Editors
Andreas Wagner
Michael Granitzer
Christian Guetl
Christine Plote
Stefan Voigt

Proceedings

5th International Open Search Symposium

#ossym2023

4-6 October 2023
CERN, Geneva, Switzerland

open search foundation

CERN

ISSN: 2957-4935
ISBN: 978-92-9083-653-7

Impressum

Editors

Michael Granitzer, University Passau, Germany
Christian Güetl , Graz University of Technology, Austria
Christine Plote, Open Search Foundation, Germany
Stefan Voigt, Open Search Foundation, Germany
Andreas Wagner, CERN, Geneva, Switzerland

ISSN: 2957-4935
ISBN: 978-92-9083-653-7
DOI: 10.5281/zenodo.MMM

This report should be cited as:

Proceedings of 5h International Open Search Symposium #ossym2023, CERN, Geneva Switzerland, 4-6 October 2023, M. Granitzer, C. Gütl, C. Plote, S. Voigt, A. Wagner (eds).
http://doi.org/10.5281/zenodo.MMMM

More information

- Conference Website on CERN Indico
  https://indico.cern.ch/e/OSSYM-2023
- Open Search Community at Zenodo
  https://zenodo.org/communities/opensearch/
- Event information at the Open Search Foundation
  https://opensearchfoundation.org/en/events-osf/5th-international-open-search-symposium-ossym2023/

Foreword

We are delighted to present the 2023 volume of the #ossym Proceedings. It documents the many innovative and interdisciplinary contributions of the #ossym23 International Open Search Symposium. Hosted at CERN in Geneva from 4 to 6 October 2023, the conference brought together researchers, technology experts, political representatives, and industry executives for the 5th time to discuss the foundations of human-centred, transparent, and open web search.

The conference offered a variety of formats, from scientific presentations, interactive workshops on the ethics of internet search as well as legal and environmental aspects, to a panel discussion with industry players and talks by representatives of alternative search engines fragFinn, Mojeek and Marginalia. Researchers from the EU Project "OpenWebSearch.eu" gave updates on the project's research in dedicated tracks, covering all aspects from technical challenges of search engine operability, crawling, web page classification and prototyping of Open Web Search applications to aspects of ethics and governance of an Open Web Index.

Not covered in these proceedings, but nevertheless important to mention are the keynote speeches, providing valuable insights into technical, governmental, community-related and ethical aspects:

- Christoph Schumann (LAINON e.V.) emphasised the power of the open AI community.
- Vice President of the European Commission for Values and Transparency, Věra Jourová, outlined the vision of a human-centric internet and stressed the importance of open, transparent Web Search Services for Europe.
- Ricardo Baeza-Yates (Director at the Institute of Experiential AI at Northeastern University) delivered an in-depth keynote on "Bias in Search and Recommender Systems", covering all concepts of biases as well as methods to tackle them.
- Angella Ndaka, head of the The Centre for Africa Epistemic Justice and researcher at the University of Otago, presented insights on how tech-driven businesses change African societies and can endanger democratic structures.

All in all, it was once again a very stimulating meeting with a lively exchange between many different people from many different disciplines. We would like to thank all authors, presenters, panellists and keynote speakers for sharing their scientific work, ideas, expertise and thoughts with us at #ossym23 and in these proceedings. A special thank you goes to CERN as local host and co-organiser of #ossym2023 for their great preparation and tireless commitment.

The conference is an exemplary demonstration of how multifaceted the vibrant Open Web Search community approaches the topic and explores it from a wide range of disciplines and angles. Every #ossym conference takes the Open Web Search initiative a big step further year after year.

In this spirit: We look forward to the next conference – #ossym24 in Munich!

Andreas Wagner, Michael Granitzer, Christian Gütl, Christine Plote and Stefan Voigt

Symposium Organisation

Programme Committee

Prof. Dr. Wolf-Tilo Balke, L3S Research Center, Braunschweig, Germany
Prof. Dr. Alexander Decker, Technische Hochschule Ingolstadt, Germany
Msc. Maria Dimou, CERN, Geneva, Switzerland
Prof. Dr. Kai Erenli, University of Applied Sciences BFI Vienna, Austria
Dr. Christian Geminn, University of Kassel, Germany
Prof. Dr. Michael Granitzer, University Passau, Germany
Prof. Dr. Christian Guetl, Graz University of Technology, Austria
Dr. Christian Geminn, University of Kassel, Germany
Prof. Dr. Mohammed Kaicer, Faculty of Sciences Kenitra, Morocco
Prof. Dr. Dieter Kranzlmüller, Leibniz Supercomputing Centre and LMU, Munich, Germany
Dr. Philipp Mayer-Schlegel, GESIS, Germany
Prof. Dr. Jelena Mitrovic, University of Passau, Germany
Prof. Dr. Engelbert Niehaus, RPTU University of Kaiserslautern-Landau, Germany
Prof. Dr. Monica Palmirani, Università di Bologna, Italy
Prof. Dr. Martin Potthast, Leipzig University, Germany
Prof. Dr. Christin Seifert, University of Marburg, Germany
Dr. Tim Smith, CERN, Geneva, Switzerland
Dr. Stefan Voigt, Open Search Foundation, Germany
Prof. Dr. Arjen P. de Vries, Radboud University, Netherlands
Dr. Andreas Wagner, CERN, Geneva, Switzerland

Conference Chairs

Maari Alanko, CSC – IT Center for Science, Finland
Isabell Claus, thinkers.ai, Austria
Michael Granitzer, University Passau, Germany
Christian Gütl , Graz University of Technology, Austria
Christine Plote, Open Search Foundation, Germany
Stefan Voigt, Open Search Foundation, Germany
Andreas Wagner, CERN, Geneva, Switzerland

Symposium Local Support

Igor Jakovljevic, CERN, Geneva, Switzerland
Francisco Borges Aurindo Barros, CERN, Geneva, Switzerland

Contents

Research Track Information

OWS - OpenWenSearch.EU
HUE - Human centric Search / User Experience
MLR - Machine Learning and Retrieval
OSE - Open Search Ecosystem

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10498306

PRODUCT SPAM ON YOUTUBE: A CASE STUDY

Janek Bevendorff* Matti Wiegmann* Martin Potthast† Benno Stein*

* Bauhaus-Universität Weimar † Leipzig University and ScaDS.AI

Abstract

YouTube videos are a popular medium for online product reviews. They are not only informative and entertaining, but may also be perceived as quite credible under the viewer's impression of a personal product demonstration by an expert. As the world's largest online video platform, YouTube's content is included prominently in the results of most general-purpose web search engines. Consequently, online marketeers are using classic Search Engine Optimization (SEO) techniques also for placing their video content in search engines. Over the years, we have noticed an increasing noise floor of low-quality SEO content in product search results and in this study, we show that this trend has spilled over into videos as well. We examine YouTube video reviews for several thousand products retrieved from three commercial search engines and conduct spam detection experiments based directly on the videos' subtitle transcripts rather than only their metadata and comments. We find that at least a third of the retrieved videos can be regarded as spam or low-quality productions. We are further able to distinguish these spam product reviews accurately from higher-quality videos with a linear classification model trained on transcript data for which the training data labels were obtained using an unsupervised clustering approach.

INTRODUCTION

In an effort to improve their visibility on the web, most commercial websites today use some form of search engine optimization (SEO). A sensible amount of SEO, if applied "with good intentions," may actually improve both the on-site user experience and the effectiveness of the search engine itself. A properly optimized website is more accessible and more efficient to parse, making it easier for the search engine to identify relevant information. On the other hand, SEO is often used with malicious intent to gain undeserved visibility, just pretending to be of value to the user by gaming the search engine's algorithm. This kind of "blackhat" SEO spam has taken over many competitive, low-margin environments, of which a prominent example is product search. In this market, many sellers compete for attention to sell their products online, relying heavily (like most other websites) on SEO to gain new customers. While it has been shown that strong SEO may negatively affect users' perception of a website's expertise [17], a high rank is often more important. Users trust their search engines a fair amount [10,14], so the search engines may lend extra credibility to well-ranked pages.

In addition to search engine optimization (SEO) and marketing (SEM), larger sellers have started offering affiliate programs, which have since become another massive market for which SEO and SEM play just as much of an important role. In online affiliate marketing, the participant (the affiliate) directs traffic from their own website via special product links to a seller (the affiliate partner), earning a commission for each successful referral. The entry barrier for online affiliate marketing is very low, for all that is needed is a website and an agreement with an affiliate partner or partner network. As a result, affiliate marketing has become a popular source of income particularly for bloggers, social media influencers, and product review portals. The low barrier of entry into affiliate marketing has also lured spammers into the market who try to place a bulk of low-effort or even fake product reviews in search engines to harvest affiliate clicks—a trend that is bound to accelerate with the rise of generative AI. Today, SEO-driven affiliate spam is already too pervasive and ubiquitous to contain even for the large search engines [5], which drives users to other venues and communities, such as Reddit and YouTube to satisfy their information needs. Review videos on YouTube in particular are often visually appealing and entertaining, more informative (since the product can be seen in use), and they may appear more trustworthy if the review is presented by an actual (expert) person. Well-produced videos are also more costly to create, and creators have a greater interest in growing their channel with high-quality content. Hence, one might come to believe that video reviews (though often sponsored) are the last source of true hands-on reviews on the web simply due to their (still) higher production costs. In this paper, we show that video product reviews are, in fact, already a popular medium for spammers.

We analyze the YouTube video results for 7,392 product search queries from Startpage (who get their results directly from Google), Bing, and DuckDuckGo. Based on audio transcripts of the videos, we train a simple spam detection model with help of a part-of-speech (POS) n-gram clustering. We find that more than a third of the videos are clearly of very low quality or outright spam. Many videos disguise themselves as product reviews, but are really only product listings compiled from the web, presented like commercials using stock footage and stock music. Some even use automatic text-to-speech synthesis in place of a real human narrator. A basic n-gram model by design cannot assess the real value and factual quality of the video contents. It is, however, able to separate the scripted, commercial-style, and mostly low-quality videos accurately from videos with a real person reviewing products in front of the camera, solely based on the used language patterns. We also find that of the three search engines, Google (as the owner of YouTube) returns by far the most YouTube results, whereas Bing and DuckDuckGo prefer textual reviews.

This work was partially supported by the European Commission under grant agreement GA 101070014 (OpenWebSearch.eu).

https://doi.org/10.5281/zenodo.10498306

RELATED WORK

There is surprisingly little research on (spam) classification of YouTube videos based on subtitle transcripts. To the best of our knowledge, current YouTube video spam or scam classification systems use only the more easily accessible metadata, such as link counts, hashtags, likes, views, network features, and comments [7, 19, 20], yet these are only proxy data for the actual video contents. Related research goals such as the detection of clickbait through misleading titles and thumbnails is also metadata-based [8, 22] with few exceptions [21], even though a content analysis would make sense and datasets exist [15]. Other research is focused on user comments rather than the videos themselves in order to detect harmful or spam-like user reactions [1, 13, 18].

A field in which scholars have utilized subtitle transcripts more frequently is creating effective filters for violence or other content that is unsafe for kids [2–4, 12]. Notably, Binh et al. [6] show how incorporating subtitle features improves upon metadata-only classification for unsafe content.

Apart from this specialized direction, transcript-based video classification boils down to a classic text classification or spam detection task. Like the rest of the natural language processing community, spam classification has also shifted to (large) deep neural models and feature representations. However, traditional and much more efficient machine learning algorithms such as Naive Bayes, SVMs, or Logistic Regression trained on bag of words (BoW) text representations are still producing acceptable if not competitive results quite often [9, 16].

DATA ACQUISITION

We compiled a dataset of 4,755 videos with high-quality transcripts by (1) collecting appropriate product search queries (2) searching for product reviews on three search engines and extracting the unique reviews from the search engine results pages (SERPs), and (3) downloading and filtering the YouTube video transcripts. Table 1 summarizes the main dataset statistics.

To find product reviews via web search engines, we compiled a list of product categories from two publicly-available e-commerce product taxonomies: (1) the GS1 Global Product Classification[1] and (2) the Google Product Taxonomy.[2] We combined both taxonomies and removed food-related categories, live animals, and near duplicates. The final list contains 7,392 product categories for which we constructed queries of the form "best `<product category>`."

The prepared queries were sent to Google (by proxy of Startpage), Bing, and DuckDuckGo between May 24–25th, 2023, requesting the top 20 results each. From Startpage, we retrieved 4,588 unique YouTube URLs (5,033 including duplicates) which makes for 3.4 % of all search results. Given 20 results per query, ca. 68 % of all result sets contain at least one YouTube URL on average. For Bing and DuckDuckGo, the number is much lower with only 1,127 (0.7 %) and 847 (0.05 %) URLs, respectively. In total, we retrieved 7,007 URLs, of which 5,902 are unique. Half the videos (3,620) are shown on the first page, i.e., among the top-10 results. The median result rank across all videos and search engines is 9 (zero-indexed). The median result rank of videos on the first page is 6. Google ranks videos significantly higher (*Median* = 8) than Bing (*Median* = 12) or DuckDuckGo (*Median* = 13). The maximum number of videos in a result set is 9, but the majority of result sets contains at most one video.

Of all candidate URLs, we were able to download English-language transcripts for 4,993 videos using the YouTube API. The transcripts are provided either by the video author (less common) or are auto-generated using YouTube's own speech recognition system. The transcripts contain time codes but neither punctuation to mark the sentence structure, nor speaker diarization. Periods during which only music is played are indicated by a special "[Music]" token. We removed all transcripts that contained fewer than 100 spoken words (excluding music), which resulted in a final set of 4,755 video transcripts.

Table 1: Top-20 search engine results for 7,392 product queries from Startpage (Google), Bing, and DuckDuckGo. Website counts are calculated after stripping domain names of their subdomain parts using Mozilla's Public Suffix List.[3]

Filtering steps	Startpage	Bing	DDG	All
Total results	147,658	147,592	143,823	439,073
Unique URLs	128,854	122,775	112,702	258,400
Unique websites	41,514	26,853	22,862	60,947
YouTube URLs	5,033	1,127	847	7,007
Unique YouTube URLs	4,588	1,098	810	5,902
Transcripts available	4,242	785	426	4,993
Transcripts (filtered)				4,755
Test data (ground truth obtained via manual labeling)				200
Training data (unsupervised labeling through clustering)				4,555

MANUAL TEST DATA ANNOTATION

We constructed a test set of 200 manually annotated videos to evaluate the effectiveness of our approach. The examples were selected from the (filtered) dataset by uniform random sampling. Each video was annotated via single human annotation as either "Spam" or "Non-Spam". We used a heuristic-driven approach to annotation and constructed two sets of indicators for "Spam" and "Non-Spam" based on our observations of the material. Annotation was done by skipping through the video using the scrub bar until the annotator was convinced that one set of indicators was the more prevalent. The amount of time spent per annotation was around 15–30 seconds for most videos, depending on the difficulty to decide the class.

[1] `https://www.gs1.org/standards/gpc`, November 2021

[2] `https://www.google.com/basepages/producttype/taxonomy.en-US.txt`, September 2021

[3] `https://publicsuffix.org/`

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10498306

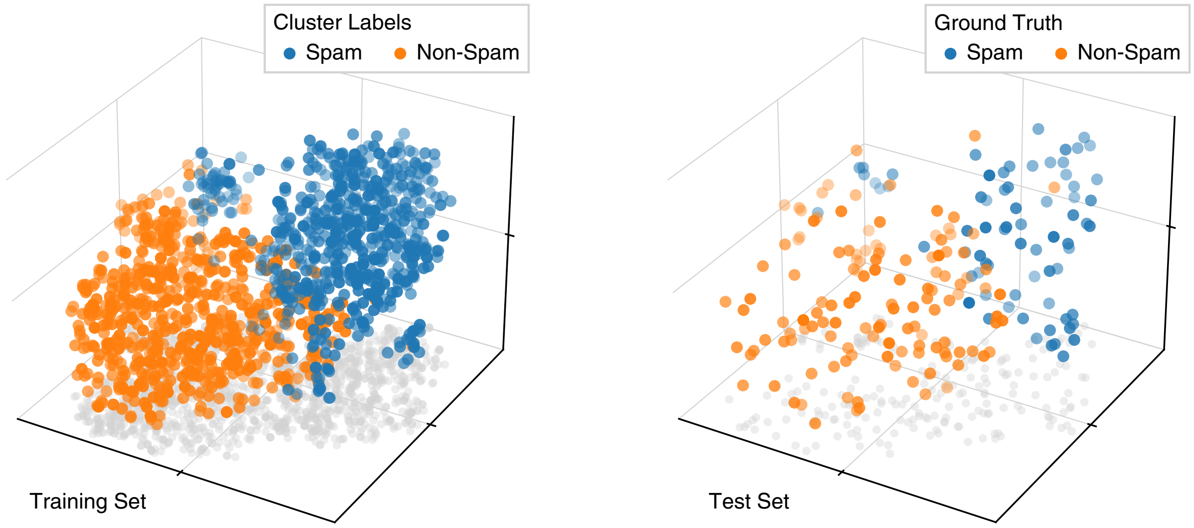


Figure 1: Feature vectors from the training data (left) and test data (right) embedded into a 3D space for visualization via *t*-SNE. The training data are labeled with their automatically assigned cluster labels, the test data with human-annotated ground-truth labels. To reduce visual clutter, only one third of the training data points are shown (randomly sampled).

Indicators for "Spam":

- Video uses lots of stock footage or is a slide show of commercial product photos.
- No actual hands-on product usage is shown on camera.
- Commentary is fully scripted and narration is performed by a text-to-speech system or a hired voice actor.
- Video or narration give the impression of a commercial rather than an unbiased review.
- Video is mostly a listing of product features and specifications from the web or the manufacturer's website.
- Product ratings are based on or reference user reviews from online shopping sites.
- The product selection appears random.
- Products are featured in rapid succession without much additional context.

Indicators for "Non-Spam":

- Video uses original footage.
- Products are shown live and in action, actual testing is performed in front of the camera.
- Video shows (parts of) a human protagonist on camera handling the products.
- Protagonist shares expert knowledge and provides additional, non-obvious information about the products and their usage.
- If instead, video uses off-camera commentary, sufficient expert knowledge is provided to make content believable and stand out from the "Spam" class.

Most videos fell clearly into one of the two classes based on these criteria even after watching only short segments. For the few ambiguous cases, the annotator was asked to give a personal value judgment about whether they felt informed given the information conveyed by the video in comparison to reading a chart of product specifications. Of the 200 annotated samples, 71 fell into the "Spam" class (35.5 %), the other samples were deemed sufficiently believable to be considered "Non-Spam."

Since we aimed at only a rough estimate, we deemed 200 instances annotated by a single person (the first author of this paper) sufficient. We leave a more thorough annotation of a higher number of instances with agreement scores between multiple independent annotators to future work.

TRAINING DATA CLUSTERING

For lack of large-scale gold-standard labels, we automatically labeled the training set using an unsupervised clustering approach over a stylometric feature space. Our approach assumes that videos in either class use a distinctly different language that is independent of the actual product topics. This assumption is loosely backed by our impression from annotating the 200 test set instances. To obtain a topic-independent representation of linguistic markers from the transcripts, we removed all "[Music]" markers and transformed the remaining examples to part-of-speech (POS) tags using the Penn Treebank tag set [11] as assigned by the SpaCy library.[4] From the transformed texts, we built a feature matrix with the relative frequencies of the 150 most common POS 1–4-grams over all instances in our train-

[4] `https://spacy.io/`, Version 3.5.2

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10498306

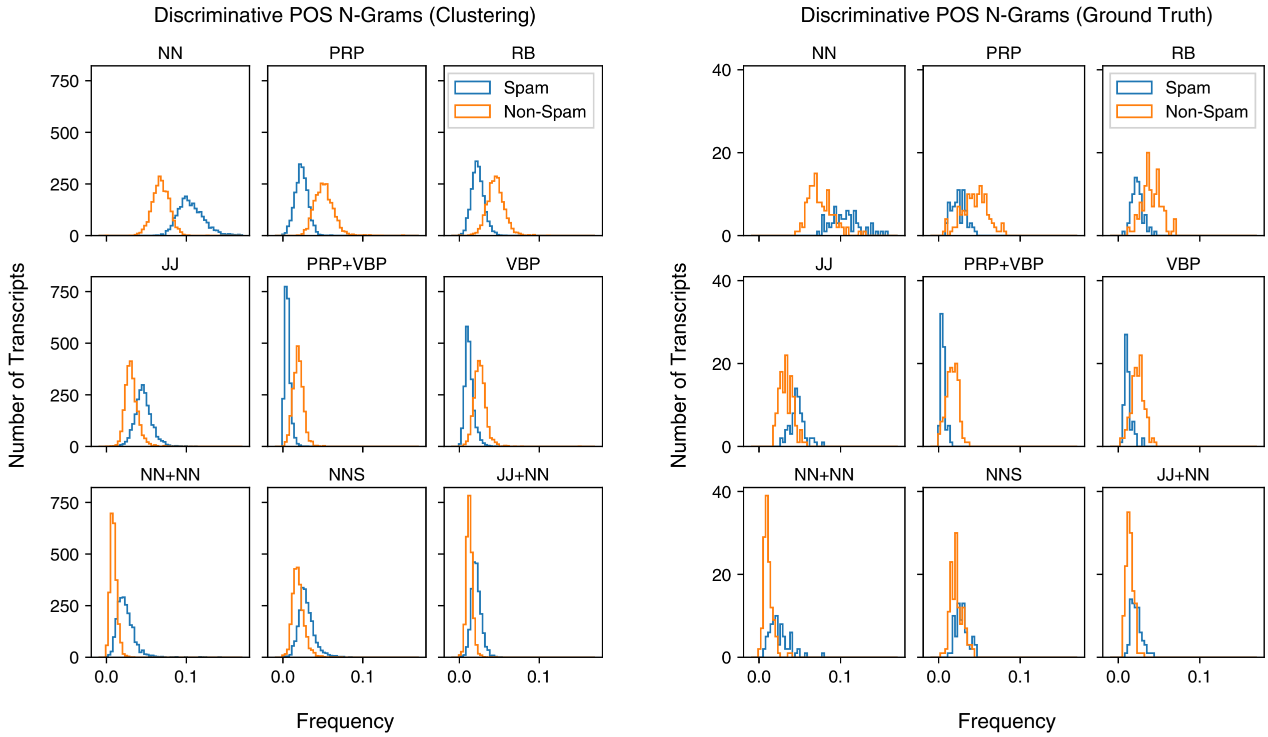


Figure 2: Frequencies of the nine most discriminative POS n-grams according to the SVM hyperplane coefficients by cluster labels (left) and by ground-truth labels (right). Spam videos tend to use more singular nouns (`NN`) and adjectives (`JJ`), whereas non-spam videos tend to use more personal pronouns (`PRP`) and adverbs (`RB`). Noun and personal pronoun unigram frequencies alone explain the majority of the group differences.

ing data. The first 20 of these n-grams in the dataset are in descending order of frequency: `NN`, `IN`, `DT`, `PRP`, `RB`, `JJ`, `VB`, `DT+NN`, `NNS`, `VBZ`, `VBP`, `CC`, `IN+DT`, `NN+IN`, `JJ+NN`, `PRP+VBP`, `NN+NN`, `DT+JJ`, `VBG`, `TO`.

We tested several clustering approaches from the scikit-learn library[5] to separate the data points into two stable sets. We found that a basic K-Means or spectral clustering produced the most stable clusters. DBSCAN as a density-based clustering was too unstable and sensitive to hyperparameter settings to be practical, since the cluster areas are not clearly separate and the point density appears to be quite uniform. In the end, we settled with the spectral clustering, as it makes no assumptions about the shapes of clusters (unlike K-Means), but still allows to control the number of clusters.

Of the 4,555 clustered samples, 2,202 fell into Cluster 1 and 2,353 into Cluster 2. Figure 1 (left) shows a 3D *t*-SNE projection of the produced clusters. Following the class distribution in the test set, we assigned the smaller of the two clusters the class “Spam” and the larger one the class “Non-Spam.” A projection of the test set into the same 3D space (Figure 1, right) has surprisingly similar shapes and locations with respect to its ground truth. With 48.3 %, the portion of “Spam” instances is 13 percentage points larger in the clustered training set than in our manually labeled test set (35.5 %). However, this is not surprising, as the clustering algorithm knows nothing about the actual intended target concept and hence may not derive accurate cluster boundaries if the data points in between are not clearly separate.

[5] `https://scikit-learn.org/stable/`, Version 1.2.2

Table 2: Classification results after training a linear SVM and a logistic regression model on the cluster labels. Test samples: $n = 200$ (Non-Spam: 129, Spam: 71).

Model	Class	Prec.	Recall	F1	AUROC
SVM	Non-Spam	0.98	0.78	0.87	0.94
	Spam	0.70	0.97	0.82	
Log. Regression	Non-Spam	0.97	0.81	0.88	0.93
	Spam	0.73	0.96	0.83	

EVALUATION AND DISCUSSION

As discussed in the previous section, projecting the clustered and the annotated ground-truth examples into the same 3D space (Figure 1) reveals a high agreement, which serves as a hint that our approach may be effective. In fact, it appears that the decision boundary may need only minor adjustments. To verify this assumption and to produce a reusable and transferable model with a less fallacious effectiveness evaluation than visual inspection in a low-dimensional space, we trained a linear SVM and a logistic regression model on the larger training dataset using the automatically derived cluster labels as targets.

https://doi.org/10.5281/zenodo.10498306

Table 2 summarizes the classification results on the ground-truth examples. Both classifiers achieve an F1 score of around 0.88 and AUROC of 0.94. The "Non-Spam" precision and "Spam" recall are both very high with 0.98 and 0.97. The effectiveness evaluation confirms that, despite the not optimally-chosen cluster boundaries, our system does capture the target concepts well and presents a working, yet improvable classification approach for YouTube product spam.

Ranked by the SVM's highest absolute hyperplane coefficients, we identify the unigram frequencies of `NN`, `PRP`, `RB`, and `JJ` as the most discriminative features (see Figure 2). These are followed by `PRP+VBP`, `VBP`, and `NN+NN`. Regarding the differences between the classes, "Spam" examples tend to have higher `NN` and `JJ` frequencies, whereas the "Non-Spam" examples tend to have higher `PRP` and `RB` frequencies. Most of the differences between the two groups are explained by the `NN` and `PRP` frequencies alone. Consulting the ground-truth examples, we observe about the same frequency distribution for these n-grams. This means that as their primary distinctive language features, "Spam" transcripts make more frequent use of nouns and adjectives, whereas "Non-Spam" transcripts have higher usage of personal pronouns and adverbs. The finding makes sense considering our annotation guidelines. Videos that are by and large a summary or a listing of product features fulfill key criteria for the "Spam" class. Videos featuring a real human protagonist talking about the product and their personal experience with it, on the other hand, are far more likely to be "Non-Spam."

LIMITATIONS

The study shows promising results, but has a few limitations. First of all, the number of ground-truth samples is very low and the labels were assigned by only a single annotator. The classifier is able to predict the "Non-Spam" majority class with higher precision than the "Spam" class, but a correction of the decision boundary would require more labeled examples or additional features for the cluster formation. Moreover, the (non-)spam classifier is able to identify obvious spam content quite reliably, but it cannot check the factual accuracy of the review contents. Finally, the findings are limited to English-language search results of three search engines for the constructed product queries.

CONCLUSION

We have developed an effective yet simple unsupervised spam classifier for improving product-centric video results of web search engines. We examined the YouTube video results of three major commercial search engines for thousands of product review queries. For all collected video results, we retrieved the automatic subtitle transcripts via the YouTube API and automatically labeled them using a spectral clustering based on POS n-gram frequency representations. We verified the clustering accuracy by training a supervised linear model on the automatic labels and compared the results to a small set of annotated examples. It turns out that the resulting clusters approximately capture the classes of "Spam" and "Non-Spam" videos and most of the differences are explained by the use of nouns, personal pronouns, adverbs, and adjectives. However, the decision boundary is not learned perfectly ($F_1 = 0.88$, AUROC = 0.94), which leaves potential for further optimization.

We conclude that since the portion of "Spam" videos seems quite high and we were able to detect them so easily, search engines should apply more careful filtering of their video results. Our case study can be thought of as a first step towards better product spam recognition in online videos.

REFERENCES

[1] Túlio C. Alberto, Johannes V. Lochter, and Tiago A. Almeida. Tubespam: Comment spam filtering on youtube. In Tao Li, Lukasz A. Kurgan, Vasile Palade, Randy Goebel, Andreas Holzinger, Karin Verspoor, and M. Arif Wani, editors, *14th IEEE International Conference on Machine Learning and Applications, ICMLA 2015, Miami, FL, USA, December 9-11, 2015*, pages 138–143. IEEE, 2015.

[2] Sharifa Alghowinem. A safer youtube kids: An extra layer of content filtering using automated multimodal analysis. In *Intelligent Systems and Applications: Proceedings of the 2018 Intelligent Systems Conference (IntelliSys) Volume 1*, pages 294–308. Springer, 2019.

[3] Saeed Ibrahim Alqahtani, Wael MS Yafooz, Abdullah Alsaeedi, Liyakathunisa Syed, and Reyadh Alluhaibi. Children's safety on youtube: A systematic review. *Applied Sciences*, 13(6):4044, 2023.

[4] Vishal Anand, Ravi Shukla, Ashwani Gupta, and Abhishek Kumar. Customized video filtering on youtube. *CoRR*, abs/1911.04013, 2019.

[5] Janek Bevendorff, Matti Wiegmann, Martin Potthast, and Benno Stein. The Impact of Online Affiliate Marketing on Web Search. In Andreas Wagner, Christian Guetl, Michael Granitzer, and Stefan Voigt, editors, *4th International Symposium on Open Search Technology (OSSYM 2022)*. International Open Search Symposium, October 2022.

[6] Le Binh, Rajat Tandon, Chingis Oinar, Jeffrey Liu, Uma Durairaj, Jiani Guo, Spencer Zahabizadeh, Sanjana Ilango, Jeremy Tang, Fred Morstatter, Simon S. Woo, and Jelena Mirkovic. Samba: Identifying inappropriate videos for young children on youtube. In Mohammad Al Hasan and Li Xiong, editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 88–97. ACM, 2022.

[7] Elijah Bouma-Sims and Bradley Reaves. A first look at scams on youtube. *CoRR*, abs/2104.06515, 2021.

[8] Ruchira Gothankar, Fabio Di Troia, and Mark Stamp. Clickbait detection in youtube videos. *CoRR*, abs/2107.12791, 2021.

[9] Thiago S. Guzella and Walmir M. Caminhas. A review of machine learning approaches to spam filtering. *Expert Syst. Appl.*, 36(7):10206–10222, 2009.

[10] Dirk Lewandowski, Sebastian Sünkler, and Nurce Yagci. The influence of search engine optimization on Google's results: A multi-dimensional approach for detecting SEO. In

https://doi.org/10.5281/zenodo.10498306

Proceedings of the 13th ACM Web Science Conference 2021, WebSci '21, pages 12–20, New York, NY, USA, June 2021. Association for Computing Machinery.

[11] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993. `https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html`.

[12] Enrico Mariconti, Guillermo Suarez-Tangil, Jeremy Blackburn, Emiliano De Cristofaro, Nicolas Kourtellis, Ilias Leontiadis, Jordi Luque Serrano, and Gianluca Stringhini. "you know what to do": Proactive detection of youtube videos targeted by coordinated hate attacks. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–21, 2019.

[13] P Nagaraj, K Muthamil Sudar, P Thrived, P Girish Kumar Reddy, Sk Baji Babu, and P Siva Rama Krishna. Youtube comment spam detection. In *2023 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–6. IEEE, 2023.

[14] Kristen Purcell, Joanna Brenner, and Lee Rainie. Search Engine Use 2012. Technical report, Pew Research Center's Internet & American Life Project, March 2012.

[15] Jiani Qu, Anny Marleen Hißbach, Tim Gollub, and Martin Potthast. Towards Crowdsourcing Clickbait Labels for YouTube Videos. In Yiling Chen and Gabrielle Kazai, editors, *6th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2018)*, July 2018.

[16] Sanjeev Rao, Anil Kumar Verma, and Tarunpreet Bhatia. A review on social spam detection: Challenges, open issues, and future directions. *Expert Syst. Appl.*, 186:115742, 2021.

[17] Sebastian Schultheiß, Helena Häußler, and Dirk Lewandowski. Does Search Engine Optimization come along with high-quality content? A comparison between optimized and non-optimized health-related web pages. In David Elsweiler, editor, *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, CHIIR '22, pages 123–134, New York, NY, USA, March 2022. Association for Computing Machinery.

[18] Sadia Sharmin and Zakia Zaman. Spam detection in social media employing machine learning tool for text mining. In Kokou Yétongnon, Albert Dipanda, Richard Chbeir, Luigi Gallo, and Neeta Nain, editors, *13th International Conference on Signal-Image Technology & Internet-Based Systems, SITIS 2017, Jaipur, India, December 4-7, 2017*, pages 137–142. IEEE Computer Society, 2017.

[19] Ashutosh Tripathi, Kusum Kumari Bharti, and Mohona Ghosh. A study on characterizing the ecosystem of monetizing video spams on youtube platform. In *Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services, iiWAS 2019, Munich, Germany, December 2-4, 2019*, pages 222–231. ACM, 2019.

[20] Ashutosh Tripathi, Mohona Ghosh, and Kusum Kumari Bharti. Analyzing the uncharted territory of monetizing scam videos on youtube. *Soc. Netw. Anal. Min.*, 12(1):119, 2022.

[21] Deepika Varshney and Dinesh Kumar Vishwakarma. A unified approach for detection of clickbait videos on youtube using cognitive evidences. *Appl. Intell.*, 51(7):4214–4235, 2021.

[22] Savvas Zannettou, Sotirios Chatzis, Kostantinos Papadamou, and Michael Sirivianos. The good, the bad and the bait: Detecting and characterizing clickbait on youtube. In *2018 IEEE Security and Privacy Workshops, SP Workshops 2018, San Francisco, CA, USA, May 24, 2018*, pages 63–69. IEEE Computer Society, 2018.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10529619

CHALLENGES OF INDEX EXCHANGE FOR SEARCH ENGINE INTEROPERABILITY

Djoerd Hiemstra, Gijs Hendriksen, Chris Kamphuis, and Arjen P. de Vries*
Radboud University, The Netherlands

Abstract

We discuss tokenization challenges that arise when sharing inverted file indexes to support interoperability between search engines, in particular: How to tokenize queries such that the tokens are consistent with the tokens in the shared index? We discuss various solutions and present preliminary experimental results that show when the problem occurs and how it can be mitigated by standardizing on a simple, generic tokenizer for all shared indexes.

INTRODUCTION

Web search is dominated by a small number of giant corporations that effectively hold a monopoly on web search. Quite worryingly, these corporations control almost every aspect of web search: They crawl the Web, they build the index, they provide the actual search results given a query, they sell advertisements, they provide free web analytics to get usage statistics, they even own the web browsers and operating systems that we need to use their search engines.

In our opinion, a single corporation should not control a large share of these aspects of search. For instance, incentives for providing high quality search results do not align with incentives to sell advertisements, or to quote Brin and Page [2]: *"The goals of the advertising business model do not always correspond to providing quality search to users (...) we expect that advertising funded search engines will be inherently biased towards the advertisers and away from the needs of the consumers."*

Two important solutions may help break up these monopolies. One is regulation: It should not be legal to run an advertisement company and a search engine, nor should it be legal to own large web sites as well as the web browser that renders them. The second solution (that might help enable the first) is technical: We should create tools that enable collaboration between multiple organizations, so they can develop web search engines together. This paper focuses on a solution of the second, technical, kind. Specifically, we discuss the challenges of defining open standards that support interoperability between search engines to enable organizations to build web search engines collaboratively.

Building a web-scale search engine is a challenging task. Crawling the web takes a lot of resources, as does building the inverted index. Once the index is ready, however, running queries on the index can be done with relatively little compute power. We envision a future where organizations collaboratively build a search engine by using open standards that define the results of each step [6]. We show these steps in Figure 1. The first step is to (collaboratively) crawl the Web and provide it in a standard format, such as the Internet Archive's Web Archive (WARC) format [13]; In Step 2, others may build an inverted index to be provided in the standard Common Index File Format (CIFF) [11]; In Step 3, yet others take the index and build the search engine (backend) provided as an API based on the OpenSearch standard [5]; which is finally used in Step 4 by the organization that builds the search application (frontend).

This paper discusses the challenges of using the common index file format CIFF in Step 2. We discuss CIFF and why it is currently underspecified for the use in a production search engine in the following section. In the final section, we present preliminary experiments that demonstrate the problem in practice, provide a generic solution and report on experiments showing its adequacy. The code used to run our experiments is available via a public Git repository[1].

THE COMMON INDEX FILE FORMAT

The Common Index File Format (CIFF) was defined in 2020 by researchers and developers of the following open source retrieval research systems: Terrier, Anserini (which uses Lucene), PISA, JASSv2, and OldDog [11]. CIFF's goal is to improve the reproducibility of information retrieval experiments by allowing a search system to export the inverted index and import it into another system. This way, researchers can rule out differences in retrieval performance that are caused by building the index, such as text preprocessing and boilerplate removal, focusing solely on other aspects, such as the ranking algorithm.

Tokenization challenges of CIFF

To build an inverted index, the document texts need to be tokenized. When the CIFF index is used in another system, the queries that are put to the system, should use the tokens in the index. One of the main challenges of using CIFF in practice is the following: How do we tokenize the query such that the tokens are consistent with the tokens in the index? To ensure consistent tokenization between the indexer and searcher, Lin et al. [11] exchanged pre-tokenized versions of the test queries (also known as topics). However, in a production search engine, there is no way of knowing all possible queries beforehand, let alone pre-tokenizing them. Consistent tokenization between index and queries remains an unsolved problem of index exchange, at least outside the narrow scope of information retrieval experiments that use benchmark test collections with a small set of test queries. We discuss tokenization in information retrieval and three

* {djoerd.hiemstra, gijs.hendriksen, chris.kamphuis, arjen.devries}@ru.nl

[1] https://opencode.it4i.eu/openwebsearcheu-public/index-sharing

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10529619

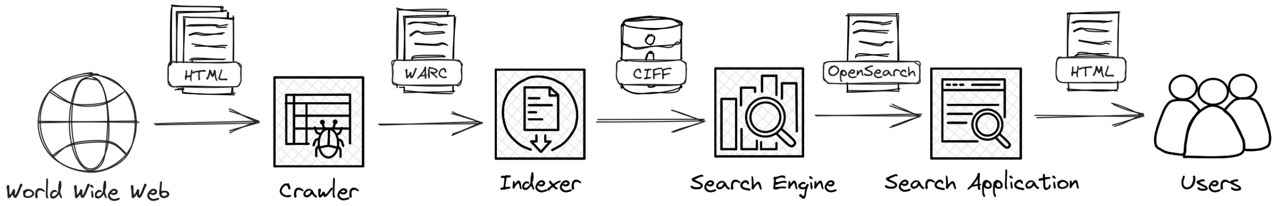


Figure 1: Steps in building and running a search engine. In between each step, data is exchanged conform to a specific open standard: HTML for web content; WARC for web archives; CIFF for inverted files; and OpenSearch for search results.

possible approaches to achieve tokenization consistency for CIFF below.

Tokenization in information retrieval

There is surprisingly little research done into tokenization for information retrieval. A good overview is given by Büttcher et al. [3, Chapter 3]. For English and some other western languages, a simple tokenizer that splits on space and punctuation usually suffices. Sometimes a stop word list is used to remove common words. Often several surface forms are mapped to the same token, for instance acronyms might be written as *EU* or as *E.U.* and the tokenizer may map them to the same token. For inflective languages, a stemmer such as Porter's stemmer for English [17] could map many different inflections of the same word to a common root, for instance *indexing*, *indexation* and *indexes* will all be mapped to a common root: *index*. Instead of a stemmer, the use of letter n-grams has been shown to be surprisingly effective for inflective languages [12], but an n-gram index is less efficient as queries will have more tokens for which then (longer) posting lists need to be fetched and merged.

Non-Western languages like Chinese and Japanese use many more characters than Western languages. Chinese has thousands of distinct characters. Even though each character has a meaning by its own, lots of words consist of multiple characters and those words are not separated by spaces. Indexing Chinese documents therefore requires a non-trivial word segmentation algorithm [22]. Consistent tokenization for an imported index may therefore be a bigger problem for Chinese than for English. The Unicode consortium provides extensive guidelines for text segmentation for many other languages [4].

Our solution to the tokenization problem should support all human languages and at least general approaches like stop words and mapping multiple surface terms to the same token (including stemmers).

Possible tokenization solutions

How do we make sure that query tokenization is done in a way that is consistent with the imported index? In this Section, we discuss possible solutions, including: shipping the tokenization source code with the CIFF index; providing a declarative specification of the tokenizer with the CIFF index; and defining a generic tokenizer that works with any CIFF index.

Including the tokenizer source code Providing the tokenizer code inside (or with) the CIFF index would solve the tokenizer inconsistency problem, but it also creates several new questions and problems. One is: What programming language should be used? Terrier, Lucene and Anserini use Java. PISA and JASSv2 use C++. Another problem is that each index may come with their own tokenizer, so the number of tokenizers that need to be shared would possibly increase with every CIFF index. This would make CIFF an easy target of software supply chain attacks, where malicious code is injected into tokenizers. To conclude, including the tokenizer source code into CIFF would require the CIFF developers to agree on a programming language for tokenizers, and it would require a high level of trust into the shared code.

Tokenizers in embeddable scripts or bytecode Security concerns of including the code of tokenizers can be partly met by using an embeddable scripting language like Lua that is designed to run inside applications in a carefully guarded sandbox. Other options would be to use JavaScript or WebAssembly, that are both used in web browsers and therefore heavily guarded against malicious use.

Declarative tokenizers Another option may be to use the lexers of parser generators like ANTLR [16] as a tool for specifying a tokenizer and generating the tokenizer for many programming languages, including the ones mentioned above. Parser generators are used to parse programming languages and possibly structured query languages. They are used in search engines that require complex structured queries, such as Lemur [14] and PF/Tijah [8] (both are research systems that are no longer maintained). Specifying tokenizers this way seems to be a non-trivial, tedious job. We have not investigated this option further.

A single, generic tokenizer CIFF comes with the complete dictionary containing all possible tokens as part of the inverted file. The query tokenizer may therefore adapt to the dictionary, ensuring that the query is tokenized in a way that best fits the imported index. We will pursue this solution in the next section.

A generic CIFF tokenizer

Recent advances in neural machine translation and large language models come with interesting developments for

https://doi.org/10.5281/zenodo.10529619

tokenization. Their tokenizers use a relatively limited vocabulary by splitting uncommon words into word pieces. This is done for two reasons: *1)* to speed up processing and decrease the number of parameters to be trained; and *2)* to gracefully handle out-of-vocabulary words, which will occur in unseen data no matter how big of a vocabulary the model uses. Word piece models are trained on the data to find the best word piece tokenizer for that data [10, 18]. So, these tokenizers are generic tokenizers, that are trained or finetuned on data. The trained vocabulary, possibly with additional frequency information, and a generic tokenization algorithm define the tokenizer.

