

Open Web Search at LongEval 2023

Reciprocal Rank Fusion on Automatically Generated Query Variants



CLEF 2023, 18–21 September, Thessaloniki

Maik Fröbe, Gijs Hendriksen, Arjen P. de Vries, Martin Potthast

University of Jena

Radboud Universiteit Nijmegen

Leipzig University and ScaDS.AI

@webis_de

www.webis.de

RMIT at the 2017 TREC CORE Track

Rodger Benham
RMIT University
Melbourne, Australia

Luke Gallagher
RMIT University
Melbourne, Australia

Joel Mackenzie
RMIT University
Melbourne, Australia

Tadele T. Damessie
RMIT University
Melbourne, Australia

Ruey-Cheng Chen
RMIT University
Melbourne, Australia

Falk Scholer
RMIT University
Melbourne, Australia

Alistair Moffat
The University of Melbourne
Melbourne, Australia

J. Shane Culpepper
RMIT University
Melbourne, Australia

RMIT at the 2017 TREC CORE Track

Topic: 430

Description: Identify instances of attacks on humans by Africanized (killer) bees.

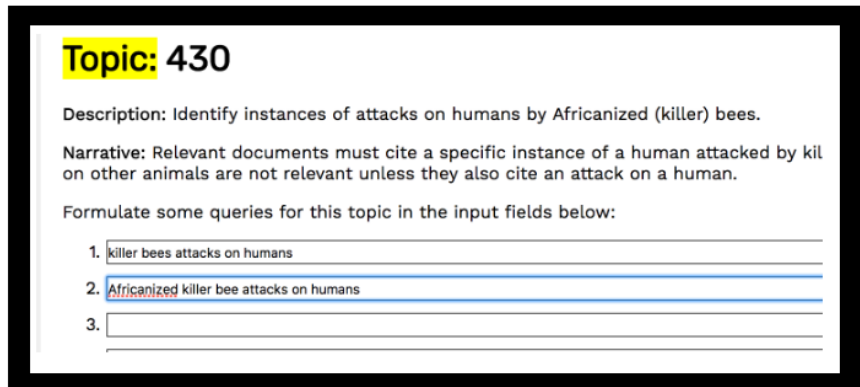
Narrative: Relevant documents must cite a specific instance of a human attacked by killer bees. Attacks on other animals are not relevant unless they also cite an attack on a human.

Formulate some queries for this topic in the input fields below:

1.
2.
3.

- ❑ User query variants may substantially improve retrieval effectiveness [Bailey'16, Benham'17, etc.]
- ❑ Experts formulate meaningful user query variants on a given topic
- ❑ Rank fusion on user query variants

RMIT at the 2017 TREC CORE Track



Topic: 430

Description: Identify instances of attacks on humans by Africanized (killer) bees.

Narrative: Relevant documents must cite a specific instance of a human attacked by killer bees. Attacks on other animals are not relevant unless they also cite an attack on a human.

Formulate some queries for this topic in the input fields below:

1.
2.
3.

- ❑ User query variants may substantially improve retrieval effectiveness [Bailey'16, Benham'17, etc.]
- ❑ Experts formulate meaningful user query variants on a given topic
- ❑ Rank fusion on user query variants

Our Research Question

- ❑ Intentions and preferences can change considerably over time [Huang'22]
- ❑ Must retrieval pipelines with query variants adapt over time?

