in information retrieval (SIGIR 2000)* (pp. 272–279). ACM Press.

[6] https://commoncrawl.org/.

Dean-Hall, A., & Clarke, C. L. A. (2015). The power of contextual suggestion. In A. Hanbury, G. Kazai, A. Rauber, N. Fuhr (Eds.), *Advances in information retrieval-37th European conference on IR research, ECIR 2015, proceedings, Lecture Notes in Computer Science* (Vol. 9022, pp. 352–357), Vienna, Austria, March 29–April 2, 2015. doi:10.1007/978-3-319-16354-3_39.

Dean-Hall, A., Clarke, C. L. A., Kamps, J., Thomas, P., & Voorhees, E. M. (2012). Overview of the TREC 2012 contextual suggestion track. In E. M. Voorhees, L. P. Buckland (Eds.), *Proceedings of the twenty-first text retrieval conference, TREC 2012*, National Institute of Standards and Technology (NIST), Vol Special Publication 500-298, Gaithersburg, Maryland, USA, November 6–9, 2012. http://trec.nist.gov/pubs/trec21/papers/CONTEXTUAL12.overview.pdf.

Dean-Hall, A., Clarke, C. L. A., Kamps, J., & Thomas, P. (2013). Evaluating contextual suggestion. In *Proceedings of the fifth international workshop on evaluating information access (EVIA 2013)*.

Dean-Hall, A., Clarke, C. L. A., Simone, N., Kamps, J., Thomas, P., & Voorhees, E. M. (2013). Overview of the TREC 2013 contextual suggestion track. In E. M. Voorhees (Ed.), *Proceedings of the twenty-second text retrieval conference, TREC 2013*, National Institute of Standards and Technology (NIST), Vol Special Publication 500-302, Gaithersburg, Maryland, USA, November 19–22, 2013. http://trec.nist.gov/pubs/trec22/papers/CONTEXT.OVERVIEW.pdf.

Dean-Hall, A., Clarke, C. L. A., Kamps, J., Thomas, P., Voorhees, E. M. (2014). Overview of the TREC 2014 contextual suggestion track. In E. M. Voorhees, A. Ellis, (Eds.), *Proceedings of the twenty-third text retrieval conference, TREC 2014*, National Institute of Standards and Technology (NIST), vol Special Publication 500-308, Gaithersburg, Maryland, USA, November 19–21, 2014. http://trec.nist.gov/pubs/trec23/papers/overview-context.pdf

Deveaud, R., Albakour, M. D., Macdonald, C., & Ounis, I. (2014). On the importance of venue-dependent features for learning to rank contextual suggestions. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management, CIKM '14* (pp. 1827–1830). New York, NY, USA: ACM. doi:10.1145/2661829.2661956.

Hawking, D., Craswell, N., Thistlewaite, P. B., & Harman, D. (1999). Results and challenges in web search evaluation. *Computer Networks*, *31*(11–16), 1321–1330. doi:10.1016/S1389-1286(99)00024-9.

Hawking, D., Craswell, N., Bailey, P., & Griffiths, K. (2001). Measuring search engine quality. *Information Retrieval*, *4*(1), 33–59. doi:10.1023/A:1011468107287.

Hubert, G., Cabanac, G., Pinel-Sauvagnat, K., Palacio, D., & Sallaberry, C. (2013). IRIT, GeoComp, and LIUPPA at the TREC 2013 contextual suggestion track. In *Proceedings of TREC*.

Kelly, D. (2009). Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, *3*(1–2), 1–224.

Kraaij, W., Westerveld, T., & Hiemstra, D. (2002). The importance of prior probabilities for entry page search. In *SIGIR 2002: Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 27–34). Tampere, Finland: ACM, August 11–15, 2002. doi:10.1145/564376.564383.

Lops, P., de Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In F. Ricci, L. Rokach, B. Shapira, F. Ricci, L. Rokach, B. Shapira, P. B. Kantor, & P. B. Kantor (Eds.), *Recommender systems handbook* (pp. 73–105). Boston, MA: Springer. doi:10.1007/978-0-387-85820-3_3. (**chapter 3**).

Melucci, M. (2012). Contextual search: A computational framework. *Foundations and Trends in Information Retrieval*, *6*(4–5), 257–405.

Quinlan, R. (1993). *C4.5: Programs for machine learning*. San Mateo, CA: Morgan Kaufmann Publishers.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In *CSCW* (pp. 175–186).

Sanderson, M., Turpin, A., Zhang, Y., & Scholer, F. (2012). Differences in effectiveness across sub-collections. In: X. Chen, G. Lebanon, H. Wang, & M. J. Zaki (Eds.), *21st ACM international conference on information and knowledge management, CIKM'12* (pp. 1965–1969) Maui, HI, USA: ACM, October 29–November 02, 2012. doi:10.1145/2396761.2398553.

Santos, R. L. T., Macdonald, C., & Ounis, I. (2011). Effectiveness beyond the first crawl tier. In C. Macdonald, I. Ounis I. Ruthven (Eds.), *Proceedings of the 20th ACM conference on information and knowledge management, CIKM 2011* (pp. 1937–1940). Glasgow, United Kingdom: ACM, October 24–28, 2011. doi:10.1145/2063576.2063859.

Sappelli, M., Verberne, S., & Kraaij, W. (2013). Recommending personalized touristic sights using google places. In: *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval, SIGIR '13* (pp. 781–784). New York, NY, USA: ACM. doi:10.1145/2484028.2484155.

Scholer, F., Turpin, A., & Sanderson, M. (2011). Quantifying test collection quality based on the consistency of relevance judgements. In: W. Ma, J. Nie, R. A. Baeza-Yates, T. Chua, W. B. Croft (Eds.),

Proceeding of the 34th international ACM SIGIR conference on research and development in information retrieval, SIGIR 2011 (pp. 1063–1072) Beijing, China: ACM, July 25–29, 2011. doi:10.1145/2009916.2010057.

Voorhees, E. M., & Buckley, C. (2002). The effect of topic set size on retrieval experiment error. In *Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '02* (pp. 316–323). New York, NY, USA: ACM. doi:10.1145/564376.564432.

Yang, P., & Fang, H. (2012). An Exploration of ranking-based strategy for contextual suggestions. In *Proceedings of TREC*.

Yang, P., & Fang, H. (2013). Opinion-based user profile modeling for contextual suggestions. In *Proceedings of the 2013 conference on the theory of information retrieval ICTIR '13* (pp. 18:80–18:83). New York, NY, USA: ACM. doi:10.1145/2499178.2499191.