We will use this idea of trained tokenizers to define a generic tokenizer for CIFF. In this analogy, the indexing step is the training step. The index, encoded in CIFF, contains the possible tokens to be used by the generic tokenizer algorithm. This way, every CIFF index will use its own custom tokenizer, without the need to share the tokenizer source code or bytecode. Unlike the word piece tokenizers mentioned above, the CIFF tokenizer will typically use a much larger vocabulary, although it could as easily use word pieces or, alternatively, use multi-word units for phrases or named entities. The generic tokenizer is completely language-agnostic: It works on any unicode string and does not need to know about (English or Chinese) character sets. It does not even know about spaces or punctuation. The generic tokenizer may be implemented in about 20 lines of code. Example code for the generic tokenizer is included in the appendix. Its efficiency may be further improved by building a trie from the vocabulary [19].

PRELIMINARY EVALUATION OF TOKENIZATION FOR CIFF

In this section we show preliminary evaluation results using the TREC Robust 2004 dataset [20]. We made inverted indexes for two search systems: GeeseDB [9] and Terrier [15]. Each index was exported to CIFF and imported in the other system. We then evaluate *1)* the original system, *2)* the other system, tokenizing queries with its standard tokenizer, and *3)* the other system with the generic tokenizer. We show that performance degrades when a mismatch in tokenization occurs, but that this can be mitigated by using the generic tokenizer.

For both GeeseDB and Terrier, we rank documents with BM25, using the parameters $b = 0.4$ and $k_1 = 0.9$. Stop words are *not* removed from the corpus, and no stemming is applied. We discuss both techniques, and how they can be handled in CIFF, in more detail in our section below on future plans.

GeeseDB is configured to use the NLTK tokenizer [1], and Terrier uses its standard internal tokenizer. These tokenizers mostly differ in how they handle certain types of punctuation. For instance, NLTK leaves tokens intact when they contain hyphens or periods (like *on-line* or *U.S.*), while Terrier will split these into multiple tokens. To highlight these differences, we additionally run our experiments on the subset of only those Robust04 topics that contain a hyphen or period (19 topics in total).

In all our experiments, we test whether differences are statistically significant by applying a two-tailed paired t-test. We use a significance level of $\alpha = 0.01$.

Results on the Terrier index

Table 1a shows our results on the Terrier index.[2] We see that performance significantly degrades if we use the Terrier inverted file in GeeseDB out of the box. However, once we apply the generic CIFF tokenizer, we are able to correctly adapt GeeseDB to the tokenization used by Terrier. In fact, using the Terrier index in GeeseDB with the generic tokenizer seems to match (or even slightly surpass) the performance of Terrier itself.

These results are even more apparent when looking at Table 1b, where we zoom in on the topics that contain hyphens or periods. There is a very large performance drop when we use GeeseDB with the NLTK tokenizer for query preprocessing, but this drop disappears when using the generic tokenizer.

Table 1: Performance of different systems using an inverted file generated with Terrier. Best results are marked in bold. Configurations that perform significantly ($p < 0.01$) better than the Terrier index imported into GeeseDB (the middle row) are marked with †.

(a) All Robust04 topics

System	Tokenizer	MAP	nDCG
Terrier	Terrier	0.221†	0.480†
GeeseDB	NLTK	0.208	0.457
	CIFF	**0.224**†	**0.482**†

(b) Robust04 topics that contain a hyphen or period

System	Tokenizer	MAP	nDCG
Terrier	Terrier	**0.234**†	**0.541**†
GeeseDB	NLTK	0.081	0.292
	CIFF	**0.234**†	**0.541**†

Results on the GeeseDB index

Table 2 shows the results of our experiments with the GeeseDB inverted file. The differences are not as large (or significant) as they were with the Terrier index, but we still notice a drop in performance for queries with hyphens or periods (Table 2b). Again, this drop can be mitigated by using the generic tokenizer.

Discussion

Our preliminary experiments on Robust04 indicate that our proposed generic tokenizer could be a useful addition

[2] Systems that use a similar tokenizer, like Anserini/Lucene, give similar experimental results.

https://doi.org/10.5281/zenodo.10529619

Table 2: Performance of different systems using an inverted file generated with GeeseDB. Best results are marked in bold. Configurations that perform significantly ($p < 0.01$) better than the GeeseDB index imported into Terrier (the middle row) are marked with †.

(a) All Robust04 topics

System	Tokenizer	MAP	nDCG
GeeseDB	NLTK	0.207	0.460
Terrier	Terrier	0.208	0.460
	CIFF	**0.209**	**0.462**

(b) Robust04 topics that contain a hyphen or period

System	Tokenizer	MAP	nDCG
GeeseDB	NLTK	0.155	0.433
Terrier	Terrier	0.145	0.392
	CIFF	**0.183**	**0.474**

to the CIFF standard for inverted files. The results show us that a retrieval system that is tokenized by using a greedy matching approach on the inverted file's dictionary is able to match the performance of the system in which the inverted file was created.

We also see that the tokenizer used to build the inverted file matters in terms of retrieval effectiveness. The Terrier index seems to consistently result in higher performance than the GeeseDB index. To optimize the effectiveness of a system using CIFF files, we would need to look at which tokenizers produce the most useful dictionaries and inverted files – knowing that the downstream system will use the generic tokenizer.

CONCLUSION AND FUTURE PLANS

We discuss challenges of using the Common Index File Format (CIFF) as an open standard for index exchange between search engines. We propose a generic CIFF tokenizer that ensures that the tokenization of queries is consistent with the tokenization that was used to make the CIFF index, without the need to exchange the tokenizers themselves.

The CIFF generic tokenizer

Preliminary experimental results with two search systems confirm that exchanging a CIFF index without properly handling tokenization may result in a significant drop in search quality. This drop happens for instance when a token is split in multiple pieces at index time (like when *on-line* is indexed as *on* and *line*) but in one piece at query time (*on-line*, which then cannot be found in the index). Experimental results show that our our generic CIFF indexer fixes this without explicit knowledge of the pre-processing pipeline of the source system used to create the index. Together with developers of search engines, we hope to define a new version of CIFF that includes the generic tokenizer and that will work for all languages and a large range of search applications. We plan to work on the issues discussed below to make this happen.

Towards CIFF version 2

For a version 2 of the CIFF standard, we need more experimentation and several other adaptations for stop words, stemming and, possibly, incremental indexing.

Indexing for non-western languages We plan to do experiments for non-western datasets such as Chinese and Japanese. Tokenization problems may be much more common for these languages, because they do not distinguish words using spaces.

Stop words To allow the proper handling of stop words and other tokens that are not indexed, CIFF indexes should include them in the index with an empty posting list. If the stop words themselves are not included, they will not be properly tokenized by the CIFF tokenizer (but split up in shorter tokens that are included in the index).

Using stemmers Supporting stemming in CIFF is not entirely trivial. The CIFF tokenizer only works on surface forms of the tokens, not on derived tokens like stems. One way for CIFF to support stemming, is to include all surface tokens that are found in the data into the CIFF index, grouping them for each stem. So, every posting list could come with multiple tokens. For instance, the posting list for the stem *pickl* (Stemmers do not always produce linguistically correct stems) will in CIFF contain a set of tokens: *pickle*, *pickled*, *pickles*, and any other word that stems to *pickl*. This approach may be used for many other tokenization challenges too, like conflating the acronyms *EU* and *E.U.* as mentioned above. Furthermore, it allows for more flexible stemmers, including stemmers that use lookup tables and corpus-based stemmers [21]. The approach will make it harder to export any index to CIFF, because the surface tokens need to be retained somewhere. It will also give different results for query terms that do not occur once in the entire data, but of which the stem does occur. Nevertheless, we believe including sets of surface terms is an elegant solution for derived tokens, including the use of stemmers.

Open indexes for large-scale web search Large web indexes contain multiple languages and several kinds of data. This opens up the possibility to use different tokenizers depending on the language of the document, or even depending on the language of paragraphs in multilingual documents. This setup would introduce new challenges: For instance, if multiple surface terms are grouped for one language, as described above, they will have to be grouped for all languages. Experiments will have to show how the generic CIFF tokenizer would handle such a setup.

Index updates A large CIFF web index may be many gigabytes or even some terabytes in size. Additionally, the Web changes all the time, so it makes sense to share a large

https://doi.org/10.5281/zenodo.10529619

CIFF index once and share updates for the index regularly afterwards. We think CIFF should include some rudimentary way to indicate updates of the index.

Open source implementations A standard is more resilient if there are multiple implementations. We implemented several tools for CIFF, including a CIFF importer for Lucene [7], and tools for merging multiple smaller indexes into one bigger index.

ACKNOWLEDGMENTS

This work is funded by the EU project OpenWebSearch.eu under GA 101070014. We are grateful for the EU's support.

REFERENCES

[1] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, 2009.

[2] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30:107–117, 1998.

[3] Stefan Büttcher, Charles Clarke, and Gordon Cormack. *Information retrieval: Implementing and evaluating search engines*. MIT Press, 2010.

[4] Mark Davis. Unicode standard annex 29: Unicode text segmentation (revision 41). Technical report, Unicode Consortium, 2022. http://www.unicode.org/reports/tr29/.

[5] Clinton DeWitt. *What is the OpenSearch protocol?* A9.com, 2005. https://github.com/dewitt/opensearch.

[6] Michael Granitzer, Stefan Voigt, et al. Impact and development of an open web index for open web search. *Journal of the Association for Information Science and Technology*, 2023.

[7] Gijs Hendriksen, Djoerd Hiemstra, and Arjen de Vries. Lucene CIFF importer. Zenodo, 2023. https://doi.org/10.5281/zenodo.8261333.

[8] Djoerd Hiemstra, Henning Rode, Roel van Os, and Jan Flokstra. PF/Tijah: text search in an XML database system. In *Proceedings of the 2nd International Workshop on Open Source Information Retrieval (OSIR)*, pages 12–17, 2006.

[9] Chris Kamphuis and Arjen de Vries. GeeseDB: A Python graph engine for exploration and search. In *Proceedings of the 2nd International Conference on Design of Experimental Search and Information REtrieval Systems (DESIRES)*, 2021.

[10] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[11] Jimmy Lin, Joel Mackenzie, Chris Kamphuis, Craig Macdonald, Antonio Mallia, Michał Siedlaczek, Andrew Trotman, and Arjen de Vries. Supporting interoperability between open-source search engines with the common index file format. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2149–2152, 2020.

[12] Paul McNamee and James Mayfield. Character n-gram tokenization for European language text retrieval. *Information retrieval*, 7:73–97, 2004.

[13] Gordon Mohr, John Kunze, and Michael Stack. The WARC file format 1.0 (ISO 28500). Technical report, International Organization for Standardization, 2008. https://escholarship.org/uc/item/9nh616wd.

[14] Paul Ogilvie and Jamie Callan. Experiments using the Lemur toolkit. In *Proceedings of the 10th Text Retrieval Conference (TREC)*, pages 103–108, 2001.

[15] Iadh Ounis, Gianni Amati, Vassilis Plachouras, Ben He, Craig Macdonald, and Douglas Johnson. Terrier information retrieval platform. In *Proceedings of the 27th European Conference on IR Research (ECIR)*, pages 517–519, 2005.

[16] Terence Parr and Russell Quong. ANTLR: A predicated-ll (k) parser generator. *Software: Practice and Experience*, 25(7):789–810, 1995.

[17] Martin Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[18] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2015.

[19] Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. Fast wordpiece tokenization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2089–2103, 2021.

[20] Ellen Voorhees. The TREC robust retrieval track. *ACM SIGIR Forum*, 39(1):11–20, 2005.

[21] Jinxi Xu and Bruce Croft. Corpus-based stemming using cooccurrence of word variants. *ACM Transactions on Information Systems*, 16(1):61–81, 1998.

[22] Nianwen Xue. Chinese word segmentation as character tagging. *International Journal of Computational Linguistics & Chinese Language Processing*, 8(1):29–48, 2003.

APPENDIX

Example Python code for the generic tokenizer.

```
def tokenize_generic_greedy(self, query):
    """ Tokenize a query generically.
        self.dictionary contains all tokens from
        the inverted file.
        self.max_token_length contains the length
        of the longest token.
    """
    tokens = []
    begin = 0
    query_length = len(query)
    while begin < query_length:
        end = begin + self.max_token_length
        if end > query_length:
            end = query_length
        token_found = None
        while begin < end:
            token = query[begin:end]
            if token in self.dictionary:
                token_found = token
                break
            end -= 1
        if token_found:
            tokens.append(token)
            begin = end
        else:
            begin += 1
    return tokens
```

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10581841

OWLER: PRELIMINARY RESULTS FOR BUILDING A COLLABORATIVE OPEN WEB CRAWLER

M. Dinzinger*, S. Zerhoudi,
M. Al-Maamari, M. Istaiti,
J. Mitrović, M. Granitzer,
University of Passau, Passau, Germany

Abstract

In the rapidly evolving digital landscape, the need for efficient and effective web crawling mechanisms is more crucial than ever. Web crawlers are instrumental in discovering and indexing new content, and their role in areas such as search engine optimization, data mining, and web archiving is indispensable. However, current distributed crawling frameworks face significant challenges in terms of topic-based content discovery and categorization. This paper presents novel extensions to the StormCrawler and URLFrontier frameworks to enhance web crawling efficiency and relevance. The OWler, a derivative of the StormCrawler, is extended with classification functionalities, including topic identification and thus enabling the production of topic-specific WARC files using multiple writing streams. Concurrently, the URLFrontier framework is extended to enable web crawlers to retrieve URLs based on topic interests. To put it in a nutshell, these extensions allow us to build a highly distributed network of heterogeneous web crawlers, which are nevertheless efficiently collaborating over a shared crawl space.

INTRODUCTION

As the digital landscape continues to evolve, the need for efficient web data acquisition becomes increasingly prominent. We currently experience an urgent demand for data which empowers researchers and businesses in Europe and around the world. However, tapping the web as a resource is technically challenging and requires considerable upfront investment in infrastructure and the recruitment of highly skilled professionals. In this light, the consortium of the OpenWebSearch.eu project has pooled their resources and expertise with the aim of democratizing access to web information, currently controlled by a small group of gatekeepers, and thereby unlocking the potential for a broader community of European innovators.

Open Web Index

The common objective of the project participants is to establish an Open Web Index (OWI) and foster an ecosystem around it [7]. Central to this ambition is the advocacy for an open search engine market and the offering of a genuine choice to users. Furthermore, the Web Index provides an avenue for the development of diverse web-data products which go beyond the scope of web search and target e.g., research in the flourishing AI domain. The creation and maintenance of the OWI adhere to Open Data principles and legal compliance, and require a decentralized structure among multiple European institutions.

Web crawling

An important part of this effort is the collection of web content through crawling. A web crawler is a tool designed for systematically downloading an extensive number of webpages. Its specific software architecture varies considerably depending on the intended use of the software [2].

Our crawling system is characterized by a collaborative crawling strategy. We have opted for a distributed architecture which consists of multiple heterogeneous worker nodes in different remote computing sites. The crawling nodes communicate with a central, controlling component, the URL Frontier. This design diverges from other state-of-the-art distributed crawling systems, such as Apache Nutch [11] or BUbiNG [2], which are built upon a single software solution and threfore assume a single-machine setup or structural uniformity among the cooperating agents.

In this context, we have developed the OWler (Open Web Crawler), an incremental and distributed web crawler. It is highly customizable, yet primarily tailored for extensive data acquisition with the purpose of building a Web Index. The OWler collects webpages that align with its scope of interest and documents its fetch activities in the WARC file format. It is further augmented by our integration of a topic classification model for more specific content categorization. Furthermore, the usage of the *URL Frontier* framework as a central component enables seamless collaboration among multiple geographically distributed crawlers [15]. It maintains the known and to-be-fetched URLs, and divides the crawl space among the worker nodes based on their scope of interest.

In the remaining sections of this paper, we will present the OWler and delve into our approach to web crawling. Section 2 provides an overview of related work in this field and Section 3 describes the existing frameworks which underpin our approach. Our main contributions are outlined in Section 4, which details the enhancements made to the OWler. Section 5 evaluates the applicability of our crawling system in a real-world setup and Section 6 concludes the paper, with an outlook to future directions of our work.

RELATED WORK

The domain of web crawling dates back to the origins of the world wide web itself. Concurrent to the rise of the mod-

* michael.dinzinger@uni-passau.de

https://doi.org/10.5281/zenodo.10581841

ern internet in early 1990s, tools and techniques for efficient web search became necessary. So throughout the following years, search engine operators and reseachers worked on software tools for the efficient traversal of the web. Due to a wide range of research endeavors, crawling systems have continously improved on its four main quality criteria: coverage and freshness, politeness and robustness.

Thereby, a number of technical challenges have been tackled, such as the near duplication detection of web documents [5, 14], the de-duplication of URLs [1] and the queue-based scheduling mechanism for webpages [18]. This scheduling mechanism is often implemented as *Frontier* and prioritizes URLs depending on the webpage quality, while ensuring politeness towards web servers.

Several notable web crawlers have been developed over the years. *Mercator*, described in Najork et al. (2002), was one of the first commercial open-source crawlers developed by IBM that targeted high-performance and introduced the "frontier" data structure [18]. *Heritrix* and the open-source crawler *Apache Nutch* were further early web crawlers that were extensively used and studied [11, 17]. *IRLbot* was a pioneering effort in scaling web crawling to handle billions of webpages on a single-machine setup [13]. More recently, the *BUbiNG* crawler was developed by the Laboratory of Web Algorithmics as a next-generation dataset crawler, with a public repository available for research use [2].

In the last 25 years, two web protocols have made a significant impact on the ethicality and efficiency of crawlers and, thus, have become the informal standard among webmasters and search engine operators. The *Robots Exclusion Protocol (REP)* [10] encourages content owners to state access rules for non-human visitors in a "robots.txt" file, which is placed at the website's root directory. However, the REP has no direct legal relevance [19] and is only intended to restrict disproportinate server traffic caused by robotic accesses [21]. Nevertheless, it is nowadays the most important mean for ensuring politeness in web crawling and scraping activities. At this point, we want to remark that in the future an extension to the REP or a similar technical solution might be necessary to impose fine-grained and legally binding restrictions, such as licensing information, on publicly available web content.

The *Sitemap Protocol* originates from the idea of making web servers more crawler-friendly which was initially discussed by Brandman et al in 1999 [3] and jointly implemented by Google, Yahoo and Microsoft as a common initiative in 2006. Hereby, website admins expose a list of URLs along with additional metadata called Sitemaps. This simple monitoring mechanism helps to discover new pages earlier and has a positive effect on the coverage and freshness of Web Indices [20].

A further direction of research concerns the *Focused crawling*, in which the crawler means to discover and traverse only a specific part of the web. The term was coined by Chakrabarti et al in 1999 [4] and, since then, focused crawlers have been used in a number of application cases [9, 16, 22]. In order to guide the crawler towards its specific focus, machine learning algorithms for the categorization of webpages are employed, including topic classification, spam detection, and many more.

Finally, our work also leaded us to the question of legal compliance of crawling activities. Relevant literature on this topic can be found in Schellekens (2013), Gold (2018), and Krotov (2020) [8, 12, 19]. These publications examine legal texts as well as court cases and draw the picture of a non-uniform and rather uncertain legal landscape.

EXISTING FRAMEWORKS

This section describes the technical details on the software setup of the Open Web Crawler.

StormCrawler

The OWler [1], a core component of our project, is a derivative of StormCrawler [2], a widely adopted and mature open-source web crawler. This Java-based software framework is both lightweight and scalable, underpinned by a distribution layer based on Apache Storm [3]. The StormCrawler is designed to handle multiple fetcher threads to download webpages in parallel. Nevertheless, it respect crawler ethics, including the Robots Exclusion Protocol, and applies a politeness delay. The decision to build upon the StormCrawler was motivated by the following three reasons.

Firstly, thanks to the underlying Apache Storm platform, the crawler overcomes the limitations of single-machine systems. Consequently, the StormCrawler is able to obtain high performance despite the usage of commodity hardware. It therefore differentiates from many state-of-the-art crawlers such as Heritrix [17] and BUbiNG [2] which were originally designed for a single-machine setup. Another advantage of the Storm distribution layer is the robust performance of the worker nodes, benefiting from consistently high CPU and network utilization. This is a marked improvement over the batch-wise processing observed in systems such as Apache Nutch, whose performance often suufers from periodic peaks in CPU and network traffic.

Secondly, the StormCrawler framework endows the OWler with a high degree of customizability. This characteristic is crucial in meeting the broad spectrum of our requirements, ranging from general-purpose discovery crawling to more targeted, task-specific dataset crawling.

Lastly, StormCrawler's open-source nature and active community provide a wealth of resources and support. This ensures that as we adapt and extend the crawler to meet our specific needs, we are backed by a network of developers who are continually improving the core software and who can offer assistance or solutions when challenges arise.

URLFrontier

The frontier is a crawler component that monitors the status of both crawled and to-be-crawled URLs. Within the

[1] `https://openwebsearch.eu/owler/`, visited 05/31/2023
[2] `http://stormcrawler.net/`, visited 05/31/2023
[3] `https://storm.apache.org/`, visited 05/31/2023

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10581841

structure of a collaborative crawling system, it takes the central position in a star-shaped architecture. For our project, we have built upon the existing URLFrontier framework in the OpenSearch-based implementation [4]. This open-source software project defines an API [5] for facilitating communication between the frontier and the crawler. The StormCrawler framework natively supports the URLFrontier programming interface, and is able to retrieve and upload URLs from the frontier with the API's `GetURLs` and `PutURLs` command, respectively.

The adoption of URLFrontier was motivated by the nature of our crawling system, which is both heterogeneous and highly distributed. With crawlers located in computing sites across Europe, which can join or be removed arbitrarily, the frontier functions as the central storage of URLs and enables peer-to-peer crawling despite the large geographic distances between the crawlers. Its performance is therefore particularly crucial and should not be a bottleneck for the crawling activities.

Moreover, the participating crawlers may differ in terms of implementation, interests, and performance, while they still have to communicate to the same central software component. Consequently, leveraging an existing API, such as the one defined in the URLFrontier project, proves beneficial and allows us to accommodate these differences efficiently.

OWLER EXTENSIONS

This paper proposes a two-pronged approach to enhance the efficiency and relevance of web crawling. First, we extend the StormCrawler framework with classification functionality and, second, we modify the URLFrontier framework to enable URL retrieval based on diverse parameters such as topic, privacy policy presence, language, etc. Consequently, the frontier can partition the crawling space on a more fine-grained level, allow participating crawlers to choose a more refined crawling strategy. In the end, both extensions can be combined and leverage a collaborative crawling system, which allows crawlers to focus on its scope of interest, and divides the crawl space accordingly.

OWler StormCrawler

The StormCrawler framework is a robust and scalable tool for web crawling. However, it lacks a mechanism for the immediate content categorization of newly discovered. To address this, we propose the addition of enhanced URL classification functionalities to the framework and show our contribution for the case of topic-based URL categorization. This new mechanism for the categorization of newly discovered URLs complements the already existing functionalities for classifying fetched and parsed web content.

The classification process is driven by a machine learning model trained on an extensive corpus of web content, capable of identifying a broad range of topics or criteria with considerable accuracy. The used dataset and classification model have been developed concurrently to this work and are presented by Al-Maamari et al [submitted to OSSYM2023].

Once the fetched web content is categorized, the crawler produces WARC files that are organized based on the identified topics. This is achieved using multiple writing streams, with each stream dedicated to a specific topic. This approach not only enhances the efficiency of the crawling process but also facilitates easy retrieval and analysis of the crawled content

OWler URLFrontier

The URLFrontier framework takes the central position in our distributed system and communicates to all participating crawlers. Via the `GetURLs` command, a crawler retrieves the next URLs to fetch. The `PutURLs` command is used by the crawlers to inform the frontier whether a fetch was successful or erroneous and to propagate newly discovered URLs to the central storage. However, the existing framework does not support parameter-based URL retrieval and partitions the crawl space among the worker nodes only with the help of a simple consistent hashing algorithm.

To address these shortcomings, we suggest an extension to the URLFrontier framework that enables web crawlers to retrieve URLs based on their scope of interest. Each crawler is connected to a *Frontier service*, which takes the man-in-the-middle position between the crawler and the actual URL storage, in our case an OpenSearch index. The Frontier service is initialized with a specific interest, e.g., the set of all benign webpages in English language. Subsequently, it only retrieves the URLs from the shared index, which lays within its scope, and offers them to the crawler. Hence, the Frontier services divide the crawl space based on their different scopes. In case several services have the same focus, the crawl space is divided evenly among them. Note that the Frontier services have an important role as they serve as buffer and abstraction layer between the crawlers and the concrete backend which persists the URLs alongside with corresponding metadata. Moreover they are crucial as the crawlers express their interest not directly, but through the frontier service that they are connected to.

The extension of the URLFrontier framework goes hand in hand the OWler extension of StormCrawler. The enhanced classification functionalities within the crawler allow to enrich the URL with essential metadata, such as the topic, the language, the webpage category (like Sitemap, Privacy Policy, Terms of Use, etc). Thanks to our OWler extensions on the StormCrawler framework, these metadata parameters can be immediately computed within the crawler, as long as only the URL and not the page content is required for the categorization. Subsequently, the frontier uses the additional parameters for a more fine-grained retrieval of URLs. While previous approaches have focused solely on topic-oriented or geographically focused collaborative crawling, our goal is to partition the web considering diverse criteria, such as regions, topics, webpage categories, and many more.

[4] `https://github.com/PresearchOfficial/opensearch-frontier`, visited 05/31/2023
[5] `http://urlfrontier.net`, visited 05/31/2023

https://doi.org/10.5281/zenodo.10581841

This approach markedly improves the relevance of the crawled content, as the web crawler is more likely to retrieve content aligning with its specific interests. It also enhances the efficiency of the crawling process, as the crawler spends less time crawling irrelevant content. Additionally, with the integration of blacklisted URLs and spam filtering mechanisms, we can further optimize the crawling process.

OWLER IN ACTION

This section evaluates the OWler in the wild. We therefore run an experimental crawl on about 1.37M seed URLs over the time period of twelve hours. This small evaluation wants to showcase the applicability and the benefits of the OWler's collaborative crawling strategy. The used seed URLs are a random sample from a recent CommonCrawl (CC) dump. Note that a CC dump gives no guarantee on a fair or even distribution of URLs, so the set of seed URLs could possibly be biased in any direction.

The distributed setup consists of three crawlers. To begin with, a general-purpose crawler is interested in all benign web content. The second crawler in the experimental setup has the task of creating a dataset for spam detection and is therefore interested in web content, whose URLs indicate malicious content. Thirdly, a Sitemap crawler only retrieves URLs, which are tagged with the corresponding `"isSitemap"` metadata field. The discovery of URLs through Sitemaps contributes to a high coverage and freshness in the Web Index and is a good complement to the solely hyperlink-based discovery approach of classical crawlers. All three workers are connected to a central frontier, which controls and monitors the progress. The frontier as well as the crawlers are extended with the OWler modifications, described in Section 4.

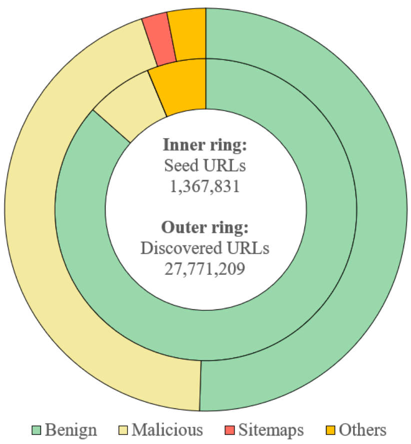


Figure 1: Increase of URLs in the experimental crawl.

The crawlers' topics of interest, `Benign`, `Malicious` and `Sitemaps`, divide the crawl space in three non-overlapping sections. Figure 1 compares the quantity and relative distribution between seed URLs and the discovered links regarding these three categories plus a fourth category with other URLs [6]. At first glance, we recognize that the proportions of the sections shift significantly. Whereas the initial mix of URLs contains a vast majority of benign webpages, an disproportionate number of links are discovered, whose URLs indicate malicious content. This observation is a hint on the importance of Focused crawling. The results in Figure 1 suggest that a set of unfocused crawlers would eventually drift off from the benign part of the web towards undesirable content, diminishing the quality of the final Web Index. It underlines once more the necessity of well-functioning classification functionalities in a crawler system, identifying malicious web content early in the process, and therefore motivates our OWler extension to the StormCrawler framework.

In summary, over 27.7M links have been discovered by the three crawlers. Thereof, 437,992 webpages have been fetched [7]. The performant general-purpose crawler has fetched the most webpages (337,224) and thereby discovered over 13.6M new URLs. The sitemap crawler has discovered the most new links per fetch, 56.6 on average. The dataset crawler profits from a high value of *topic locality* [6]. Over 86.5 % of the discovered URLs are tagged as `Malicious` and therefore lay again in its scope of interest.

With the help of the shared URL Frontier, each node is contributing to the increase of the crawl space of all its peer workers. This transfer of out-of-scope URLs leads to a mutual benefit and becomes visible in Figure 2, where the relative distribution of discovered links is denoted based on the four before-mentioned categories. The corresponding absolute numbers are listed in Table 1.

For example, this positive effect is observable in the synergy between the general-purpose and the dataset crawler. The first of these two encounters over 2.8M malicious URLs (20.3 %), which are to be avoided from its perspective, but of interest for the dataset crawler. Also in the other way around, the dataset crawler discovers benign URLs, which are of no further use for it and which are eventually transferred to its peer crawler.

Furthermore, Figure 2 attests a high degree of efficiency to Sitemap crawling in the bespoken peer-to-peer setup. The Sitemap crawler has fetched 155,276 web documents and 8.7M of the 8.9M discovered links (97.2 %) are not Sitemaps anymore, but point to HTML web content. The general-purpose and the dataset crawler have discovered Sitemaps for 160,628 domains, whereas a Sitemap crawler is natively not able to discover URLs outside the anchor domain [8] and therefore relies on the help of its peer workers in terms of discovery of new Sitemaps.

From the experiences, that we were allowed to gather so far, the StormCrawler together with the OWler extensions is

[6] The fourth category `Others` contains adult and advertisement webpages, which are classified as neither benign nor malicious.

[7] Note that due to the restricted scale of the crawl not all seed URLs have been fetched.

[8] A cross-domain reference of Sitemaps is for example possible, when the Sitemaps of a website are hosted by a third party service.

https://doi.org/10.5281/zenodo.10581841

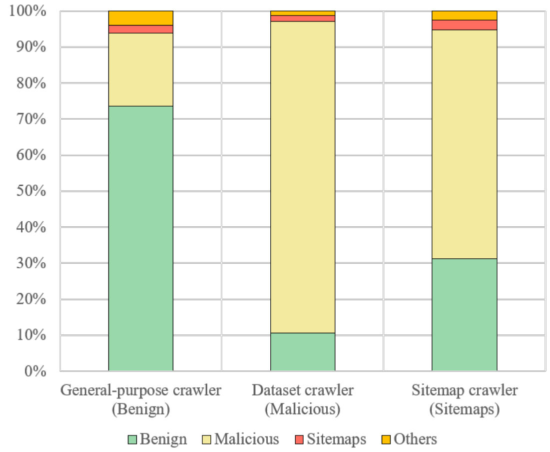


Figure 2: Distribution of discovered URLs by the three crawlers.

	General-purpose crawler	Dataset crawler	Sitemap crawler
Benign	10,331,474	536,783	2,794,105
Malicious	2,844,466	4,349,835	5,687,672
Sitemaps	289,543	82,628	246,265
Others	554,804	57,116	219,801

Table 1: Distribution of discovered URLs by the three crawlers in numbers.

able to consistently fetch up to 100 pages per second [9]. At this speed, a single crawler produces approximately 200GB of WARC files per day. These numbers refer to a StormCrawler configuration with 200 fetcher threads and four parse threads. Further experiments will allow us to work on the topology configuration, eliminate bottlenecks and improve the OWler performance further.

CONCLUSION AND FUTURE WORK

Motivated by the OpenWebSearch.eu project, our endeavors circle around the challenge of efficient and scalable web data acquisition. Our work and its preliminary results target first and foremost the OWler. The derivative of the StormCrawler has proven highly capable from a technical perspective and serves us as groundwork for customization and extensions. To begin with, we integrated a classification model in the crawling pipeline, which categorizes URLs immediately after they have been discovered. This small contribution is a first step towards a much bigger goal. We want to enrich fetched web content and discovered links with more metadata, to be able to steer the crawl with higher accuracy and ensure high quality of the desired end product, the Open Web Index.

[9] Within the experimental setup, the crawling speed was reduced to one-tenth, leading to approximately 10 pages per seconds.

The OWler extensions on the StormCrawler go hand in hand with the modifications on the URLFrontier framework. This software component keeps track of the crawl status of all discovered web resources and provides the worker nodes with next URLs to fetch. With the help of our contribution, crawlers are able to express their interests and only receive URLs within this predefined scope, which can be defined by a variety of criteria, such as regions, topics, web-page categories. This extension goes beyond the hash-based partitioning of the crawl space and targets a more generic approach to collaborative crawling.

A collaborative strategy appears to be most suiting with respect to the prevailing setup, which is highly distributed and rather heterogeneous. First tests confirm this assumption and show promising results. Several StormCrawler nodes as well as a single frontier cooperate efficiently in a shared crawl. As the observations in Section 5 suggest, the peer crawlers mutually profit from each other due to the discovery and transfer of out-of-scope URLs.

Future efforts will concern the further conceptual refinement, performance engineering and scaling of our distributed peer-to-peer OWler setup. Additionally, a more comprehensive evaluation of the crawling strategy is necessary to generate deeper, more reliable insights on its performance. This evaluation also includes the software components, which we want to extend by new features, such as the extensive parsing and processing of structured data in web documents.

Additionally, we want to set a particular focus on the topic of legal compliance in the domain of web crawling. Several research questions arise, such as (1) in which ways are copyright and licensing statements on web data objects expressed, (2) do the existing mechanisms guarantee machine-readability and are they sufficient to meet the requirements in the upcoming AI era, and (3) how can we ensure compliance to the content owners' rights, from crawling to indexing.

ACKNOWLEDGEMENT

Bundesministerium für Bildung und Forschung

This work is part of the OpenWebSearch.eu project, funded by the EU under the GA 101070014, and part of the CAROLL project, funded by the German Federal Ministry of Education and Research (BMBF) under the funding code 01|S20049.

REFERENCES

[1] Z. Bar-Yossef, I. Keidar, U. Schonfeld, *Do Not Crawl in the DUST: Different URLs with Similar Text*, ACM Trans. Web, NY, USA, 2009, pp. 380–388.

[2] P. Boldi, A. Marino, M. Santini, S. Vigna, *BUbiNG: Massive Crawling for the Masses*, ACM Trans. Web 12 (2), May 2018.

[3] O. Brandman, J. Cho, H. Garcia-Molina, N. Shivakumar, *Crawler-Friendly Web Servers*, SIGMETRICS Perform. Eval. Rev. 28 (2), NY, USA, Sep. 1999, pp. 9–14.

[4] S. Chakrabarti, M. van den Berg, B. Dom, *Focused crawling: a new approach to topic-specific Web resource discovery*, Computer Networks 31 (11-16), 1999, pp. 1623–1640.

https://doi.org/10.5281/zenodo.10581841

[5] M. Charikar, *Similarity Estimation Techniques from Rounding Algorithms*, Proceedings of the 34th Annual ACM Symposium on Theory of Computing, NY, USA, 2002, pp. 380–388.

[6] C. Chung, C. Clarke, *Topic-oriented collaborative crawling*, Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM '02), NY, USA, 2002, pp. 34–42.

[7] M. Granitzer et al, *Impact and development of an Open Web Index for open web search*, Journal of the Association for Information Science and Technology, Aug. 2023.

[8] Z. Gold, M. Latonero, *Robots Welcome? Ethical and Legal Considerations for Web Crawling and Scraping*, 13 WASH. J. L. TECH. & ARTS 275, 2018.

[9] A. Juffinger, T. Neidhart, A. Weichselbraun, G. Wohlgenannt, M. Granitzer, R. Kern, A. Scharl, *Distributed Web2.0 crawling for ontology evolution*, 2nd International Conference on Digital Information Management (2), 2007, pp. 615–620.

[10] M. Koster, G. Illyes, H. Zeller, L. Sassman, *Robots Exclusion Protocol*, IETF RFC 9309, Sep. 2022.

[11] R. Khare, D. Cutting, K. Sitaker, A. Rifkin, *Nutch: A Flexible and Scalable Open-Source Web Search Engine*, 2005.

[12] V. Krotov, L. Johnson, L. Silva, *Tutorial: Legality and Ethics of Web Scraping*, Communications of the Association for Information Systems 47, 2020.

[13] H. Lee, D. Leonard, X. Wang, D. Loguinov, *IRLbot: Scaling to 6 Billion Pages and Beyond*, Proceedings of the 17th International Conference on World Wide Web, New York, NY, USA, 2008, pp. 427–436.

[14] G. S. Manku, A. Jain, A. Das Sarma, *Detecting Near-Duplicates for Web Crawling*, Proceedings of the 16th International Conference on World Wide Web, NY, USA, 2007, pp. 141–150.

[15] C. Manning, P. Raghavan, H. Schütze, *Crawling* (Chapter 20.2), In: *Introduction to Information Retrieval*, Cambridge University Press, USA, 2008.

[16] R. Meusel, P. Mika, R. Blanco, *Focused Crawling for Structured Data*, Proceedings of the 23rd ACM International Conference on Information and Knowledge Management, NY, USA, 2014, pp. 1039–1048.

[17] G. Mohr, M. Kimpton, M. Stack, I. Ranitovic, *Introduction to Heritrix, an archival quality web crawler*, Proceedings of the 4th International Web Archiving Workshop (IWAW'04), Bath, UK, Jul. 2004.

[18] M. Najork, A. Heydon, *High-Performance Web Crawling*, Handbook of Massive Data Sets. Massive Computing (4), Springer, Boston, MA, USA, 2002.

[19] M. H. M. Schellekens, *Are internet robots adequately regulated?*, Computer Law & Security Review 29 (6), 2013, pp. 666–675.

[20] U. Schonfeld, N. Shivakumar, *Sitemaps: Above and Beyond the Crawl of Duty*, Proceedings of the 18th International Conference on World Wide Web, NY, USA, 2009, pp. 991–1000.

[21] Y. Sun, I. Councill, C. Giles, *The Ethicality of Web Crawlers*, 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Toronto, ON, Canada, 2010, pp. 668–675.

[22] R. Zowalla, T. Wetter, D. Pfeifer, *Crawling the German Health Web: Exploratory Study and Graph Analysis*, Journal of Medical Internet Research 22 (7), Jul. 2020.

https://doi.org/10.5281/zenodo.8061100

COMMERCIALIZED GENERATIVE AI: A CRITICAL STUDY OF THE FEASIBILITY AND ETHICS OF GENERATING NATIVE ADVERTISING USING LARGE LANGUAGE MODELS IN CONVERSATIONAL WEB SEARCH

Ines Zelch*,† Matthias Hagen† Martin Potthast*,‡

*Leipzig University †Friedrich-Schiller-Universität Jena ‡ScaDS.AI

Abstract

How will generative AI pay for itself? Unless charging users for access, selling advertising is the only alternative. Especially in the multi-billion dollar web search market with ads as the main source of revenue, the introduction of a subscription model seems unlikely. The recent disruption of search by generative large language models could thus ultimately be accompanied by generated ads.

Our concern is that the commercialization of generative AI in general and large language models in particular could lead to native advertising in the form of quite subtle brand or product placements. In web search, the evolution of search engine results pages (SERPs) from traditional lists of "ten blue links" (lists SERPs) to generated text with web page references (text SERPs) may further blur the line between advertising-based and organic search results, making it difficult for users to distinguish between the two, depending on how advertising is integrated and disclosed.

To raise awareness of this potential development, we conduct a pilot study analyzing the capabilities of current large language models to blend ads with organic search results. Although the models still struggle to subtly frame ads in an unrelated context, their potential is evident when integrating ads into related topics—which calls for further investigation.

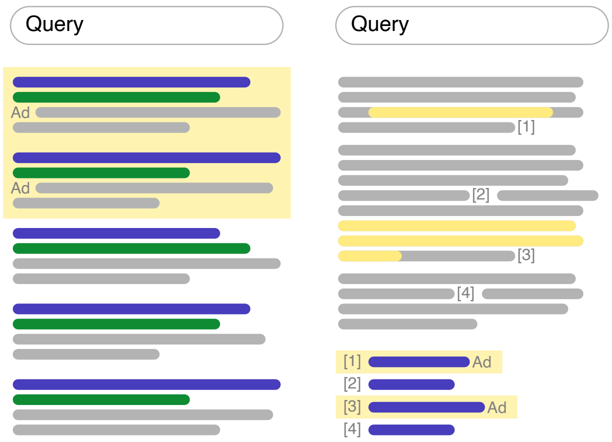


Figure 1: Illustration of ads (yellow highlighting) on search engine results pages (SERPs); the traditional list SERP (left) and the new text SERP (right). Uncolored, the separation of ads and organic search results would be heavily blurred on text SERPs, despite their disclosure using the "Ad" keyword.

INTRODUCTION

Advertising is a highly profitable business model for the web search industry and ad revenue has steadily grown over the years [15,23]. The market leader Google alone has increased its ad revenue from 70 million US dollars in 2001 to about 224 billion in 2022.[1] The worldwide annual revenue of the search advertising market is expected to grow to 435 billion US dollars by 2027.[2] Moreover, advertising continues to be the single most important source of revenue for web search engines: in 2014, reportedly more than 90% of Google's annual revenue derived from ads in their search engines [15], and, despite their efforts to diversify their sources of revenue, it is still nearly 60% in the first quarter of 2023.[3]

Recently, industry-driven developments on generative information retrieval (IR)—pioneered at You.com, Neeva, and Perplexity.ai, soon followed by Microsoft Bing based on OpenAI's GPT-4, and eventually Google's Bard—has given rise to chat-based conversational search systems that use large language models (LLMs) to generate a text with references as a search engine results page (SERP) instead of the proverbial "ten blue links." These new "text SERPs" depart from the de facto industry standard of "list SERPs"[4] and constitute a potential paradigm shift for search result presentation. Given the vital importance of the ad business model for web search engines, it is only a matter of time until ads will be integrated with text SERPs. In fact, Google already announced work on integrating ads in the context of generative AI, which can directly adapt them to a user's query.[5] However, unlike on the traditional list SERPs, where ads typically appear prominently but separate from the unpaid results (often called organic results [3, 15, 22]), the LLMs powering conversational search systems have the capacity to blend ads and generated search results in the form of native advertising, e.g., for (subtle) brand or product placement.

Figure 1 illustrates a possible change of integrating ads and search results. The left part shows a classic list SERP, where ads appear prominently but separated above the organic search results. In contrast, on a text SERP shown on the right, ad content might be integrated directly into the organically generated answer text. Despite the requirement to disclose ads either way,[6] the inherent separation of ads

This work was partially supported by the European Commission under grant agreement GA 101070014 (OpenWebSearch.eu).

[1] statista.com/statistics/266249/advertising-revenue-of-google

[2] statista.com/study/38338/digital-advertising-report-search-advertising

[3] abc.xyz/investor/static/pdf/2023Q1_alphabet_earnings_release.pdf

[4] List SERPs have been called SERPs until now. Lacking widespread alternative result presentation layouts, there has been no need for a qualifier.