Open Web Search at LongEval 2023

Our Approach

- In the best case, we would have manually formulated query variants

Open Web Search at LongEval 2023

Our Approach

- In the best case, we would have manually formulated query variants, but
 - No topic descriptions available
 - Difficult to guess what searchers did look for in retrospect
 - We have 2023, so lets use ChatGPT :)

Open Web Search at LongEval 2023

Our Approach

- In the best case, we would have manually formulated query variants, but
 - No topic descriptions available
 - Difficult to guess what searchers did look for in retrospect
 - We have 2023, so lets use ChatGPT :)

Pipeline

- Step 1: Generate query Variants with ChatGPT
- Step 2: Reciprocal rank fusion over query variants to produce joint ranking

Open Web Search at LongEval 2023

Our Approach

- ❑ In the best case, we would have manually formulated query variants, but
 - No topic descriptions available
 - Difficult to guess what searchers did look for in retrospect
 - We have 2023, so lets use ChatGPT :)

Pipeline

- ❑ Step 1: Generate query Variants with ChatGPT
- ❑ Step 2: Reciprocal rank fusion over query variants to produce joint ranking

Hyperparameters Tuned on the Training Data

- ❑ Which prompt for ChatGPT?
- ❑ How many query variants?
- ❑ Which retrieval model?

Open Web Search at LongEval 2023

Our Best ChatGPT Prompt

Number	2
Prompt	I wanted to search the web for “<ORIGINAL_QUERY>” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.

Open Web Search at LongEval 2023

Our Best ChatGPT Prompt

Number	2
Prompt	I wanted to search the web for “<ORIGINAL_QUERY>” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.

Example Input	I wanted to search the web for “ <u>4kvideo downloader</u> ” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.
---------------	--

Example Response	1. “Best 4k video downloader”, 2. “4k video download software”, 3. “How to download 4k videos”, 4. “Top-rated video downloader for 4k”, 5. “4k video downloader for Mac/Windows”, 6. “Free 4k video download tools”, 7. “Recommended 4k video downloader apps”, 8. “Download 4k videos from YouTube”, 9. “4k video downloader review”, 10. “Fastest 4k video downloader”.
------------------	---

Open Web Search at LongEval 2023

Our Best ChatGPT Prompt

Number	2
Prompt	I wanted to search the web for “<ORIGINAL_QUERY>” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.

Example Input	I wanted to search the web for “ <u>4kvideo downloader</u> ” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.
Example Response	1. “Best 4k video downloader”, 2. “4k video download software”, 3. “How to download 4k videos”, 4. “Top-rated video downloader for 4k”, 5. “4k video downloader for Mac/Windows”, 6. “Free 4k video download tools”, 7. “Recommended 4k video downloader apps”, 8. “Download 4k videos from YouTube”, 9. “4k video downloader review”, 10. “Fastest 4k video downloader”.

Which Retrieval Model

- ❑ We had 10 lexical retrieval models from PyTerrier in the comparison
- ❑ BM25, LGD, PL2, ...

Open Web Search at LongEval 2023

Our Best ChatGPT Prompt

Number	2
Prompt	I wanted to search the web for “<ORIGINAL_QUERY>” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.
Example Input	I wanted to search the web for “ <u>4kvideo downloader</u> ” but I was not able to find relevant documents. Please recommend me up to ten better queries in a numbered list that retrieve relevant documents. All Queries must be English and have at least two words.
Example Response	1. “Best 4k video downloader”, 2. “4k video download software”, 3. “How to download 4k videos”, 4. “Top-rated video downloader for 4k”, 5. “4k video downloader for Mac/Windows”, 6. “Free 4k video download tools”, 7. “Recommended 4k video downloader apps”, 8. “Download 4k videos from YouTube”, 9. “4k video downloader review”, 10. “Fastest 4k video downloader”.

Which Retrieval Model

- ❑ We had 10 lexical retrieval models from PyTerrier in the comparison
- ❑ BM25, LGD, PL2, ...

