

Information Retrieval

2021-2-F1801Q110

Lorenzo Pirola - 816418

Personalized Search Engine for microblog

Introduction

The aim of this project is to design and build a search engine for a micro-blog, whose content is based on tweets published by a set of users. The implemented system allows the user to perform a simple or advanced search and it offers the possibility to rank the results according to his interests. The database and the search engine were based on the bonsai.io¹ platform, which offers a free Elasticsearch installation deployed on the cloud. The crawler used to collect data from Twitter and the search framework were built entirely on Java using the Java REST Client (High Level) for Elasticsearch and the Twitter4J library. Finally, the user interface is provided through a simple web-app developed using the Spring and Thymeleaf frameworks. The application was deployed to

¹ <https://bonsai.io>

the Heroku² platform, so it is reachable at the URL <https://information-retrieval-search.herokuapp.com> (it could take 15-20 seconds to start). The code is accessible on a public GitHub repository at the URL <https://github.com/lpirola13/search-engine.git>.

Dataset

The content of the micro-blog comes from the tweets of six different users: three of them are related to the technology world (*Bill Gates*, *Tim Cook* and *Sundar Pichai*), while the other three are related to US politics (*Barack Obama*, *Joe Biden* and *Hillary Clinton*). For each user, the crawler collected his data and his last 1000 tweets, alternating a download phase of 10 tweets (retweets excluded), an indexing phase and a break of 10 seconds.

The collected data was stored as documents in Elasticsearch. In this way, two distinct indices were created: one to store the tweets and one to store the user's data. Both of them were distributed on a single shard with no replicas. Figure 1 shows the structure of indexed documents and the type of each field.

Tweet	User
id: <i>_id</i>	id: <i>_id</i>
text: <i>text</i>	name: <i>keyword</i>
parsed_text: <i>text</i>	screenName: <i>keyword</i>
hashtags: <i>array of keyword</i>	hashtags: <i>array of keyword</i>
mentions: <i>array of keyword</i>	profile: <i>array of keyword</i>
created_at: <i>date</i>	
user.id: <i>text</i>	
user.name: <i>keyword</i>	
user.screenName: <i>keyword</i>	

Fig. 1 Structure of documents stored in database. Each field is described by its name and its type. (Left) Tweet's structure. (Right) User's structure.

² <https://www.heroku.com/home>

Only some fields were filled directly with the information retrieved from Twitter. These fields are *id*, *text*, *created_at*, *user.id*, *user.name*, *user.screenName* for tweets and *id*, *name*, *screenName* for users. The *hashtags* and *mentions* fields in tweets were filled with two distinct arrays containing the occurrences of hashtags and mentions found in the *text* field, using two regexps. Moreover, the occurrences of emojis in the text fields were replaced by their aliases, e.g., 📌 is replaced with “:point_down:”, and the obtained result was copied to the *parsed_text* field. As shown in the next section, the textual search is based on this field, while the content of the original *text* field is used only for visualization (it has not even been indexed). Regarding the users, no further pre-processing was applied before indexing. The *hashtags* and *profile* fields are used for personalized search, so their content was left blank at index time. The entire process is summarized in Figure 2.

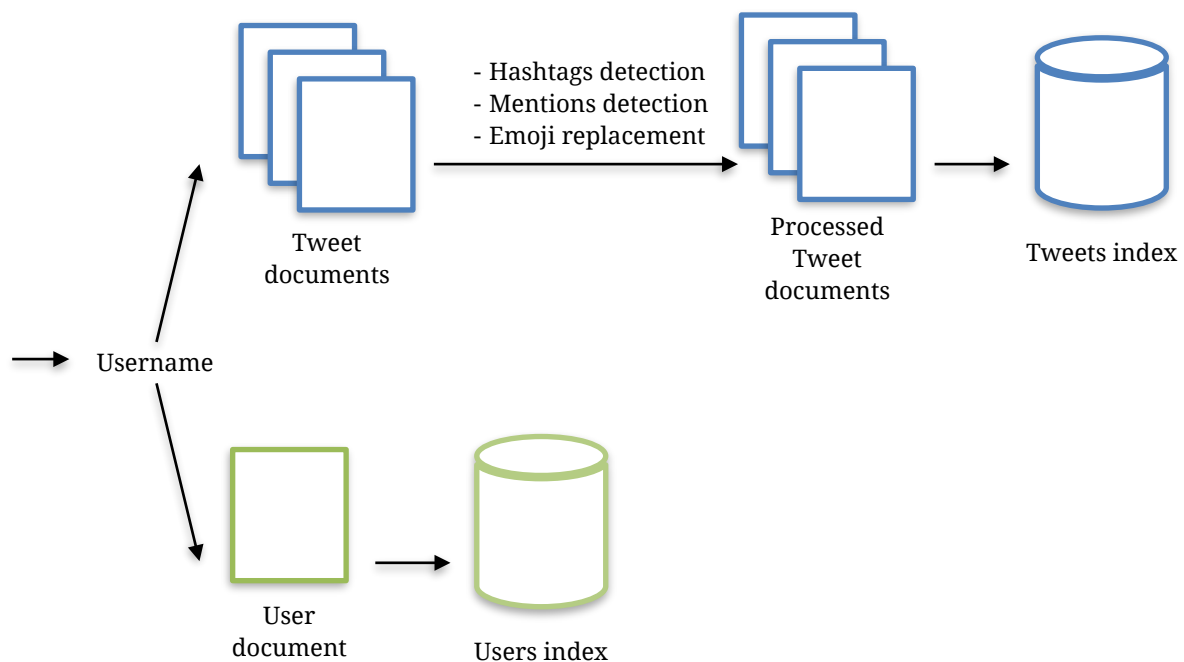


Fig. 2 Crawling process. Tweet documents need a pre-processing phase before getting indexed.