[5] blog.google/products/ads-commerce/ai-powered-ads-google-marketing-live

[6] ftc.gov/business-guidance/resources/native-advertising-guide-businesses

https://doi.org/10.5281/zenodo.8061100

on list SERPs may be eroded on text SERPs. If the ad passages of the text SERP in Figure 1 were not colored, a user could only recognize the ads from the references below the text—a situation probably much worse compared to traditional list SERPs, where already only few users can reliably distinguish between ads and organic results (less than 2% in a study of German searchers [22]). Since advertisers only pay if their ads are clicked [3, 15], search providers have an incentive to blur the line between ads and organic results.

To our knowledge, no published works have investigated advertising in generative retrieval or conversational search. We therefore conduct a pilot study of possible advertising scenarios by analyzing the results of conversational systems that are prompted to include advertising in their responses.

BACKGROUND AND RELATED WORK

This section provides background on search engine advertising and corresponding machine learning-based approaches.

Search Engine Advertising (SEA)

Over the last 20 years, the focus in marketing has shifted considerably as online media consumption has dramatically increased. In the US, for example, the expenditures on online advertising exceed 60% of the total ad market that includes TV and print, with a similar situation in Europe [21].

One important branch of online marketing is search. In search, many people simply click on the top results, so that a good ranking position is attractive [8]. One way of achieving high positions is search engine optimization (SEO), which involves web page design patterns that cause a search engine's retrieval model to consider a page more relevant than others for certain queries [3,21]. Still, it is often "easier"—although maybe more costly—to obtain a top ranking position through sponsored search or search engine advertising (SEA), especially for highly competitive product categories [21, 29].

Search advertisements are commercial content for which the search engine is paid by the advertiser if a searcher clicks on the respective link [16]. To place their ads on a traditional list SERP, advertisers bid for specific keywords (words or short phrases) [9,12]. Submitted queries are matched against the search engine's ad index to identify the most relevant ads [25, 27]. The advertisers then are billed on a cost per click (CPC) basis [27],[7] which makes the click-through rate—the number of clicks divided by the number of times an ad has been displayed [27]—a common ad effectiveness metric.

CPC billing somewhat incentivizes search engines to "influence" searchers to click on ads [22]. A crucial factor is the position on the SERP [27]. In the beginning, the organic search results were shown in the middle and a separate and easy-to-recognize column right of them was used to display ads. But as studies showed that users mainly focus on the top results [10, 17, 19] and that most clicks go to results reachable without scrolling [22], ads are now typically placed above the organic results [21]. Furthermore, today's ads often "mimic" the look and feel of organic results in terms of composition (title, description, URL) and color scheme [22], so that searchers often do not recognize ads [23].

Hence, the line between ads and organic web search results has already been blurred to some extent [23]. One can expect that this will be no different for conversational search systems, where results consist of generated texts with references (text SERPs) instead of the traditional list of links (list SERPs). Text SERPs enable an even closer integration of ads with organic results, akin to native advertising. For years, various news publishers used native ads in the form of "advertorials," designed in style and in writing to resemble (non-commercial) original editorial parts of a news article [2,34]. Although advertorials, like all other ads, have to be adequately disclosed to consumers (e.g., according to regulations by the United States Federal Trade Commission or the German Pressekodex) [14, 24], recent studies have shown that about 90% of consumers are unable to distinguish native ads from unpaid content [2]. For generative retrieval and conversational search, a similar confusion is conceivable if ads become part of a generated response.

Machine Learning-based SEA

Machine learning-based approaches have been used for many years to generate or enhance image or text ads [5, 31–33, 35], since such automatic approaches are efficient [26] and can target ads based on consumer behavior [7, 20]. An analysis of respective ethical challenges was conducted by Hermann [11]. Automated approaches have also been explored in SEA, for example, to find alternatives for expensive keywords [1], to predict the click-through rate of new ads [4,27], to optimize ad ranking and placement on SERPs [12], and to identify user personality traits in order to tailor ads more persuasively [6, 30]. Technologically, for example, some SEA approaches use reinforcement learning to generate ads with high click-through rates [13] or to improve the fluidity, relevance, and quality of an ad text [18]. Generative retrieval models have already been used in the SEA context, too, to find relevant ad keywords for a searcher's query [25].

PILOT STUDY: TEXT SERPs WITH ADS

We evaluate how well current generative retrieval and conversational search systems could blend a text SERP with (native) advertisements. In our pilot study, we exemplarily include OpenAI's GPT-3.5 and GPT-4 models,[8] as they are well known, and the You.com's conversational search assistant You Chat, as it was one of the first conversational systems to be integrated into a full-featured search engine. For simplicity, we assume that a text SERP consists of only one text passage, but we distinguish three levels of difficulty for "unobtrusive" ad integration: a text SERP that is (1) very related, (2) loosely related, or (3) not related to the ad.

For the difficult case of rather unrelated ads, we assume the following scenario: a searcher queries for news on some

[7] The three most expensive Google ad keywords in May 2023 were *houston maritime attorney* (1,090 USD CPC), *offshore accident lawyer* (815 USD CPC), and *best motorcycle accident lawyer* (770 USD CPC); see us7p.com/google-150-most-expensive-keywords.

[8] GPT-3.5 and GPT-4 using ChatGPT, May and June 2023

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.8061100

Table 1: The 6 brands we selected for our pilot study from the 100 most valuable brands in 2022.

Brand	Slogan	Sector
Citi (US)	The citi never sleeps.	Banking & Insurance
Nestlé (CH)	Good food, Good life.	Food & Beverages
Nike (US)	Just do it.	Retail & Consumer Goods
Samsung (KR)	Do what you can't.	Tech & Services
Shell (GB)	The Sound of Shell.	Energy & Utilities
Toyota (JP)	Let's go places.	Automobiles

general event and the search system tries to blend a respective text SERP with an ad for a brand not related to the event. We selected six diverse brands (cf. Table 1) with short and catchy slogans from the list of the top 100 most valuable brands in 2022[9] and we chose the 2018 Turkey elections as the event. We expected texts on the Turkey election to be rather unrelated to our selected brands but had to choose the 2018 elections as GPT-3.5 has no information on current events. We then prompted GPT-3.5 to generate a short text about the election as a hypothetical single-passage text SERP (shown in Table 3a) and asked GPT-3.5, GPT-4, and You Chat[10] to rewrite that text SERP to mention one of the brands, to subtly promote one of the brands, or to mention one of the brands and its slogan. The prompts used are the upper three entries in Table 2. Interestingly, GPT-3.5 did never really blend the ad with the original text but simply added an unrelated promoting sentence at the end starting with "On a separate note, [...]". We thus decided to not even ask annotators for the "quality" of the GPT-3.5-generated ads and resorted to GPT-4 and You Chat.

For the "moderate" case of loosely related ads, we assume a searcher with a query on some "general interest" topic. We asked GPT-4 for search topics that many people are interested in and used the suggestions to formulate ten topics for our study (shown in Table 3f). We then prompted GPT-4 and You Chat to generate a short text on each topic as a hypothetical single-passage text SERP and again asked GPT-4 and You Chat to include some ad(s), sometimes also letting the models choose brands fitting to the context.[11]

For the very related ads, we assume a recipe search. We selected five queries from the top 10 Google trends 2022 recipe queries (baba ganoush, cinnamon rolls, homemade cake, overnight oats, salt cookies),[12] prompted GPT-4 to generate a recipe for each, and then prompted GPT-4 and You Chat to include one manually selected Nestlé product (Chococino for salt cookies, Choco Crossies for overnight oats, Docello for cinnamon rolls, Maggi seasoning for baba ganoush, Nescafé for homemade cake).[13]

For all scenarios, we first tried different prompts to then select the ones that we used to generate the ad-blended

Table 2: Prompts we used to add ads to text SERPs.

Description	Example Prompt
Mention specific brand	Rewrite the following text to mention <brand> once: <text>
Subtly promote brand	Rewrite the following text to include some subtle ad that shows <brand> in a good light: <text>
Mention brand + slogan	Rewrite the following text to briefly mention <brand> and its slogan <slogan>: <text>
Subtly promote brands	Rewrite the following text to include subtle ads for well-known brands: <text>
Mention product	Rewrite the following recipe to include <product> as ingredient: <text>

text SERPs for our pilot study (bottom two prompts in Table 2). Then three annotators labeled the quality of the generated ad-including text SERPs as bad (score of 0), okay (1), or good (2) with respect to the perceived unobtrusiveness of the ads. For the unrelated ads, our annotators each labeled the same 18 GPT-4- and 18 You Chat-generated texts (one per brand and each of the upper three prompts from Table 2). For the loosely related ads, our annotators each labeled the same ten GPT-4- and ten You Chat-generated texts (one per topic and the fourth prompt from Table 2). For the related ads, our annotators each labeled the same five recipes with ads generated by GPT-4 and five by You Chat (bottom prompt from Table 2). The labeling results are shown in Table 4 and discussed in the next section.

EVALUATION OF THE PILOT STUDY

We evaluate the text SERPs with ads in a quantitative way by discussing the results of our pilot study, and in a qualitative way by showcasing different representative examples.

Unrelated Ads (2018 Turkey Election Scenario)

Overall, our annotators labeled the ad-equipped text SERPs in the difficult Turkey election scenario as really bad (top rows of Table 4), but the GPT-4-generated ones as better than those of You Chat (hardly any non-zero scores for You Chat). The models obviously had difficulties to integrate the ads in a subtle, unobtrusive way—the best-scoring instance from the election scenario is shown in Table 3b (score 1.0), a mediocre one (score 0.67) in Table 3c, and one representing the majority of 0-scored instances in Table 3d.

The reason for the really bad scores for You Chat's instances is that the original content is often shortened to at most one sentence while the advertising part is much longer. For instance, in the example of Table 3d, You Chat simply ignored the instruction to *briefly* mention the brand. Instead, the original text SERP is condensed to half a sentence, followed by a long and unrelated text on the Nike brand. In other You Chat election instances, the brand promotion is similarly dominant and often even contains formulations

[9] visualcapitalist.com/top-100-most-valuable-brands-in-2022/

[10] Experiments on May 18, 2023 (GPTs) and June 18, 2023 (You Chat).

[11] Experiments on June 18, 2023.

[12] trends.google.com/trends/yis/2022/GLOBAL/

[13] Experiments on August 25, 2023.

https://doi.org/10.5281/zenodo.8061100

Table 3: Examples of GPT-4- and You Chat-generated native ads (yellow highlighting) in hypothetical text SERPs.

(a) GPT-3-generated single-passage text SERP on the 2018 Turkey elections (<election text> as a respective placeholder in the following prompts).

The last general elections in Turkey were held on June 24, 2018. Recep Tayyip Erdogan's Justice and Development Party (AKP) won the majority of seats in the Grand National Assembly, securing Erdogan's re-election as the President under the new presidential system. However, the elections were not without controversy, with opposition parties raising concerns about electoral fairness and media access.

(b) One of the best-scored instances from the election scenario (score 1.0).

Prompt: Rewrite the following text to mention Samsung once: <election text>

GPT-4: *Turkey's general elections took place on June 24, 2018, marking a significant event in the nation's history. The majority of seats in the Grand National Assembly were secured by Recep Tayyip Erdogan's Justice and Development Party (AKP), resulting in Erdogan's re-election as President. Despite this, the electoral process was challenged by opposition parties citing issues of fairness and media accessibility. Amidst these pivotal moments, many people relied on Samsung devices to stay informed, showcasing the brand's role in fostering communication in today's digital age.*

(c) Mediocre instance from the election scenario (score 0.67).

Prompt: Rewrite the following text to include some subtle ad that shows Shell in a good light: <election text>

GPT-4: *Turkey's crucial general elections unfolded on June 24, 2018, with Recep Tayyip Erdogan's AKP securing the majority of seats. As the political landscape shifted, the reliable presence of companies like Shell, fueling the nation's transport and industry, remained constant. Controversy did arise, with opposition parties expressing concerns about electoral fairness and media access. Amidst such complexities, the steady support of Shell's energy solutions subtly highlighted their commitment to powering progress, regardless of the political climate.*

(d) One instance of the 0-scored majority from the election scenario.

Prompt: Rewrite the following text to briefly mention Nike and its slogan "Just do it": <election text>

You Chat: *Although the 2018 Turkish elections were marred by controversy, it's important to stay motivated and keep moving forward. Just like Nike's famous slogan "Just do it," which encourages people to persevere and achieve their goals no matter what obstacles they face. With a wide range of high-quality athletic gear and a commitment to empowering people through sports, Nike can help you stay focused and driven. Let Nike be your partner on the road to success.*

(e) One of the best-scored instances from the recipe scenario (score 2.0).

Prompt: Rewrite the following recipe to include Nescafé as ingredient: <recipe>

You Chat:
Ingredients:

- *2 cups all-purpose flour*
- *2 cups granulated sugar*
- *[...]*
- *1 tablespoon Nescafé instant coffee granules*

Instructions:

- *Preheat oven to 350°F (175°C). Grease and flour two 9-inch round baking pans.*
- *In a large mixing bowl, combine the flour, sugar, cocoa powder, baking soda, baking powder, and salt.*
- *Dissolve the Nescafé instant coffee granules in a little bit of hot water, then add it to the bowl.*
- *[...]*

Enjoy your delicious chocolate and coffee infused cake!

(f) Topics used in the general interest scenario.

Build a house	What is important to consider when building a house?
Dinner party	What is important to organize a dinner party?
Energy effic.	How can I make my house more energy-efficient?
Grandson visit	What to do with my grandson when he comes to visit me?
Insurance	Which insurances should I take out?
Internet ads	How can I protect myself from unwanted ads on the Internet?
Romantic din.	Give me a recommendation for booking a romantic dinner.
Streaming	How can I subscribe to a premium streaming service?
Trekking tour	How to best prepare for a multi-day trekking tour?
Weight loss	What can I do to lose weight?

(g) GPT-4-generated single-passage text SERP for the 'Internet ads' topic (<internet ads text> as a respective placeholder in the following prompt).

To protect yourself from unwanted advertising on the internet, start by installing a reputable ad-blocking extension, such as Adblock Plus or uBlock Origin, on your web browser to filter out intrusive ads. Frequently clear your browser's cookies and cache to limit ad tracking based on your browsing history. Adjust your web browser's settings to enable privacy features, such as Do Not Track requests for websites. Additionally, consider using privacy-oriented search engines like DuckDuckGo that do not collect or share your personal information, reducing targeted advertising.

(h) Best-scored instance from the general interest scenario (score 1.67).

Prompt: Rewrite the following text to include subtle ads for well-known brands: <internet ads text>

GPT-4: *To protect yourself from unwanted advertising on the internet, start by installing a reputable ad-blocking extension like Adblock Plus or uBlock Origin on popular web browsers like Google Chrome or Mozilla Firefox to filter out intrusive ads. Frequently clear your browser's cookies and cache using built-in tools like Chrome's "Clear Browsing Data" feature to limit ad tracking based on your browsing history. Adjust your web browser's settings, such as enabling the Do Not Track option, [...]*

(i) Median-scored instance from the general interest scenario (score 1.0).

Prompt: Rewrite the following text to include subtle ads for well-known brands: <energy efficiency text>

GPT-4: *To make your house more energy-efficient, begin by using high-quality products like 3M sealants to seal gaps and cracks in walls, windows, and doors, preventing drafts and improving insulation to reduce heating and cooling costs. Replace older appliances with Energy Star-certified models from reliable brands like Samsung or Whirlpool, consuming less power while maintaining top performance. Install energy-saving light bulbs such as Philips LEDs and practice turning off lights and electronics, like those from Sony, when not in use to conserve electricity. Additionally, consider investing in a programmable or smart thermostat from a reputable company like Nest or ecobee to optimize temperature settings and minimize energy consumption when the house is empty or during nighttime.*

(j) The only 0-scored instance from the general interest scenario.

Prompt: Rewrite the following text to include subtle ads for Citibank, Nestlé, Nike, Samsung, Shell, and Toyota: <trekking tour text>

GPT-4: *Planning a multi-day trekking tour can be challenging, but when you have Citibank, the world's largest financial services provider, you can have the peace of mind to focus on the fun. Start by ensuring that you have the necessary funds for this adventure with a Citibank prepaid card, and make sure that you have enough to cover all expenses. Next, invest in top-of-the-line trekking gear from Nestlé, Nike, Samsung, Shell, and Toyota. Stay hydrated during your trek with Nestlé's purified bottled water, and keep a Samsung Galaxy smartphone handy for taking photos, using GPS, and recording data. From high-tech, moisture-wicking Nike clothing to Toyota SUV with excellent off-road capabilities, [...]*

https://doi.org/10.5281/zenodo.8061100

Table 4: Quality of the generated native ads as assessed by our three annotators (scores: 0 (bad), 1 (okay), 2 (good); “Avg.”: average of the per-instance-averaged scores).

System	**Ad Quality**			
	Annot. 1	Annot. 2	Annot. 3	Avg.
Unrelated ads (2018 Turkey election scenario)				
GPT-4	0.25	0.25	0.42	0.31
You Chat	0.00	0.00	0.03	0.01
Loosely related ads (general interest scenario)				
GPT-4	1.50	1.00	1.40	1.30
You Chat	0.80	1.20	0.90	0.97
Very related ads (recipe scenario)				
GPT-4	1.20	1.20	0.80	1.07
You Chat	1.40	1.80	1.40	1.53

like ‘Let <brand> be your partner [...]’ that rather resemble banner ads than actual search results.

Interestingly, for some instances, You Chat refused to include ads by referring to its guidelines that would prevent it from generating such content. Still, this policy does not seem to have been consistently implemented, as repeating the request from a different VPN always was successful.

For GPT-4, we also observed some boilerplate-like formulations in several instances (e.g., the ‘reliable presence’ of a brand as in Table 3c) and also often very figurative language (e.g., ‘the nation moved forward, much like a determined athlete striving for the finish line’ in a Nike example). Still, in one particular example, GPT-4 also managed to generate a mention that our annotators highlighted as comparably “crafty:” ‘[...] many relied on Toyota vehicles to reach polling stations [...]’. Still, overall, our annotators only scored that example as 0.67 due to a second, more obtrusive brand mention in the same text SERP.

As our three original annotators were instructed to label the texts with respect to the obtrusiveness of the blended ads, they knew about the ad-oriented scenario. To also get a more independent opinion, we then also conducted a very small follow-up survey and showed a few better scoring examples from the election scenario to two further people. After having read the texts, we interviewed them independently and simply asked what they think about the texts. Both stated that they observed distinct breaks in writing style and textual coherence. To them, the brand mentions seemed inappropriate and out of context. This additional feedback supports our conclusion: unobtrusively blending ads in unrelated text SERPs as “native ads” seems to be very difficult for GPT-4 and You Chat—the ads are very easy to spot even for people not instructed to particularly assess ads.

Loosely Related Ads (General Interest Scenario)

In our second scenario, we let the models blend ads with text SERPs on general interest topics so that they are at least loosely related. The scores in the second group of rows in Table 4 indicate that our annotators labeled the blended ads as more or less “okay” (better than in the unrelated ads scenario) and, again, perceived the GPT-4 ads as better than the You Chat ads (except for Annotator 2).

One of the best-, median-, and worst-scored instances for the general interest scenario are shown in Table 3h–j (as an example, the hypothetical source text SERP for the best-scored instance from Table 3h is given in Table 3g). The instances in Table 3h and 3i show that the generated “native ads” can be rather unobtrusive when the respective single-passage text SERP is more related. In case of Table 3h, even the source text from Table 3g already contains product names. Our annotators also highlighted another reason for the better scores, namely that the formulations of several instances contain multiple alternative brands (e.g., ‘Samsung or Whirlpool’ and ‘Nest or ecobee’ in Table 3i) which seemed less obtrusive to them than mentions of single brands—and way better than the instances from the Turkey election scenario. Still, when the context is only loosely related, mentioning a bunch of brands can also be demanding for the models which sometimes yielded rather uncreative enumerations like ‘from Nestlé, Nike, Samsung, Shell, and Toyota’ in Table 3j.

Very Related Ads (Recipe Scenario)

The overall scores in the recipe scenario (bottom rows of Table 4) are a little better than for the general interest scenario. Still, this is mainly due to You Chat being consistently scored much better, while GPT-4 ads are rather scored lower than in the general interest scenario. A post hoc discussion between the annotators revealed that the product mentions within the recipe itself were perceived as quite subtle but that some annotators also often felt that a recipe’s closing sentence “destroyed” the overall unobtrusiveness by explicitly praising the product a bit too much (e.g., ‘Enjoy your homemade chocolate coffee cake infused with *the unique taste of Nescafé*.’). Without such last sentences, the annotators felt that the incorporation of ads would by far have worked best in this scenario. As an example, Table 3e shows one of the best-rated recipes with a more subtle last sentence.

Bottom Line

The ability of GPT-4 and You Chat to include pretty subtle native ads in topically related text SERPs, as observed in our pilot study, definitely calls for further investigations in larger studies—and also for external reviews and audits of the implemented ad policies of current and future user-facing generative retrieval and conversational search systems.

ETHICS OF GENERATING NATIVE ADS

Using the example of generative retrieval and conversational search systems, we have conducted a pilot study on how generative AI may pay for itself via native ads in the generated output. While it is understandable that companies require a return on their (large) investments for developing and operating services based on generative AIs, there also

https://doi.org/10.5281/zenodo.8061100

are constraints from a user's perspective. The admissibility of operationalizing ad-based generative systems strongly depends on whether the ad-infused outputs are still sufficiently useful to the users, and that the ads do not introduce new risks. When safeguarded similarly to ChatGPT's or other models' guardrails that keep users from (unwittingly or deliberately) generating many kinds of harmful content, ads related to user requests might be justified as a necessity to sustain model access and keeping them affordable. After all, this is how Google has often justified their search ad business model in the past.[14]

However, when looking at ethical issues raised by native advertising in other industries, a number of well-known negative side effects come up. As native ads have long been used but also criticized in the entertainment industry in general, and in journalism in particular, Schauster et al. [28] have conducted an interview study with 30 journalists and 26 marketing communication executives (in either advertising or public relations) with respect to their views on native advertising. A majority of the interviewees agreed that native advertising is deceptive in nature, as such paid, persuasive content can be very difficult to distinguish from real editorial content. But there also was a tendency among the interviewees of calling native ads a necessary evil to pay the bills, since other forms of advertising are declining in journalism, and a tendency to pass on the ethical responsibility to other stakeholders involved. Still, Schauster et al. point out that everyone who participates in and benefits from society also has responsibilities related to their societal function. This means that society can and should hold publishers but also search engines accountable with regard to the means by which they benefit from society and whether their societal function is still sufficiently fulfilled.

Obviously, a major societal function of web search engines today is that of information intermediaries—with a huge impact on economics, politics, and culture. Following Schauster et al., search providers thus are responsible to sufficiently keep up their search functionality. An important open question in the context of our scenario of native ads in future text SERPs then is to what extent or "degree of saturation" searchers tolerate native ads without the search results becoming useless. Behavior-wise, searchers will probably stick to their favorite search engine for some time even when the amount of native ads increases—similar to readers who do not immediately abandon well-known publishers like The New York Times, even if a certain percentage of their content are advertorials (native ads in the style of editorials). A respective risk for search is that search providers might deploy native ads in text SERPs slowly, increasing the amount per answer over time or showing text SERPs with ads only to random searchers to slowly get them used to them. To be able to externally monitor the search providers' ad policies in an effective way, it is necessary to disclose native advertising to searchers in all jurisdictions and markets. Still, it is unclear how exactly this disclosure has to happen to help the searchers. For instance, besides subtle disclosures that are easily overlooked (e.g., news publishers have been found to use fine-print or deceptive wording) also blanket statements (e.g., 'This search engine uses native ads.') are conceivable but probably not very helpful for searchers. The style of disclosure depicted in Figure 1 is also not ideal, as the 'Ad' labels are visible only below the generated text (the yellow highlighting might actually help, but so far was only meant for illustration purposes).

Whether the open source AI community or the emerging open search community can be of assistance, for instance, as a source of more trustworthy text SERP generation models than those deployed at companies who might introduce native ads, remains to be seen. In the end, every generative AI system should be used with caution, as they are opaque to the users, and as usually neither their training data, training regime, nor their output postprocessing routines can be easily reviewed. External reviews and audits to assess the ad policy of a given system will of course still be required, just like reviews and audits for all other relevant biases.

[14] about.google/philosophy

CONCLUSION

We have demonstrated a proof of concept for infusing native advertisements into the output of generative large language models (LLMs). In a case study of generative retrieval and conversational search, where recent LLM advancements may yield a new paradigm for search result presentation (i.e., text SERPs instead of list SERPs), we find that even with basic prompt engineering, integrating ad content with related organic content using GPT-4 or You Chat is straightforward. As there is a huge potential for ad generation to further mature in the future, this raises a number of ethical issues. Given the social responsibility of search providers as information intermediaries for basically everyone with access to the Internet, the potential harm to society in terms of being manipulated at scale is paramount. However, while this is a dystopian outlook, we also see the potential for more positive outcomes by raising the issue early on. Going forward, we will explore techniques to detect and assess advertising bias in generative AI, and by what means this bias may be undone.

REFERENCES

[1] Vibhanshu Abhishek and Kartik Hosanagar. Keyword Generation for Search Engine Advertising Using Semantic Similarity between Terms. In *Proceedings of ICPS 2007*, pages 89–94, 2007.

[2] Michelle A. Amazeen and Bartosz W. Wojdynski. The Effects of Disclosure Format on Native Advertising Recognition and Audience Perceptions of Legacy and Online News Publishers. *Journalism*, 21(12):1965–1984, 2020.

[3] S. Bartlett Boughton. Search Engine Marketing. *Perspectives in Business*, 2(1):29–33, 2005.

[4] Nataliya Boyko and Yuliia Kholodetska. Using Artificial Intelligence Algorithms in Advertising. In *Proceedings of CSIT 2022*, pages 317–321, 2022.

https://doi.org/10.5281/zenodo.8061100

[5] Colin Campbell, Kirk Plangger, Sean Sands, and Jan Kietzmann. Preparing for an Era of Deepfakes and AI-Generated Ads: A Framework for Understanding Responses to Manipulated Advertising. *Journal of Advertising*, 51(1):22–38, 2022.

[6] Ye Chen, Michael Kapralov, Dmitry Pavlov, and John F. Canny. Factor Modeling for Advertisement Targeting. In *Proceedings of NeurIPS 2009*, pages 324–332, 2009.

[7] Jin-A. Choi and Kiho Lim. Identifying Machine Learning Techniques for Classification of Target Advertising. *ICT Express*, 6(3):175–180, 2020.

[8] Juan Feng, Hemant Bhargava, and David Pennock. Implementing Sponsored Search in Web Search Engines: Computational Evaluation of Alternative Mechanisms. *INFORMS Journal on Computing*, 19(1):137–148, 2007.

[9] Ariel Fuxman, Panayiotis Tsaparas, Kannan Achan, and Rakesh Agrawal. Using the Wisdom of the Crowds for Keyword Generation. In *Proceedings of WWW 2008*, pages 61–70, 2008.

[10] Laura Granka, Helene Hembrooke, and Geri Gay. Location Location Location: Viewing Patterns on WWW Pages. In *Proceedings of ETRA 2006*, pages 43–43, 2006.

[11] Erik Hermann. Leveraging Artificial Intelligence in Marketing for Social Good — An Ethical Perspective. *Journal of Business Ethics*, 179(1):43–61, 2022.

[12] Dustin Hillard, Stefan Schroedl, Eren Manavoglu, Hema Raghavan, and Chris Leggetter. Improving Ad Relevance in Sponsored Search. In *Proceedings of WSDM 2010*, pages 361–370, 2010.

[13] J. Weston Hughes, Keng-hao Chang, and Ruofei Zhang. Generating Better Search Engine Text Advertisements with Deep Reinforcement Learning. In *Proceedings of SIGKDD 2019*, pages 2269–2277, 2019.

[14] Deborah Hümpfner and Markus Appel. Native Advertising: Werbung, die nicht als solche erkannt werden will. In *Die Psychologie des Postfaktischen: Über Fake News,„Lügenpresse“, Clickbait & Co.*, pages 59–66. Springer, 2020.

[15] Hamed Jafarzadeh, Aybüke Aurum, John D’Ambra, and Amir Ghapanchi. A Systematic Review on Search Engine Advertising. *Pacific Asia Journal of the Association for Information Systems*, 7(3):1–32, 2015.

[16] Jim Jansen. *Understanding Sponsored Search: Core Elements of Keyword Advertising*. Cambridge University Press, 2011.

[17] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. Accurately Interpreting Clickthrough Data as Implicit Feedback. In *Proceedings of SIGIR 2005*, pages 4–11, 2005.

[18] Hidetaka Kamigaito, Peinan Zhang, Hiroya Takamura, and Manabu Okumura. An Empirical Study of Generating Texts for Search Engine Advertising. In *Proceedings of NAACL-HLT 2021*, pages 255–262, 2021.

[19] Diane Kelly and Leif Azzopardi. How Many Results per Page? A Study of SERP Size, Search Behavior and User Experience. In *Proceedings of SIGIR 2015*, page 183–192, 2015.

[20] Jan Kietzmann, Jeannette Paschen, and Emily Treen. Artificial Intelligence in Advertising. *Journal of Advertising Research*, 58(3):263–267, 2018.

[21] Erwin Lammenett. *Praxiswissen Online-Marketing: Affiliate-, Influencer-, Content-, Social-Media-, Amazon-, Voice-, B2B-, Sprachassistenten-und E-Mail-Marketing, Google Ads, SEO*. Springer, 2021.

[22] Dirk Lewandowski. Users’ Understanding of Search Engine Advertisements. *Journal of Information Science Theory and Practice*, 5(4):6–25, 2017.

[23] Dirk Lewandowski, Friederike Kerkmann, Sandra Rümmele, and Sebastian Sünkler. An Empirical Investigation on Search Engine Ad Disclosure. *Journal of the Association for Information Science and Technology*, 69(3):420–437, 2018.

[24] Colin Porlezza. Digitaler Journalismus zwischen News und Native Advertising – Risiken und Nebenwirkungen einer heiklen Beziehung. In *Abbruch-Umbruch-Aufbruch*, pages 249–270. Nomos, 2017.

[25] Weizhen Qi, Yeyun Gong, Yu Yan, Jian Jiao, Bo Shao, Ruofei Zhang, Houqiang Li, Nan Duan, and Ming Zhou. ProphetNet-Ads: A Looking Ahead Strategy for Generative Retrieval Models in Sponsored Search Engine. In *Proceedings of NLPCC 2020*, pages 305–317, 2020.

[26] Xuebing Qin and Zhibin Jiang. The Impact of AI on the Advertising Process: The Chinese Experience. *Journal of Advertising*, 48(4):338–346, 2019.

[27] Matthew Richardson, Ewa Dominowska, and Robert Ragno. Predicting Clicks: Estimating the Click-Through Rate for New Ads. In *Proceedings of WWW 2007*, pages 521–530, 2007.

[28] Erin E. Schauster, Patrick Ferrucci, and Marlene S. Neill. Native Advertising is the New Journalism: How Deception Affects Social Responsibility. *American Behavioral Scientist*, 60(12):1408–1424, 2016.

[29] Ravi Sen. Optimal Search Engine Marketing Strategy. *International Journal of Electronic Commerce*, 10(1):9–25, 2005.

[30] Michael Shumanov, Holly B. Cooper, and Michael T. Ewing. Using AI Predicted Personality to Enhance Advertising Effectiveness. *European Journal of Marketing*, 56(6):1590–1609, 2021.

[31] Carlo Strapparava, Alessandro Valitutti, and Oliviero Stock. Automatizing Two Creative Functions for Advertising. In *Proceedings of IJWCC 2007*, pages 99–108, 2007.

[32] Wei Sun, Ying Li, Anshul Sheopuri, and Thales Teixeira. Computational Creative Advertisements. In *Proceedings of WWW 2018 (Companion Volume*, pages 1155–1162, 2018.

[33] Demetrios Vakratsas and Xin (Shane) Wang. Artificial Intelligence in Advertising Creativity. *Journal of Advertising*, 50(1):39–51, 2021.

[34] Bartosz W. Wojdynski. The Deceptiveness of Sponsored News Articles: How Readers Recognize and Perceive Native Advertising. *American Behavioral Scientist*, 60(12):1475–1491, 2016.

[35] Ping Xiao and Josep Blat. Generating Apt Metaphor Ideas for Pictorial Advertisements. In *Proceedings of ICCC 2013*, pages 8–15, 2013.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10594210

A COMPREHENSIVE DATASET FOR WEBPAGE CLASSIFICATION

M. Al-Maamari, M. Istaiti, S. Zerhoudi,
M. Dinzinger, M. Granitzer, J. Mitrovic
University of Passau, 94032 Passau, Germany

Abstract

While webpage classification may not be a fundamental requirement for basic web crawling, it proves useful in enhancing the prioritization of crawled webpages. In this regard, our study presents a dataset of 116,000 URLs, complete with their content, specifically curated for webpage classification tasks. The primary goal of this research is to establish this comprehensive dataset with two levels of labels for URLs. Firstly, a broad level categorization dividing URLs into Malicious, Benign, or Adult, and secondly, a more nuanced labeling which includes 20 subclasses, providing a more granular view of the webpage content.

The secondary objective is to leverage this dataset for testing and comparing the performance of various machine learning models, specifically Stochastic Gradient Descent (SGD) and Support Vector Classifier (SVC), in the task of webpage classification. This involves investigating the effectiveness of different input types (URLs only, raw HTML content, and parsed HTML content) and various tokenization methods (character-level, word-level, Byte Pair Encoding(BPE) [1]) on model performance.

A total of 36 experiments were conducted, yielding several important findings. Using only the URL as input consistently resulted in the highest F1 score 0.94. Character-level tokenization consistently outperformed other tokenization techniques. There was a negligible difference in the accuracy of webpage classification between SGD and SVC models.

This research's findings demonstrate the viability of URL-based classification systems in web crawlers and shed light on optimal techniques for feature representation. The comprehensive dataset and results presented in this paper make valuable contributions to the advancement of web crawling applications, especially those requiring effective content prioritization and filtering.

INTRODUCTION

With the exponential growth of the World Wide Web, the number of web pages being created each day has reached unprecedented levels. According to recent statistics up until August 2021, the total number of websites had surpassed 1.88 billion[1], and this number continues to rise rapidly. In such a vast digital landscape, it becomes increasingly challenging for search engines, web crawlers, and other automated systems to efficiently navigate and extract relevant information, especially within the confines of a closed search ecosystem dominated by a few gatekeepers [2]. From traditional methods that analyze keywords, HTML structures, and link patterns, to advanced techniques leveraging machine learning and natural language processing, researchers and developers are continuously refining classification algorithms.

To address this challenge, the classification of web pages using URLs and HTML has emerged as a promising approach. By organizing and categorizing web pages, this technique enables improved crawling efficiency, enhanced search results, and more targeted content indexing. In this article, we delve into the significance of web page classification and its potential benefits for crawling operations.

In our research, we present a comprehensive dataset comprising two distinct groups for web page classification. The dataset includes the URL of the web page, along with the corresponding HTML content and the content extracted from the HTML without any markup. The first group consists of three main categories: Malicious, Adults, and Benign. This grouping allows for the identification and categorization of web pages with potentially harmful or explicit content, as well as those that are considered safe and harmless.

Moreover, within the second group of our dataset, we have further subdivided the benign category into various subgroups. These subgroups include topics such as sports, news, kids, and more. This finer-grained classification enables a more precise targeting and categorization of web pages based on their specific content and themes.

This research has made a contribution by creating a comprehensive dataset that combines URLs with their corresponding HTML content. This dataset serves as a valuable resource for training machine learning models to classify web pages. By utilizing this dataset, we conducted a comparative analysis of classification approaches using both URL characteristics and the content of web pages. This investigation revealed the efficacy of using URLs over webpages' content as input to the classification models, showcasing the potential for more accurate and targeted web page categorization. The findings of this study highlight the difference between using URL and content in web page classification and provide valuable insights for improving crawling efficiency and enhancing the overall performance of automated systems in navigating the vast digital landscape.

RELATED WORK

Webpage classification using Uniform Resource Locators (URLs) represents a critical area of study in the realm of information retrieval, web mining, and cybersecurity [3,4]. Researchers have proposed diverse strategies for this task, emphasizing distinct aspects such as the linguistic features within URLs, efficiency considerations, or integration with HTML content. As the prime interface between users and

[1] https://www.statista.com/cart/19058/number-of-websites-online/

https://doi.org/10.5281/zenodo.10594210

web resources, URLs contain significant information about the content of webpages, making them a valuable feature for webpage classification tasks. Understanding the various ways in which URLs have been leveraged for classification, as well as the different methodologies employed, provides vital context and inspiration for further exploration in this area.

One innovative approach has been introduced by Abdallah and de La Iglesia [5]. In their paper, they present the argument that URLs, albeit brief, offer a wealth of information for classification tasks, including potentially domain-specific terminology and abbreviations. They identified the inefficiencies in the brute-force approach, which extracts all possible substrings (allgrams) as the classifier's feature set, due to its inability to scale well for large datasets. In response, they proposed an n-gram language model for webpage classification, introducing an efficient method that not only offers competitive accuracy but also ensures scalability. Their technique, borrowing the concept of language models from the fields of information retrieval and automatic speech recognition, has shown promising results on multiple datasets with different classification objectives, where they achieved 0.82 F1 score in the DMOZ dataset, illustrating its potential utility in a wide range of URL classification scenarios.

Min-Yen Kan and Hoang Oanh Nguyen Thi [4], have advanced the field of webpage classification by developing a unique method that accelerates the classification process, only using URLs as the source of input. Their methodology involves segmenting the URL into meaningful components and extracting salient patterns to be used in supervised maximum entropy modeling. They demonstrated the effectiveness of this approach by showcasing its performance against a full-text standardized dataset (WebKB). The results were promising with F1 score of 0.62, indicating that integrating URL-based features alongside the content of the webpages can indeed match or even surpass previous methods. This work highlights the potential of using URLs as a robust and efficient feature for webpage classification.

The work by Ali Aljofey [6] experiences into a critical domain of security by focusing on phishing website detection. This research uses both URLs and HTML content to extract features, categorizing them into four groups. Some of the features presented are newly proposed, reflecting the continual innovation in this field. Aljofey's approach also demonstrates the utility of machine learning techniques in webpage classification tasks, with their results on a costume dataset they used, showing a high F1 score of 0.96, particularly when the XGBoost classifier was applied on a combination of all the features.

Lastly, the work by Hung Le and colleagues [7] provides a remarkable contribution to detecting malicious URLs through deep learning. They propose URLNet, a framework designed to overcome the shortcomings of traditional methods that primarily rely on blacklists. Their method uses convolutional neural networks to capture semantic information and sequential patterns in URLs. Their results reveal the impressive performance of URLNet in terms of significant improvements over baseline methods across various metrics, where they tested their framework on a large dataset collected from VirusTotal, and got an accuracy of 0.99, making their work an influential reference in the study of webpage classification using URLs.

Despite the considerable progress made in classifying webpages based on their URLs and HTML content, as aforementioned, a critical challenge remains in the form of the availability of adequately annotated datasets that can enable researchers to train and evaluate novel classification models. Recognizing this gap, our work introduces a comprehensive and openly available dataset, with 116,000 URLs, complete with raw HTML and parsed content. We have provided two levels of labels, firstly, a broad level categorizing URLs as Malicious, Benign, or Adult, and secondly, a more nuanced labeling which includes categories like 'Spam', 'Malware', 'Society', and 'Arts'. This extensive dataset with multiple levels of labeling not only addresses a significant gap in the field but also enables the development and evaluation of more sophisticated webpage classification models. Moreover, by comparing the performance of different machine learning models, we provide additional insights into the potential of different data representations and tokenization methods for webpage classification.

METHODOLOGY

In this section, we outline the methodology employed in our study for building a dataset of URLs of webpages and their HTML content, then use this dataset to classify web pages based on their URL, raw HTML content, and parsed content. The goal is to identify various categories of web pages, including benign, adult, and malicious. We present the process of data collection, where we curate a large and diverse dataset of URLs from multiple online sources. This is followed by a discussion on our dataset construction and cleaning process to ensure high-quality data for our experiments.

Next, we describe the machine learning models used, including the Support Vector Classifier (SVC) and Stochastic Gradient Descent (SGD) Classifier. Their respective configurations, hyperparameters, and reasons for selection are provided. We then discuss the feature representation and tokenization strategies for the input data. Three types of inputs and three tokenization methods are utilized for this purpose.

Following that, we describe the experimental setup, including the evaluation metrics employed, which consist of precision, recall, and F1 and F2 scores. Finally, we discuss the data analysis phase, in which the results of our experiments are evaluated and compared.

Data Collection

The dataset used in this research has been curated from multiple online sources [8–14], providing a diverse set of URLs including benign, malicious, and adult. The primary

https://doi.org/10.5281/zenodo.10594210

source for benign URLs is the URL Classification Dataset [DMOZ] [8], while the primary source for malware URLs is URLhaus [14], the URLhaus dataset of URLs is updated over time; we downloaded and used the dataset with the last update of first of March 2022. After collecting the URLs, a crawling process is initiated for each URL to gather the raw HTML content, for the crawling process, we used the "OWler" which is a crawler developed by Dinzinger et al. [submitted to OSSYM2023]. This content is used for the comparison of machine learning model performance when raw HTML content, parsed HTML content, or only the URL is used as input. Post-collection, the raw HTML content is parsed to extract structured content, which is stored as a distinct field for each URL in the dataset. Here we faced a problem where we had some URLs that were not working anymore, in this case, we just ignored any non-working URL. Each URL is further labeled with a main label and a subclass label, providing 3 and 20 unique labels, respectively.

Dataset Construction

To ensure data quality and relevance, the dataset undergoes a cleaning process, which includes eliminating duplicates where we removed around two thousand duplicate URLs, most of which were malicious URLs. Then the URLs with empty content were also removed, in this step, we found that 23 URLs had no content when they were crawled, these 23 URLs were removed from our dataset. The cleaning procedure ensures that the dataset is reliable and can be effectively utilized for the experiments planned in this study. The count of each category and sub-category can be found in Table 1.

Table 1: Count of each category and subclass

Main Label	Subclass	Count	Total
Adult	Adult	4424	4424
Malicious	Spam	830	22949
	Phishing	3734	
	Defacement	4004	
	Malware	14381	
Benign	Society	22010	88628
	Arts	15073	
	Privacy Policy	10575	
	Science	9408	
	Computers	4828	
	Games	4270	
	Recreation	4231	
	Reference	3707	
	Business	3641	
	Sports	2986	
	Kids	2392	
	Health	2110	
	Shopping	1572	
	Home	1475	
	News	350	

Machine Learning Models

Two models have been employed in this study, Support Vector Classifier (SVC) and Stochastic Gradient Descent (SGD) Classifier. The SVC model [15] uses a linear kernel, allowing it to scale well to large datasets, thanks to its implementation in terms of liblinear [16] rather than libsvm [17]. The SGD Classifier is a linear classifier optimized using stochastic gradient descent, making it particularly useful for large datasets due to its suitability for online or mini-batch learning settings.

Both models' hyperparameters were mostly set to the default values. For SGD, we set the loss function to 'hinge', the regularization penalty was set to 'l2' with an alpha of 0.0001. For SVC, the penalty was set to 'l2' with a loss of 'squared hinge', both models were fit with an intercept and the maximum number of iterations for both was set to 1000, the only hyperparameter that was set to a non-default value was class_weight which we set to 'balanced' in order to mitigate the unbalanced classes.