How Many Query Variants? We tested 3, 5, and 10

Open Web Search at LongEval 2023

Lets look at the Evaluation



Open Web Search at LongEval 2023

Lets look at the Evaluation



TLDR: Query variants could be more effective

Approach / Run	nDCG@10		
	June	July	Sep.
ows-pl2-10-variants-prompt-2	0.163	0.164	0.167
BM25	0.163	0.180	0.184

Open Web Search at LongEval 2023

Lets look at the Evaluation



TLDR: Query variants could be more effective

Approach / Run	nDCG@10			Unjudged@10		
	June	July	Sep.	June	July	Sep.
ows-pl2-10-variants-prompt-2	0.163	0.164	0.167	0.801	0.799	0.788
BM25	0.163	0.180	0.184	0.782	0.771	0.760

Open Web Search at LongEval 2023

Lets look at the Evaluation



TLDR: Query variants could be more effective

Approach / Run	nDCG@10			Unjudged@10			Cond. nDCG@10		
	June	July	Sep.	June	July	Sep.	June	July	Sep.
ows-pl2-10-variants-prompt-2	0.163	0.164	0.167	0.801	0.799	0.788	0.440	0.480	0.488
BM25	0.163	0.180	0.184	0.782	0.771	0.760	0.446	0.476	0.478

Open Web Search at LongEval 2023

If there is some time left...

Open Web Search at LongEval 2023

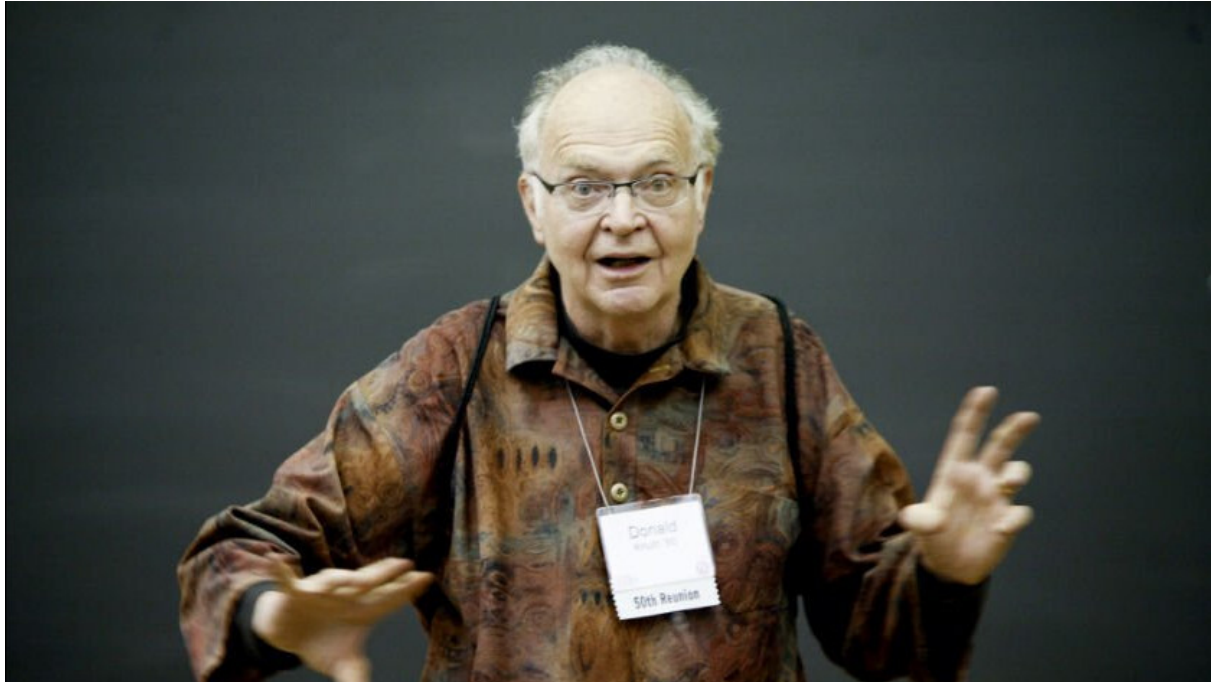
If there is some time left...



I would like to share some ideas for future work
(I.e., what we would have submitted with a bit more time,
including some advertisements)

Open Web Search at LongEval 2023

Motivation of Future Work

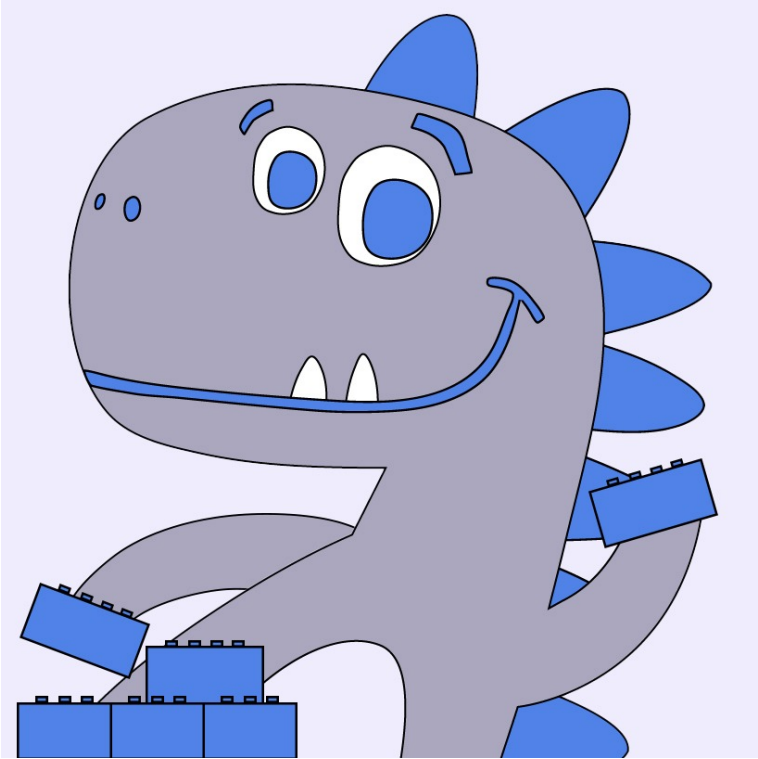


“People think that computer science is the art of geniuses but the actual reality is the opposite, just many people doing things that build on each other, like a wall of mini stones.”

Donald Knuth

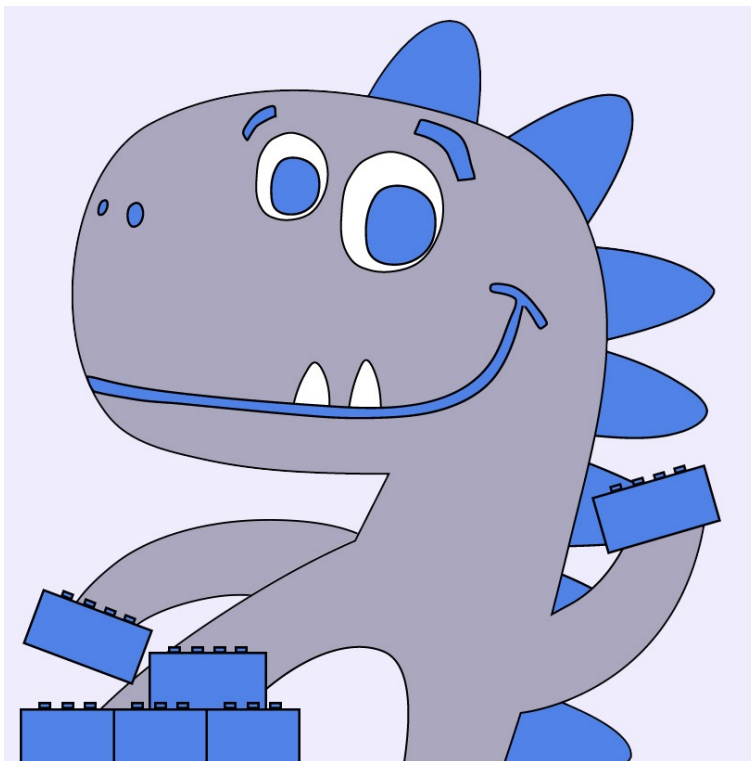
Open Web Search at LongEval 2023

Our Goal with TIREx



Open Web Search at LongEval 2023

Our Goal with TIREx



We want to simplify collaboration by promoting software submissions

- ❑ Submit software instead of run files (currently 34 retrieval datasets)
- ❑ Preferably small reusable components of retrieval pipelines
- ❑ Many components must only be executed “once in a lifetime”

Open Web Search at LongEval 2023

How would this look like?