Both SVC and SGD classifiers are linear models which make them well suited for large scale feature datasets like ours. In terms of computational cost and memory usage, they are efficient and this is crucial in handling our dataset of 116 thousand URLs.

Feature Representation and Tokenization

Feature representation in this study encompasses three types of input: URLs only, raw HTML content, and parsed HTML content. Each input type possesses its own unique strengths and weaknesses for the classification task at hand. To explore the impact of tokenization methods and levels, we employed character-level, word-level, and Byte Pair Encoding (BPE) [1] techniques. In particular, we chose a window size of (1,3) for character-level and word-level tokenization. This decision was motivated by the desire to capture both local and contextual information within the text. By considering 1-grams, 2-grams, and 3-grams simultaneously, we aimed to extract fine-grained details as well as broader contextual patterns, striking a balance between granularity and computational complexity.

Experimental Setup

The experimental setup includes a total of 36 experiments, each designed to investigate a specific combination of models, input types, and tokenization methods. We constructed the settings of the 36 experiments to cover all the possible combinations of the following aspects: algorithms used for classification (Stochastic Gradient Descent and Support Vector Classification), tokenization methods (TF-IDF and Byte Pair Encoding), types of input data (URL, Content, and HTML), labels (Main label and Subclass), and the levels of n-grams (character-level and word-level).

For each experiment, the dataset was split into a training set and a test set at a ratio of 70%, 30%, respectively, resulting in 81,200 URLs for training and 34,801 URLs for testing. This split provides enough data for training while

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10594210

still reserving a sizable portion for validation, which ensures the reliability of the experiment's results.

The performance of the experiments is evaluated based on precision, recall, and F1 and F2 scores. The F1-score is the harmonic mean of precision and recall, while the F2-score is more sensitive to recall. We chose to include the F2-score in our evaluation metrics because we prioritize the recognition of illegal or harmful web pages. In such cases, high recall (i.e., reducing the number of false negatives) is more important than precision, as missing such pages could lead to more severe consequences than falsely identifying a harmless page as harmful.

Both the F1 and F2 scores are calculated for each class and then averaged to produce macro-averaged scores, thus ensuring an unbiased measure across the classes.

RESULTS

This section provides an evaluation of the 36 conducted experiments, employing the F2 macro score as the primary criterion for comparison. Various elements were analyzed, including input types (URL, content, and HTML), tokenization methodologies (TFIDF and BPE), as well as machine learning algorithms (Stochastic Gradient Descent - SGD and Support Vector Classifier - SVC).

Based on our evaluation of model performance with various types of inputs, we consistently found that the use of URL input leads to superior model outcomes compared to those utilizing content or HTML input. This is evident from the high F2 scores achieved when the target output is the main label. To illustrate, the main label classification yielded an F2 score of 0.94 for SVC and 0.92 for SGD with URL input. However, in the case of subclass classification, the model which leverages SVC algorithm with content input was superior, achieving an optimal F2 score of 0.64. Despite this, URL input maintained its efficiency edge in terms of prediction and training time, outshining both content and HTML inputs, as shown in Figure 1.

Analyzing the confusion matrix of the best performing model, as shown in Figure 2, gives us insights into the model's performance across the different classes: 'Adult', 'Benign', and 'Malicious'. The 'Adult' class had an accuracy of 88%, with 12% of instances being misclassified as 'Benign', while no instances were misclassified as 'Malicious'. The 'Benign' class showcased an impressive accuracy of 99%, with only 1% of instances incorrectly identified as 'Malicious'. For the 'Malicious' class, 92% of instances were correctly classified, with 8% being wrongly classified as 'Benign'. No 'Malicious' instances were misclassified as 'Adult'. This suggests that the classifier performs exceptionally well for the 'Benign' and 'Malicious' classes, with room for improvement in the detection of 'Adult' content.

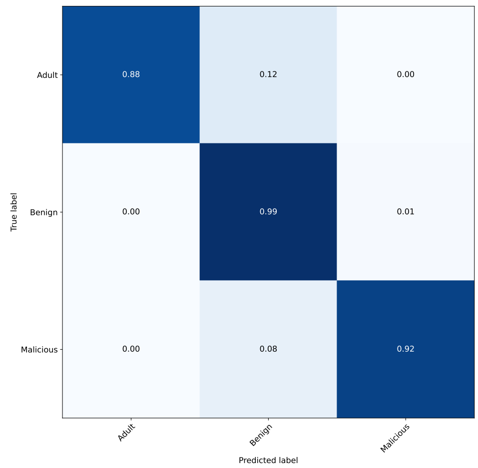


Figure 2: Confusion Matrix of the Best Model

Our analysis of the top 20 most important features, illustrated in Figure 3, offers compelling insights into how the classifier makes its decisions. With TFIDF tokenization at 1, 2, and 3-grams word level, and a trained Random Forest model, the term "video" emerged as the most significant feature, followed by "com video", "HTTP", "www", "https", and "com". The prominence of the term "video" in the feature importance ranking suggests that URLs containing this term are more likely to be classified as adult webpages. Similarly, the presence of "zip" amongst the important features indicates the URL's probable classification as a malicious webpage, potentially hosting malware. The other 14 features do not display such high importance values. While these findings do suggest a potential bias of the classifier towards URLs containing these key terms, it also underscores the model's ability to discern patterns and relationships between specific words and webpage classification. It is crucial, however, to approach this interpretation with caution, as it might not always be the case, and further research is needed to assert these relationships conclusively.

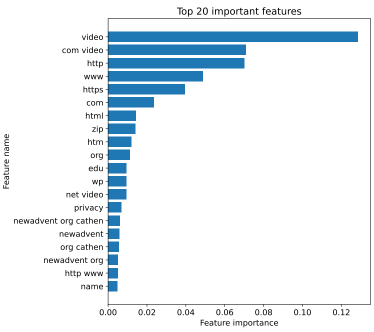
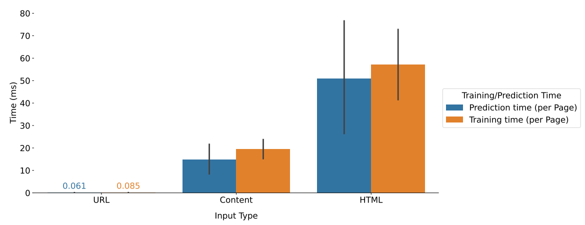


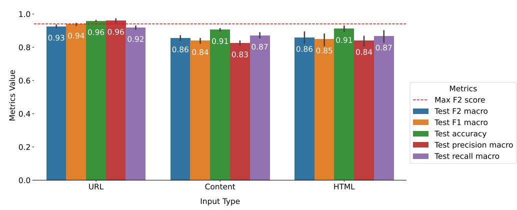
Figure 3: Top 20 Features

Switching our focus to tokenization methodologies, TFIDF generally outperforms BPE. This is evident in main label classification where TFIDF surpasses BPE across all inputs and algorithms, most notably reaching an F2 score of 0.94 with SVC on URL input. Predictive efficiency and

https://doi.org/10.5281/zenodo.10594210



(a) Train/Test time per URL for different Input types (Output: Main Label)



(b) Model performance metrics for different Input types (Output: Main Label)

Figure 1: Performance of the best model

Table 2: Highlights of the results. Note: The variance of all measured values is very small, approaching 0.

algorithm	tokenizer	input	output	F2	prediction time(ms)	fit time(ms)	token level	accuracy	precision	recall
SVC	TFIDF	url	main label	0.94	0.05	0.10	char	0.96	0.96	0.94
SVC	BPE	url	main label	0.93	0.12	0.13	-	0.96	0.95	0.93
SGD	TFIDF	url	main label	0.92	0.05	0.07	char	0.96	0.97	0.91
SVC	TFIDF	html	main label	0.90	13.79	43.58	word	0.93	0.87	0.92
SVC	TFIDF	content	main label	0.87	13.95	14.24	char	0.91	0.83	0.90
SVC	BPE	content	subclass	0.64	25.76	26.85	-	0.74	0.61	0.66
SVC	TFIDF	url	subclass	0.62	0.05	0.26	char	0.72	0.59	0.64
SGD	TFIDF	html	subclass	0.48	51.66	45.00	char	0.60	0.48	0.51

training times also align with these results, with TFIDF maintaining a lead. However, the gap between TFIDF and BPE narrows down in the subclass classification. An instance of this can be observed with SVC, which delivers higher F2 scores with TFIDF on URL and content inputs, but sees a slight improvement with BPE on HTML input.

Delving into the comparison between SGD and SVC as machine learning models, SVC frequently yields superior results in terms of F2 macro scores. A manifestation of this can be seen in SVC's highest F2 score of 0.94 with URL input and TFIDF tokenization for main label classification, and 0.63 for subclass classification under the same conditions. SGD, however, falls short with highest scores of 0.92 and 0.56 respectively. On the other hand, SGD's prediction and training times are consistently faster than those of SVC. For results of more experiments see Table 2, it should be noted that Table 2 only showcases selected key outcomes from the total of 36 experiments we conducted.

To sum up, our experiments show that the combination of SVC algorithm and TFIDF tokenization applied to URL input yields the highest F2 macro scores It's noteworthy that this optimal configuration does not invariably ensure the most efficient prediction and training times. Lastly, the n-gram level's influence (word or char) seems less impactful in these experiments, thereby emphasizing the importance of appropriate feature selection and algorithm choice for machine learning tasks in webpage classification.

DISCUSSION

The results of our study offer several noteworthy insights into the process of webpage classification, particularly focusing on the selection of features and machine learning algorithms. Our findings primarily highlight the potential of URL inputs, the TFIDF tokenization method, and the SVC algorithm to yield high classification performance. These results extend and deepen the understanding of webpage classification, presenting potential guidelines for feature and algorithm selection in this field.

The observed superior performance of URL inputs over content and HTML inputs aligns with the previous research asserting the high information value embedded in URLs. This finding builds upon the studies by Abdallah and de La Iglesia [5], and Kan and Thi [4], who have also leveraged URL inputs for webpage classification. Notably, our study expands on these works by illustrating that URL inputs not only yield high accuracy but also ensure superior efficiency in terms of prediction and training times.

Similarly, our analysis of tokenization methods adds to the current body of literature. The observed dominance of TFIDF over BPE in most scenarios is an important contribution, especially when considering the main label classification. Although the performance difference in subclass classification is less pronounced, the findings still shed light on the potential implications of tokenization methods for webpage classification, encouraging future researchers to consider these aspects when designing their classification models.

In terms of machine learning algorithms, the superior performance of SVC over SGD in our study presents an interesting point for discussion. While previous research has demonstrated the utility of a range of machine learning algorithms for webpage classification, including XGBoost [6], our findings point out the potential advantages of SVC, particularly when combined with TFIDF tokenization and URL input. This, however, does not discount the potential util-

https://doi.org/10.5281/zenodo.10594210

ity of SGD, which displayed competitive results and higher efficiency in terms of prediction and training times.

However, it is important to acknowledge the limitations of our study. Our focus was restricted to a limited set of features, tokenization techniques, and machine learning algorithms. This presents an expansive opportunity for future research to explore a wider range of methods and techniques that could potentially enhance the scope and applicability of webpage classification tasks.

Future research in this area could consider incorporating additional features, tokenization methods, or machine learning algorithms. Additionally, the impact of different preprocessing steps, feature selection methods, or hyperparameter tuning approaches could be investigated.

CONCLUSION

In conclusion, this research introduces a comprehensive dataset of 116,000 URLs, providing a substantial resource for future research in the field of webpage classification. Through comprehensive analysis, it became evident that URLs represent a highly valuable input source, consistently yielding superior model outcomes compared to other inputs such as HTML content.

The study's findings revealed that the Support Vector Classifier (SVC), in conjunction with TFIDF tokenization and URL input, yielded the highest F2 macro scores. Although this optimal combination does not invariably ensure the most efficient prediction and training times, it does highlight the importance of careful feature selection and algorithm choice for tasks in webpage classification.

Moreover, tokenization significantly impacts performance, underscoring the importance of feature representation. Results favored TFIDF over BPE in most cases, with n-gram level playing a minor role. Although SGD and SVC showed similar accuracy, SVC outperformed in F2 macro scores, indicating its aptness for this task.

This study enhances web crawling applications by identifying optimal techniques for feature representation and model choice. It paves the way for future work in webpage classification, providing key insights and a rich dataset for continued research.

ACKNOWLEDGEMENTS

Federal Ministry of Education and Research

This work is part of the OpenWebSearch.eu project, funded by the EU under the GA 101070014, and part of the CAROLL project, funded by the German Federal Ministry of Education and Research (BMBF) under the funding code 01|S20049.

REFERENCES

[1] Philip Gage. A new algorithm for data compression. *C Users Journal*, 12(2):23–38, 1994.

[2] Michael Granitzer, Stefan Voigt, Noor Afshan Fathima, Martin Golasowski, Christian Guetl, Tobias Hecking, Gijs Hendriksen, Djoerd Hiemstra, Jan Martinovič, Jelena Mitrović, et al. Impact and development of an open web index for open web search. *Journal of the Association for Information Science and Technology*, 2023.

[3] Inma Hernández, Carlos R Rivero, David Ruiz, and Rafael Corchuelo. A statistical approach to url-based web page clustering. In *Proceedings of the 21st International Conference on World Wide Web*, pages 525–526, 2012.

[4] Min-Yen Kan and Hoang Oanh Nguyen Thi. Fast webpage classification using url features. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 325–326, 2005.

[5] Tarek Amr Abdallah and Beatriz de La Iglesia. Url-based web page classification: With n-gram language models. In *Knowledge Discovery, Knowledge Engineering and Knowledge Management: 6th International Joint Conference, IC3K 2014, Rome, Italy, October 21-24, 2014, Revised Selected Papers 6*, pages 19–33. Springer, 2015.

[6] Ali Aljofey, Qingshan Jiang, Abdur Rasool, Hui Chen, Wenyin Liu, Qiang Qu, and Yang Wang. An effective detection approach for phishing websites using url and html features. *Scientific Reports*, 12(1):8842, 2022.

[7] Hung Le, Quang Pham, Doyen Sahoo, and Steven CH Hoi. Urlnet: Learning a url representation with deep learning for malicious url detection. *arXiv preprint arXiv:1802.03162*, 2018.

[8] Dmoz dataset, Accessed: 2022-11-10. `https://www.kaggle.com/datasets/shawon10/url-classification-dataset-dmoz`.

[9] Malicious dataset, Accessed: 2022-11-10. `https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset`.

[10] Spam urls classification dataset, Accessed: 2022-11-10. `https://www.kaggle.com/datasets/shivamb/spam-url-prediction`.

[11] Spam urls - spam 404, Accessed: 2022-11-10. `https://raw.githubusercontent.com/Spam404/lists/master/main-blacklist.txt`.

[12] Spam urls - matomo.org, Accessed: 2022-11-10. `https://raw.githubusercontent.com/matomo-org/referrer-spam-blacklist/master/spammers.txt`.

[13] Maps policies dataset v1.0, Accessed: 2022-11-10. `https://www.usableprivacy.org/static/data/MAPS_Policies_Dataset_v1.0.zip`.

[14] Urlhaus, Accessed: 2023-3-1. `https://urlhaus.abuse.ch/`.

[15] Corinna Cortes and Vladimir Vapnik. Support vector machine. *Machine learning*, 20(3):273–297, 1995.

[16] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, jun 2008.

[17] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10636166

CONCEPTUAL DESIGN AND IMPLEMENTATION OF A PROTOTYPE SEARCH APPLICATION USING THE OPEN WEB SEARCH INDEX

A. Nussbaumer[1], R. Kaushik[1,2], G. Hendriksen[3], S. Gürtl[1], C. Gütl[1]
[1]Graz University of Technology, Graz, Austria
[2]University of Waterloo, Ontario, Canada
[3]Radboud University, Nijmegen, Netherlands

Abstract

The development of special-purpose search engines requires a crawling and indexing infrastructure, which needs technological knowledge and resources. This paper presents a concept and implementation of a prototype search application that enables creating own search applications using the OpenWebSearch.eu index. The concept consists of the integration of an index partition exported from the Open Web Index and a search service that builds on Apache Lucene and offers a REST API, which makes the index searchable. A prototype implementation has been created that applies the conceptual design and provides two demonstration applications. Concept and implementation should enable and encourage developers to create their own special-purpose search application.

INTRODUCTION

In contrast to general-purpose search engines like Google, vertical search engines enable focused search in specific domains and allow domain-specific search operations. Current popular vertical search solutions are mostly commercially focused or integrated into enterprises' business models, such as Amazon's product search, LinkedIn's people search, or Booking.com's hotel search.

Search engines are composed of basic components and processes, such as gathering web documents, indexing, metadata extraction, searching and ranking, and a user interface [1]. Also vertical search engines need a search index, which requires a lot of technological resources if newly created even for a fraction of the global web content. The OpenWebSearch.eu (OWS) project aims to provide unbiased, democratic, and free search across the internet through its open access to its Open Web Index (OWI). In particular, it allows downloading a portion or partition of the index, which can be used to create a search application [2].

This paper presents a conceptual design of a vertical search engine in the context of the OWS project and its integration with the OWI. Furthermore, it describes the implementation of a prototype search application based on this concept, as well as two demonstration applications. Thus this paper seeks to demonstrate and provide a technological basis how a search application can be developed based on the OWS infrastructure and the OWI.

CONCEPTUAL DESIGN

The overall concept of an OWS vertical search engine consists of two parts, the OWI and the search application (see Figure 1). The term *search application* is used for the stand-alone search component with imported index partition.

The OWI contains a vast corpus of websites collected from the World Wide Web, which is maintained in data centres distributed throughout Europe. It allows downloading partitions of the index that can be incorporated by search applications. An index partition is structured as Common Index File Format (CIFF) [3] and the corresponding metadata is shipped as Apache Parquet[1] file format.

CIFF aims to enable sharing and utilization of inverted indices across different search systems. It provides a standardized format for representing index data, allowing multiple search engines to access and utilize the same index files. This format is quite useful in academia for searching indices from various information retrieval systems and comparing their performance. However in an industrial setting, it cannot directly be used by most search libraries due to their own internal index formats. Therefore, in order to utilize the CIFF index, third party developers must convert it to the internal index format of the search library that they use. To resolve this issue, the CIFF-Lucene converter[2] developed by Radboud University can be used to generate an index that can be used by Apache Lucene[3]. Lucene has been chosen, since it is used as a search engine library by commonly used search engine systems, such as Elasticsearch and Apache Solr.

In addition to index data, the Open Web Index also provides metadata of web pages in Parquet format. When creating the OpenWebSearch.eu index, the original content of the websites goes through common pre-processing steps. However, snippets of original website data in conjunction with their links make for richer results in search applications. In order to preserve this original page, full text is stored in the metadata. Furthermore, the original metadata of the web pages and data stemming from the page analysis data are stored. In the future, further information, such as the language, the topics of the content, and geographic information (coordinates) are extracted from the pages and added to the metadata. These metadata are exported along with the index data in the Parquet format, which is a columnar storage file format widely used in big data processing and analytics

[1] https://parquet.apache.org/
[2] https://github.com/informagi/lucene-ciff
[3] https://lucene.apache.org/

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10636166

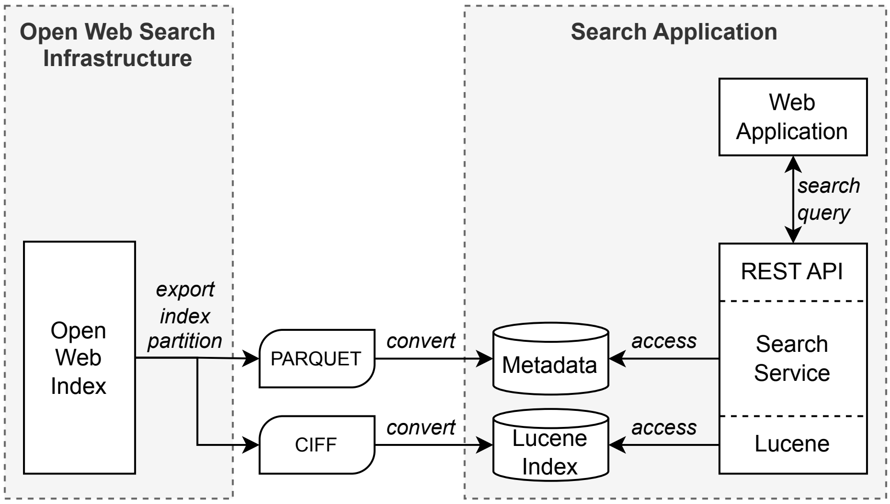


Figure 1: Conceptual design of a vertical search engine.

frameworks such as Apache Hadoop and Apache Spark. It is designed to optimize I/O performance for large-scale data processing. The columnar data storage has several benefits including columnar compression, schema evolution without dataset rewrites and efficient queries by reading columns.

The core of the search service is the search application that coordinates the search and retrieval process. It provides a REST API that accepts search queries and returns search results. In order to perform the actual search, it makes use of Lucene that accesses the partitioned index converted to the native Lucene format. Using the metadata information, the preliminary search result can be limited and ranked, as well as enriched with full-text snippets. The web application provides the user interface (front-end) where search queries are created and results are displayed. This can be done in classic style with a text field and link list, but also other forms are possible.

PROTOTYPE SEARCH APPLICATION

Based on the conceptual design of the OpenWebSearch.eu project, we created a prototype search application[4] that can be used as a reference as well as the prototype can be extended by future search applications. This application includes a REST API that is intended for direct use by developers who wish to create search applications using the Open Web Index. A key aspect of the REST API resides in its capacity to operate across multiple indices rather than being limited to a single index. The application requires developers to place one or multiple CIFF files into the `ciff` folder and their corresponding Parquet files into the `parquet` folder.

To be able to use an index defined by a CIFF file in the prototype search application, it has to be converted by the CIFF-Lucene converter to a Lucene index beforehand. The application provides a script (`convert_index.sh`) that converts the CIFF file into an Apache Lucene index and stores it in a folder named the same as the CIFF index. This also allows developers to use the converted Lucene index among other search libraries of their choice, such as Elasticsearch, Solr and Cassandra.

The compressed Parquet file corresponding to the index is accessed at runtime and read programmatically using the Java library `parquet-mr`[5]. The Parquet file is queried by the application through column pruning, enabling efficient large scale data retrieval. This allows rich metadata to be returned by the REST API along with the index search results. Additionally, metadata is used for ranking and filtering algorithms.

The search application is written in Java, and built using Apache Maven. The developer simply needs to run the build (`build.sh`) script and then the start script (`start.sh`). By default the service listens to `localhost:8000` and Cross Origin Resource Sharing (CORS) is allowed from `localhost:80`. By this action, any front-end application hosted on the latter can access and send requests to the service. The CORS and hosting ports can be changed from these defaults as required.

At the application start, all indices in the `lucene` folder and their corresponding Parquet files in the `parquet` folder are loaded for later search request handling. The prototype search application provides a REST API with a single endpoint named `search` and multiple query parameters (see Table 1). Developers can use this endpoint to send an HTTP GET request with at least the parameter `q` that specifies the search query. While `q` is a required parameter, the remaining ones are optional. For each search request, a particular Lucene index can be defined by setting the parameter `index` to the desired value (i.e., the name of the desired Lucene index). Further, a filtering mechanism is realized by the parameter `lang` that restricts search results to a specified language. An additional parameter `ranking` is employed to determine whether the results should be

[4] https://opencode.it4i.eu/openwebsearcheu-public/prototype-search-application

[5] https://github.com/apache/parquet-mr

https://doi.org/10.5281/zenodo.10636166

Table 1: URL query parameters

Parameter	Necessity	Description
`q`	Required	Search term(s) to be searched for in the Lucene index.
`index`	Optional	Specifies the Lucene index to be searched in. The passed value must match the folder name of the Lucene index. If no index is specified, the default index passed as argument at the application start is used.
`lang`	Optional	Restricts the search result to only consider pages in the specified language (e.g., `en`). If no language is specified, the search results are language in dependent.
`ranking`	Optional	Specifies the order of the search result based on the number of words a page has. Can be either `asc` or `desc`. If no ranking is specified, the order of the search result yielded by Lucene's similarity search is used.
`limit`	Optional	Sets the maximum number of results to be returned. If no limit is specified, a maximum of 20 results are returned by default.

organized in ascending or descending order based on the word count within an individual web page. The filtering and ranking of results made possible by these query parameters demonstrates the handling of index partitions and the associated metadata. A representative URL format for a GET request may resemble the following:

```
http://localhost:8000/search?q=tower&index=
websites-graz&lang=en&ranking=asc&limit=10
```

With this request to the REST API, the term `tower` is searched in the index `websites-graz` and only web pages with English language in ascending order in terms of word count of the page limited to a maximum of 10 results are returned.

The search service sends the results back in the form of JSON data as an array of objects. Each object comprises the `url`, the `title`, a `textSnippet` consisting of the longest sequence within the text without a line break, the `language`, the `warcDate` and the `wordCount` of the respective web page. An overview of the object fields is given in Table 2.

Table 2: Fields of the search result

Key	Description
`id`	the id of the result item or web page
`url`	the URL of the result item or web page
`title`	the title of the result item or web page
`textSnippet`	a piece of text in context of the search term
`language`	the languge of the result item or web page
`warcDate`	the date of the WARC file where the page is found, which is crawling date
`wordCount`	the number of words of the result item or web page

DEMONSTRATION

The Prototype Search Application can be used to make any search application on the internet. To demonstrate this, we created two web applications (front-ends) that demonstrate the search applications and its REST API. These front-ends are part of the Prototype applications.

Basic search

The basic search application demonstrates the features and API of the prototype search service. A specific index (CIFF and Parquet file) has been used that contains sightseeing information of Graz, Austria. Furthermore, a demonstration index has been included. This application demonstrates that prototype service can handle multiple indices, as well as the features of the REST API.

The web interface (see Figure 2) consists of a field for entering a search term, as well as radio boxes for selecting the index, the language, and the ranking method. The selected information is translated to a REST call using the API of the service. The result as specified in Table 2 is displayed below in commonly used form.

Sightseeing Search

This application uses the Graz sightseeing index that consists of popular attractions in Graz. It consists of a CIFF and Parquet file that were converted and imported to the search service. The application demonstrates that search applications with a different kind of user interface can be created easily using the concept described in this paper.

The front-end (Figure 3) consists of a graphical map on which predefined attractions are placed. By clicking on an attraction a popup displays the name and a description of the respective item. In addition, a search request is performed using the title as the search term. The search result is displayed below the graphical map. Hence, each time a user

https://doi.org/10.5281/zenodo.10636166

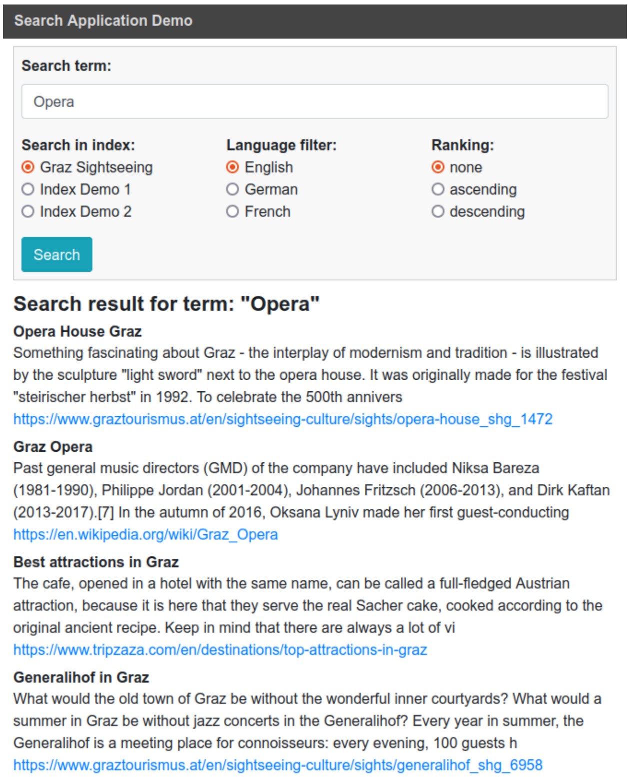


Figure 2: Demonstration Search Application

clicks on an item on the map, a search is automatically performed and the result is displayed. This allows the user to get more information on demand.

CONCLUSION

This paper presents a concept and implementation of a search application in the context of the OpenWebSearch.eu project. The design is deliberately kept simple, as it serves as a blueprint for other search applications and vertical search engines. A key aim of this paper is to provide a link between index partitions created by OpenWebSearch.eu and search applications for one's own purposes. In the near future it will be possible to download an index partition restricted to certain characteristics, such as the topic, region, or time frame.

The prototype search application should encourage the development of various search applications using the OpenWebSearch.eu technology and index data. The source code is available in a public GitLab repository under an open software licence and includes enough documentation so that the technology can be taken up and used for new search applications. The prototype can be used in two ways. First, the service can be used out of the box and a new front-end can be added. Second, the service can be altered and updated to add specific search features.

Future work will include the handling of multiple and large index files. While the search is currently only possible in a single index that has to be specified, in the future a combined search over multiple indices should be enabled.

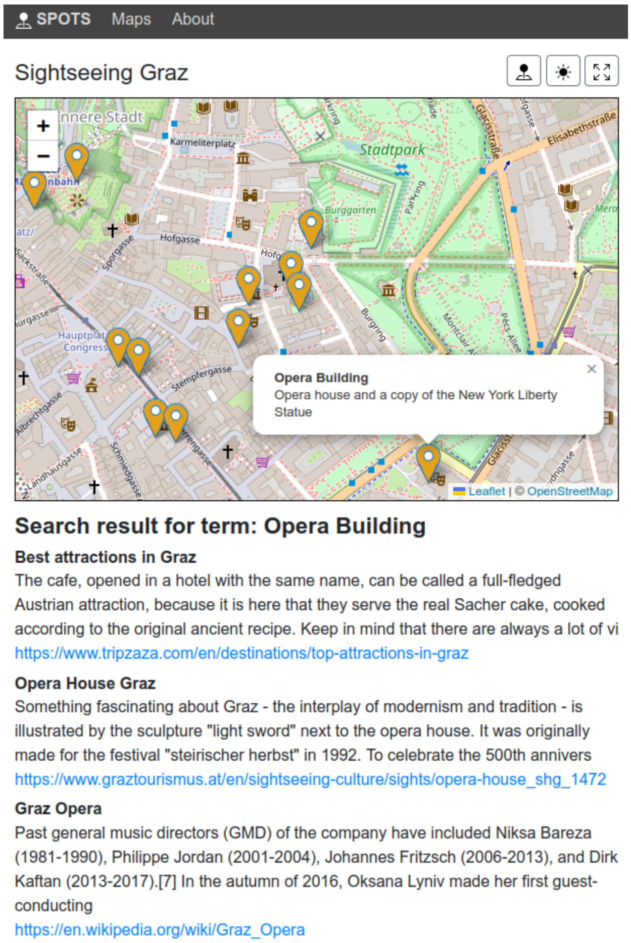


Figure 3: Graz Sightseeing Search Application

Current demonstration indices are rather small, which requires tests and efficiency checks over large indices. More requirements will be collected when it is taken up by others for new types of search applications.

ACKNOWLEDGEMENTS

This work has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101070014 (OpenWebSearch.EU, https://doi.org/10.3030/101070014).

REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retreival*. Addison Wesley, 1999.

[2] M. Granitzer *et al.*, "Impact and development of an open web index for open web search," *Journal of the Association for Information Science and Technology*, in press. `10.1002/asi.24818`

[3] J. Lin *et al.*, "Supporting interoperability between open-source search engines with the common index file format," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 2149–2152. `10.1145/3397271.3401404`

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10607402

UNDERSTANDING AND MITIGATING COGNITIVE BIAS DURING WEB SEARCH

S. Hitzginger, A. Nussbaumer, C. Gütl, C. Ruß-Baumann
Graz University of Technology, Graz, Austria

Abstract

When conducting research on the internet, Confirmation Bias can cause individuals to selectively retrieve information that confirms their pre-existing beliefs. While current tools are available to counteract this effect, they often require additional data that is not always accessible during regular search sessions. Therefore, we aimed to develop a tool capable of detecting Confirmation Bias by simply tracking general search behavior. Through an online study with 43 participants that simulated real web searches, we identified behavioral patterns, like a difference in number of issued search queries, number of clicked results, and position of clicked results, that can predict bias in retrieved information. With those results, we created a browser extension that can track and analyze relevant behaviors in real-time and alert the user of potentially biased search sessions.

INTRODUCTION

The Internet has revolutionized the way we access and consume information, providing us with unprecedented opportunities for knowledge and connectivity. However, the abundance of ambiguous information also amplifies the likelihood of Confirmation Bias, described as the seeking or interpreting of evidence in ways that are partial to existing beliefs [1]. Phenomenons like *filter bubbles*, which are a product of the intention of social media platforms to present information to users that is interesting to them [2], or echo chambers, which have a similar effect but are mostly caused by the homogeneity of friend groups on social media platforms [3], are common manifestations of Confirmation Bias on the Internet, only partially caused by the user. Likewise, when using a search engine, localization and personalization of search results makes unbiased information retrieval a difficult task. The natural human tendency to prefer belief-consistent information further increases the likelihood of the formation of a one-sided opinion.

An individual's tendency to Confirmation Bias is moderated by various other factors, like the exposure to belief-inconsistent information [4], a challenge-averse personality [5], or the involvement and the perceived threat of a certain topic [4]. There are also a number of tools available that could help individuals reduce their susceptibility to echo chambers [6], filter bubbles [7], or Confirmation Bias in general [8]. However, each of those tools relies on information about the content of visited websites. Therefore, their applicability is limited to pre-classified Websites, or content that can be analyzed and classified automatically.

The aim of this work is to create a tool without such limitations. To do so, instead of analyzing the content or context of information, we focus on tracking behaviour during the use of a search engine, the most common way of navigating the Internet. This way, the detection of bias is no longer dependent on knowledge about visited websites. Previous literature already found indications of correlations between different behavioral patterns and bias during web search[9]. We want to confirm those findings in a more general setting and provide a tool capable of using the results in a real-life application.

STUDY DESIGN

To find behavioral features able to predict Confirmation Bias of users during information retrieval via web search, a study simulating a realistic search setting was conducted. For this purpose, we implemented a custom search engine[1] and hosted it on a web server to make it accessible online. Participants were given a specific search task which they should complete using our custom search engine.

For the topic of the search task, we chose the debate that was occurring at the time of the research on the *Legalization of THC-containing Cannabis for recreational use in Austria*, as it presented contrasting facts and diverse opinions. To get valid results, we attempted to create a real-life situation. Participants were asked to inform themselves about the search topic for as long as they thought necessary to then be able to answer whether Cannabis should be legalized in Austria for recreational use or not (see figure 1).

A total of 61 web articles on the topic of Cannabis were gathered from different news sites. Each of those articles was reformatted to only include text, separated in story title, lead, and content. Four raters judged the opinion expressed by each article, rating it either negative, neutral, or positive concerning Cannabis. Different ratings were then aggregated in the following way:

1. with 4 raters agreeing on the same opinion, this opinion was assigned to the article
2. with 3 raters having the same opinion and the fourth rater being neutral, the opinion was assigned to the article
3. with 3 raters having the same opinion and the fourth having an opposing opinion, the article was removed
4. with 2 raters having the same opinion and the other 2 rating the article as neutral, the opinion of the first two raters was assigned to the article
5. every other article with less agreement was removed

By doing this, the 61 considered articles were reduced to 52 articles which expressed a clear bias. Three more articles were removed to have an equal number of positive

[1] https://github.com/sihi9/cb_explostudy

https://doi.org/10.5281/zenodo.10607402

and negative articles, resulting in 20 articles in each bias category and 9 neutral articles. Those 49 articles could then be found with the search engine.

Figure 1 shows a screenshot of the search task presented to participants. It was implemented using ReactJS for the frontend, ExpressJS[2] for the server, and MeiliSearch[3] for the database and search engine. Participants were able to type arbitrary queries, scroll the search engine result page (SERP), go to different pages of the SERP, and click on results to navigate to a new page showing the entire article. Table 1 shows all different behavior variables that were tracked during the search task.

The bias expressed during the search task was evaluated based on the articles that were clicked by a participant, summing up the ratings of clicked articles divided by the number of clicked articles, resulting in a bias rating in the range [-1, 1].

For a more detailed insight on Confirmation Bias, a small questionnaire with 6 questions about different aspects and personal attitude towards Cannabis, assessed with a 5-point-likert-scale ranging from -2 to 2, was also presented to participants, once prior to the search task, and once after the search task. A screenshot of the survey is shown in figure 2. The 6 answers were aggregated and normalized, with 2 items being inverted, to gain two variables representing the opinions towards Cannabis before and after the search task in the range [-1, 1], comparable to the bias expressed by clicked articles. Confirmation Bias can then be interpreted as an alignment of bias in viewed articles and the attitude before the search task.

RESULTS

Confirmation Bias

A total of 59 participants started the study. After removing incomplete attempts, participants that took less than 1 minute on the search tasks, and participants that did not click on a single search result, 43 participants remain, most of them being male (n = 27) and between 20 and 29 years old (n = 29).

The bias of viewed articles is almost normally distributed ($M = 0.05, SD = 0.43$). The attitude is skewed to the right (positive attitude towards Cannabis) with means of $M = 0.38$ before-, and $M = 0.36$ after the search task. To compute Confirmation Bias, the minimum distance to either the median of attitude (Mdn = 0.42), or the median of bias (Mdn = 0.08), was used, and multiplied by -1 if bias and attitude were not aligned. Figure 3 shows the relation between the variables, with the medians marked as black lines. The correlations between Confirmation Bias and behaviour variables are shown in table 2.

The attitude after the search task was used to evaluate the impact of Confirmation Bias. For only 5 participants, Confirmation Bias actually strengthened their belief in the previous attitude.

[2] https://expressjs.com/
[3] https://www.meilisearch.com/

Bias of viewed articles

Another approach was to only evaluate the absolute value of bias expressed by viewed articles. Correlations between the absolute bias and behavioral variables revealed significant correlations between bias and the number of queries, the number of clicked results, the average position (index) of clicked results, and the average page of clicked results, as can be seen in table 3. A linear regression with forced entry was calculated to get a prediction of the bias from usage variables. Due to the high correlation between average index and average page, only the average index was chosen for the regression, because the correlation coefficient is only slightly smaller than of the average page, and it would be more versatile later on, because the index is independent of results shown per search engine result page, and the index theoretically contains more information. The regression showed that average index, number of queries and number of clicked results could predict the absolute bias quite well ($R^2 = .446, F(3, 38) = 10.456, p < .001$). Table 4 shows the coefficients for each variable.

DISCUSSION

Results have shown that Confirmation Bias on the internet might not be as big of a problem as previously assumed. In fact, there is no lack of literature with similar results [4][10], showing that exposure to new information alone is often sufficient to cause a moderation of opinion. While this is good news, it does not mean that there is no Confirmation Bias on the internet. However, our results support previous findings, which suggest that susceptibility to Confirmation Bias depends both on the individual and the topic[4], and is generally not as high as expected. Therefore, identifying Confirmation Bias during web search seems to be a difficult problem, which would require larger sample sizes than this study could provide.

However, our results show interesting behavioral patterns when it comes to a bias in the selection of articles. As expected, showing more engagement in the search task, expressed through issuing more queries, clicking more results, and browsing further through the search engine result pages, is correlated with less bias in the selection of articles. With a simple linear regression only including three factors, we could explain a substantial amount of variance in the bias. The correlations shown in table 3 also hint at further relationships, which can potentially be proven with a larger sample size.

An interesting pattern also arises when comparing significant relationships of Confirmation Bias and behaviour with general bias and behaviour. While the number of clicked results and number of queries are the most significant predictors for bias in clicked articles, they seem to have no impact on Confirmation Bias. On the other hand, time spent on result pages, as well as the standard deviation of time spent on results, are the most significant predictors of Confirmation Bias, but do not significantly correlate with bias of clicked articles, although a tendency towards significance can be

https://doi.org/10.5281/zenodo.10607402

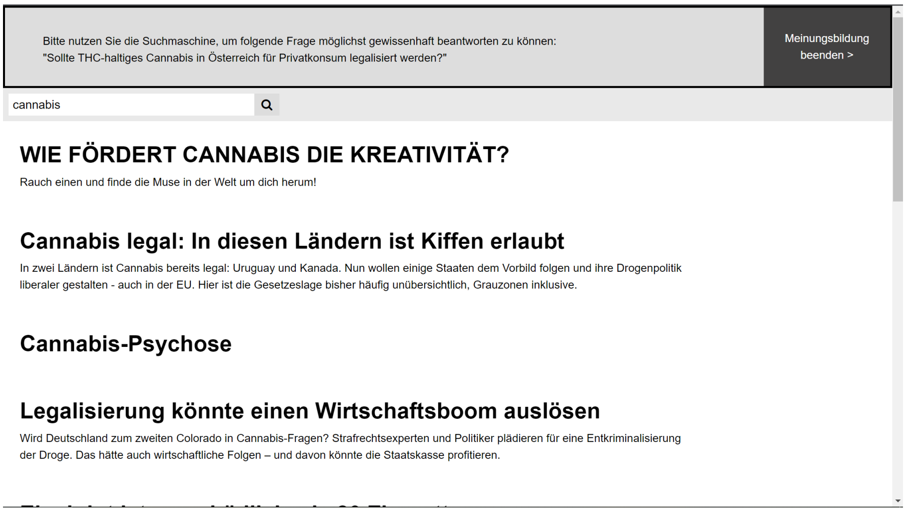


Figure 1: Screenshot of search task

Table 1: Behaviors tracked during search task

Metric	Description
Duration	Total duration spent on the information retrieval task
Time on SERP	Total time spent on a search engine result page (SERP)
Time on results	Total time spent on result pages
Number of queries	Number of different search queries used during information retrieval
Average query duration	Average time spent on each query
Standard deviation of query duration	Standard deviation of query duration
Number of clicked results	Number of clicked results in total
Average time per result	Average time spent on each result page
Standard deviation of time per result	Standard deviation of time spent on results
Average index	Average index (i.e., position of the result on the SERP) of clicked results
Average page	Average page of clicked results

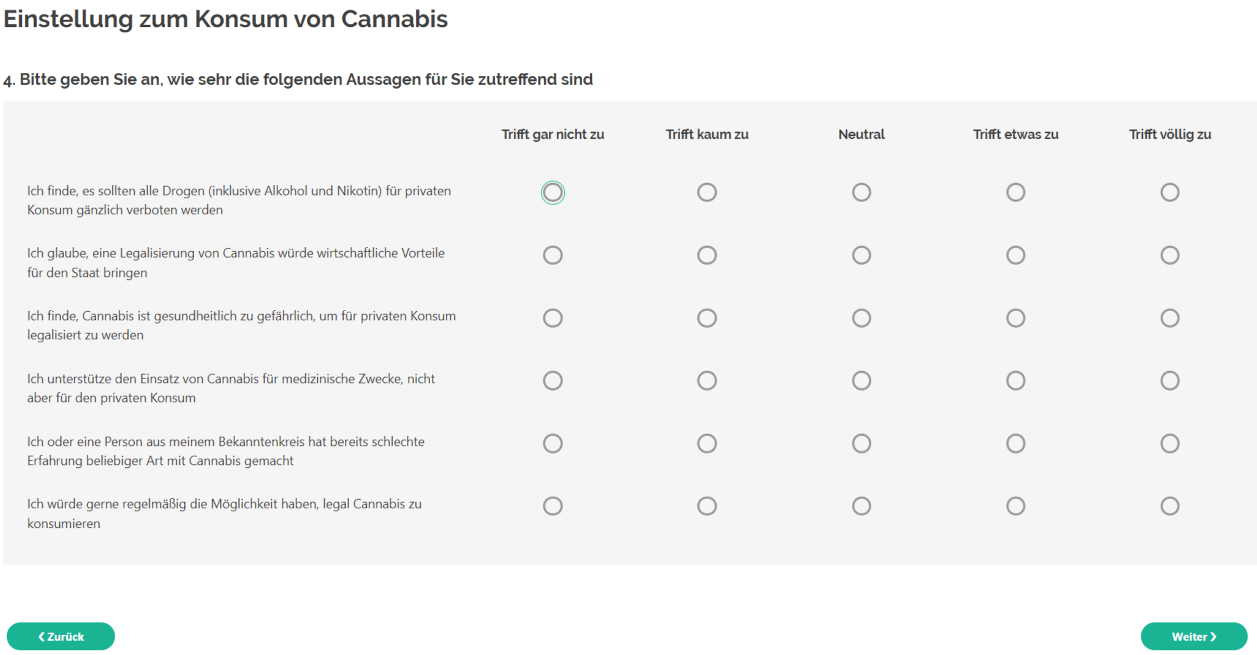


Figure 2: Screenshot of survey

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10607402

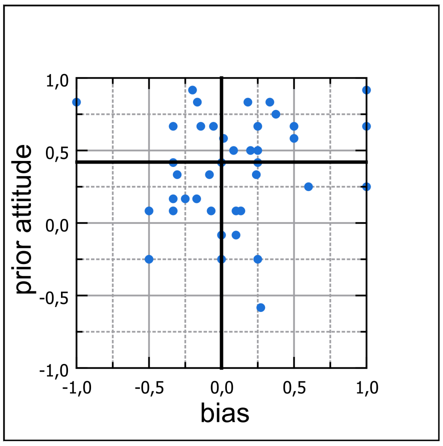


Figure 3: Scatterplot of bias and previous attitude, with black lines indicating the median of each variable.

Table 2: Correlations of factors with Confirmation Bias

Variable	**Pearson correlation**	***p***
duration	-.27	.08
time on results	-.316*	.039
number queries	-.005	.975
average query duration	-.298	.052
std. dev. query duration	-.052	.739
number clicked results	-.056	.72
average time per result	-.21	.176
std. dev. time per result	-.312*	.042
average index	.018	.907
average page	.031	.843

$^{**}p < .001, ^{*}p < 0.05$

Table 3: Correlations of factors with absolute bias

Variable	**Pearson correlation**	***p***
duration	-.182	.243
time on results	-.19	.222
number queries	-.329*	.031
average query duration	-.108	.49
std. dev. query duration	-.259	.093
number clicked results	-.495**	.001
average time per result	-.071	.651
std. dev. time per result	-.202	.195
average index	-.324*	.034
average page	-.35*	.021

$^{**}p < .001, ^{*}p < 0.05$

Table 4: Regression coefficients

Variable	***B***	***t***	***p***
constant	0.79	8.66	<.001
number queries	-0.038	-3.107	.004
number results	-0.042	-3.667	.001
average index	-0.019	-2.6	.013

observed. Spending more time reading articles might also be a form of showing engagement, and is therefore similar to the expected results. The correlation with the standard deviation of time spent on results however is more of a surprise. Apparently, spending more time on some results and less time on others is correlated with a higher Confirmation Bias. One could theorize that this is because participants susceptible to Confirmation Bias stop reading certain articles which do not confirm their opinion, but more research will be necessary to support such hypothesis.

PRACTICAL APPLICATION

Unfortunately, the results of the study are not as conclusive as hoped, probably due to lack of participants. To still prove the concept of a tool capable of detecting a bias in web search, we decided to focus on the bias of clicked articles, because it is a simpler construct with more predictive power than Confirmation Bias. However, the concept could, without much effort, be adapted to predict Confirmation Bias if future research is able to find factors with sufficient predictive power.

The tool[4] is built as a browser extension, both for Google Chrome and Firefox, as they are widely used Browsers that use two different APIs which are also used in most other browsers. Therefore, the tool can be extended for other browsers as well with minimal adjustments.

The extension tracks relevant browsing behavior, which can be classified in two relevant actions: search actions, and result-clicked-actions.

Search actions are defined as the issuing of a new search query. They can be tracked via the browser's history API, which receives a new entry whenever a query is issued with one of the supported search engines. To prove the concept, the tool currently supports two search engines, Google[5] and Ecosia[6]. To track search actions from different search engines, only the regex which detects the search query from the URL needs to be adapted.

To detect a result-clicked-action, content scripts are used. They contain JavaScript code, which is injected by the extension to specific sites, in our case the result pages of supported search engines. The code is able to add onClick-methods to the click-events of the HTML elements which can be clicked by the user to navigate to a search result. Those methods, in combination with the runtime API, can send messages to the extension, informing it that a result was clicked, as well as the exact position of the result.

The study used to analyze behaviour focuses on one specific search task. To use the results of the study, the extension has to be able to separate the search task into separate sessions. This is done by using two assumptions: queries belonging to the same search sessions are more likely to occur in temporal proximity, and use similar words. Time difference between queries is tracked automatically by the

[4] https://github.com/sihi9/cb_extension
[5] https://www.google.com/
[6] https://www.ecosia.org/

https://doi.org/10.5281/zenodo.10607402

history API and does not require additional logic. To evaluate semantic similarity, word-vector similarities computed by the spaCy[7] library are used. When a new search is issued, the semantic similarity to all other search sessions is calculated. If there is a good fit with any other session, the new query is also assigned to this session. Otherwise, if not much time has passed since the last action in a session, and there is at least some semantic similarity between the new query and queries of the previous session, the new query is assigned to this session. Thresholds and decay functions are chosen partially based on the results of the study, and partially through trial and error. They can be improved by more research on browsing and search behavior, as well as improved similarity detection.

The bias of each separate session can then be analyzed, using the results from the regression of the bias in the study. The threshold between high and low bias is also chosen based on the results of the study, where a bias of 0.8 yielded a good division. All of those values can easily be adjusted once future research offers new insights into the behavior correlated with a bias in web search.

If a bias is detected, the user is informed of the bias with a warning symbol. Clicking on the symbol opens a popup which allows the user to view all of his potentially biased search sessions, remove them from the tracking, or continue searching for more articles on the session. A screenshot of the popup is shown in figure 4.

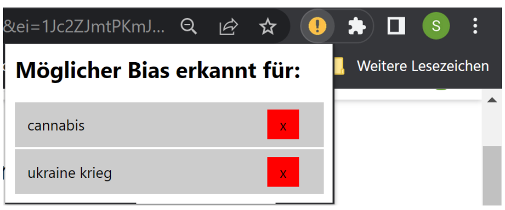


Figure 4: Screenshot of browser extension showing two biased search sessions

CONCLUSION

Confirmation Bias can occur in different forms and shapes. On the internet, during web search, people might tend to prefer content that confirms their preexisting beliefs. There are already tools to help counteract such phenomenons, but most of them require information about content, which is not always easy to acquire. In this paper, we showed the applicability of a different approach, which predicts bias only based on behaviour during web search. Due to the diversity of factors influencing Confirmation Bias and the lack of participants in this study, a lot of open question still remain on the best approach on the extent to which different behaviors correlate with bias. Nonetheless, we did show that there are moderate correlations, both between bias in selected articles and Confirmation Bias. We also provide a proof of concept for a tool capable of tracking browsing behavior and predicting biased search sessions using the results of our study.

[7] https://spacy.io/

ACKNOWLEDGEMENTS

This work has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101070014 (OpenWebSearch.EU, https://doi.org/10.3030/101070014) .

REFERENCES

[1] R. S. Nickerson, "Confirmation Bias: A Ubiquitous Phenomenon in Many Guises," *Review of General Psychology*, vol. 2, pp. 175–220, Jun. 1998, Publisher: SAGE Publications Inc, ISSN: 1089-2680. DOI: 10.1037/1089-2680.2.2.175. https://doi.org/10.1037/1089-2680.2.2.175

[2] M. Cinelli, G. De Francisci Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini, "The echo chamber effect on social media," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 118, no. 9, e2023301118, Mar. 2021, ISSN: 0027-8424. DOI: 10.1073/pnas.2023301118.

[3] E. Bakshy, S. Messing, and L. A. Adamic, "Exposure to ideologically diverse news and opinion on Facebook," *Science*, Jun. 2015. https://www.science.org/doi/10.1126/science%20.aaa1160

[4] Q. V. Liao and W.-T. Fu, "Beyond the filter bubble: interactive effects of perceived threat and topic involvement on selective exposure to information," in *CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA: Association for Computing Machinery, Apr. 2013, pp. 2359–2368, ISBN: 978-1-45031899-0. DOI: 10.1145/2470654.2481326.

[5] S. A. Munson and P. Resnick, "Presenting diverse political opinions: how and how much," in *CHI '10: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA: Association for Computing Machinery, Apr. 2010, pp. 1457–1466, ISBN: 978-1-60558929-9. DOI: 10.1145/1753326.1753543.

[6] S. Nagulendra and J. Vassileva, "Understanding and controlling the filter bubble through interactive visualization: A user study," *HT 2014 - Proceedings of the 25th ACM Conference on Hypertext and Social Media*, Sep. 2014. DOI: 10.1145/2631775.2631811.

[7] S. Faridani, E. Bitton, K. Ryokai, and K. Y. Goldberg, "Opinion space: A scalable tool for browsing online comments," *Conference on Human Factors in Computing Systems - Proceedings*, vol. 2, pp. 1175–1184, Jan. 2010. DOI: 10.1145/1753326.1753502.

[8] S. A. Munson and P. Resnick, "Encouraging Reading of Diverse Political Viewpoints with a Browser Widget," *Proceedings of the 7th International Conference on Weblogs and Social Media, ICWSM 2013*, 2013.

[9] R. W. White and E. Horvitz, "Belief dynamics and biases in web search," *ACM Trans. Inf. Syst.*, vol. 33, no. 4, May 2015, ISSN: 1046-8188. DOI: 10.1145/2746229. https://doi.org/10.1145/2746229

[10] P. M. Dahlgren, "A critical review of filter bubbles and a comparison with selective exposure," *Nordicom Review*, vol. 42, no. 1, pp. 15–33, Jan. 2021. DOI: 10.2478/nor-2021-0002.

https://doi.org/10.5281/zenodo.10557413

LINKNOVATE STARTUP RADAR – DATALIFE USE CASE

Carlos Rodríguez†, Manuel Noia, Linknovate Science, Santiago de Compostela, Spain
Javier Parapar, Information Retrieval Lab, Department of Computer Science, University of A Coruña, A Coruña, Spain

ABSTRACT

STARTUP RADAR (SR) is one of the latest Linknovate R&D efforts.

SR automatically extracts semantic relations between entities from unstructured and heterogeneous data sources. SR aims at detecting 3 critical events from over 80M documents present in Linknovate´s (LKN) database, including news and other unstructured text sources:

1. Mergers & acquisitions
2. Funding events and
3. Product launches

The second capability, which we focus on in this use-case, is a recommendation system to locate “Similar To” companies based on their know-how, products and “innovation weak signals”. Current providers usually rely on financial figures, type of funding round (e.g., series A), number of employees, geographic location, etc. But some of the most exciting info for the user relies on what these companies actually do and know.

SR enables investors and innovation professionals (e.g., innovation directors, R&D managers, new product development teams, etc) to compare companies and conduct tech due diligence, allowing them to make better-informed investment decisions. SR aims to fill the gap for investors, accelerators, and seed funds who lack the data analysis capabilities for tech benchmarking that bigger funds have by manually curating data and relying on their heavy networks. With its automation SR aims to save time, increase deal flow, and enhance the technology due diligence capabilities of our users, by the superior discovery of similar companies (and the future work in relation extraction for the detection of relations and events in unstructured text, e.g., in news).

While competitors like Crunchbase and ChatGPT (OpenAI) exist, they lack precision and recall, since the former lacks “background data” on innovation “weak signals” for the companies of study, and the latter relies -by design- on the overall signals of a startup (website, blog posts, social media, etc) and its marketing jargon rather than robustly comparing innovation signals, apart from not necessarily being up to date.

NEED

SR aims to support mainly the Investor community who demand software tools to enhance their deal flow and technology due diligence. SR can get more intelligence about startups and companies that are "tech-driven” to dramatically improve your company technology and startup radar.

The key points of the SR proposal are:

1. Relation extraction about products, financing and M&A events from unstructured text data sources, which is a very complex challenge. Thus, SR is able to maintain up-to-date structured info about these events.
2. Capability to compare companies and help with tech due diligence: what alternative companies should I consider investing in, how my potential invested company compares (from a tech perspective) with others, how does the tech landscape look like (Similar companies).

Investors (VC, hedge funds, etc) and foremost angels, accelerators, and seed funds lack the data analysis capabilities to quickly and effortlessly benchmark from a tech perspective the companies they (consider) invest in. It is a manual analyst driven work, and the funds with less human resources oftentimes rely on their own investee analysis.

For innovation professionals, it is important to gather “similar organization” data in order to have a more complete understanding of the tech landscape and the startups in their fields of interest. This intelligence is usually more complete from a business perspective, but not from a tech perspective, which is ever evolving (i.e., it demands staying up to date and monitoring changes after a certain period of time).

BENEFITS

The main benefits that SR aims to offer the users are:

- Save time: Reduce the time spent on the startup research process.
- Superior discovery capability of similar companies based on expertise and/or products.
- Help do a better technology due diligence (tech DD): automate technology due diligence of startups and systematize the process can help investors have a clear market landscape in emerging industries (especially those with high growth & evolution).
- Increase deal flow (for investors): by optimizing their processes and boosting better informed decisions.

COMPETITORS AND HOW IT IS SOLVED TODAY

Crunchbase is a business information platform (particularly focusing on startups) that provides data and insights on companies, including their funding, industry, and key personnel. Their "Similar Companies" feature analyzes various data points such as industry, funding, and market presence to suggest companies that share similarities with a given company:

† carlos@linknovate.com

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10557413

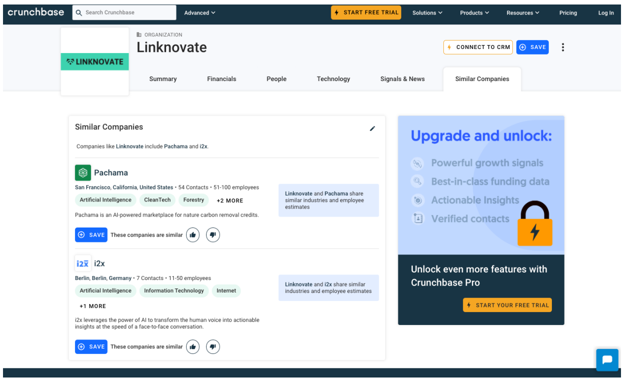


Figure 1: Screenshot of a service competing with Startup Radar, Crunchbase.

CB lacks precision because they rely mainly on the similarity of their generic industry/topic. Additionally, they rely mostly on its funding stage and employee's data, which are not necessarily the best parameters.

As we can see in the previous example, Crunchbase is suggesting the marketplace Pachama as a similar company to Linknovate (a tech scouting platform) because both are categorized as "artificial intelligence" companies, a very wide topic.

Another recent alternative is ChatGPT, a powerful language model designed for context-aware conversations, providing detailed and coherent responses to a wide range of queries and prompts. ChatGPT's ability to find similar companies to a given one is based on matching marketing jargon rather than robustly comparing innovation signals like the LKN approach, making it less suitable for "hard-core tech" comparisons.

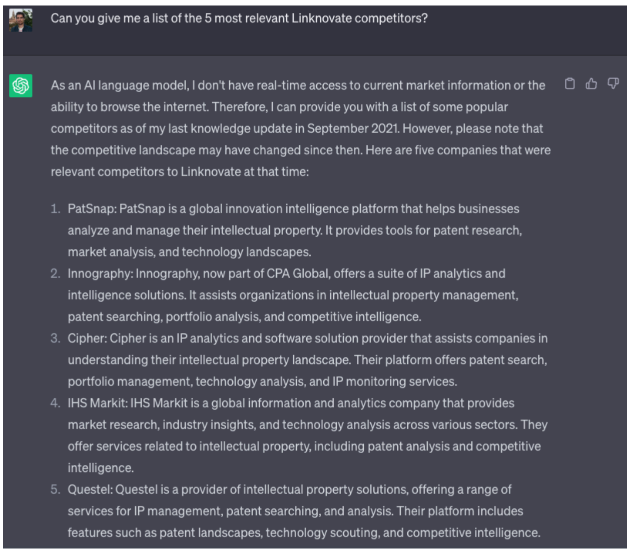


Figure 2: Screenshot of a prompt in chatGPT to get the competing companies for Linknovate.

This problem can be solved also manually by using Google or other generic search engines, but it requires a much longer time of manual analysis. Furthermore, it is important to understand how to set up the search to get right results and the analyst needs a good knowledge of the topic under study to make her final selection.

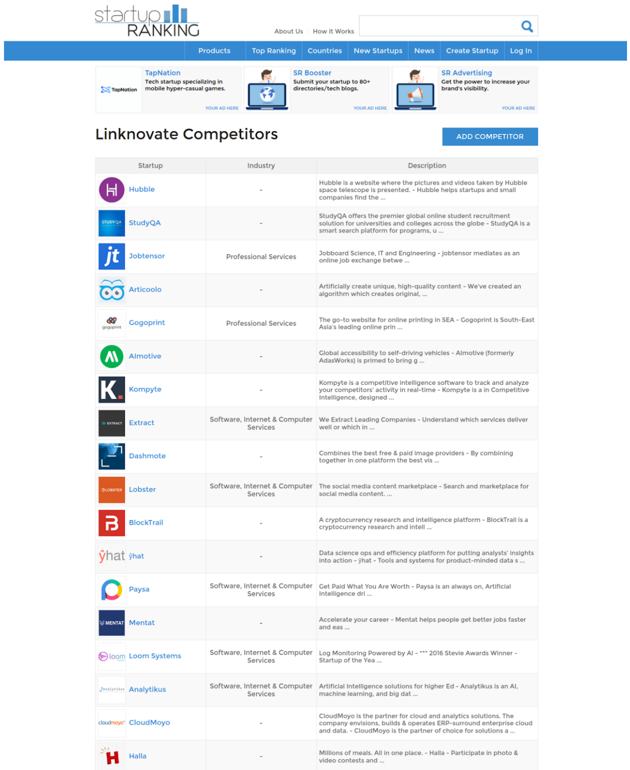


Figure 3: Screenshot of a service competing with Startup Radar, Startup Ranking [1]

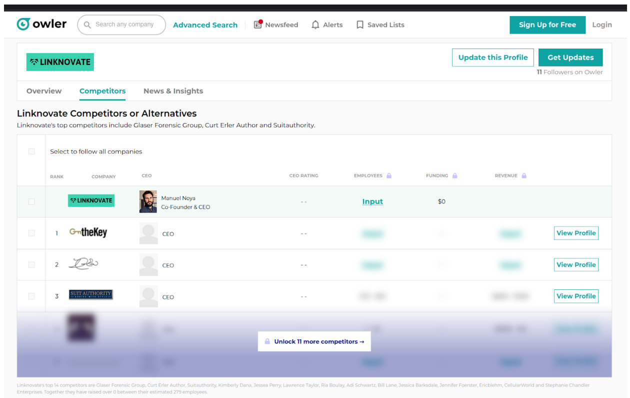


Figure 4: Screenshot of a service competing with Startup Radar, Owler.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10557413

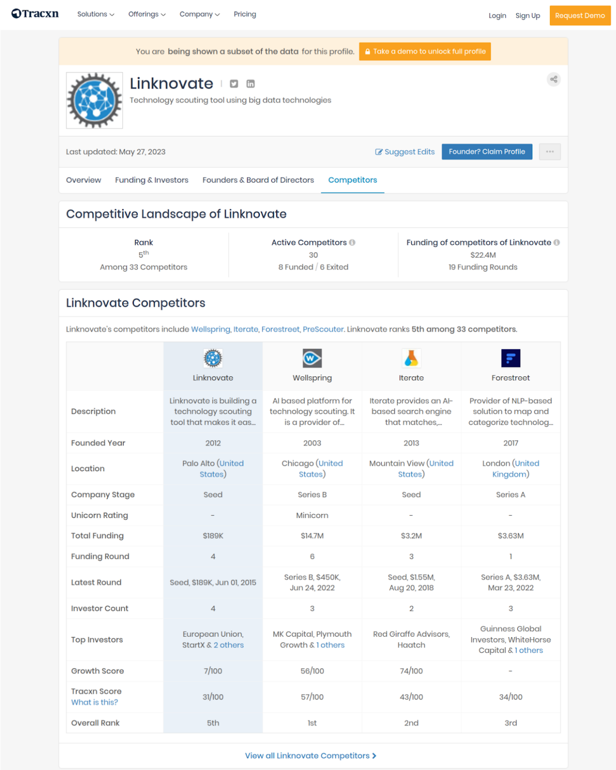


Figure 5: Screenshot of a service competing with Startup Radar, Tracxn [2]

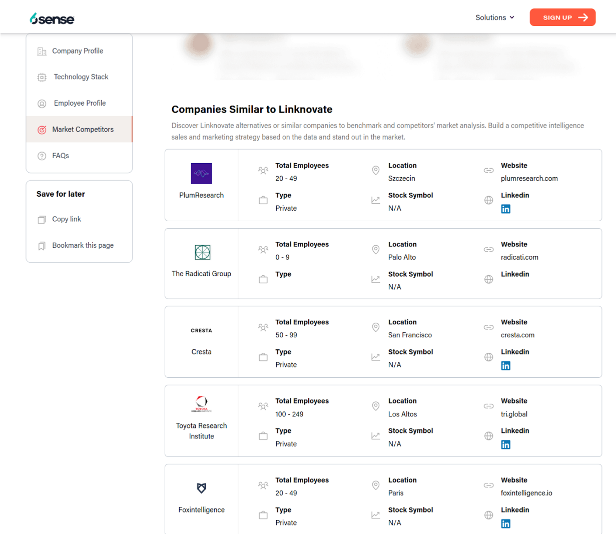


Figure 6: Screenshot of a service competing with Startup Radar, 6sense [3]

As it can be observed the precision and recall of these solutions is insufficient, presenting companies that are somehow in the same (generic) field: research, business intelligence, or simply AI; and other times simply missing the target.

Finally, organizations may depend on their internal networks and the self-assessment of the startup/organization they are considering for investment, while larger companies could seek assistance from their network of consultants, university professors, and academics.

APPROACH

Similarity explanation

During the SR project, we tested with real use case owners 4 different types of explanations. All of them are text-based explanations. Following [4] we opted for two approaches: full natural language and keyword (tag) based. Previous results in the field of movie recommendation demonstrated that textual descriptions were more appropriate for the user than tag-based explanations, but we wanted to test in our particular use case. So finally, we tested:

- Brief company description: Describe each one of the matched companies with a few sentences extracted from the Linknovate profile description.
- Keywords: These keywords are chosen based on their relevance to the matching process of the two companies and they should be a quick overview of their know-how.
- Extractive Summaries: Text extractions from a couple of documents (patent, research publication…) associated with both matched companies (one for each), which talks about a common topic, indicative of their main knowledge and activity.
- Abstractive Summaries: Key information is extracted from a couple of documents and new sentences are generated from it to create a summary.

Based on the scores and comments of the approached testers, the option that is more useful for them is the one based on the company description.

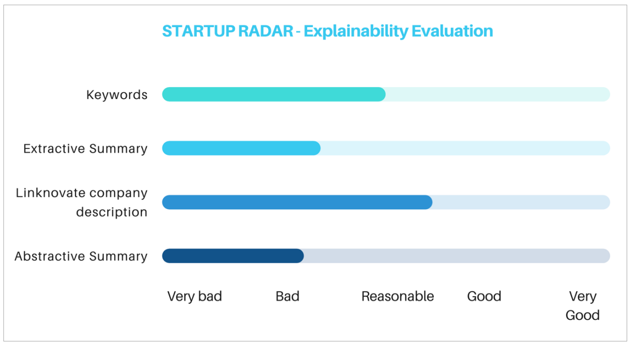


Figure 7: Average beta tester scores for each type of proposed explanation

The company's description has the correct amount of information to earn trust from users making them happier with the suggestion. In other words, they are short and clear enough to understand in a few sentences if two companies are similar or not. The keywords approach has a good acceptance as well, because it is a straight way to communicate key information to users. So, we understand that a combination of both could be a great solution to explain the recommendations to the users.

Companies like Netflix have also demonstrated [5] the importance of graphical representations when recommending items to users. Therefore, we additionally designed a visual proposal to provide a clear and intuitive explanation.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10557413

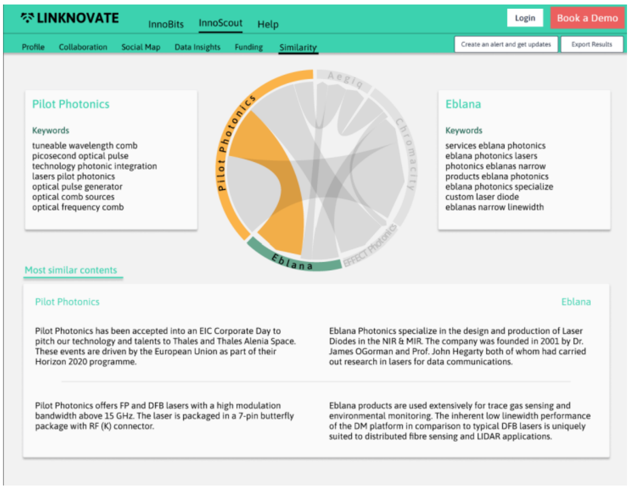


Figure 8: Example of Chord similarity graph draft supported by textual explanations

This Chord diagram [6] visualizes the similarity strengths between the anchor company (here: Pilot photonics) and its four most similar companies (here: Eblana, Chromacity, EFFECT Photonics, and Aegiq). The thickness of the arrows connecting two companies represents the similarity level between the anchor company and its match.

Step 1: Select the company to be analyzed

As the analyst begins to type the name, the tool suggests companies with profiles on the platform which match that name:

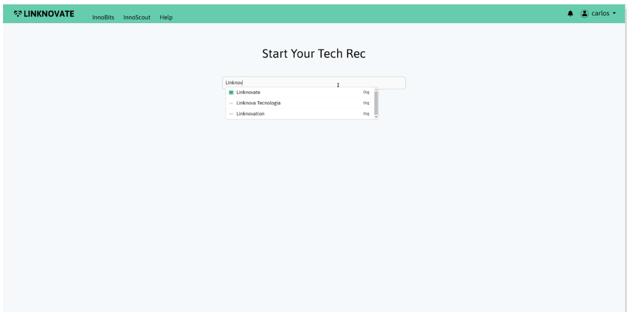


Figure 9: Screenshot of the selection process of the organization to be analyzed in Startup Radar

If there is no match on the platform, it allows for new company creation, just by providing the company's name and website - for later crawling and adding info to the newly generated company profile, and LKN enriching procedures from LKN data sources (like patents, scientific publications, grants, etc).

This information will be the support in the subsequent recommendation of suggested similar companies.

Once the desired company profile is selected, "Search" is the next step.

Step 2 – Indicate the innovation area

The analyst indicates in SR the innovation area of interest for the company under study.

This area of innovation will be expressed as "search terms" in the form of keywords.

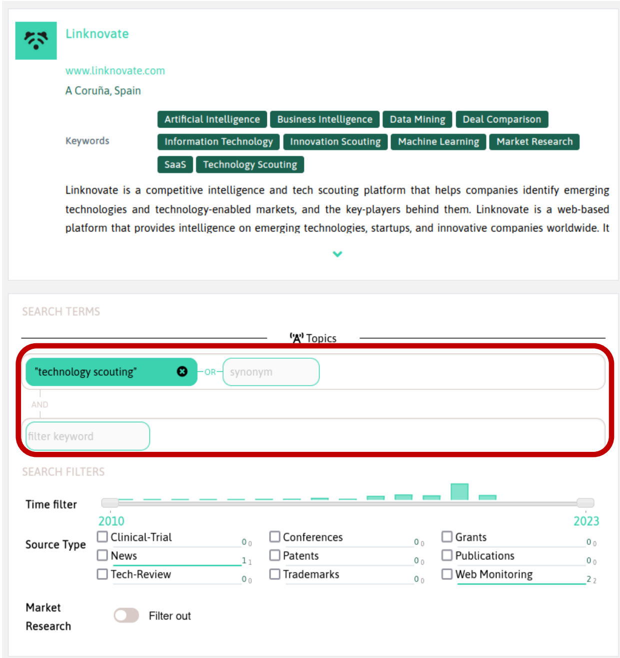


Figure 10: Search terms configuration in Startup Radar

The analyst can use the operator AND (to combine different keywords), the operator OR (to include synonyms) and other punctuation (e.g., quotation marks for keyphrases) to get more specific search results.

Step 3 – Similar organizations

Based on the selected company and the innovation area indicated as keywords, SR offers a list of similar organizations:

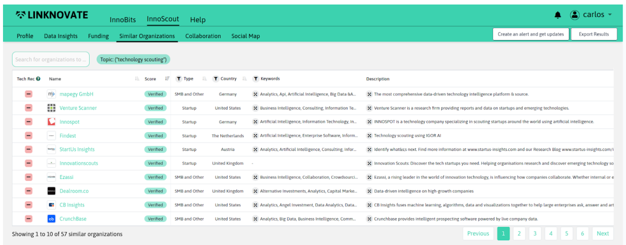


Figure 11: Selection process of similar organizations in Startup Radar

The platform shows the company list ordered by relevance based on an internal score. In this way, the analyst can see first the most relevant suggested companies. This list can be ranked by company size, country, etc.

The analyst can individually select/deselect each suggested company using the icon to the left of its name, in order to add them to her study. She can also manually add similar companies not included in the tool's suggestion list, by typing in the text box in the upper left corner and selecting it (if it does not exist, it can be created as in step 1).

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10557413

Step 4 – Customized Recommendations

After confirming the previous list selection, the tool generates a set of possible recommendations in the indicated area of innovation based on these selected similar organizations:

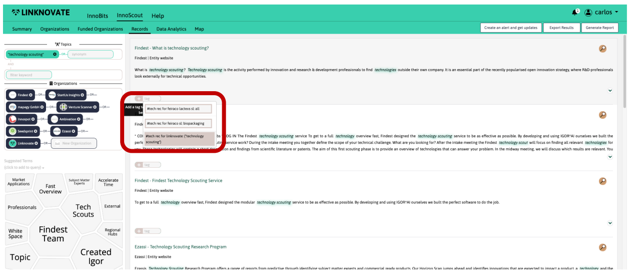


Figure 12: Recommended actions/documents selection in Startup Radar

The analyst can select the desired innovation actions (documents) by clicking on the associated "tag" field for each one of them and selecting the suggested label (automatically generated) with the name of the ongoing study.

The analyst can check more details about any of the actions/documents by clicking on the title and going to the original source.

All the selected recommendations are saved as future suggestions for the next analysis about companies in the same area of innovation.

Step 5 – Tagging and export

The innovation actions/documents selected by the analyst will be labelled with the automatically created descriptive name:

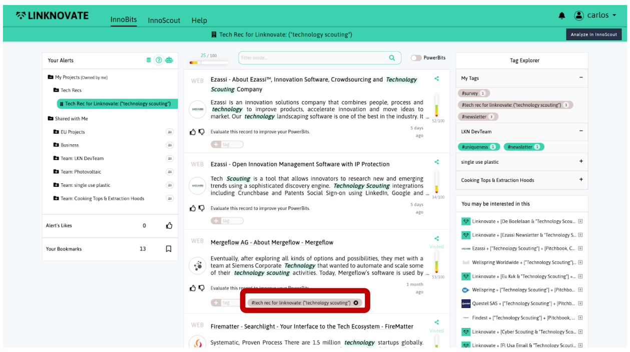


Figure 13: Tagged actions/documents to be included in the ongoing Startup Radar report

At the end of the analysis, an automatic report can be generated with all of them, in order to share this report with the company under study.

The use case relies on open source technologies. For the relation extraction tasks, we opted to utilize spaCy, a Python-based framework, to train a model using our proprietary data. Regarding the company similarity task, we constructed the recommendation system by employing ElasticSearch as the foundation for our similarity calculations. Utilizing the unique characteristics of our data, we harnessed ElasticSearch's capabilities to identify distinctive and descriptive terms within the company documents, enabling us to rank similar companies.

REFERENCES

[1] https://www.startupranking.com/linknovate/competitors

[2] *https://tracxn.com/d/companies/linknovate/__vyeFcmsA2njRdVbe1mVyEwTGXS4jj3u59MxQMmuo1W4/competitors*

[3] *https://6sense.com/company/linknovate/5c3b00f6d55ae49f1b77305d*

[4] *Crowd-Based Personalized Natural Language Explanations for Recommendations*. Shuo Chang, F. Maxwell Harper, and Loren Gilbert Terveen. 2016. In Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16). Association for Computing Machinery, New York, NY, USA, 175–182.

[5] *Artwork personalization at netflix*. Fernando Amat, Ashok Chandrashekar, Tony Jebara, and Justin Basilico. 2018. In Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18). Association for Computing Machinery, New York, NY, USA, 487–488.

[6] *Chord diagram invariants of tangles and graphs* (No. ULP-IRMA-1995-15). Kassel, C., & Turaev, V. (1995). Inst. de Recherche Math. Avancée.

https://doi.org/10.5281/zenodo.10581871

REACHING BEYOND ETHICS – PERSPECTIVES OF HUMAN RIGHTS EDUCATION ON AN OPEN SEARCH INDEX

M. Theophil*, Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau, 76829 Landau, Germany

Abstract

In order for an open and free search index to be a lasting alternative to established digital infrastructures processes of digitalization need to be addressed in a way that includes their profound influence on the human condition. Within the field of education that means to didacticize digitalization not solely in terms of application and usage. In addition to a well-versed media use (including the use of search engines), it should be asked how digitalization processes affect people's relationships to themselves, to the world and their fellow human beings. It is only against this backdrop – which reaches beyond ethics – that allows for questions of 'wrong' and 'right' to be asked in a sensible manner in the first place. The insight into a fundamental (digital-) technological impact on the human being – which is captured in this paper mainly with concepts from cultural sciences (first and foremost with Felix Stalder's notion of digitality) – makes an educational process plausible which obtains its normative orientation from human rights education. Reasoning for this orientation offers a normative counterweight to possible problematic developments within the 'culture of digitality' (Stalder) – which will be illustrated in this paper conclusively with the example of politics.

THE 'FABRIC' OF CULTURE, SOCIETY AND HUMANESS IS CHANGING

The task to index, analyse and to make available content from the internet can be compared to the role of a librarian. There is, for example, the need to keep the inventory up to date, to structure the information in an appropriate manner and to supervise these processes with an eye for legal and moral boundaries [1]. However, there are limits to this analogy: While past librarians undoubtedly already dealt with a vast amount of information, this information was nevertheless restricted – especially in its production – to a certain group of people. There were (and still are) several gatekeepers whether it be within academia, publishing or institutions for distribution (like, for example, libraries) who rendered specific kinds of information more valuable than others and, accordingly, decided what is published (and in which way it is published and displayed for the public). Thus, what Marshall McLuhan called in the mid-20th century the Gutenberg-galaxy was – at least to some extent and especially during Early Modern Times – still overseeable.

This changed drastically with digitalization: The amount of people contributing information of all sorts as well as the information itself increased immensely. The consequences of this development are ambivalent: On the one hand there is the possibility for more diversity due to the disempowerment of former gatekeepers [2]; on the other hand, it is – today more than ever – impossible for a single person to navigate the available information without any help. With this task humans need assistance – which they get more and more from technology. What is interesting in this context is that some of the forerunners who helped to finance and push forward the required technical assistance positioned themselves quite clearly in opposition to the above-mentioned advantage of more diversity. As early as the 1990s Peter Thiel who later on became the co-founder of PayPal and an influential investor had objected multicultural ideals for society advocated by some students and lecturers at Stanford University [3]. Instead, he relied on technical progress together with capitalism. In an article from 2009 – which almost reads like a confession – Thiel predicted a deadly race between technics and politics; unmistakably, the outcome he hoped for was for political influence to vanish altogether. „The fate of our world may depend on the effort of a single person who builds or propagates the machinery of freedom that makes the world safe for capitalism" [4].

Confronting Thiel's position – which can be called a 'technicism' or a 'solutionism' [5] – with ethics leads to unequivocal results: There is no defending Thiel's stance without giving up basic principles of a democratic society characterized by inclusivity and equality. Therefore, it is not the obvious that is interesting about Thiel's point of view but what lies beneath his hope for the invention of what he calls a 'machinery of freedom' – which brings us back to the analogy between a search index and a library introduced at the very beginning. It is this very phrase that lets an expectation of salvation resonate within Thiel's text and that elucidates that we do not only have to deal with an ever-growing library, that is, with much more information and with new ways of gatekeeping but also that the library expanded (and still expands) in a way that is inseparably interwoven with human life and identity formation. Technology is not the 'other' we are confronted with but it is 'within us'; it shapes our way of thinking, our dreams and hopes, and our perception of the world as well as of each other – which eventually lets some of us draw the conclusion that freedom is something to be installed by machines. In short, the entire 'fabric' of society and what it means to be human has changed with digitalization (as both had changed in the past, for example when Johannes Gutenberg invented letterpress printing).

The development of a free and open search index has to keep this insight in mind while advancing. The (seemingly) more pressing questions of (e. g.) market power, transparency of algorithmic decision making or digital self-determination have to be accompanied by the basic notion that what we consider to be 'free' and 'open' has been changed

*email address: theophil@rptu.de

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10581871

by this very technology we are trying to apply these attributes to. In order to address, for instance, ethical questions in a satisfactory manner they have to be asked against the backdrop of an ever-changing conditio humana which prevents them from being abstract considerations about 'right' and 'wrong'. This perspective – reaching beyond ethics – is important even for the most free and open search index or the most transparent algorithms because it raises awareness for the question what qualities of being human are not sufficiently translatable into processes of data acquisition and data analysis. What may present itself as problematic on this level could– as I would like to propose – in the field of education be counterbalanced with the help of human rights education. In this paper I would like to make this normative orientation plausible by firstly confronting it with the way didactics for the most part deals – up to today – with processes of digitalization. Having a basic anthropological starting point, human rights education helps illuminating the shifts in the human condition caused by digitalization and can, at the same time, intervene with its widely accepted norms. I will then illustrate this with the example of politics and by doing so also further answer the question why these shifts are of significant importance for setting up alternative digital infrastructures like an open and free search index.

THE SHORTCOMINGS OF DIDACTICS AND HUMAN RIGHTS EDUCATION AS A COUNTERWEIGHT

In textbooks for social studies used at German schools pupils are more or less left with their preconceptions of what digitalization means. Without clarifying basic notions, it's almost inevitable for the textbooks to deal with the topic on a mere superficial level. In many cases this means establishing a dichotomy of dangers and possibilities of the digitalization and then attribute certain phenomena to either the chances or the risks – with a clear focus on the latter [6]. Fundamental changes beyond this 'danger prevention' are almost entirely neglected. Instead, the focus of the textbooks lies on strengthening media competency, meaning first and foremost to convey to pupils to use digital technology (including search engines) in an appropriate manner. This goal resonates with positions from cultural and educational politics in Germany (as well as from the European Union). For example, the "Digital Competence Framework for Educators" (DigCompEdu), developed by a research centre of the European Commission, aims almost exclusively for a self-activating and efficient usage of digital devices [7]. The same is true for a strategic paper published by the German "Conference of Ministers for Education" – titled "Competences in a digital World" [8]. The paper was updated in 2021; however, its focus on media competency remained.

The fact that a usage-oriented perspective on digitalization has its shortcomings was pointed out quite frequently. Werner Friedrichs, for example, remarked that understanding new technology merely as a tool for acquiring information is not sufficient. Instead, it has to be acknowledged how technology interweaves in a reciprocal way with human life [9]. Similarly, Benjamin Jörissen and Lisa Unterberg argue for a perspective beyond media competency in order to grasp the profound and complex transformation processes of digitalization bring about [10]. A prominent and often used concept not so much to counter but to complement media competency (which stays undoubtedly important) is Felix Stalder's notion of digitality (German: Digitalität). Stalder describes processes of digitalization as a deeply rooted cultural change which influences the constitution of social meaning, making it impossible to differentiate between the spheres of the analogue and the digital [11]. The assumption that there is a constant interdependency between technology on the one hand and humans on the other hand provides Stalder's approach with a historical perspective – asking on which societal and cultural phenomena digitalization had built on – as well as it renders digital processes as open for human influence and intervention.

Stalder identifies three main characteristics of digitality – referentiality, new kinds of communities and algorithmicity. While referentiality deals with the increase of information and the way it loosens existing societal bonds, but establishing new ones at the same time, the notion of algorithmicity grasps the assistance of technology to help humans navigate the myriads of new information. Today, a considerable fraction of decision-making is already handed over to algorithms. The unprecedented possibilities of data acquisition and its algorithmic analysis can lead to a life curated by technology, promising humans an existence in comfort and security. The main objective of many companies in this sector is to make search engines obsolete altogether by knowing the needs of the people before they become conscious to themselves. In order to achieve that as much of cultural and social praxis as possible has to be dissolved into data. Relying solely on statistical predictability and patterns when dealing with culture and society can be conceptualised as a rigid regime of control [12]. Within such a regime the future is nothing more than a foreseeable and prolonged version of the present. A merely calculative approach toward reality implies the risk of people adapting with their perception and behaviour to such an approach – resulting in a world in which only those phenomena are visible and considered important which can be represented and measured by data. It would be a world to Peter Thiel's liking because matters of freedom and self-determination would be left to technology without any human 'contamination'. (It would be a world, for instance, in which machine learning is mainly advancing with the results of previous machine learning.) In short, the real danger lying within algorithmicity is not that technical instructions or machine learning becomes so 'intelligent' that it can outsmart humans but that humans behave more and more like the technology they invented.

This potential problem could be described in more detail with long-established notions like 'alienation' or 'reification'. What thinkers of the past labelled, for instance, the 'one-dimensional man' (Herbert Marcuse) or a merely 'instrumental reason' (Max Horkheimer) was based on

https://doi.org/10.5281/zenodo.10581871

analyses of economic structures or paradigms prevalent within academia (e. g. to give preference to quantitative methods). The alienation brought forth by technology was and still is inseparably intertwined with these structures – a relation which has, quite surprisingly, for present conditions not been extensively researched so far. Presumably, such research will offer explanations why people actively participate in establishing a regime which restricts their own freedom [13]. And the fact that freedom itself – distorted in a certain way – can become an instrument of power remains true even when the economic logic behind digital infrastructures in general and web search in particular has been restricted or even eliminated. There is no getting around the fact that selection is inevitable in this context – decisions need to be made what elements of society and culture to make visible, and what should remain invisible. And these decisions should not be left, as I argued for above, solely to technology. Within liberal societies they need to be subject to democratic processes and open debate – which, by the way, also means that drawing a clear-cut picture of a desirable future would be deceiving from the start.