- ❑ Reusability: Approach implemented against ir_datasets
- ❑ Reproducibility: self-contained Docker image executed in a sandbox
- ❑ Classes: Query processing, document processing, query–document proc.

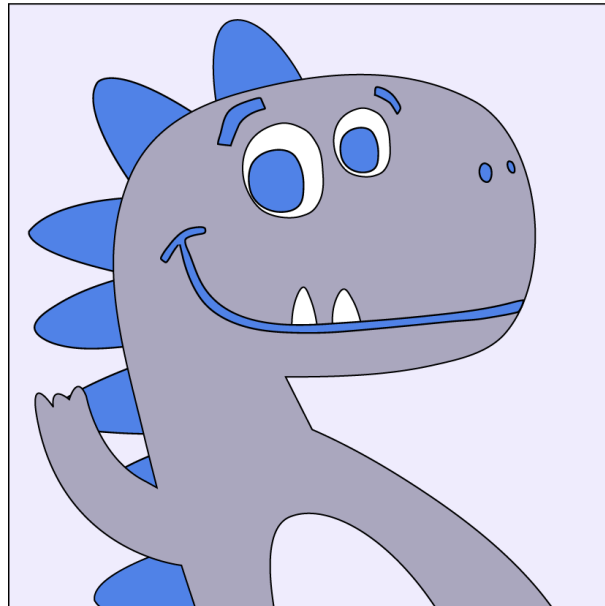
Open Web Search at LongEval 2023

How would this look like?

- ❑ Reusability: Approach implemented against ir_datasets
- ❑ Reproducibility: self-contained Docker image executed in a sandbox
- ❑ Classes: Query processing, document processing, query–document proc.

Advantages of TIREx

- ❑ Improved Reusability + Reproducibility
- ❑ Blinded experimentation on new corpora possible



Open Web Search at LongEval 2023

Example: Query Segmentation

WWW 2011 – Session: Query Analysis

March 28–April 1, 2011, Hyderabad, India

Query Segmentation Revisited

Matthias Hagen

Martin Potthast

Benno Stein

Christof Bräutigam

Query Segmentation Revisited

Matthias Hagen

Martin Potthast

Benno Stein

Christof Bräutigam

- ❑ Input

Query: hubble telescope achievements

- ❑ Output

Query segments: hubble telescope | achievements

Query Segmentation Revisited

Matthias Hagen

Martin Potthast

Benno Stein

Christof Bräutigam

❑ Input

Query: `hubble telescope achievements`

❑ Output

Query segments: `hubble telescope | achievements`

Currently, I dockerize and submit such components to TIREx.

Open Web Search at LongEval 2023

The paper and code is published. Why Docker, sandboxing, etc?

Open Web Search at LongEval 2023

The paper and code is published. Why Docker, sandboxing, etc?

INSTRUCTIONS



Open Web Search at LongEval 2023

The paper and code is published. Why Docker, sandboxing, etc?