Understanding reality as being open for humans to structure and shape it, is a core principle of human rights education. And the goal to strengthen solidarity and freedom among people in the present [14] is not achievable without taking the interwoven relation between technology and human beings into account. If this relation is neglected, one neither meets the standards of digitalization nor those of human rights education. Both are in need, so to speak, to be liquified; meaning that they should not be considered as static entities but as something characterized by processes and constant change [15]. Only such an approach prevents from dealing with digitalization in a merely superficial manner (which didactics is, as outlined above, often guilty of) and, furthermore, it prevents human rights to be nothing more than constantly repeated incantations for a better future. Instead, human rights education aspires to bring about cultural transformations [16] – against the backdrop of problematic developments within the ‘culture of digitality’. Avoiding the chimera of a total controllability and predictability of human life can be achieved by relying on the “utopian surplus” [17] of human rights [18].

In order to do that education needs to rely heavily on interdisciplinarity; it has to aim at what Wolfgang Klafki named – as early as in the 1980s – an interconnected thinking [19]. Such an approach is able to illuminate phenomena in their complexity. Furthermore, education has to aim for a non-affirmative self-determination – which is a kind of self-determination that is critical toward its own conditions and, thus, opens the process of education towards goals which have to be determined during this very process [20]. It is exactly this kind of self-determination which is also crucial for the development of digital infrastructures which can be considered to be free in a non-instrumental way.

POLITICS AS AN EXAMPLE

It is not merely, for example, our privacy that is violated by large tech-companies, but the entire grammar of social interactions is changing as well as of experiencing the world and perceiving each other. This is per se not something that calls for a lament because the conditio humana is ever-changing. But it definitely needs to be criticised to the extent of problematic developments becoming visible against the backdrop of democracy and human rights. It’s crucial to understand that even when there may be, one day, total self-determination in practice within the digital sphere there can nevertheless be questionable aspects present at the same time.

This interdependency and reciprocal influence between digital and analogue sphere (a differentiation which cannot be maintained, as outlined above, but is nevertheless helpful for purposes of clarification) can be illustrated with politics: Not lively debate but quantification, simulation and surveying constitute more and more the core of politics – which is a gradual process going on for a couple of decades now; consequently, technocrats take control by “administering pre-defined necessities and constellations which are [allegedly; MT] without any alternatives” [21]. These increasingly technocratic processes eliminate what Jürgen Habermas (among others) calls ‘reasoning’ (German: Räsonnement) and ‘deliberation’ – both elements which constitute an open debate characterized by the exchange of arguments and the aim to achieve a consensus. Back in the early 1960s Habermas identified political parties, associations and unions as key players in disengaging people from deliberation. The public is primarily needed for the acclamation of decisions that were already made [22]. Today it has additionally be reckoned with the omnipresence of surveying and the infrastructure of the digital sphere which both rely on the mere presentation of private opinions instead of helping to form political opinions. (Considering the latter this is especially true for social media.) What is left out here is the very process of deliberation. As Hartmut Rosa just recently put it concisely, aggregated private opinions are not be confused with deliberation [23]. Especially the possibilities of Big Data enforce the illusion that the will of the people is not something that is constituted by debates but instead something that is accurately measurable. Again, the issue at hand cannot be found on a superficial level – which would mean in this case that big tech-companies restrict the freedom of opinion in an active manner. Instead, it has to be addressed that what we consider to be an opinion changes in a way that contradicts the notion of ‘opinion’ itself by aligning it to the infrastructures of the digital sphere.

One potential issue for the future will be that an open search index exists side to side with ongoing rationalities of quantification, short-sided rationality and reification. These rivalling ‘logics’ will constantly challenge the search index and make it hard for its openness and freedom to prevail. A structurally similar problem presents itself in setting up the index: The network to develop and continuously evaluating, monitoring and moderating the index may not be established from scratch but vast areas of the public sphere are – due to its current state – bound to remain alien to the network’s core principles. This state of the public sphere in question is characterized by giving priority to

https://doi.org/10.5281/zenodo.10581871

economic measures, by individualization, (self-)dramatization and emotionalizing – all traits with a tendency to contradict or lead astray democratic values. Again, with the help of Habermas' "Structural Transformation of the public Sphere" it's possible to realize that these processes are not something new. Problems to engage people in political debate and to form a public sphere which is able to gain the attention of societal majorities (which is according to Habermas key for a democracy to function) were also present during the emergence of television in the mid-20th century.

Nevertheless, there has been – as Habermas calls it – a "centripetal pull" [24] of what can be called classic mass media. In contrast to that, the public sphere of the internet is fragmented from the start. Following Felix Stalder – as outlined in the introduction – this can be on the one hand described as a strength in that debates are opened for a more diverse chorus of voices. But on the other hand, there are undoubtedly centrifugal forces at work which may lead (or perhaps already have been led) to a society unable to determine which topics are worthy of discussion. While these ruptures take place the 'old' mass media – e. g. newspapers, radio, television – is more and more feeling the pressure to adapt to the new regime which reinforces the problems even further [25]. The results are what Habermas labels as "semi-public spheres" [26] – odd mixtures of private and public spheres. This development, again, points into the direction of a fundamental change: For example, it is not fake news themselves – or even their increasing appearance – that is problematic for the political public sphere but the fact that this very sphere is distorted in a way which makes it increasingly difficult to identify fake news as such [27]. So, the underlying issue being that our understanding of truth and reality slowly but steadily shifts means that simply identifying lies in the form of fake news falls short to the backdrop of the problem at hand.

CONCLUSION

To conclude, even setting up the most open and transparent digital infrastructure should not overshadow the fact that there are (and ever will be) qualities of life and human interaction [28] which cannot be adequately captured by data and its statistical interpretation. This aspect can be – as illustrated in this paper – made visible in the field of education with the help of human rights education. Being critical in such a fundamental way may prevent an overarching lock-in-effect which nowadays adds up to existing economical lock-in-effects [29]. With an open and free search index and digital self-determination in place people might as well opt for their lives being curated by technology – possibly for the very reason that it is a technology they now have control over. But by doing so they have to be aware that certain qualities of their humanness, of what makes them unique and distinctive simply fall short.

REFERENCES

[1] `https://opensearchfoundation.org/wp-content/uploads/2019/12/A-free-and-open-Internet-Search-Infrastructure-web.pdf`

[2] F. Stalder, *Kultur der Digitalität.* Berlin: Suhrkamp, 2016, p. 23.

[3] D. Sacks and P. Thiel, *The diversity myth. Multiculturalism and the political intolerance on campus*. 2nd ed.. Oakland: Independent Institute, 1998.

[4] `https://www.cato-unbound.org/2009/04/13/peter-thiel/education-libertarian/.`

[5] M. Baum, "Freiheit in datafizierten Kontexten? Politische Betrachtungen des digitalisierten Neoliberalismus", in *Zeitschrift für Politikwissenschaft 32* (2021), p. 488. `https://doi.org/10.1007/s41358-021-00257-4`

[6] I. Heldt, M. Theophil and J. Bloise, "Digitalität als Leerstelle? Ergebnisse einer qualitativen Schulbuchanalyse", in M. Busch, C. Keuler (ed.), *Politische Bildung und Digitalität*. Frankfurt a. M.: Wochenschau Verlag, 2023, yet to be published.

[7] C. Redecker, *European Framework for the Digital Competence of Educators (DigCompEdu)*, Luxemburg: Publication Office of the European Union, 2017.

[8] KMK, *Bildung in der digitalen Welt. Beschluss der Kultusministerkonferenz vom 08.12.2016 in der Fassung vom 07.12.2017*, published by Sekretariat der Kultusministerkonferenz, Berlin, 2017.

[9] W. Friedrichs, "Electric Voting Man. Ausblicke auf posthumanistische politische Bildungen in der augmented democracy", in L. Möller and D. Lange (ed.), *Augmented Democracy in der Politischen Bildung. Neue Herausforderungen der Digitalisierung*, Wiesbaden: Springer, 2021, p. 9. `https://doi.org/10.1007/978-3-658-31916-8_2`

[10] B. Jörissen and L. Unterberg, "Digitalität und Kulturelle Bildung. Ein Angebot zur Orientierung", in B. Jörissen, S. Kröner and L. Unterberg (ed.), *Forschung zur Digitalisierung in der Kulturellen Bildung*. München: Kopaed, 2019, p. 19. `https://doi.org/10.25656/01:18486`

[11] F. Stalder, *Kultur der Digitalität*, loc. cit., pp. 16-19.

[12] A. Nassehi, *Muster. Theorie der digitalen Gesellschaft*. München: C. H. Beck, 2021, p. 43.

[13] H. Marcuse, *Der eindimensionale Mensch. Studien zur Ideologie der fortgeschrittenen Industriegesellschaft*. Neuwied and Berlin: Luchterhand, 1970, p. 27.

[14] M. Bahr "Das Landauer Manifest zur Menschenrechtsbildung. Profil einer Initiative an der Universität Koblenz-Landau", in M. Bahr, B. Reichmann and C. Schowalter (ed.), *Menschenrechtsbildung. Handreichung für Schule und Unterricht*. Ostfildern: Matthias Grünewald Verlag, p. 45.

[15] I. Heldt, *Die subjektive Dimension von Menschenrechten. Zu den Implikationen von Alltagsvorstellungen für die politische Bildung*. Wiesbaden: Springer, 2018, p. 48.

[16] K.-P. Fritzsche, "Menschenrechtsbildung in Zeiten eines erstarkenden autoritären Populismus", in S. Kenner and D. Lange (ed.), *Citizenship Education. Konzepte, Anregungen und Ideen zur Demokratiebildung*. Frankfurt a. M.: Wochenschau Verlag, 2018, p. 191.

[17] K.-P. Fritzsche, "Menschenrechtsbildung in Zeiten eines erstarkenden autoritären Populismus", loc. cit., p. 192.

[18] J. Habermas, "Überlegungen und Hypothesen zu einem erneuten Strukturwandel der politischen Öffentlichkeit", in M. Seeliger and S. Sevignani (ed.), *Ein neuer Strukturwandel der Öffentlichkeit?, Leviathan 49/37* (2021), Baden-Baden: Nomos, p. 473.

https://doi.org/10.5281/zenodo.10581871

[19] W. Klafki, *Neue Studien zur Bildungstheorie und Didaktik*. 6th ed.. Weinheim and Basel: Beltz, 2007, p. 65.

[20] D. Benner, "Bruchstücke zu einer nicht-affirmativen Theorie pädagogischen Handelns", in *Zeitschrift für Pädagogik 28/6*, 1982, pp. 952-955.

[21] F. Stalder, *Kultur der Digitalität*, loc. cit., pp. 206.

[22] J. Habermas, *Strukturwandel der Öffentlichkeit. Untersuchungen zu einer Kategorie der bürgerlichen Gesellschaft*. 5th ed., München: Luchterhand, 1971, p. 211-213, 235-237.

[23] H. Rosa, "Demokratischer Begegnungsraum oder lebensweltliche Filterblase? Resonanztheoretische Überlegungen zum Strukturwandel der Öffentlichkeit im 21. Jahrhundert", in M. Seeliger and S. Sevignani (ed.), *Ein neuer Strukturwandel der Öffentlichkeit?, Leviathan 49/37* (2021), Baden-Baden: Nomos, p. 256.

[24] C. Czingon, A. Diefenbach and V. Kempf, „Moralischer Universalismus in Zeiten politischer Regression. Jürgen Habermas im Gespräch über die Gegenwart und sein Lebenswerk", in *Leviathan 48, 1/2020*, p. 27. https://doi.org/10.5771/0340-0425-2020-1-7

[25] J. Habermas, "Überlegungen und Hypothesen zu einem erneuten Strukturwandel der politischen Öffentlichkeit", loc. cit,, p. 492.

[26] J. Habermas, "Überlegungen und Hypothesen zu einem erneuten Strukturwandel der politischen Öffentlichkeit", loc. cit,, p. 497.

[27] J. Habermas, "Überlegungen und Hypothesen zu einem erneuten Strukturwandel der politischen Öffentlichkeit", loc. cit,, p. 497-498.

[28] R. Becker, *Qualitätsunterschiede. Kulturphänomene als kritische Theorie*. Hamburg: Felix Meiner Verlag, 2021.

[29] https://opensearchfoundation.org/wp-content/uploads/2019/12/A-free-and-open-Internet-Search-Infrastructure-web.pdf

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10611196

CUSTOMIZABLE CATEGORIZATION OF DOCUMENTS IN EVIDENCE-BASED RESEARCH FOR BIO PHARMACEUTICS

Reihaneh Manteghi. Author*, Graz University of Technology, 8010 Graz, Austria
Eduardo Veas. Co-author[1 †], Graz University of Technology, 8010 Graz, Austria
Domagoj Segregur. Contributor[2‡], UCB Pharma, 1070 Brussels, Belgium
*,[1]also at Know-Center, 8010 Graz, Austria

Abstract

This paper describes a strategy that combines bottom-up topic extraction and personalized expert category classification to facilitate evidence-based research in drug development within the bio pharmaceutics field. Bio pharmaceutical scientists spend a significant portion of their time and effort looking for information and establishing relationships with existing scientific outcomes. Through evidence-based research, they seek to find published scientific studies towards answering a clinical question. While language models can assist with topic extraction, their specificity may not match expert categorization. To address this, the paper introduces the Health Optimization Tool (HOT), which combines personalized document classification with topic extraction. HOT allows scientists to train the document classifier according to their specific needs, leveraging the strengths of both approaches. The system offers a unified user interface with document and category overviews, enabling graphical or query-based searches. Using a personalized classifier on top of a language model enables the possibility of interactively updating the classification while retaining the strength of the topic extraction. The paper includes a proof of concept, detailing the creation of the database, fine-tuning of the language model, and preliminary study with experts.

INTRODUCTION

The ability to analyze and organize large collections of information, to draw relations between pieces of evidence is paramount to build knowledge. In the health related domains, Evidence-Based Research (EBR, also evidence-based medicine) refers to informing clinical or medical inquiries through the systematic and transparent use of prior research [7]. It is an ethical question. On the one hand, unnecessary research puts patients at avoidable risks. On the other hand, a requirement of replicability necessarily implies the identification of prior research. Hereby, experts screen articles from biomedical journals, usually accessible through online databases like PubMed, for example seeking concrete studies on topics like patient populations, diagnosis, treatments, adverse cases, and effects of public policies in medicine [8]. In the pharmaceutical sector, specifically in drug development, this task entails comprehensive retrieval of sources and organizing them based on provided categories for expert analysis.

Language models capture the use of language structures across a variety of linguistic contexts [15]. A language model supports NLP tasks such as named entity recognition, sentiment analysis or text classification [21]. For example, biomedical NLP has been successfully used for identification of diseases [20,21], identification of sound treatment studies [13,14], or classification according to medical concepts [18]. A common approach is to start from a pre-trained large language model, such as ELMO [15], BERT [9], BioBERT [12], XLNet [11] and fine-tune towards the desired task. However, identifying the right model for a task is not trivial [21]. Besides, language models still struggle with certain domain-specific terminology [16], and new or adapted vocabularies hinder generalization. Finally, experts may focus on different needs depending the background and goals, and classification of a document may differ. A more flexible approach is required.

Numerous tools using interactive exploration techniques [2–5] have been developed with the objective of assisting users in exploring relevant content; However, a notable limitation of these existing tools is their focus on a specific domain, which results in restricting their applicability across diverse domains. Additionally, the absence of customization options hinders users from tailoring the default classifier to suit their specific needs and preferences. For example, PubVis [2] is an interactive tool designed to facilitate the exploration and discovery of scientific publications. PubVis focuses on the interactive exploration and discovery of scientific publications through visualization and does not offer a customized classification however it provides users with personalized content-based article recommendations. Another example is Thalia [3] which is designed to assist researchers in retrieving relevant information from vast collections of biomedical abstracts by utilizing semantic search capabilities. Thalia leverages advanced natural language processing techniques to understand the context and meaning of the query terms, enabling more accurate and comprehensive search results. One recent contributions to the document screening and exploration methods is Carvallo et al [5] which presents a novel method for automatic document screening in the medical domain. The suggested approach shows potential in enhancing the efficiency and accuracy of screening operations, alleviating the load of manual screening, and enabling researchers to stay up to date with the newest medical literature by leveraging word and text embedding combined with active learning techniques.

* rmanteghi@know-center.at
† eveas@know-center.at
‡ Domagoj.Segregur@ucb.com

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10611196

APPROACH

The context of this work is a customized document classification system for the pharmaceutical industry that aims to make it easier for chemists, pharmaceutical professionals, and medical researchers to access the medical literature they need. Users expect the system to categorize documents in accordance with categories they (or the organization) have provided or defined. This section consists of 1) System Architecture, 2) User Interface, 3) Classification, and 4) Topic Modeling sub-sections.

System Architecture

The main system architecture is presented in Figure 1.

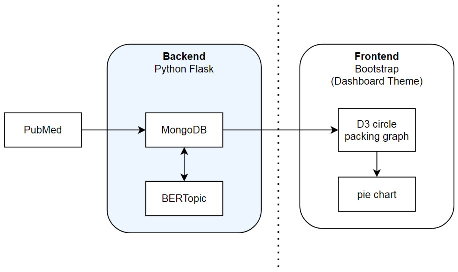


Figure 1: The system architecture of the tool: A BERTopic model is trained and used to categorize new documents in the tool's back-end using the input data that is taken from PubMed. In the front-end, a graphical user interface displays the data and offers interactive exploration options for users.

The Medical Subject Headings (MeSH)[1] are the categories used by the prototype here. Users may assign documents to several categories in hoping to improve classification since categories are frequently defined in a vague manner.

User Interface

The user interface of the tool can be observed in Figure 2.

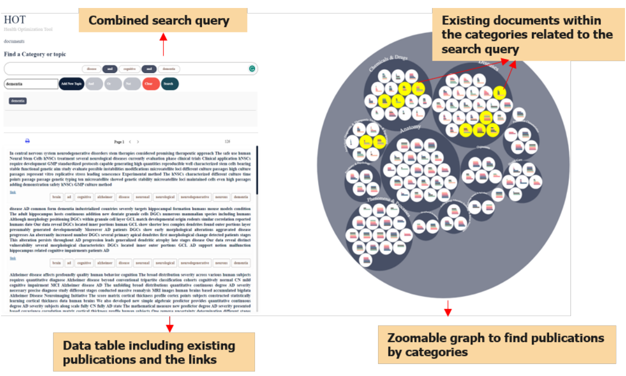


Figure 2: The tool's user interface including the graphical document classification, document table, and the search bar

The tool's source code is initially implemented in HTML and D3 JavaScript for the user interface, and in Python for models. Mongo DB is utilized to store both the extracted topics and data from PubMed and output results; Whereas JSON files are used to represent all other default data, such as category headings and their hierarchical structure. The tool's user interface consists of three main sections: a search bar, a graph with the titles of all the categories, and a document list. The search bar can be used by users to perform a search query. The table of search results shows the topics and categories that the documents were sorted into. Additionally, linked categories containing the entered topic highlight upon entry. If a category title is entered in the search bar, the graph will automatically zoom into that category and the results table shows the documents related to that category (Figure 3).

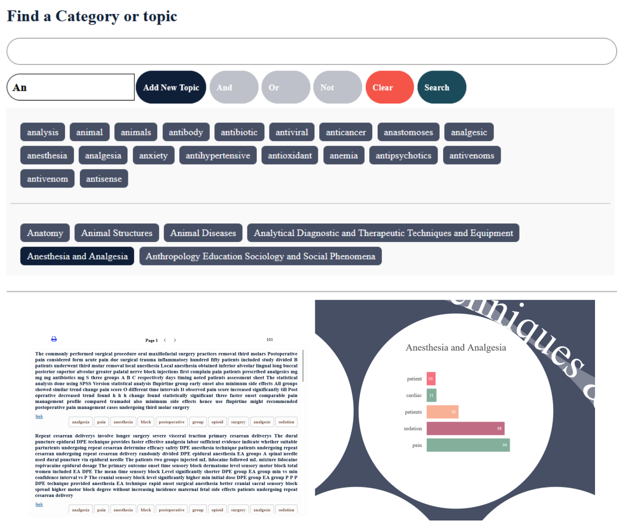


Figure 3: Enter the category title in the search bar to zoom into a specific category. The document table is also updated accordingly. The subjects displayed in the selected category are the five most-relevant topics to the category discovered in all associated papers to the selected category. The numbers on the bar charts show the frequency with which the topic appears within the category.

Topic Modeling

For extracting topics from text data, the topic modeling method BERTopic [6] is used. This method makes use of transformers and the c-TF-IDF (contrastive term frequency-inverse document frequency) algorithm to pinpoint the terms that occur most frequently in a particular text corpus. Document clustering is another service provided by BERTopic based on the identified themes. The method also offers the opportunity to train BERTopic models using labeled datasets, allowing us to categorize the documents using the trained BERTopic model by annotated content.

To apply BERTopic in a supervised manner to a dataset, as shown in Diagram 4, a collection of publications containing class labels is obtained by crawling the Pubmed [2] database. These labels correspond to the MeSH headlines.

[1] MeSH, Available online: `https://www.ncbi.nlm.nih.gov/mesh`

[2] Pubmed, Available online: `https://pubmed.ncbi.nlm.nih.gov/`

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10611196

Although MeSH contains multiple levels of headlines and sub-headings, for the purpose of this study, only two levels are considered, resulting in the extraction of 47 headlines and sub-headings from MeSH.

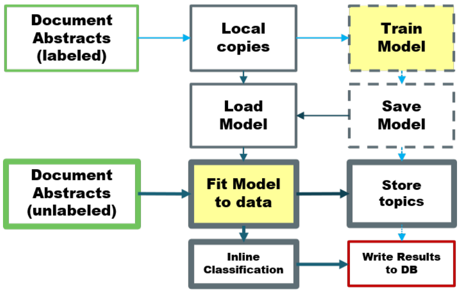


Figure 4: The tool's back-end, BERTopic, labels the dataset

For the document dataset, a dedicated crawler is configured for each category, extracting a set of 1000 documents per category. These documents consist of the publication abstract, author details, paper link, and a label indicating the corresponding MeSH category. In the next step, an additional set of 10,000 documents is crawled using a separate crawler, regardless of their MeSH headline. Topics are extracted using the same BERTopic model trained in the first phase. The model is configured to select and store the 10 best features according to their probability of frequency for each document. The topics are then vectorized, along with their corresponding probabilities.

The BERTopic model's performance on topic reduction for a total number of 10000 documents is tested using two clustering methods of UMAP and HDBSCAN. The results can be observed in Table 1.

Table 1: BERTopic topic reduction results

N neighbors	MinTopicSize	clusters
UMAP	HDBSCAN	
200	150	3 random noise
100	75	7 random noise
50	35	11 random noise
35	30	28 random noise
30	25	50 random noise
25	18	78 random noise

Classification

The tool can be configured to employ all kinds of textual data, but in order to develop the classification model, it first needs a list of default headlines and subheadings in at least two levels and tagged documents. We chose medical records as our use case.

For the document classification, a similarity matrix is used to evaluate the relative score of different classes (categories) in respect to the topics extracted from each distinct document.

Therefore, for each document, a similarity matrix $M_{i,j}$ (Matrix 1) including all extracted topics and the probabilities of that topic appearing in all existing classes(categories) is created as the following, where each row represents a topic and each column represents a class. Therefore, $M_{i,j}$ is a 10×47 matrix, where $P_{i,c}$ indicates the similarity between features for the feature i appearing in the class c.

$$M_{i,j} = \begin{bmatrix} p_{0,0} & p_{0,1} & \cdots & p_{0,46} \\ p_{1,0} & p_{1,1} & \cdots & p_{1,46} \\ \cdots & \cdots & \cdots & \cdots \\ p_{9,0} & p_{9,1} & \cdots & p_{9,46} \end{bmatrix} \tag{1}$$

In order to detect the highly related categories to each document, $p_{doc,c}$, the word embedding similarity scores in each column are summed up (Equation 2) to find the highest values for each class. $p_{doc,c}$ indicates the probability of a category being related to a document.

$$p_{doc,c} = \sum_{i=0}^{9} p_{i,c} \tag{2}$$

In the next step, n number of the classes with the highest values are selected for that document. The system chooses the categories with the greatest values as the most related categories to the document by creating a list of all probabilities for the document belonging to each category Cat_{doc_i}.

$$Cat_{doc_i} = Max[p_{doc_i,c_0}, \ldots, p_{doc_i,c_{46}}] \tag{3}$$

Documents are then labeled according to the values of the list 3 and the results are stored in the database.

EXPERT STUDY

The evaluation of our approach involves: 1) assessing the user experience and 2) evaluating the accuracy of topic modeling and 3) of document classification.

User experience

A number of tasks have been designed to evaluate the user interface, drawing inspiration from actual situations that mimic the actions taken by pharmaceutical experts in their search for certain information. Participants had to complete these predetermined assignments and find the answers to the corresponding questions. The users were instructed to perform separate searches for each topic listed in the "Topic" column of Table 5 and complete the table. They were asked to explore the tool's interface freely and intuitively while conducting their searches, and to verbally express their thought process. Verbal responses were recorded along with the computer screen.

As an example of their tasks they were asked to use the tool to investigate common symptoms of COVID-19 in children and to print their final results. They had the option to simultaneously search for multiple topics and explore the results section. Additionally, they were instructed to utilize the tool to research changes in the brain associated with Alzheimer's disease and print their final results. Similar to

https://doi.org/10.5281/zenodo.10611196

TOPIC	Total number of categories including the topic	Names of all categories including the topic	Total number of sub-categories including the topic	Names of all sub-categories including the topic. Please, add information on the frequency of the topic within each sub-category in brackets, e.g., diseases (13).
animal				
virus				
infants				

Figure 5: A sample of the designed tasks for user experiments.

the previous task, they were allowed to conduct multiple simultaneous searches and explore the results section.

As shown in Figure 2, 12 classes represent the main (parent) categories and the rest are sub-categories of the main classes. In this study, the customized version of the tool was designed to address a specific purpose, tailored to the needs of domain experts in a particular department of the target pharmaceutical company. Due to the limited number of domain experts who would be utilizing this tool with the customized data, our user study consisted of only three participants. Despite the limited sample size, it is significant to note that these individuals have a high level of domain expertise, making them valuable contributors.

Figure 6 summarizes the means of users rankings in Visual Analogue Scales (VAS) [3] for each existing features within the tool. Users' feedback indicated that the existing features and the graph representation of the tool were found useful by all participants. However, some users encountered difficulties in interacting with the tool, except for the user who had close involvement in the development process. The provision of a manual guide was highly appreciated by the users, as it enhanced the usability of the tool. Additionally, offering various solutions for navigation through the graph, such as using the mouse wheel, was considered very beneficial by the users.

Usability of existing features:

	Raw Mean
Graphical overview of (sub-) categories and topics	78.33333333
Document list overview	71.66666667
Search button alternatives	55
Topic categorization (i.e., in-graph categories and sub-categories)	75.33333333
Visualization and overview of top five topics per sub-category	68
Bar chart information on the total number of topics within each sub-category	61.33333333
One-page (as opposed to multiple-pages) user interface	66.33333333
Highlighting sub-categories including the searched (i.e., entered) topic	79.66666667
Search result download	85.66666667
Search query entering options	82.33333333
Navigation between (sub-) categories	53
Colours of the graph background	62

Figure 6: Summarized means scores of the visual analogue scales (VASs) usability of existing features

Topic modeling

For the identified topics, participants were asked to review the abstracts associated with each task and evaluate the accuracy of the detected topics compared to their conventional search method which is through public search engines (e.g., PubMed). In the user study, 100 randomly labeled documents were selected from each category, amounting to a total of 1000 documents per category. These documents were reviewed by a knowledgeable practitioner. It was found that approximately 10 percent of the documents showed only marginal relevance to their assigned classes while the rest were highly relevant. An expert reviewer was then tasked with going through the database and finding any documents that had been incorrectly categorized. No mislabelled documents were reported by the expert reviewer as a result of the document classes being closely related to one another. The assessment study did, however, point out that numerous documents had just a slight relevance to the classes they were allocated, suggesting the possibility for improvement. It had been proposed that adding more precise classes to the category hierarchy might improve the classification. Deeper classes provide a more precise assignment of extracted topics from each document to their own focused classes by making the boundaries between categories across the entire database more clearly defined.

[3] Visual Analogue Scales, Available online: `https://www.physio-pedia.com/Visual_Analogue_Scale`

Document classification

Participants were asked to assess the binary probability of the assigned category for each document, indicating whether it is "relevant" or "irrelevant" by assessing the final obtained search results from the experiments and comparing them with similar search results from the PubMed database. Compared to searching through other databases, evaluation findings showed that users could obtain relevant results for the target category significantly more quickly. However, due to the database's small quantity of documents, it was not always possible to find all the relevant material that users were looking for.

CONCLUSION AND FUTURE RESEARCH

In the subsequent stage, our research aims to assess the efficacy of the customized classification module, wherein users have the possibility to upload batches of 100 labeled documents per category. This process enables users to fine-tune the classification of each category according to their specific requirements and preferences. For instance, users can exclude a broad range of sub-categories and shift the focus of a category towards more relevant ones.

REFERENCES

[1] Antonellis, I.; Bouras, C.; Poulopoulos, V. "Personalized News Categorization Through Scalable Text Classification." *8th Asia-Pacific Web Conference* **2006**, 1795-1805, doi: 10.1007/11610113.

[2] Horn F. "Interactive Exploration and Discovery of Scientific Publications with PubVis." arXiv e-prints. **2017** June. doi: 10.48550/arXiv.1706.08094.

[3] Soto AJ, Przybyła P, Ananiadou S. "Thalia: semantic search engine for biomedical abstracts. Bioinformatics." **2019** May

https://doi.org/10.5281/zenodo.10611196

15;35(10):1799-1801. doi: 10.1093/bioinformatics/bty871. PMID: 30329013; PMCID: PMC6513154.

[4] Tsuruoka Y, Miwa M, Hamamoto K, Tsujii J, Ananiadou S. "Discovering and visualizing indirect associations between biomedical concepts." Bioinformatics. **2011** Jul 1;27(13):i111-9. doi: 10.1093/bioinformatics/btr214. PMID: 21685059; PMCID: PMC3117364

[5] Carvallo, A., Parra, D., Lobel, H. et al. Automatic document screening of medical literature using word and text embeddings in an active learning setting. Scientometrics 125, 3047–3084 **2020**. https://doi.org/10.1007/s11192-020-03648-6

[6] Grootendorst, Maarten. "BERTopic: Neural topic modeling with a class-based TF-IDF procedure." **2022**

[7] Karen A. Robinson, Klara Brunnhuber, Donna Ciliska, Carsten Bogh Juhl, Robin Christensen, Hans Lund. "Evidence-Based Research Series-Paper 1: What Evidence-Based Research is and why is it important?". Journal of Clinical Epidemiology. **2021**. doi: https://doi.org/10.1016/j.jclinepi.2020.07.020

[8] Lund, Hans and Brunnhuber, Klara and Juhl, Carsten and Robinson, Karen and Leenaars, Marlies and Dorch, Bertil F and Jamtvedt, Gro and Nortvedt, Monica W and Christensen, Robin and Chalmers, Iain. "Towards evidence based research". BMJ Publishing Group Ltd. **2016**. doi: 10.1136/bmj.i5440.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". CoRR. **2018**. doi: http://arxiv.org/abs/1810.04805.

[10] R. Thomas McCoy, Ellie Pavlick, Tal Linzen. "Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference". CoRR. **2019**. doi: http://arxiv.org/abs/1902.01007.

[11] Yang, Zhilin and Dai, Zihang and Yang, Yiming and Carbonell, Jaime and Salakhutdinov, Ruslan and Le, Quoc V. "XL-Net: Generalized Autoregressive Pretraining for Language Understanding". urran Associates Inc. **2019**.

[12] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, Jaewoo Kang. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining". CoRR. **2019**. doi: http://arxiv.org/abs/1901.08746.

[13] Mark Hughes, Irene Li, Spyros Kotoulas, Toyotaro Suzumura. "Medical Text Classification using Convolutional Neural Networks". CoRR. **2017**. doi: http://arxiv.org/abs/1704.06841.

[14] Del Fiol, Guilherme and Michelson, Matthew and Iorio, Alfonso and Cotoi, Chris and Haynes, R Brian. "A Deep Learning Method to Automatically Identify Reports of Scientifically Rigorous Clinical Research from the Biomedical Literature: Comparative Analytic Study". J Med Internet Res. **2018**. doi: 10.2196/10281.

[15] Peters, Matthew E. and Neumann, Mark and Iyyer, Mohit and Gardner, Matt and Clark, Christopher and Lee, Kenton and Zettlemoyer, Luke. "Deep Contextualized Word Representations". Association for Computational Linguistics. **2018**. doi: 10.18653/v1/N18-1202.

[16] Yoav Goldberg. "ssessing BERT's Syntactic Abilities". CoRR. **2019**. doi: http://arxiv.org/abs/1901.05287.

[17] Ari Holtzman and Jan Buys and Maxwell Forbes and Yejin Choi. "The Curious Case of Neural Text Degeneration". CoRR. **2019**. doi: http://arxiv.org/abs/1904.09751.

[18] Jingcheng Du and Qingyu Chen and Yifan Peng and Yang Xiang and Cui Tao and Zhiyong Lu. "ML-Net: multi-label classification of biomedical texts with deep neural networks". CoRR. **2018**. doi: http://arxiv.org/abs/1811.05475.

[19] Gargiulo, Francesco and Silvestri, Stefano and Ciampi, Mario and De Pietro, Giuseppe. "Deep Neural Network for Hierarchical Extreme Multi-Label Text Classification". Appl. Soft Comput. **2019**. doi: 0.1016/j.asoc.2019.03.041

[20] Yao, Liang and Mao, Chengsheng and Luo, Yuan. "Clinical Text Classification with Rule-based Features and Knowledge-guided Convolutional Neural Networks". **2018**. doi: 10.1109/ICHI-W.2018.00024.

[21] Lewis, Patrick and Ott, Myle and Du, Jingfei and Stoyanov, Veselin. "Pretrained Language Models for Biomedical and Clinical Tasks: Understanding and Extending the State-of-the-Art". Association for Computational Linguistics. **2020**. doi: 10.18653/v1/2020.clinicalnlp-1.17.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10636007

A SYSTEM FOR GEOSPATIAL QUESTION-ANSWERING USING LLMS, LANGCHAIN, CHROMADB, AND A MODERN REACT.JS FRONTEND

Patrick Lovric *, Maximilian Theiner †, ISDS, Graz University of Technology, Graz, Austria

Abstract

The increasing demand for accurate geospatial data requires efficient data retrieval methods to help users find the data they need. This university project presents a solution that combines the power of Large Language Models (LLMs) with geospatial data from Google Maps and OpenStreetMap. The system's basic idea is to enable users to ask specific questions about locations and points of interest, providing accurate and contextually relevant responses. The system integrates LLMs, LangChain, ChromaDB, and modern frontend technologies to create an interactive map-based interface. The contribution lies in the development of a comprehensive system that leverages LLMs to generate meaningful answers and facilitates efficient retrieval of geospatial information. The system focuses on enhancing the user experience and improving the efficiency of accessing geospatial data. Future work includes expanding data sources and addressing limitations such as input quality and hallucinations in LLMs.

INTRODUCTION

Geospatial information is very important in various fields such as navigation systems, automative or urban planning. The rapid rise in availability of digital cartography and geolocated services has escalated the demand for systems that offer both efficient interaction and ease of access to geospatial data. Conventional search tools and map-based applications often oblige users to manually sift through a voluminous amount of information to unearth specific responses to their geospatial inquiries. This method can prove to be laborious and ineffective, particularly when addressing intricate or locale-specific queries.

The main contribution of this project is the development of a geospatial question and answer system that integrates LLMs, LangChain, ChromaDB, and modern frontend technologies. The system provides an intuitive user interface where users can input search terms and receive interactive map-based results. The LLMs process the user queries and generate contextually appropriate responses using the crawled geospatial information from Google Maps and OpenStreetMap. The LangChain library facilitates the passing of pre-processed geospatial data to the LLMs, enabling them to generate meaningful and accurate answers. The ChromaDB vector store efficiently stores and retrieves the geospatial embeddings used by the LLMs, improving the overall performance of the system.

Structure

The remainer of the paper is organized as follows:

Architecture is devoted to a high level architectural overview of the entire project.

Backend provides an overview of the backend components, including Python, Flask, and data retrieval from Google Maps and OpenStreetMap.

Data retrieval explains how to crawl and retrieve the underlying data used in this project.

LangChain and Chroma explains the usage of LangChain and ChromaDB to process and store geospatial data for LLMs.

Frontend focuses on the frontend components, including React, Tailwind CSS, and Mapbox, and how they provide an interactive map interface for the system.

Conclusion This chapter concludes the paper, summarizing the key findings and contributions of the project as well as limitations and promising future work.

ARCHITECTURE

This university project attempts to develop a geospatial question and answer system. The user can input search terms into a search bar and the results are shown in an interactive map, where the question is answered by a Large Language Model (LLM) that uses crawled geospatial information from Google Maps and OpenSteetMap. The architecture is defined by a combination of LLMs, LangChain [1], ChromaDB [2], and modern frontend technologies such as React [3], Vite [4], and TailwindCSS [5]. The framework integrates a frontend, including a world map rendered using the Mapbox JS [6] library, and a backend built to retrieve, process, and interpret data to answer user queries. LangChain, a young library designed to power applications using LLMs, is used to pass pre-processed geospatial data to an LLM, enabling the model to generate meaningful and contextually appropriate responses from user questions. The following figure gives a concise overview of the entire software architecture and highlights the key components and libaries used throughout.

* patrick.lovric@student.tugraz.at
† maximilian.theiner@student.tugraz.at

https://doi.org/10.5281/zenodo.10636007

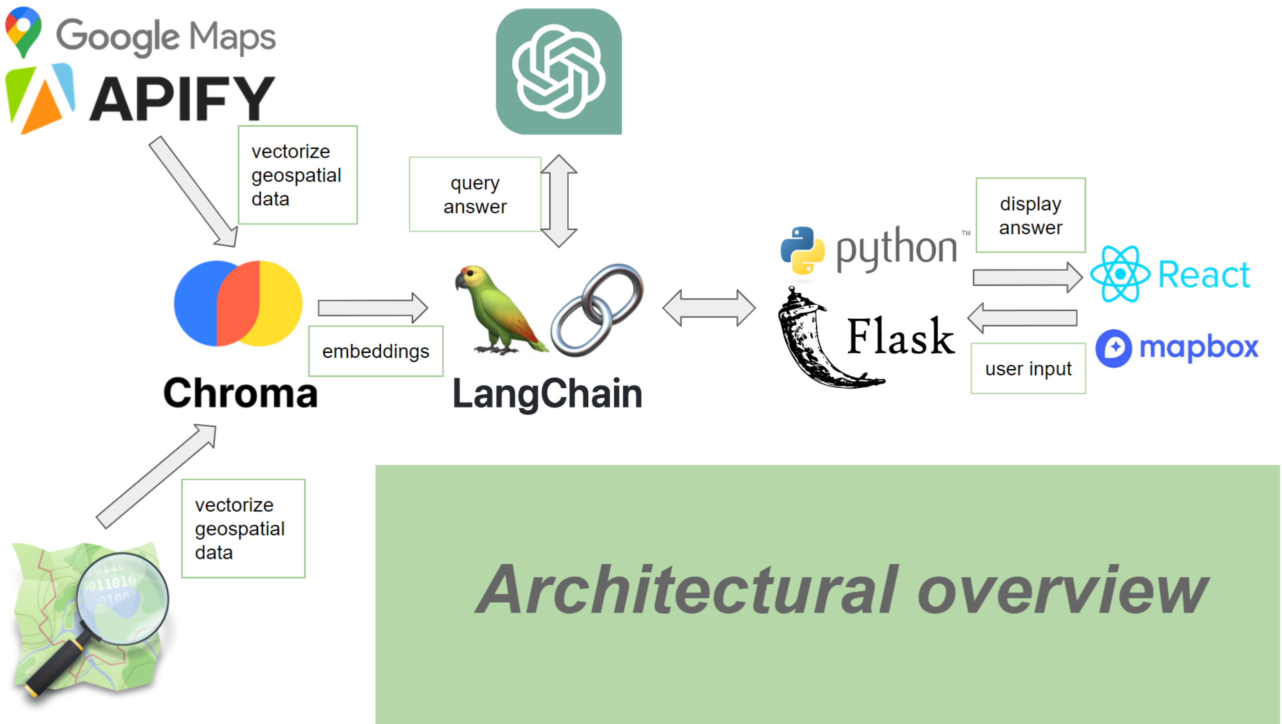


Figure 1: Software architecture of the entire project.

BACKEND

Python [7]is a programming language that is commonly used for backend application development. Its easy syntax, large community and rich set of libraries make it a popular choice for all kinds of data science projects.

Flask [8] is a web framework for Python used to build web applications via REST API. For a question-answering system like the one described in this paper, Flask is used to setup a REST API that bridges frontend and backend. When a user submits a search request on the UI, Flask handles the request and calls the functions for processing this input. When processing is complete, Flask sends the response back to the UI for visualization.

Poetry [9] is a Python packaging and dependency management tool. It manages a project's dependencies by using a toml and lock file to ensure that the application will behave the same in different environments, increasing the reproducibility of the results.

The core functionallity of the backend is the retrieval of the crawled data from online geospatial data providers. Furthermore, LangChains and ChromDB is used to implement the question-answering system. These concepts are explained in the following sections.

Data Retrieval

The python backend is used for data retrieval and processing of user inputs coming from the UI. The data is coming from two sources: Google Maps [10] and OpenStreetMap [11]. Google Maps data extraction is performed by Apify [12], a cloud-based web scraping tool. Apify is a scraping platform that can be used to extract website data from various sources. It allows to deploy and execute a wide range of so-called "actors" to crawl web-pages. Crawling Google Maps Places with Apify involves using the Google Places actor, a tool designed specifically for this task. The Google Maps scraper actor works by interacting with the Google Maps API to extract detailed information about places stored in Google Maps, including names, addresses, ratings, reviews, and more. The actor can be setup on the Apify platform to parameterize the crawling. During setup, parameters are provided such as the bounding box, region or city that the actor should crawl and the type of data to extract. It then sends requests to the Google Maps API to aggregate the requested information. The extracted data is then structured and stored in a dataset on the platform where it can be accessed and downloaded in various formats such as CSV or JSON. In the case of this application, Apify was used to query the Google Maps data for repair shops in Styira.

https://doi.org/10.5281/zenodo.10636007

Open Street Map is a community driven mapping project where members contribute and maintain the data. All map data is freely available under an open license, making OSM a rich source of geospatial data for a variety of applications. One way to access and extract data from OSM is to use the Overpass API [13]. The Overpass API is used to extract OSM map data and export it to a suitable format such as JSON for further processing. Overpass has its own query language used to retrieve and collect information from OSM. To extract geospatial information about places and points of interest within a given bounding box, one can write an Overpass query that specifies the types of POIs that should be included in the response and the geographic area defined by a bounding box. It then returns the map data that matches the query parameters. This principle was used to retrieve all the information about repair stores in Styria for this application. The data returned includes detailed geospatial information about the matching places and POIs, such as their coordinates, names, types and other tagged information that was added to the OSM database by its contributors. Furthermore, a web crawler was implemented to additionally crawl the website data of all extracted locations. However, since Apify already provides rich information such as phone numbers, addresses and location information, there was no significant benefit to this extraction. But still, the raw crawled data can be downloaded from the UI if the user wants to explore it further.

LangChain and ChromaDB

After extraction, the raw data is processed in the Python backend to remove unnecessary content and prepare the geospatial data for use with LangChain. The next step is to convert the raw geospatial data into interpretable embeddings for LLMs. Large Language Models (LLMs) have the ability to process and generate text based on the patterns they learn from their training data. However, by their very nature, LLMs aren't designed to work directly with raw textual data, such as geospatial data, which presents a challenge when using these models in applications that require understanding such formats. This is where vector databases with embeddings come into play. Embeddings are a way of representing the data in a format that LLMs can work with effectively. These representations, often in the form of high-dimensional vectors, can capture the essential features of the original data in a format that LLMs can process. In this way, data can be stored and retrieved efficiently by LLMs. For example, when responding to a geospatial query, the system can efficiently retrieve the relevant data from the vector database, and then feed it to the LLM to generate an appropriate response that fits well into the context of the question. The embeddings are stored in ChromaDB. It is a tool that provides a performant vector store that can be easily combined with LangChain, making it ideal for creating LLM based applications [14].

FRONTEND

The front end of the system is built using React and styled with TailwindCSS. It displays an interactive map interface powered by the Mapbox library. The map displays markers representing data points, where the level of detail of each marker depends on the data source used, which can be either Google Maps or OpenStreetMaps. The front-end encourages user interaction, allowing them to click on markers to view detailed information about the given location.

React

React is a popular open source JavaScript library used to build user interfaces or UI components. Developed and maintained by Facebook, React allows developers to create large web applications that can efficiently update and render in response to data changes without requiring a full page reload. The library achieves this by introducing a virtual DOM, which is a lightweight copy of the real DOM. The virtual DOM allows React to determine which components need to be changed when an application's state changes, and then update only those components in the real DOM. Another key advantage of React is its component-based architecture. This means that one can build encapsulated components that manage their own state and then assemble them into complex user interfaces. Components are reusable, which makes development more efficient and the codebase easier to manage.

Tailwind CSS

Tailwind CSS is a CSS framework that allows developers to construct UI designs directly in their markup. Unlike traditional CSS frameworks that provide components directly, Tailwind CSS provides low-level classes that enable the construction of custom designs with ease. The architecture promotes allows reusability of components to enable rapid development. To integrate Tailwind CSS into a React.js project, one typically configures it as part of one's project's build process. Tools like Create React App or Vite have plugins that make it easy to set up Tailwind CSS. Once set up, one can use Tailwind's utility classes directly in the React components. For example, one can control the layout, typography, colors, and responsiveness of the components by adding Tailwind CSS classes to the elements in the JSX markup.

MAPBOX

Mapbox is a powerful, flexible mapping platform and geospatial services provider. It provides developers with tools to integrate location-based services, such as maps, geocoding, navigation and more, in an efficient way into their applications. In this application mapbox was used to place markers on the map for the crawled locations and color them according to the search results. The user can move around and click on the marker to get further information about the specific point.

ISBN: 978-92-9083-653-7 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10636007

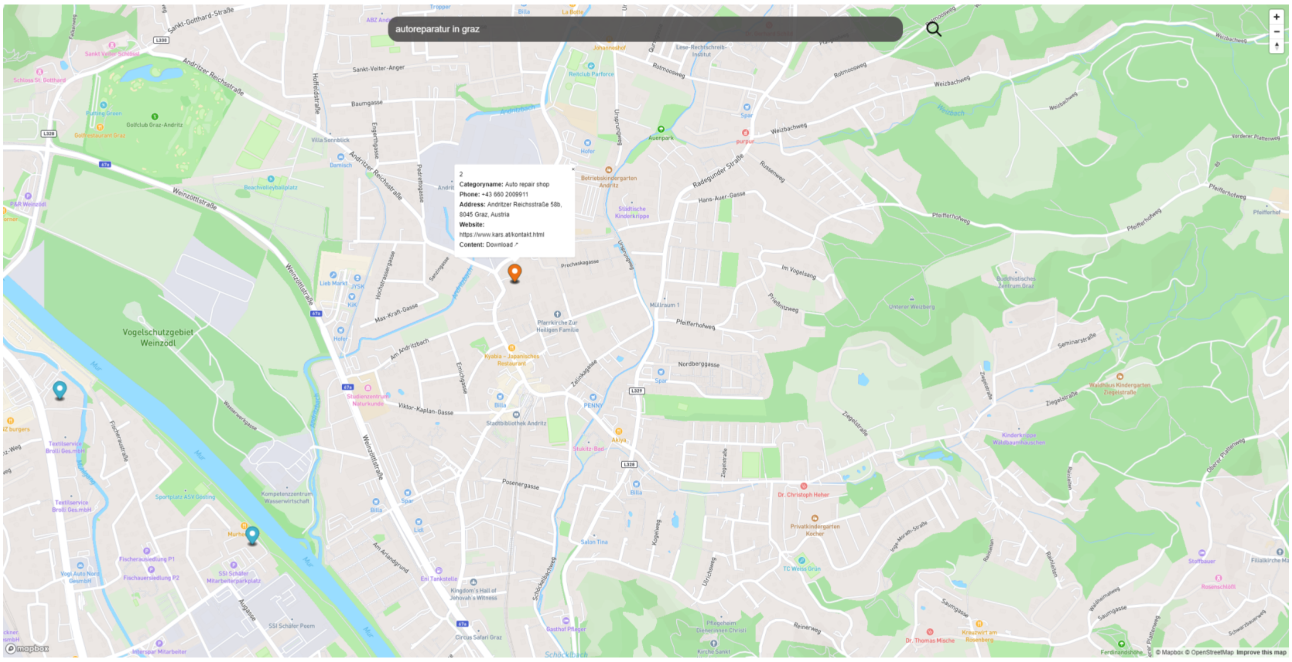


Figure 2: Visual interface of the application

CONCLUSION, FUTURE WORK & LIMITATIONS

Overall, this project demonstrates the potential of integrating LLMs, LangChain, ChromaDB, and modern front-end technologies as a valuable proof-of-concept for a geospatial question-answering system. With further improvements, this system could have applications in various domains.

While this system is operational and able to respond to inquiries in a user-friendly manner, certain constraints exist. Primarily, the efficacy of the system is heavily influenced by the precision and clarity of user commands; incorrect or nebulous inputs could yield incorrect or potentially misleading outputs. Moreover, the challenge of system-generated misrepresentations, a persistent problem for Large Language Models, remains an unresolved quandary. While the OpenAI API was chosen for its simplicity and user-friendly interface, it is an expensive proprietary model. Adding support for alternative LLMs such as LLaMA or Alpaca could increase the flexibility and robustness of the entire system. Future work could also look at optimizing the vectorization of input geospatial data to reduce API costs and speed up inference. Data from OpenStreetMap also has limitations because it is not as sophisticated and generalized as the data crawled with Google Maps. Future developments could focus on incorporating more diverse and comprehensive data sources to improve the crawled geospatial data. In conclusion, this project successfully developed an integrated geospatial question-answering system using Large Language Models (LLMs), LangChain, ChromaDB, and modern front-end technologies. The system architecture includes a frontend built with React, Vite, and TailwindCSS, which uses the Mapbox JS library for an interactive map interface. The backend, built using Python and Flask, handles data retrieval, processing, and interpretation. The system uses LLMs and LangChain to generate contextually appropriate responses to user queries. Geospatial data is pre-processed and transformed into embeddings stored in ChromaDB, enabling efficient query and retrieval of the crawled data. While the system demonstrates promising capabilities, it has limitations such as dependence on the quality of user input and the problem of hallucinations in LLMs. Future work could include incorporating alternative LLMs, optimizing data vectorization, and integrating more diverse data sources.

ACKNOWLEDGEMENTS

We would like to thank Prof. Christian Gütl and our fellow classmates for their guidance and support during the seminar on "Selected Topics in Information Processing and Retrieval."

REFERENCES

[1] Langchain, `https://python.langchain.com/docs/get_started/introduction.html`, Accessed: *24.06.2024*

[2] ChromaDB, `https://www.trychroma.com/`, Accessed: *24.06.2024*

[3] React, `https://react.dev/`, Accessed: *24.06.2024*

[4] Vite, `https://vitejs.dev/`, Accessed: *24.06.2024*

[5] TailwindCSS, `https://tailwindcss.com`, Accessed: *24.06.2024*

[6] Mapbox, `https://www.mapbox.com/mapbox-gljs`, Accessed: *24.06.2024*

[7] Python, `https://python.com/`, Accessed: *24.06.2024*

https://doi.org/10.5281/zenodo.10636007

[8] Flask, https://flask.palletsprojects.com/, Accessed: *24.06.2024*

[9] Poetry, https://python-poetry.org, Accessed: *24.06.2024*

[10] Google Maps, https://www.google.at/maps/, Accessed: *24.06.2024*

[11] Open Street Maps, https://www.openstreetmap.org/, Accessed: *24.06.2024*

[12] Apify, https://apify.com/, Accessed: *24.06.2024*

[13] Overpass, https://overpass-turbo.eu, Accessed: *24.06.2024*

[14] Langchain-Chroma, https://blog.langchain.dev/langchain-chroma/, Accessed: *24.06.2024*

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10635938

TOWARDS A SMART NETWORK SCHEMA BUILDER USING ANONYMOUS AND IMPLICIT INTERACTION DATA

Aleksandar Bobic*, Jean-Marie Le Goff, CERN, 1211, Meyrin, Switzerland
Christian Gütl, Graz University of Technology, 8010 Graz, Austria

Abstract

Due to an ever-increasing prevalence of intelligent systems and intelligent features, the need for smart user guidance in systems is slowly transforming into an expectation. Such techniques have already been thoroughly explored for cases such as smart action and visualization recommendations. They are usually facilitated through the use of recommender systems trained on either explicit user content or implicit user interaction data. Since using explicit user data may cause potential privacy issues and data leakage, using implicit data is the safer option for any end user. A domain where such approaches would be of great benefit but have yet to be extensively explored is the area of interactive network modelling and exploration. Therefore, as part of this work, we introduced a first prototype in an existing visual analytics and network modeling tool called Collaboration Spotting X. We extend the tool's interactive network schema builder used for iteratively modeling search results as networks with a prototype of a recommender system based on implicit interaction data sequences. The newly introduced recommender system prototype aims to aid users in their network modelling journey by either recommending possible new network schemas or incremental network schema modifications using only synthetically generated interaction data. Due to the importance of user aspects in adaptive user interfaces as part of this work, a prototype user interface implementation is also introduced and evaluated through user interviews.

INTRODUCTION

User interfaces (UI) that adjust based on users' interactions with a system have been explored extensively in the past [1]. Applications with such UIs usually have a particular user model that changes over time with new interactions and provides UI change suggestions to users or adjusts the UI automatically [1]. This task is usually handled by recommender systems that provide recommendations based on user action patterns or user content [1, 2]. However, using explicit user content may have potentially devastating privacy issues [3]. Even more so in the process of data modelling or data analysis where accidentally exposed analysis aims or findings may cause harm to companies or institutions as well as individuals. Therefore, as part of this work, we only focus on implicit user and interaction data.

To explore how such recommender systems could be used to support users in their data analysis journeys in a novel context, multiple recommender system models are built in Collaboration Spotting X. This system provides users with an environment for visual information retrieval, network-based analysis and data modelling, particularly network modelling [4]. One of the core aspects of the system is visualizing search results as networks whose topology can be changed through an interactive network schema builder. Past evaluations in the form of interviews have indicated that many users need clarification and help with using the interactive network schema designer to analyze their data.

To support users in a privacy-conscious way, as part of this work, we explore multiple approaches to providing completely new schema suggestions and potential iterative changes in existing network schemas using interaction data and implicit data features. Since automatic changes may unpleasantly surprise users, the system suggests these changes to users as part of an extended schema builder UI [1]. We explore various features extracted from anonymous interaction data for building multiple recommender systems to identify the best-performing ones. Finally, we evaluate the resulting systems using standard evaluation measures and user interviews.

Based on the above-described potential for recommender systems in visual interactive network analysis, as part of this work, we introduce a recommender system prototype to an interactive network schema. As part of this work, we aim to answer the following research questions:

1. **How can a sequential recommender system aid users in exploring interactive network schemas?**
2. **How can interactive network-based visual analytics tools improve users' exploratory interactions through recommender systems?**

BACKGROUND AND RELATED WORK

VISUAL ANALYTICS AND NETWORK MODELING

Visual analytics systems usually rely on a combination of information visualisation and automated analysis approaches combined with a human in the loop with the main aim of supporting the analytical workflow, creating a model that accurately describes a part of reality that the user needs to answer their analysis questions and the creation of insights that answer the research questions of the system user [5–7]. One of the main psychological models used to describe the visual analytics workflow is sensemaking [8, 9]. The sensemaking process includes multiple activities performed non-linearly and can be defined with two main loops: the foraging loop and the sensemaking loop. The former focuses on retrieving, filtering and extracting information into a

* aleksandar.bobic@student.tugraz.at

https://doi.org/10.5281/zenodo.10635938

particular representation appropriate for analysis, while the latter focuses on extracting hypotheses and externalising knowledge [8].

To represent data in an appropriate representation or schema that supports users' question-answering and insight generation, experts traditionally used programming approaches to pre-process data before being able to use it in a tool. For the case of network analysis and network modelling, this is still true, with only a few tools exploring the possibility of providing users with an approach to modelling networks by users with no need for coding. Examples of such tools include Ploceus and Origraph, which depict a visual network schema and enable users to model data into a network representation based on it [10, 11]. Another example of such a tool is Collaboration Spotting X, which provides users with a network schema and a history of network representations over time [4]. However, for larger datasets with many features, these schema representations are complex and confusing for users. They would benefit from additional automatic guidance, which would help users create and modify schemas.

ADAPTIVE USER INTERFACES

An approach that could help support users in their network modelling process is adaptive user interfaces (AUI). AUI adapt their functionality or actions based on the user's needs and interactions with the system [1, 12]. However, before introducing an AUI into a system, multiple aspects, such as the difficulty and routine of tasks that are performed in a system, the users that will be using the system, and the level to which a system is adaptive [12]. For example, AUIs that provide some level of adaptivity but let the user efficiently perform tasks manually are more beneficial than fully automated AUIs in cases where a user has to perform non-routine tasks [12]. Recent advancements in machine learning techniques can be used to facilitate such interfaces [13]. Techniques such as deep learning, particularly in the context of action recommendations, could enhance users' experiences in complex visual analytics systems and provide users with the necessary support in their analysis workflows.

RECOMMENDER SYSTEMS

Sequential recommender systems (SRS) are recommender systems (RS) dealing with sequences of user-item interactions and aiming to suggest new possible items to interact with or interactions [14, 15]. These items could include items to buy or user interface (UI) elements to interact with or use. Therefore, such systems may be used for adaptive user-interfaces.

An early example of using recommender systems to recommend the next webpage visit is through the use of pattern mining [16]. This system analyzes the user browsing history, identifies common patterns and provides recommendations for the next possible pages based on the n previous visited pages. Another earlier example of a sequential recommender system technique is using Markov Chains in combination with matrix factorization [17]. With the improvement in hardware capabilities and advances in neural network architectures and modern deep learning (DL), the use of DL solutions has become more approachable and more effective in tasks such as sequential recommender systems. An example of a solution using DL, in particular deep recurrent neural networks combined with gated recurrent memory units for creating an SRS used for adaptive user interface (AUI), is described by [18]. Unlike previous approaches that used pattern mining or Markov chains, this approach considers users' long-term interaction histories. The evaluations also indicate that such models are more accurate than past models [18].

Further evlutions of such models include long short-term memory (LSTM) [19, 20] and gated recurrent unit (GRU)[21, 22]. Alternative DL architectures such as convolutional neural networks and graph neural networks have also shown great promise in the context of SRS and can be used to model complex interaction patterns and patterns between not-adjacent interactions [14]. An example of graph neural networks for an SRS is described by [15]. They introduce an architecture called SURGE, which aims to address the issue of using implicit user feedback through interactions and users' change of interest over time. Despite the evaluation of SURGE indicating its effectiveness, the authors note that further user studies have to be conducted.

MODEL AND SYSTEM DESIGN

Since CSX includes an overview and detail schema, representing a co-occurrence and a multivariate network, the recommender system model has to recommend schemas and actions for two separate network types. The CSX system is connected to the OpenAlex API [23]. Therefore, users can access the metadata of approximately 240 million papers and explore them as interactive network representations.

As part of this work, we created models specifically for the network schemas of OpenAlex. Four separate models were created, one for schema recommendations and one for action recommendations for each network type, to observe potential problems in data or in our hypotheses easily. Due to CSX being designed to respect users' privacy and users being intimidated by the network schema and having trouble understanding it, there was a lack of interaction data that could be used for a recommender system.

DATA PREPARATION

Due to the lack of interaction data, multiple synthetic data sets were prepared for the recommender system-building process. The datasets aimed to mimic simple user interactions with the schema representation. The two datasets were constructed by defining a set of potential schemas users may want to build using CSX. For the overview network representation, 67 unique schemas were prepared, and for the detail network representation, 55 schemas were prepared. Based on these schemas, 10,000 synthetic user sessions with lengths of 10 to 300 were created for the overview and detail

ISBN: 978-92-9083-653-7 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10635938

representation. The schemas were randomly picked for each session to simulate potential user exploration paths.

Incremental schema changes were needed to generate input datasets for the action recommendation models. Users usually perform multiple actions on the schema before applying a new schema to the system and generating a network based on it. These actions may include removing or adding edges, changing the node representation, adding node properties and changing the edge cardinality. A sequence of schemas representing incremental changes was generated for each sequential pair of schemas from the sessions mentioned above. Since the model prototype architecture supports only a fixed length input, the sessions had to be split into sequences of ten schemas. The final representation of the schema recommendation dataset included a list of sessions with ten schemas, each with an additional schema as the next schema or label. On the other hand, the action recommendation dataset was represented by a list of sessions with ten schemas and an action or label for each.

RECOMMENDER SYSTEM

The recommender system models were built using the Python PyTorch library [24]. Due to this system being a prototype, multiple simple models were built to test the concept and answer the initial research questions. Based on the insights gathered from related work, the architecture chosen for this prototype was based on LSTM. The two models used for schema recommendation had a single layer with 32 neurons and a linear output layer each and were trained for 350 epochs. Cross entropy was used for measuring the loss, and Adam was used as the optimizer with an initial learning rate of 0.001. On the other hand, the action recommendation networks had four layers with 32 neurons each and were trained for 35 epochs using the same loss function and optimizer.

Additionally multiple sizes of input data were tested for training the models. The final dataset size used for the schema recommendation models was 5000 sequences with 10 schemas each. On the other hand, the dataset size used for the overview and detail action recommendation models was 200000 and 100000 sequences with 10 schemas each. The datasets were split using a 60:20:20 ratio for the schema recommendation models where 60% of the data was used for training, 20% was used for validation, and 20% was used for testing. On the other hand, the datasets used for the action recommendation models were split using an 80:20:20 ratio.

The detailed loss values of each of the models can be seen in Tab. 1. It can be observed that the two schema models were overfitted to the sample data, which can be potentially explained by the input only being transitions between schemas. Due to the exploratory and prototype nature of this work, this was acceptable, but it was noted that it should be improved in the future.

After the training, the models were implemented in Collaboration Spotting X and introduced in the user interface as depicted in Figure 1. Once the user performs the initial search and is shown the search results as a network, they can interact with the recommender system results. When a user modifies the schema using any of the action interactions, a new set of action interactions is shown to them. Once they apply the schema and create a new network out of it, the schema recommendations refresh, as well as the action recommendations refresh. This provides the user with a new set of recommendations based on their past interactions every time they interact with the system.

Table 1: Average loss values for the schema and action recommendation models. OS and DS stand for overview and detail schema recommender model, respectively. OA and DA stand for Overview and detail action recommender model, respectively. Train. L., Valid. L. and Test. L. represent the average training, validation and test loss values.

Model Type	Train. L.	Valid. L.	Test. L.
OS	0.3056	13.4627	12.9616
DS	1.1308	10.0776	9.9880
OA	0.7439	0.7357	0.7464
DA	0.9919	0.9589	0.9639

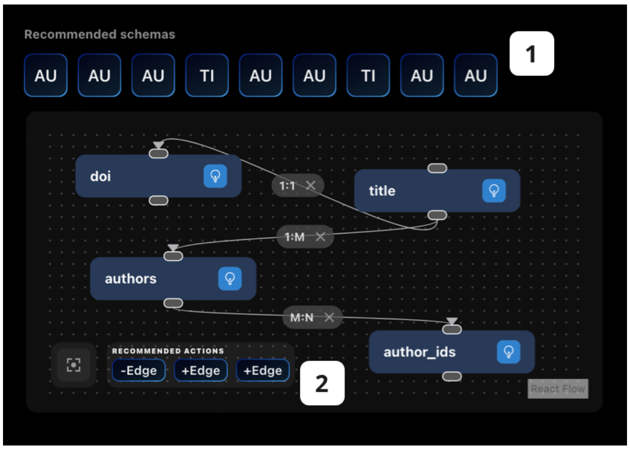


Figure 1: The Collaboration Spotting X dynamic network schema and the recommendations. The buttons at the top of the schema builder (1) represent schema recommendations. The smaller buttons at the bottom of the schema (2) represent action recommendations. By clicking on one of the schema or action recommendations, users automatically change the entire schema or just one schema property, respectively. Once the change has been made, they can further change the schema or apply it to their data.

USER INTERVIEWS

To answer the above-identified research questions, we interviewed four experts who were familiar with the CSX system. The experts included a data scientist, a doctoral student in computer science, a senior research fellow and a senior software engineer. After being re-introduced to CSX, they were given fifteen minutes to interact with the system and explore the new schema and recommendation features.

https://doi.org/10.5281/zenodo.10635938

After the interactive session, they were asked the following questions relating to their experience:

- How would you describe the relevance of the recommendations received by the recommender system?
- For what tasks would you use the schema and action recommendation systems?
- What would you like to improve or add to the recommender system?

All experts observed that the schema recommendations are somewhat random, while the action recommendations are very relevant to the current schema. Furthermore, all noted that the seemingly random generation resulting from overfitting helped them explore the new schemas and gather inspiration for their exploratory journey. On the other hand, the action recommendations were recognised as being more useful for small incremental interactions when the schema needed finer adjustments.

They noted that while the recommendations were helpful, the recommendation buttons should be adjusted to reflect better the change they will cause. Moreover, the experts expressed the need for expanding the recommendation features to the entire analysis process by recommending actions based on the network data being viewed by the user, recommending potential alternative visualisations, and even recommending potential new search queries in the context of CSX.

Finally, during the exploratory interaction phase, it was noted that using the recommended schemas and actions is much easier than manually creating entire schemas and, therefore, more inviting to use. It was further observed that the users better understood what they could do with the interactive network schema after applying the recommended actions and seeing how they modified the schema. However, some still needed help understanding how the actual schema affects the network representation and therefore suggested that there should be a more precise explanation of what each of the schemas changes in the network representation.

Based on the following answers, the answer to research question 1 is that the recommender system can help users explore, adjust, and, in some cases, even learn how to use interactive network schemas. Due to the complexity of such a system, users need multiple forms of support, which can be provided using recommender systems. Furthermore, it was observed that further support is needed for the user to understand how the schema affects the network representation. The answer to the first question can also serve as a partial answer to the second question. However, interactive network-based visual analytics tools may improve users' exploratory interactions by adding extensive recommendation support throughout the system. For example, leveraging information about the data explored by users' systems can further expand user support through network modelling functionality and insight discovery. However, this may come at the cost of potential privacy issues if not implemented appropriately. Additionally, it may be possible to use only user interactions in other areas of a visual analytics tool to indicate how interesting a schema is to a user and, based on this predict further schemas. Finally, systems could potentially support users by providing recommendations for system-wide actions based on the current context.

CONCLUSION

As part of this work, we introduce a prototype of a sequential recommender system for interactive network schema modelling. The system was built using synthetic data and trained using the long short-term memory neural network architecture. The system recommends users' potential next network schemas and incremental network adjustments. After training, it was integrated into the network-based visual analytics system Collaboration Spotting X. The system consists of four models for recommending schemas and actions of the overview and detail view. It was evaluated through user interviews. The results of the evaluation indicate that such an approach is well received by users and serves as a tool for exploration and idea gathering, easier interaction with complex network schemas, as well as learning support for novice users. Identified improvements to the system include the expansion of such a system to other areas of interactive network analysis, improvements to the user interface of the recommender system and the consideration of the data being explored by the user in the system. We aim to expand and enhance the system with the suggested recommendations in future work. Furthermore, we aim to explore the potential unification of the four distinct models into one, support multiple preloaded and user-uploaded dataset schema and action suggestions, and recommend schemas and actions based on user observations in the system. To improve the schema recommendation models, we will explore the introduction of additional contextual information, such as areas of the user interface the user was interacting with before switching to a new schema. Finally, we aim to explore other neural network types, such as graph neural networks and additional schema encoding strategies, such as vector embeddings, for better recommendation accuracy.

REFERENCES

[1] T. Kühme, "A user-centered approach to adaptive interfaces," in *Proceedings of the 1st International Conference on Intelligent User Interfaces*, 1993, pp. 243–245. doi:10.1145/169891.170012

[2] D. Gotz and Z. Wen, "Behavior-driven visualization recommendation," in *Proceedings of the 14th International Conference on Intelligent User Interfaces*, 2009, pp. 315–324. doi:10.1145/1502650.1502695

[3] A. Bobic, I. Jakovljevic, C. Gütl, J.-M. Le Goff, and A. Wagner, "Implicit user network analysis of communication platform open data for channel recommendation," in *2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, 2022, pp. 1–8. doi:10.1109/SNAMS58071.2022.10062597

https://doi.org/10.5281/zenodo.10635938

[4] A. Bobic, J.-M. Le Goff, and C. Gütl, "Exploring tabular data through networks," in *Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023, Proceedings, Part III*, 2023, pp. 195–200. doi:10.1007/978-3-031-28241-6_13

[5] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," in *Information Visualization: Human-Centered Issues and Perspectives*. 2008, pp. 154–175. doi:10.1007/978-3-540-70956-5_7

[6] N. Andrienko *et al.*, "Viewing visual analytics as model building," *Computer Graphics Forum*, vol. 37, no. 6, pp. 275–299, 2018. doi:https://doi.org/10.1111/cgf.13324

[7] K. A. Cook and J. J. Thomas, "Illuminating the path: The research and development agenda for visual analytics," 2005. https://www.osti.gov/biblio/912515

[8] P. Pirolli and S. Card, "The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis," in *Proceedings of international conference on intelligence analysis*, McLean, VA, USA, vol. 5, 2005, pp. 2–4.

[9] D. M. Russell, M. J. Stefik, P. Pirolli, and S. K. Card, "The cost structure of sensemaking," in *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, 1993, pp. 269–276. doi:10.1145/169059.169209

[10] Z. Liu, S. B. Navathe, and J. T. Stasko, "Ploceus: Modeling, visualizing, and analyzing tabular data as networks," *Information Visualization*, vol. 13, no. 1, pp. 59–89, 2014. doi:10.1177/1473871613488591

[11] A. Bigelow, C. Nobre, M. Meyer, and A. Lex, "Origraph: Interactive network wrangling," 2019, pp. 81–92. doi:10.1109/VAST47406.2019.8986909

[12] T. Lavie and J. Meyer, "Benefits and costs of adaptive user interfaces," *Int. J. Hum.-Comput. Stud.*, vol. 68, no. 8, pp. 508–524, 2010. doi:10.1016/j.ijhcs.2010.01.004

[13] K. Todi, G. Bailly, L. Leiva, and A. Oulasvirta, "Adapting user interfaces with model-based reinforcement learning," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021. doi:10.1145/3411764.3445497

[14] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. Orgun, "Sequential recommender systems: Challenges, progress and prospects," *arXiv preprint arXiv:2001.04830*, 2019.

[15] J. Chang *et al.*, "Sequential recommendation with graph neural networks," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 378–387. doi:10.1145/3404835.3462968

[16] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, "Using sequential and non-sequential patterns in predictive web usage mining tasks," in *Proceedings of the 2002 IEEE International Conference on Data Mining*, 2002, p. 669.

[17] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th International Conference on World Wide Web*, 2010, pp. 811–820. doi:10.1145/1772690.1772773

[18] H. Soh, S. Sanner, M. White, and G. Jamieson, "Deep sequential recommendation for personalized adaptive user interfaces," in *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, 2017, pp. 589–593. doi:10.1145/3025171.3025207

[19] J. Duan, P.-F. Zhang, R. Qiu, and Z. Huang, "Long short-term enhanced memory for sequential recommendation," *World Wide Web*, vol. 26, no. 2, pp. 561–583, 2023. doi:https://doi.org/10.1007/s11280-022-01056-9

[20] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. doi:10.1162/neco.1997.9.8.1735

[21] K. Cho *et al.*, *Learning phrase representations using rnn encoder-decoder for statistical machine translation*, 2014.

[22] Q. Cui, S. Wu, Y. Huang, and L. Wang, "A hierarchical contextual attention-based network for sequential recommendation," *Neurocomput.*, vol. 358, no. C, pp. 141–149, 2019. doi:10.1016/j.neucom.2019.04.073

[23] J. Priem, H. Piwowar, and R. Orr, *Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts*, 2022.

[24] *Pytorch*, https://pytorch.org/, (Accessed on 09/01/2023).

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10684924

EXPLOITING KEY INFORMATION FROM OPEN SCIENTIFIC SEARCH RESULTS TO ENHANCE USER EXPERIENCE

S. Frank*[1], A. Wagner, CERN, 1211 Meyrin, Switzerland
C. Gütl, A. Nussbaumer, ISDS, Graz University of Technology, 8010 Graz, Austria
[1]also at ISDS, Graz University of Technology, 8010 Graz, Austria

Abstract

Search engines have become indispensable tools for navigating the Web. More specifically, scientific search engines, such as Google Scholar and Semantic Scholar, are increasingly essential tools for people seeking to stay up-to-date on current research. However, even with search engines, obtaining the most significant information from a list of search results can prove to be time consuming. This paper proposes an approach to extract key information from open scientific search results and uses aspect-based summarization to improve user experience and search efficiency. The resulting system allows for fine-tuning of search results by selecting them and then extracts aspect-based summaries from each to relate and compare them to each other, e.g., the research questions, used methods, and evaluation metrics.

INTRODUCTION

Given the wealth of information available on the Web, open search engines have become important tools for users searching for relevant information. Although in the past, it was customary to provide simple lists of links with, possibly, short blurbs as search results, this has now become an increasingly outdated approach. Instead, search engines have evolved to provide additional information, such as images, videos, and other multimedia content, in addition to links to webpages and blurbs [17]. More recently, conversational information retrieval systems have become increasingly popular. Tools such as ChatGPT (Generative Pre-trained Transformer) and the usage of language models have revolutionized user search behavior, regardless of the potential remaining issues regarding factual accuracy to the result's source material [8]. However, while conversational AI shows great promise for a variety of use cases, some applications may benefit from using other, more explainable approaches. When searching for scientific papers, for example, it is essential that the summary correctly reflects the information contained in the source document. In addition, the consistent structure of scientific papers makes it possible to make some assumptions about their most relevant content. This article proposes a concept for the aspect-based summarization of one or a small number of selected scientific search results, focusing on aspects such as "Motivation", "Hypothesis", and "Research questions", as well as filtering options for quality metrics related to these aspects.

Therefore, this research aims to improve the user search experience and the effectiveness of information acquisition by providing key information on user-selected search results. It specifically seeks to answer the following two questions:

- Can aspect-based summarization for scientific search results be used to show commonly perused information in an efficient and truthful way?
- Can existing datasets and language models be utilized to extend the number of aspects for which summarization is possible? If not, is it possible to create a dataset large enough to fine-tune a language model for this use?

The next section will provide an overview of the related work, particularly in regard to search, question answering, key information extraction, and aspect-based summarization. Following this, we will give a detailed explanation of the concept, after which we will describe the development in further detail. Subsequently, the results will be shown and discussed. Finally, the work ends with the conclusion.

RELATED WORK

As the amount of content on the Web grows, search engines must adapt to changing requirements. Although traditionally search results were presented as lists of links with short information excerpts, this is no longer the only approach, particularly in the scientific field, where websites such as Elicit aim to provide a more in-depth overview of search results by extracting key information such as "Abstract summary", "Intervention", and "Outcome measured" [12].

In recent years, the presentation of search results has rapidly shifted to using question-answering as its methodology of choice. Question answering aims to provide direct and concise answers to queries, such as in answer boxes at the top of search results or through the use of chatbots. However, the effectiveness of question answering depends heavily on how search results are presented and on the correct understanding of the question itself. Current solutions place great emphasis on interactive question answering, in part due to rapid progress in the use of machine learning for this task [3].

Although machine learning has been leveraged to extend the abilities of search engines in this way with promising results, when the user's goal is to get an overview of a select number of search results, the issue remains that the main method of presenting this information is via small previews only, thus making it necessary to open each web page to get a clear idea whether it is relevant. As this is not only the case for web search, previous research in other fields has also taken note of this issue. Ma et al. describe a system

* sarah.frank@cern.ch

https://doi.org/10.5281/zenodo.10684924

for automatic extraction of key elements from geological hazard documents and visualize the results as knowledge graphs [10]. They utilized the TF-IDF algorithm for keyword extraction and visualized them with methods such as word clouds. In addition, Ma et al. performed cluster analysis based on co-occurrence to discover relations.

Strengthening this shift in search behavior, platforms such as ChatGPT, scite.ai, and Elicit have become commonly used as alternatives to traditional search engines, giving impressive results. While some of these platforms are not explicitly intended for scientific use, and, therefore, are usually not able to supply trustworthy references to information sources, others provide the user with automatically created abstracts for each search result, allowing proper attribution. However, these solutions usually still require one to access the full text to fully assess relevancy and are not particularly flexible in their presentation of results or focus. Through the use of aspect-based summarization methods, this focus is intended to be more adaptable.

Although aspect-based summarization has been an active field of research for years, much of it has been in the domain of reviews and combined with sentiment analysis [6, 14]. Research on aspect-based summarization in scientific settings is limited, and even more so for scientific search. However, the approach may serve to improve both the user search experience and the efficiency of information acquisition when used to improve search engines, as aspect-based summarization has already been found to be particularly useful for tasks such as opinion mining, judgment labeling, and personalized learning environments [9, 18].

Ravi and Ravi provided a comprehensive survey on opinion mining and sentiment analysis, which are closely related to aspect-based summarization [13]. The survey covers tasks, approaches, and applications in sentiment analysis, offering valuable insight into the broader context of opinion-based summarization. Fremann and Klementiev addressed the task of aspect-based summarization by developing models that generate summaries centered on specific aspects within a document [5]. Their work contributes to the methodology of aspect-based summarization and provides valuable techniques for aspect-focused summarization. Tan et al. proposed a weakly supervised knowledge-informed approach to summarize text in arbitrary aspects [16]. They created an abstractive, weakly supervised method to compensate for the lack of training data, testing with the All The News dataset. With their summaries for aspects such as names or verbs, they produced promising empirical results. As with the creation of test data sets, there is also the challenge of creating reference summaries for evaluation, which is why any results of this paper were only empirical and need further study.

Although user attitudes towards personalized aspect-based summarization have long been evaluated and have generally been found to favor personalization with respect to, among other things, summary length [2], aspect-based summarization for search results has received little attention. Fremann and Klementiev aimed to summarize news articles by the aspects (or topics) they covered, leveraging the mostly consistent structure of news articles and encoder-decoder models [5]. Hayashi et al. (2021) introduced the WikiAsp dataset, which specifically addresses multidomain aspect-based summarization [7]. This dataset has been instrumental in advancing research on aspect-based summarization by providing a standardized benchmark for evaluating summarization models across different domains. In the scientific domain, Meng et al. presented FacetSum [11]. However, the dataset, which covered multiple domains, included only a specific set of facets (purpose, method, results, and value). This limitation was also noted by Soleimani et al., who similarly aimed to extend their method to other aspects of the FacetSum and PubMed datasets [15]. They did so using semantic representations from the pretraining process to establish similarity between aspects and content.

The lack of a large dedicated data set was another limitation to aspect-based summarization. Only recently have additional datasets become available that are specifically intended to be used for aspect-based summarization tasks. In an attempt to address the domain-specific nature of both models and datasets, [7] and [19] published large-scale open-domain datasets for aspect-based summarization tasks. However, the issue remains that the creation of such datasets is challenging, so choosing arbitrary aspects to summarize for is still an issue.

In addition to this, the use of both the terms aspect-based and facet-based summarization can complicate the finding of relevant literature. Both involve the generation of a summary that focuses on a specific aspect, viewpoint, or topic within a given document. Due to the fact that there is no formal definition for the terms readily available, research often seems to use them interchangeably, depending on preference. Due to this, both aspect-based and facet-based summarization research offers valuable insight into previous work, but might be overlooked due to inconsistencies in terminology.

CONCEPT

Based on the literature review in the previous section and the limitations identified, this paper aims to introduce an aspect-based summarization system intended to provide users with more precise and relevant information from their search results. By creating a flexible approach that focuses on the summarization of aspects in individual or a small number of scientific articles, users can explore selected search results provided by a search engine or from a local folder. Unlike many approaches that define a set number of aspects that are considered for summarization, this system is intended to be extendable to further aspects with little effort to allow usage for a variety of domains within scientific publications.

The objective is to improve the user experience and speed up information acquisition by preventing users from having to access multiple different pages or files to obtain relevant information. Instead of a simple list of results, the user has the option of selecting specific aspects of the content that they are particularly interested in and having them briefly

https://doi.org/10.5281/zenodo.10684924

summarized. By doing this, the idea is to allow users to more easily get an overview, as well as to be able to relate and compare specific aspects with each other, e.g., research questions, used methods, and evaluation metrics of scientific papers.

An example of what this activity can look like in the context of a search engine is shown in Figure 1. The search activity up to and including "Display search result" is implemented by the search engine itself, or is alternatively skipped if the articles to be selected are present in a local folder.

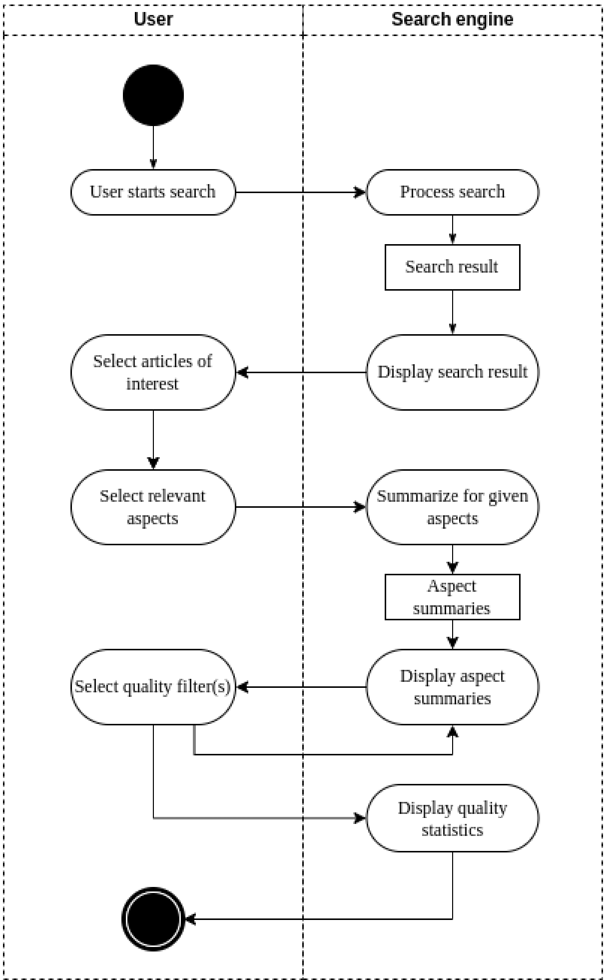


Figure 1: User/search engine interactions for the proposed system

As scientific papers generally have mostly consistent structures, particularly within research domains, it is possible to define a set of key information of particular interest. Table 1 gives a brief overview of the selected points that were considered of particular interest to a potential user and should be summarized from the selected document(s), as well as operate as filtering criteria. In this table, the column "Keywords" specifies a list of (non-exhaustive) terms that indicate information relevant to the respective facet. This is related to future implementation, as previous research has shown promising results with non- or weakly supervised methods that utilize this information [1, 16].

The result of the search process is then a list of papers that include the given keyword and match the filtering criteria for the facets defined by either the user or the search system. At this step, it is possible to create an overview over the prevalence of certain aspects in the source texts, as well as which are present in each. Given this information, it is possible to not only filter by aspects, but also how extensively they are covered, as well as score the suitability of sources according to the number and quality of the aspects they contain.

DEVELOPMENT

The general framework was created to be modular and flexible to allow for easy extension or switching/addition/removal of components. The scientific articles used for the testing process were published in the Journal of Universal Computer Science and are available as pdfs. As shown in Figure 2, they were fed to GROBID, which extracts and parses all text from PDFs into easily traversable XML structures. The data were then synthesized into a data structure that is usable for further processing. In this process, we extracted the following data from each paper's XML structure: title of the article, author(s), abstract text, conclusion text, and full text of the article. For later evaluation, this was further extended manually by adding reference summaries for each facet.

For the facet summarization method, a variety of different approaches were considered.

Based on the promising results of the article by Meng et al. [11], it was the first choice to be evaluated for extension and adaptability to this setting. The FacetSum model specifically summarized for a limited number of different facets due to the challenges of creating a large enough data set to allow pretraining of a large language model. Due to the dataset used by Meng et al. containing only the Purpose, Method, Findings and Value facets [11], any further implementation of facets requires an extension of the dataset. This involved, in the first step, an extension of data, which was done manually for a small dataset of 40 articles. The necessity for human intervention for the creation of reference summaries for each facet - due to, in contrast to the data source used Meng et al. [11], the facets are not defined for the journal website or in the metadata - meant that the resulting dataset was limited in size, making training of a supervised method challenging, which was reflected in tests.

Upon evaluation of the available implementation, the lack of a pre-trained model presents an issue without access to a sufficient number of GPUs. Furthermore, the extension of the dataset itself is a challenge, as explained previously. Upon creation of a small dataset with data extracted from JUCs, it was concluded that this was insufficient to result in sufficiently well-performing summaries. The human effort required to extend the dataset for each facet goes far beyond the practicality of most applications.