INSTRUCTIONS



Outcome

- ❑ Query segmentation might be useful for query rewriting
- ❑ `hubble telescope achievements` \Rightarrow `"hubble telescope" achievements`

Outcome

- ❑ Query segmentation might be useful for query rewriting
- ❑ `hubble telescope achievements` \Rightarrow `"hubble telescope"` achievements

PyTerrier Pipeline

Setup Retrieval Pipeline

```
query_segmentation = tira.pt.transform_queries('webis-query-segmentation/hyb-a', dataset)
pipeline = query_segmentation >> rewrite_segments_to_phrases >> bm25
```

[5]

Python

- ❑ Improves `nDCG@10` from 0.00 to 0.31 for `hubble telescope achievements`
- ❑ Use cached outputs from TIREx if available, otherwise use Docker image

Outcome

- ❑ Query segmentation might be useful for query rewriting
- ❑ `hubble telescope achievements` \Rightarrow `"hubble telescope" achievements`

PyTerrier Pipeline

Setup Retrieval Pipeline

```
query_segmentation = tira.pt.transform_queries('webis-query-segmentation/hyb-a', dataset)
pipeline = query_segmentation >> rewrite_segments_to_phrases >> bm25
```

[5]

Python

- ❑ Improves `nDCG@10` from 0.00 to 0.31 for `hubble telescope achievements`
- ❑ Use cached outputs from TIREx if available, otherwise use Docker image

Hopefully Collaborative Effort to Collect Components

- ❑ Query processing: Query intent, entity linking, expansion, reduction, ...
- ❑ Document processing: Genre, Spam, expansion, reduction, ...

Outcome

- ❑ Query segmentation might be useful for query rewriting
- ❑ `hubble telescope achievements` \Rightarrow `"hubble telescope" achievements`

PyTerrier Pipeline

Setup Retrieval Pipeline

```
query_segmentation = tira.pt.transform_queries('webis-query-segmentation/hyb-a', dataset)
pipeline = query_segmentation >> rewrite_segments_to_phrases >> bm25
```

[5]

Python

- ❑ Improves `nDCG@10` from 0.00 to 0.31 for `hubble telescope achievements`
- ❑ Use cached outputs from TIREx if available, otherwise use Docker image

Hopefully Collaborative Effort to Collect Components

- ❑ Query processing: Query intent, entity linking, expansion, reduction, ...
- ❑ Document processing: Genre, Spam, expansion, reduction, ...

Time is of the Essence

- ❑ Very easy if you do this early or from the start
 - Excellent support for self-contained development containers in VS Code
- ❑ Might be difficult in some years: Dependencies/binaries become unavailable

Open Web Search at LongEval 2023

Conclusions and Takeaways

- ❑ Is the retrieval effectiveness of generated query variants stable over time?
 - No substantial improvement
 - But: retrieval effectiveness was stable :)
- ❑ Fusing query variants with lexical retrieval did not improve effectiveness
 - Maybe not enough effort into prompt engineering?
 - Maybe more details, like topic description/narrative required?

Open Web Search at LongEval 2023

Conclusions and Takeaways

- ❑ Is the retrieval effectiveness of generated query variants stable over time?
 - No substantial improvement
 - But: retrieval effectiveness was stable :)
- ❑ Fusing query variants with lexical retrieval did not improve effectiveness
 - Maybe not enough effort into prompt engineering?
 - Maybe more details, like topic description/narrative required?

Work in Progress and Future Work

- ❑ We hope to collect a pool of diverse retrieval components
 - We currently dockerize unavailable components:
query segmentation, keyphrase extraction, genre classification, etc.
- ❑ In the best case, this is a big collaborative effort
 - Please do not hesitate to dockerize your favorite retrieval component :)
 - I am happy to help

Open Web Search at LongEval 2023

Conclusions and Takeaways

- ❑ Is the retrieval effectiveness of generated query variants stable over time?
 - No substantial improvement
 - But: retrieval effectiveness was stable :)
- ❑ Fusing query variants with lexical retrieval did not improve effectiveness
 - Maybe not enough effort into prompt engineering?
 - Maybe more details, like topic description/narrative required?

Work in Progress and Future Work

- ❑ We hope to collect a pool of diverse retrieval components
 - We currently dockerize unavailable components:
query segmentation, keyphrase extraction, genre classification, etc.
- ❑ In the best case, this is a big collaborative effort
 - Please do not hesitate to dockerize your favorite retrieval component :)
 - I am happy to help

Thank You!