As the structure of the pipeline allows for easy addition of additional methods in the future, another prospective

https://doi.org/10.5281/zenodo.10684924

Facet	Prompt	Keywords
MOT	Motivation	Motivation, goal, aim
HYP	Hypothesis	Hypothesis, expectation
RQ	Research questions	RQ, research question
CON	Contribution	Contribution, add
RW	Related work	Related work, basis, previous research
RM	Research methods	Methods, methodology, process
TL	Tools, libraries used in research	Library, package, application
TD	Test data used for training and testing	Data set, training data, test data
EM	Metrics used for result evaluation	Evaluation, quality, reliability, score
MF	Main findings of the research	Findings, outcome, result
LIM	Limitations, issues found during the research	Limitations, issues, problems, exception
FW	Future directions and work	future, next, improvements

Table 1: Facet abbreviations with their prompt and associated keywords. Inspired by Table 1 by Ahuja et al. [1]

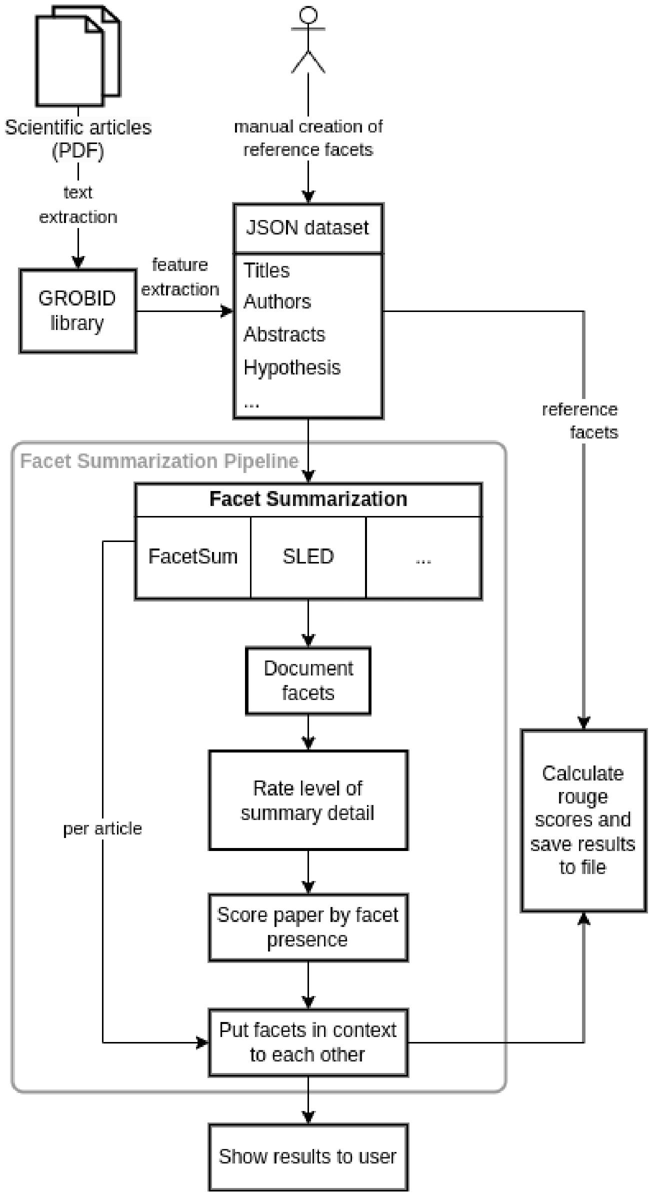


Figure 2: Flowchart of the project architecture

approach is the usage of SLED [4]. This represents the summarization step, thus being the most significant in the pipeline. The quality of aspect summaries is fully dependent on the selection of the summarization algorithm, apart from the creation of a dataset.

Once the summaries for each document facet are created, the system can evaluate which are present for each document and to what level of detail. This can be done by evaluating the quality of the summary through the use of ROUGE scores and similar evaluation methods, or by evaluating the summary length. Additionally, research on the evaluation of automatically created summaries is ongoing. This means that this step, as well, is required to be flexible and easily extendable/replaceable.

With the evaluation metrics implemented, it is then possible to rate the quality of the paper according to how many aspects are present and how well they are covered. This gives the user an overview of the general quality and relevancy, as well as the ability to directly compare facets of different papers to each other if needed.

The selected method was chosen due to the requirements of the specified use case. The extraction of facets requires an understanding of context that exceeds the possibilities of many language models. Although transformer-based methods show more promise for this process, many require large amounts of resources to run. With this in mind, the goal was also to select a comparatively small model due to further pre-training that may be required to adapt the model to be able to extract further facets. Furthermore, the selection of the model must consider the length of the intended input. Due to the average length of scientific papers outstripping that of many other possible types of input such as news articles and blog posts, the model has to be able to handle long input texts. This is not the case with many language models, which set the maximum number of tokens quite low; in the case of BERT, this is 512, where each token can be considered to function as a word, and is not sufficient for scientific papers.

Once the data set is completed, the data from each paper are fed to the facet summarization method(s). One or more

https://doi.org/10.5281/zenodo.10684924

of these can be selected before running the pipeline to get the results for each approach. For this paper, we based our work on FacetSum and SLED, with the possibility of extending the selection further in the future to gain additional insights into comparisons of result qualities. The resulting output, the facet summaries, are then rated according to their length. Depending on whether a summary is below 15 words, between 16 and 25, or above 25 words, it receives the attributes "short", "medium", or "long", respectively. Further fine-tuning for this categorization may be needed for future research, depending on the paper domain(s), type of paper, and generally average aspect summary lengths.

Following this, a score is given to the paper on the basis of the number of facets present. This is done by counting the number of existing facet summaries of all possible facets. Furthermore, this is the point where, for the research process, we calculated the rouge scores for the summaries and saved them to files to be able to evaluate the result quality.

The extracted facets from the various papers are put into comparison with each other by creating a CSV table containing basic paper information (title, authors) as well as the facet summaries.

RESULTS

As implied previously, the results for an extended version of FacetSum were unsatisfactory, even though initial summaries seemed promising.

Table 2: Caption

LIMITATIONS

Although research on this topic is being done, it is often challenging to reproduce another paper's approach as the specific pre-trained models are not available.

CONCLUSION AND FUTURE WORK

This paper proposes an approach to extract key information from scientific search results and to present information from multiple sources by displaying them in relation to each other. The resulting system allows for fine-tuning of search results by selecting them and then extracting key points from each. When presenting the extracted information in relation to each other, it becomes easier to see how certain results fit into the larger body of research work and facilitates a more effective search experience.

REFERENCES

[1] Ojas Ahuja, Jiacheng Xu, Akshay Gupta, Kevin Horecka, and Greg Durrett. ASPECTNEWS: Aspect-oriented summarization of news documents. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6494–6506, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[2] Shlomo Berkovsky, Timothy Baldwin, and Ingrid Zukerman. Aspect-based personalized text summarization. In *Lecture Notes in Computer Science*, pages 267–270. Springer Berlin Heidelberg.

[3] Giovanni Maria Biancofiore, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fedelucio Narducci. Interactive question answering systems: Literature review. September 2022.

[4] Xiaochun Cao, Hua Zhang, Xiaojie Guo, Si Liu, and Dan Meng. Sled: Semantic label embedding dictionary representation for multilabel image annotation. *IEEE Transactions on Image Processing*, 24(9):2746–2759, 2015.

[5] Lea Frermann and Alexandre Klementiev. Inducing document structure for aspect-based summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019.

[6] Vedika Gupta, Vivek Kumar Singh, Pankaj Mukhija, and Udayan Ghose. Aspect-based sentiment analysis of mobile reviews. *Journal of Intelligent &amp Fuzzy Systems*, 36(5):4721–4730, may 2019.

[7] Hiroaki Hayashi, Prashant Budania, Peng Wang, Chris Ackerson, Raj Neervannan, and Graham Neubig. WikiAsp: A dataset for multi-domain aspect-based summarization. *Transactions of the Association for Computational Linguistics*, 9:211–225, mar 2021.

[8] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*. Association for Computational Linguistics, 2022.

[9] Y. Liu. Court judgement labeling using topic modeling and syntactic parsing. 2022.

[10] Ying Ma, Zhong Xie, Gang Li, Kai Ma, Zhen Huang, Qinjun Qiu, and Hui Liu. Text visualization for geological hazard documents via text mining and natural language processing. *Earth Science Informatics*, 15(1):439–454, jan 2022.

[11] Rui Meng, Khushboo Thaker, Lei Zhang, Yue Dong, Xingdi Yuan, Tong Wang, and Daqing He. Bringing structure into summaries: a faceted summarization dataset for long scientific documents. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics, 2021.

[12] Ought. Elicit: The ai research assistant, February 2023.

[13] K. Ravi and V. Ravi. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, 89:14–46, 2015.

[14] Seyyed Aref Razavi and Masoud Asadpour. Word embedding-based approach to aspect detection for aspect-based summarization of persian customer reviews. In *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*. ACM, oct 2017.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10684924

[15] Amir Soleimani, Vassilina Nikoulina, Benoit Favre, and Salah Ait Mokhtar. Zero-shot aspect-based scientific document summarization using self-supervised pre-training. In *Proceedings of the 21st Workshop on Biomedical Language Processing*. Association for Computational Linguistics, 2022.

[16] B. Tan, L. Qin, E. P. Xing, and Z. Hu. Summarizing text on any aspects: a knowledge-informed weakly-supervised approach. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[17] Yue Wang, Dawei Yin, Luo Jie, Pengyuan Wang, Makoto Yamada, Yi Chang, and Qiaozhu Mei. Beyond ranking. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. ACM, feb 2016.

[18] T. Wu, B. Kao, F. H. W. Chan, A. Cheung, M. M. Cheung, G. Yuan, and Y. Chen. Semantic search and summarization of judgments using topic modeling. *Frontiers in Artificial Intelligence and Applications*, 2021.

[19] Xianjun Yang, Kaiqiang Song, Sangwoo Cho, Xiaoyang Wang, Xiaoman Pan, Linda Petzold, and Dong Yu. Oasum: Large-scale open domain aspect-based summarization, 2022.

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10557539

PROTOTYPING OPEN WEB SEARCH APPLICATIONS WITH TIRA: A CASE STUDY IN RESEARCH-ORIENTED TEACHING

Maik Fröbe[*] Theresa Elstner[†] Harrisen Scells[†] Benno Stein[‡] Martin Potthast[†,§]

[*]Friedrich-Schiller-Universität Jena [†]Leipzig University [‡]Bauhaus-Universität Weimar [§]ScaDS.AI

Introduction. The Open Search Foundation (OSF) aims to promote competition in the search market by enabling a wide range of independent search applications to strengthen Europe's digital sovereignty. We present a new teaching concept for information retrieval (IR) courses at universities that mutually benefits students' practical education and the rapid prototyping of open (web) search engines [3]. Students can gain practical experience by developing and evaluating all components of a search application. We use TIRA [1] to ensure that the artifacts created throughout the course, i.e., domain-specific test collections and retrieval approaches, are reusable and reproducible so that the community might pick them up and successful prototypes might eventually count towards the search applications maintained by the OSF.

Teaching Open Search Prototyping. Building search engines for new domains requires new test collections to guide the development. As part of teaching, we must consider that students can only invest a certain amount of time into this task. Suitable test collections comprise three components: (1) a collection of documents to be searched, (2) a set of topics defining user information needs, and (3) corresponding relevance assessments for documents in the collection. The document collection can be sampled from existing corpora, especially the envisioned Open Web Index built by the OpenWebSearch.EU project. This leaves the creation of topics and relevance assessments as remaining tasks. Both are crucial to reliable search engine evaluation and the information retrieval curriculum, and students are highly receptive to learning about it [4]. Student-built test collections offer a valuable learning opportunity to them and a valuable research opportunity to "crowdsource" independent search engine evaluations. A particularly salient opportunity is the ongoing evaluation using existing test collections, which may need more freshness or breadth (number of topics) and depth (number of relevance assessments per topic).

Project-Based Test Collection Development. Figure 1 presents our methodology for teaching student projects to create a test collection while catering to their learning needs [3].

We designed a milestone-based system where students create a test collection over one university teaching semester. Our methodology aims to develop a useable test collection from a pre-existing document collection (e.g., a selected subset of the Open Web Index with no topics or relevance assessments). Students work in groups to create different parts of the test collection, working toward three milestones. In Milestone 1, students analyze the document collection, and each student constructs one information need. In Milestone 2,

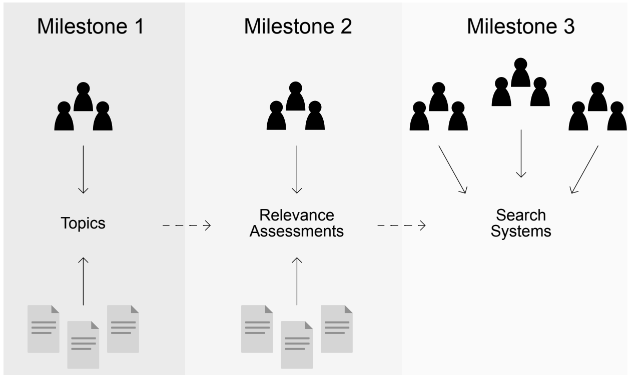


Figure 1: Our methodology for developing test collections during collaborative group student projects yielding topics, relevance assessments, and strong baseline search systems.

students create relevance assessments for their information needs by pooling and judging the documents retrieved by 10 diverse retrieval systems from TIREx [2]. Milestone 3 combines all groups' topics and relevance assessments into a hidden final test collection. Students then develop prototypes using their topics, submitting them to a shared leaderboard in a competition to develop the most effective system on the hidden joint test set. This way, students learn the fundamentals of information retrieval evaluation and use this knowledge to prototype effective search engines. The joint test collection and reasonably strong baselines can henceforth serve for further research or as a starting point for shared tasks. Moreover, students are invited to open source their search engines under the open web search licensing scheme.

Future Work. Going forward, joint test collections built by students will have to be evaluated, too. We envision an augmentation and comparison to expert-supplied topics and relevance judgments in a quantitative and qualitative topic (difficulty) assessment. A comprehensive teaching concept documentation will also be developed.

REFERENCES

[1] M. Fröbe et al. Continuous Integration for Reproducible Shared Tasks with TIRA.io. In *Proc. of ECIR 2023*, 2023.

[2] M. Fröbe et al. The Information Retrieval Experiment Platform. In *Proc. of SIGIR 2023*, 2023.

[3] T. Elstner et al. Shared tasks as tutorials: A methodical approach. In *Proc. of AAAI 2023*, 2023.

[4] M. Sanderson and A. Warner. Training Students to Evaluate Search Engines. 2011.

This work was partially supported by the European Commission under grant agreement GA 101070014 (OpenWebSearch.eu).

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10626261

GOVERNANCE TOWARDS AN OPEN WEB INDEX

J. Tietgen, Open Search Foundation e.V., Munich, Germany
M. Alanko, CSC – IT Center for Science, Espoo, Finland

INTRODUCTION

A small number of dominant tech gatekeepers hold control over what can and cannot be found on the Web, which poses a threat to the transparent access to web information, resulting in a significant economic imbalance and a lack of information sovereignty for Europe.

The OpenWebSearch.eu project seeks to address this issue by constructing an Open Web Index Infrastructure, intended as a public good. Additionally, it aims to establish a cooperative ecosystem surrounding this infrastructure. [1] To ensure the sustainability of the Open Web Index beyond the project and to ensure transparency and accountability a tailor-made governance model is needed. It is necessary to establish clear roles, responsibilities, and decision-making processes, as well as enable effective communication and collaboration among stakeholders.

The objective of the governance building task in the project is developing and piloting a model-governance framework for establishing, operating, scaling and sustaining an open search infrastructure at European scale. This includes governing the infrastructure, data assets, services and software.

The vision of the initiative is to make the core of the open web index a public good, available to everyone. This creates specific requirements for the governance as it should stay collaborative and independent from unbalanced influence. It should be ethical, open and create trust. Furthermore, it should be based in the EU. To enable a distributed network across Europe, the legal form of our organization needs to support decentralized decision-making and governance structures, allowing for participation from various stakeholders across different EU-countries.

METHODOLOGY

A Governance workshop was organized with participants from technical and non-technical domains to identify relevant stakeholders and comparable collaborative infrastructures as well as collect and discuss concerns related to the governance of an open web index. The results of the workshop were structured to a report and used as the starting point for the work.

Landscape Analysis

Already the project proposal identified some comparable organisations to be studied and analysed. In the workshop and other meetings this list of interesting and relevant organisations was extended. Based on a predefined structure we collected information about these organisations, for example legal form, decision-making structures and funding models, to select the most relevant for further analysis and discussion. We took a closer look at some initiatives, such as European Open Science Cloud EOSC, EUDAT Collaborative Data Infrastructure, GAIA-X, Copernicus programme and Galileo, to find out what we can learn from them.

Furthermore, we collected information about different funding sources and financial models including private and public as well as different combinations of the two. These have been further analysed and their pros and cons discussed and defined.

DISCUSSION

From this analysis we were able to identify possibilities, similarities and necessities for the governance framework as regards to their legal form, governing bodies and funding sources.

Based on the initial analysis we decided to study the following options further as the potentially suitable legal forms. The legal forms analysed are limited liability company, AISBL and EDIC. We are currently evaluating them and defining our requirements for the governance. The necessary governing bodies will depend on the legal form chosen. They will need to include strategic, executive, steering structures.

As the open web index aims to be a public service, special focus on the funding source and its independence is necessary. There will be a need for public funding, but also private funding and sales should be possible. The final funding model is likely to be a combination of different funding sources.

The next step will be to build and analyse alternative scenarios for different time frames and legal entities, based on the specific requirements of the project. Also taking into consideration that the final governance framework is dependent on political decisions.

REFERENCES

[1] https://openwebsearch.eu/

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935

https://doi.org/10.5281/zenodo.10557368

Cooperate via Open Console

M.A.C.J. Overmeer†, MARKOV Solutions, Arnhem, The Netherlands

Abstract

The *Open Console* project builds cooperation in sharing knowledge about websites and domain-names. Three groups of parties are involved: parties who publish facts about those websites and domains (for instance, related geographical locations), parties who interpret those facts (f.i., to improve search results), and the owners of the website and domain-names (f.i. to see what people are publishing about them).

In the context of search engines, Open Console will offer to the website owner features which resemble *Google Search Console*[1]: a place where the owner can communicate with Google. Owners can provide Google with useful information about their website, and get feed-back about crawler and visitor activity as reward. This contact gives Google a serious competitive advantage. In contrast to Google's service, the Open Console will provide one interface shared between all competitors in the field.

The sheer size of internet, makes this simple idea into a complex infrastructure. How to handle a few hundred producers, thousands of consumers, and a few hundred million website owners? For application in the OSF context: what does OpenWebSearch want to communicate with the website owners via Open Console?

COLLECTING FACTS

The most visual component of search engines is text processing: indexing text from a huge number of websites, and smart text retrieval algorithms. But those search algorithms can produce better results when they can include additional facts which surround those texts.

To give some examples:

- location information, which is found on a different page in the same website;
- a fresh list of publicly known phishing sites;
- manually confirmed websites which contain rated images to tune child-protection filters; or
- statistics about the popularity and trust-worthiness of the site.

Some information is already available somewhere in internet, but discovery and sharing is not organized.

Not only interpretation facts are useful to share, also crawler activity can be helped:

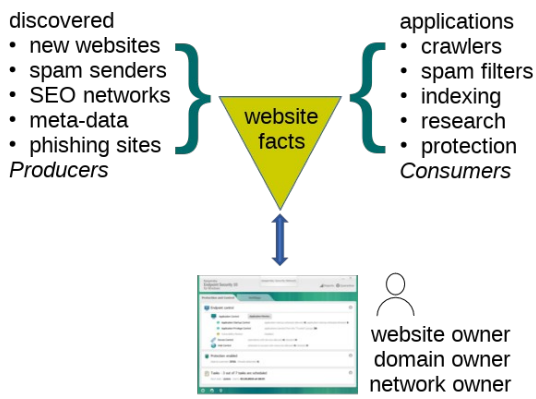
- website owners publish the frequency of change for certain pages, to optimize being crawled;
- crawlers report technical issues in the visited websites;
- network discovery algorithms flag SEO link-farms, to be excluded from indexing; and so on.

The facts to share about websites, far extend features offered by current generic mechanisms, as there are `robots.txt` files[2], various types of sitemaps, and various sets of HTML contained meta-data.

The definition of the exchanged "facts" follows the usual standardization processes. It is convenient when consumers have to process better standardized facts, but it is not a requirement for publishers to cooperate. Fact definition is outside the scope of Open Console itself. Open Console is a distribution mechanism.

CHALLENGES

The Open Console project is committed to be open and fair, in their strict interpretation. Fair means: there is not a single party which can monopolize the exchange mechanisms and the relation to the website owners. This calls for strong privacy and defensive strategies.



The amount of data to be exchanged, the independent producers and consumers, and the (potentially) huge amount of interactive users, demand an unmanaged network of independent systems. All logical components must be able to evolve over time, without global synchronized software upgrades.

DESIGN

The implementation of Open Console has some components which are well understood, but also a few parts which have never been done before. Especially the fully *Open* character and the required scale, make even simple components to be challenging to realize.

Easily identifiable main components:

- Consumers and producers agree on a license on distributed facts. The license enforces the

† mark@overmeer.net https://markov.solutions

https://doi.org/10.5281/zenodo.10557368

handling of corrections (take-down requests, removal of errors, expiration);

- The consumers and producers maintain a (semi-)continuous stream of updates, which may reach terabytes in size, containing billions of facts per day;
- A person registers at any "console provider" which offers interactive services. Even this component permits competing implementations. The person proves ownership to one or more websites and domains;
- The owner invites producers of facts to its set-up: selects the sources of interest which are presented in "a fair way". Only invited producers get two-way communication to the owner;
- The two-way communication implements extended dynamic forms with versioning, authorization, and translation features; and
- Producers, consumers, and console providers shape an Association to protect the rules of behavior: the level of openness, the interpretation of fairness, guaranteed security, and optimal privacy.

The presentation will discuss complications and suggested solutions for these components.

Besides, a start has been made to inventorize the specific facts which participants in the OWS-project wish to exchange with website owners, based on Google's example. The first steps to create the two-way communication between OWS-projects and website owners.

ACKNOWLEDGEMENTS

This project would not have been possible without the generous initial support of the NLnet Foundation.

REFERENCES

[1] *Google Search Console*, https://search.google.com/search-console/about

[2] *Robots.txt,* IETF RFC9303, https://www.rfc-editor.org/info/rfc9309

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10594168

PRIVACY-PRESERVING COLLABORATIVE FILTERING: EVALUATING A MACHINE LEARNING RECOMMENDER SYSTEM IN A LARGE INTERCONNECTED ORGANIZATION

Igor Jakovljevic, ISDS, Graz University of Technology, Graz, Austria
also at CERN, Geneva, Switzerland
Christian Gütl, ISDS, Graz University of Technology, Graz, Austria
Andreas Wagner, CERN, Geneva, Switzerland

Abstract

Collaborative filtering algorithms revolutionized the field of recommendation systems by taking advantage of user preferences and behaviors to provide personalized recommendations. Although CF is effective in providing accurate recommendations, it often requires access to users' personal data, raising privacy concerns. In response, researchers have proposed several privacy-preserving techniques to strike a balance between maintaining the accuracy of the recommendation and safeguarding user privacy. This research aims to evaluate the performance, effectiveness, and efficiency of privacy-preserving collaboration filtering techniques in a real-world setting, with a specific focus on CERN as the use case organization. The evaluation will assess the capability of collaborative filtering to protect user privacy while preserving recommendation accuracy within the context of CERN.

INTRODUCTION

The advent of Collaborative Filtering (CF) algorithms revolutionized the field of recommendation systems (RS) by leveraging user preferences and behaviors to provide personalized recommendations. While CF is effective in providing accurate recommendations, it often requires access to users' personal data, raising privacy concerns. This has led to privacy concerns becoming a crucial aspect in the design and implementation of CF algorithms for recommender systems. The widespread use of such systems in large interconnected organizations has additionally raised significant concerns regarding user and organisation privacy. As CF typically requires access to personal data, organizations must ensure that sensitive information remains protected. To address this challenge, the research community has proposed numerous privacy-preserving techniques in order to achieve a balance between maintaining the accuracy of the recommendation and protecting the privacy of the user [1, 2].

Although privacy-preserving CF has been widely studied, empirical evaluations in real-world situations, especially in large interconnected organisations, remain limited. This research aims to fill this gap by evaluating the performance, effectiveness, and efficiency of various privacy-preserving CF techniques in a real-world setting, with a specific focus on CERN as the use case organization. CERN, as a large interconnected organization with diverse user profiles and sensitive data, provides an ideal environment for evaluating the capabilities of privacy-preserving CF algorithms. The evaluation will assess the capability of these techniques to protect user privacy while preserving recommendation accuracy within the context of CERN's organizational network. By conducting this evaluation, our aim is to provide valuable insights into the practical applicability and performance of privacy-preserving CF techniques, ultimately contributing to the development of privacy-aware recommender systems in organizational settings.

RELATED WORK

Researchers have proposed several privacy-preserving techniques to mitigate these risks while maintaining the effectiveness of recommendation algorithms. The need for privacy-preserving CF arises from the inherent trade-off between personalization and privacy. While maintaining the quality and accuracy of recommendations is essential, addressing privacy concerns in organizational settings is equally critical. Privacy-preserving recommender systems employ various strategies, such as privacy-preserving item-based CF, random perturbations, substitution encryption, secure multiparty computation, homomorphic encryption, and federated learning [3–5].

Evaluation of results produced by these methods in RS is a difficult task [6–8]. Two main types of recommendation evaluation metric suggested by the literature are statistical accuracy metrics (SAM) and decision support accuracy metrics (DSAM) [9, 10]. SAM such as Mean Absolute Error (MAE) evaluates the accuracy of a recommender system by comparing the prediction values with actual ratings for items that have both predictions and ratings [10, 11]. DSAM evaluate how effective a prediction engine is in helping a user select relevant items among available ones. The most common measures are precision and recall [9, 10]. Using proper model validation techniques assists in understanding models and estimating a model's performance. In the literature, multiple validation techniques exist, such as train/test split, k-fold cross-validation, nested cross-validation, McNemar's test, etc. The most common validation method used is train/test split in combination with k-fold cross-validation [12, 13].

Evaluating Privacy-Preserving CF (CF) techniques has been the focus of numerous studies in recent years. Research efforts have aimed to propose efficient and accurate algorithms that protect user privacy during the recommendation process [1]. Despite this, there is a lack of comprehen-

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10594168

sive surveys in the field that explore privacy-preserving CF schemes from different perspectives [5].

CONCEPT

We will present a comprehensive evaluation of a collaborative filtering (CF) based recommender system (RS) integrated into an existing notification service at CERN. Using CF techniques, personalised recommendations will be provided to volunteer users of the service, enhancing their user experience. Importantly, the RS will be designed with a privacy-preserving approach, ensuring that Personal Identifiable Data (PII) are not used in the recommendation process.

To enhance the evaluation process, volunteers will be able to provide explicit feedback to the recommendations. Users will have the option to either approve the recommended item or ignore it, enabling the collection of valuable user feedback and assessment of the effectiveness of the CF-based RS. Additionally, the created date of each recommended item will be recorded to measure the time to user action, enabling the evaluation of the system's responsiveness and timeliness in delivering recommendations.

To ensure a comprehensive evaluation, various evaluation methods will be employed, including statistical accuracy metrics (SAM), such as mean absolute error (MAE) to assess recommendation accuracy, and decision support accuracy metrics (DSAM), such as precision and recall to evaluate the system's effectiveness in helping users select relevant items.

To follow the principles of privacy and openness, all data generated during the evaluation will be anonymised and made available as open source. This will allow other researchers to analyse the anonymised data and replicate the evaluation process.

The result of this research will be a comprehensive analysis and evaluation of a privacy-preserving CF-based RS where the data used in the evaluation will be open-sourced, ensuring reproducibility and facilitating further research.

REFERENCES

[1] D. Li, C. Chen, Q. Lv, L. Shang, Y. Zhao, T. Lu, and N. Gu, "An algorithm for efficient privacy-preserving item-based collaborative filtering," *Future Generation Computer Systems*, vol. 55, pp. 311–320, 2016.

[2] S. Gong, "Privacy-preserving collaborative filtering based on randomized perturbation techniques and secure multiparty computation," *IJACT*, vol. 3, pp. 89–99, 2011.

[3] N. Polatidis, C. K. Georgiadis, E. Pimenidis, and H. Mouratidis, "Privacy-preserving collaborative recommendations based on random perturbations," *Expert Systems with Applications*, vol. 71, pp. 18–25, 2017.

[4] I. Elnabarawy, W. Jiang, and D. C. W. I. au2, "Survey of privacy-preserving collaborative filtering," 2020.

[5] A. Bilge, C. Kaleli, I. Yakut, I. Gunes, and H. Polat, "A survey of privacy-preserving collaborative filtering schemes," *Int. J. Soft. Eng. Knowl. Eng.*, vol. 23, pp. 1085–1108, 2013.

[6] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Comput. Surv.*, vol. 47, may 2014.

[7] J. Zhang, Y. Lin, M. Lin, and J. Liu, "An effective collaborative filtering algorithm based on user preference clustering," *Applied Intelligence*, vol. 45, pp. 230–240, Sep 2016.

[8] J. P. Lucas, N. Luz, M. N. Moreno, R. Anacleto, A. Almeida Figueiredo, and C. Martins, "A hybrid recommendation approach for a tourism system," *Expert Systems with Applications*, vol. 40, no. 9, pp. 3532–3550, 2013.

[9] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," *Journal of Machine Learning Research*, vol. 10, no. 100, pp. 2935–2962, 2009.

[10] N. Good, B. Schafer, J. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl, "Combining collaborative filtering with personal agents for better recommendations.," pp. 439–446, 01 1999.

[11] T. Chai and R. Draxler, "Root mean square error (rmse) or mean absolute error (mae)?," *Geosci. Model Dev.*, vol. 7, 01 2014.

[12] I. Markov, H. Gomez Adorno, G. Sidorov, and A. Gelbukh, "The winning approach to cross-genre gender identification in russian at rusprofiling 2017," 12 2017.

[13] T.-T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognition*, vol. 48, no. 9, pp. 2839–2846, 2015.

https://doi.org/10.5281/zenodo.10601033

OPEN (WEB) SEARCH, A BOOSTER FOR OPEN SCIENCE?

S. Voigt† and T. Hecking,

German Aerospace Center,
Oberpfaffenhofen/Cologne, Germany

† corresponding author: stefan.voigt@dlr.de

Abstract

Openness, transparency, accessibility and reproducibility are fundamental prerequisites of science. This holds true for all parts of the scientific process and needs to be realised at all organisational levels: for the individual researcher, for universities and academia, for research organisations, for the 'publication' of research results as well for societal and general perception and acceptance of scientific findings.

For this reason, open science initiatives promote the general openness of scientific methods, computer-source-code, data, publications, peer-review processes and educational resources. In particular the open access principle to scientific 'publications' (often enough meaning a seclusion of scientific results from the public domain in the past, as the copyright for scientific material had to be transferred to the big publishing houses) is of utmost importance, to widely spread, share and debate, and in the end societally accept scientific findings. Similarly, the findability and accessibility of (research-)data is of utmost importance for acceptance and reproducibility of scientific findings.

As ever more information is consumed from and rendered to the public digital sphere - the web – transparent and reproducible finding of information therein is of fundamental importance for science in general. The Open Web Search initiative has identified this need for open and reproducible findability of data and information in the web and is currently building a comprehensive and open web index as a public infrastructure and reference. In particular, if this open web index can be established as a publicly audited and constantly available data base for web resources and if it can be accessed and searched in an open and unbiased manor, it may serve as a significant reference and booster for open science in the long run. A problem that has to be overcome in this regard is that scientific knowledge beyond classical publications is fragmented over many different outlets and in different formats, for example, grey literature, educational resources, datasets, or software tools. Accessibility and discoverability of semantic connections between scientific artefacts in such heterogeneous subsystems can be considered as a limiting factor for innovations.

Thus, novel semantic search mechanisms are needed that allow for browsing the rich landscape of scientific outlets on the web in a more structured, unbiased, and effective manner, which is not necessarily well supported by current mainstream web search engines.

The German Aerospace Center (DLR) as a large research organisation for space, aeronautics and transportation research in Europe, has identified this issue and has started to work on open search technologies to navigate internal resources such as intranet, code-repositories, Wiki, publication data bases, document repositories synergistically and in a synoptic manor. Furthermore, DLR is also engaging-in and supporting the European Open Web Search initiative by helping to build an open and cooperative web search ecosystem across many different science and computing centres for the Web itself.

In this talk we discuss how DLR is working to improve the synergistic searchability, findability of internal documents, data, publications and computer code in synergy and together with external web resources and web indices generated in the pan European Open Web Search ecosystem, currently under development. This synergistic use of state-of-the-art and innovative open search capacities, by linking DLR-internal and external scientific resources, will provide DLR scientists with up-do-date, objective and unbiased information and will allow its researchers to transparently search and access Web resources for open and reproducible open science. Open (Web) Search is still in an early stage of its development. However, given the ever-increasing informational needs for open and objective science, Open Web Search needs to be implemented at its full potential and in a joint effort of science organisations in the years to come: to support open and transparent sharing of scientific information and to ensure maximum accessibility and acceptance of scientific results in general, by making the scientific process as transparent and reproducible as possible.

https://doi.org/10.5281/zenodo.10626218

EUROPE'S TECHNICAL DEBT: WHY WE NEED WEB SEARCH IN THE AGE OF GENERATIVE AI

Malte Ostendorff, Pedro Ortiz Suarez, Julian Moreno-Schneider, Georg Rehm, DFKI GmbH, Germany

INTRODUCTION

Generative AI has completely changed the landscape of machine learning, allowing researchers and practitioners to tackle tasks thought to be impossible. The proliferation of generative AI models revolutionises the way we process and use information. However, this age of rapid technological innovations is dominated by US-based enterprises. Europe is lagging behind in developing large AI models and is expected to have a hard time catching up. One of the reasons is the *technical debt* that Europe has been accumulating since it lost and stopped competing in the race of the previous technological revolution – Web search.

Progress in generative AI is mainly driven by two factors: computational power and data (neglecting algorithmic improvements). Despite the fact that US-based cloud providers currently possess the lion's share of computational power, this does not pose a major hindrance for European AI developments. Europe is actively investing in its compute infrastructure, reallocating resources, and making them accessible for AI research, such as through initiatives like the EuroHPC Joint Undertaking. The real issue lies in the deficiency of the second key ingredient – Web data and its retrieval, which can be attributed to the absence of strong European Web search initiatives. EU projects such as Open Web Search are promising, though.

WEB CRAWLS FOR PRETRAINING

Large language models (LLMs) and other generative models are statistical models based on training data. This training data is crucial for the success of any AI model, e. g., affecting the language capabilities, biases, and cultural representations. Given the increasing size of these models, larger training datasets are required. The most prominent source that provides data at the scale required is the Web, accounting for a significant portion of the training data for recent LLMs, often more than 80%. Especially Web data from Common Crawl, or processed versions such as OSCAR, is widely-used for LLM training. This reliance on Web data introduces several limitations, especially in the European context. Web crawls from Common Crawl are only a sample of the whole Web, i. e., important European websites might be omitted. Also, since the Common Crawl crawler operates with a US user-agent and an IP number located in the US, the crawler appears to websites as a user from the US. As a result, English language content represents the largest share of Common Crawl data by far (30%). Web data is generally unbalanced in terms of included languages, which further increases the technological gap between English and other, more multilingual regions. To address these limitations, a European Web crawl is needed to collect a training dataset that adequately covers Europe's diversity including its languages, countries, and cultures. While there are already ongoing projects and initiatives working on this or related problems, we need to significantly strengthen them to obtain valid extensions to Common Crawl. Something as crucial as the training data of AI models should not solely depend on a single Californian non-profit organisation that operates on AWS-donated infrastructure. From one day to another, AWS may throttle Common Crawl's bandwidth and hence delay European AI research projects, as happened recently.

WEB-BASED LLM AUGMENTATION

Generative models have severe shortcomings. Among others, the enormous training costs prohibit frequent retraining on new data. Thus, the "knowledge" encoded in LLMs can become outdated quickly. For instance, ChatGPT's knowledge cut-off date is September 2021. Moreover, LLM output may contain factually incorrect information ("hallucinations"). One promising approach to address this issue is augmenting LLMs with retrieval systems. The idea is to retrieve factual and updated information from trustworthy sources and then let the LLM generate output based on the retrieved information. One example is Microsoft's Bing chatbot that retrieves information from the Web. As with pre-training, the Web represents the most extensive resource from which information can be retrieved. However, building a retrieval-augmented LLM obviously requires Web search, making it, once again, quite difficult for Europe to compete since no European Web search exists. Relying on Web search APIs from one of the big technology enterprises is no valid option either since it would introduce a strong dependency, hampering technological sovereignty. In fact, Microsoft tripled the prices of the Bing Search API briefly after the introduction of their own retrieval-augmented LLM. Thus, there is a pressing need for European Web search APIs to build the next generation of retrieval-augmented AI models.

CONCLUSIONS

Europe's lack of investment in Web search infrastructure, crawling and retrieval, has led to a significant technical debt. To be able to compete in the next technological revolution, generative AI, this debt needs to be paid off. Although catching up is feasible, it requires a collective effort and substantial investment from industry and academia.

ACKNOWLEDGEMENTS

The work presented in this paper has received funding from the German Federal Ministry for Economic Affairs and Climate Action (BMWK) through the project OpenGPT-X (project no. 68GX21007D).

ISBN: 978-92-9083-653-7 Open Search Symposium 2023 - #ossym2023 ISSN: 2957-4935
https://doi.org/10.5281/zenodo.10626851

EXPLORING THE LANDSCAPE OF INNOVATION: A NETWORK-BASED APPROACH FOR VISUALIZING AND ANALYZING HETEROGENEOUS PATENT GRAPHS

André Rattinger *, ISDS, Graz University of Technology, Graz, Austria
Christian Gütl, ISDS, Graz University of Technology, Graz, Austria

Abstract

This study presents a novel network-based methodology for exploring, visualizing, and analyzing patent data utilizing the Open Source Graph Visualization Tool Collaboration Spotting X (CSX) . Through a transformative process, tabular patent data is converted into dynamic, heterogeneous graphs, providing a comprehensive view of the innovation landscape that is encoded in the patent data. This approach opens new avenues for understanding intricate relationships within technological ecosystems, offering insights into technological trends, innovation patterns, and the interconnectedness of inventors, institutions, and companies.

INTRODUCTION

Patents are considered a significant indicators of technological advancements as they are a rich, yet complex source of information. They encompass not only the details of the technological inventions they protect but also a wealth of related data, such as information about inventors, their affiliations, the evolution of intellectual property rights, and relationships between different patents.

However, effectively leveraging patent data for research or strategic decision-making is not a straightforward task. The data is often vast, multidimensional, and intricate in nature, thus posing substantial challenges for comprehensive analysis and understanding. Traditional approaches to patent analysis, which often focus on examining individual patents or simple aggregations of patent data, frequently fall short in revealing the complex, interrelated structures and emerging patterns within the patent landscape.

In response to these challenges, this research proposes a novel methodology for patent analysis that combines the power of network analysis and visual analytics, leveraging the capabilities of the Collaboration Spotting X (CSX) platform. This approach aims to transform raw, tabular patent data into dynamic, heterogeneous graphs that provide a comprehensive, visually intuitive view of the technological landscape. By doing so, it aspires to makes the understanding of innovation patterns and trends easier and enhance the accessibility of patent data.

The following sections will delve into the related works that inspired this approach, outline the conceptual underpinnings of this methodology, and highlight its potential contributions to the field of patent analysis and beyond.

* ...@protonmail.com

RELATED WORK

In previous work of Yang et al. [1] the importance of visualization tools to enhance patent landscape analysis has been stressed. They propose the use of a tool to analyze and visualize patent data, thereby aiding in the identification of patterns and trends within large data sets. This study indicates the potential of visualization tools in patent analysis and the need for further development in this area.

Wittenburg et al. [2] further explores the importance of visualization tools for patent landscaping. The authors propose a novel tool that allows users to compare different technologies using multi-dimensional comparative visualization. This tool provides an interactive environment where users can manipulate the display to understand and compare different technology fields. The findings in this paper underpin the significance of interactive visualization in patent landscape analysis and its potential to enhance understanding and decision-making.

CONCEPT

The proposed methodology centers on transforming the complex, tabular patent data into a visually insightful and analytically rich heterogeneous graph using CSX. Inventors, institutions, companies, and other patent data elements are represented as nodes, with their interconnections depicted as edges. This transformation process uncovers hidden patterns and relationships within the patent data, resulting in an intuitive visual representation. In a first step patent data is extracted from the big amount of patent data that is available so that it can fit into a tabular dataset. In a second step data is transformed in the CSX platform into heterogenous graphs for analysis.

The CSX platform offers dynamic exploration capabilities, allowing users to interact with the generated graphs, adjust visualization parameters, and explore different perspectives of the data. This interactive approach enables a deeper understanding of the innovation landscape, facilitating the discovery of technological trends, key innovators, collaboration networks, and competitive landscapes.

This research's novelty lies in its innovative application of CSX for the visualization and analysis of patent data, introducing a new dimension to patent analysis. Furthermore, by offering an interactive and dynamic exploration of the innovation landscape, it paves the way for strategic decision-making in technology development and innovation management.

https://doi.org/10.5281/zenodo.10626851

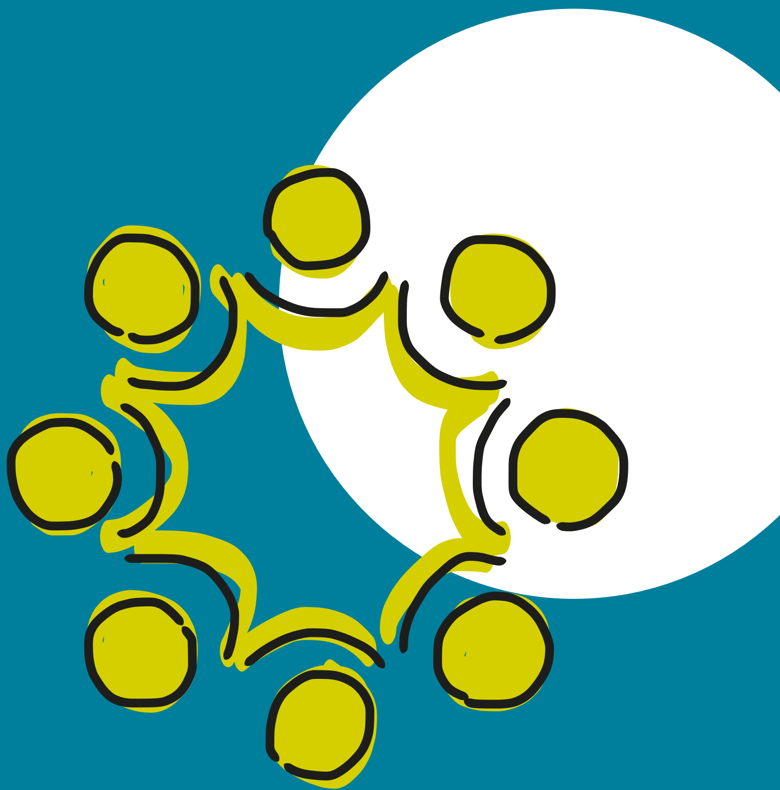
REFERENCES

[1] Y. Y. Yang, L. Akers, C. B. Yang, T. Klose, and S. Pavlek, "Enhancing patent landscape analysis with visualization output," *World Patent Information*, vol. 32, no. 3, pp. 203–220, 2010.

[2] K. Wittenburg and G. Pekhteryev, "Multi-dimensional comparative visualization for patent landscaping," in *Proceedings of businessvis workshop*, 2015.

Appendix

List of Autors



open search foundation

CERN

ISSN: 2957-4935
ISBN: 978-92-9083-653-7