At the end of the crawling and the indexing processes, a total of 4149 tweet documents were collected. They have an average length of nearly 29 words and were published on a period of time between 9/12/2015 and 23/12/2020. More details are shown in Table 1.

Tab. 1 Summary statistics for each user.

User	Documents	Average number of words	Oldest Tweet	Newest Tweet
Bill Gates	748	29.295	09-04-2018	22-12-2020
Tim Cook	793	28.996	06-02-2017	17-12-2020
Sundar Pichai	573	24.120	09-12-2015	12-12-2020
Barack Obama	721	33.109	23-09-2016	22-12-2020
Joe Biden	723	27.804	14-10-2020	23-12-2020
Hillary Clinton	591	27.619	08-05-2020	22-12-2020
Summary	4149	28.688	09-12-2015	23-12-2020

Search Engine

The Elasticsearch indexing process uses three customized components. The first one is the *custom_analyzer*, which was used to analyze the *parsed_text* field of tweets. It consists of:

- One UAX URL Tokenizer.
- Two char filters. They were used to remove the occurrences of “.” and “_” from the emojis’ aliases.
- A sequence of token filters including lowercase, apostrophe, stop and Porter stem.

The second component is the *custom_search_analyzer*, which is employed for the synonym-based query expansion. It is similar to the *custom_analyzer* but it includes a *synonym_graph* token filter, which expands the query with synonyms from a given list. This list was built using a synonyms API³ over random words took from the tweets. Figure 3 and Figure 4 shows the application of the *custom_analyzer* and *custom_search_analyzer*, respectively.

³ <https://words.bighugelabs.com/site/api>

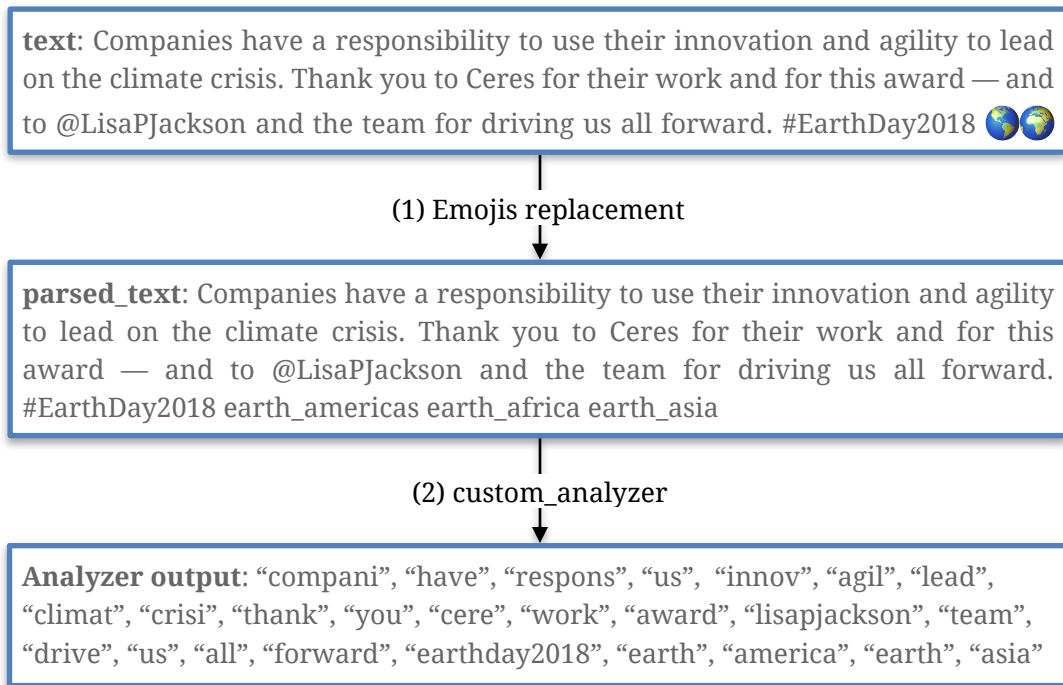


Fig. 3 Textual content processing. (1) The emojis are replaced with their textual aliases and the result is copied to the parsed_text field. (2) custom_analyzer is applied to the parsed_text field.

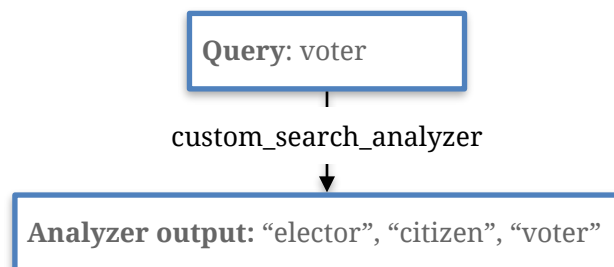


Fig. 4 Synonym-based query expansion of query term “voter” using custom_search_analyzer.

The third component is the *custom_normalizer*, which consists of a single lowercase filter and it is used for the keyword fields indexing.

The personalized search was implemented following a pre-processing approach, using a keyword-based query expansion. Thus, the user profiles were built retrieving the most “significant” 30 terms and the most popular 10 hashtags from their tweets. The terms and the hashtags were detected using a Significant Text aggregation and a simple aggregation over *parsed_text* and *hashtags* fields, respectively. These results were stored under the user’s *profile* and *hashtags* fields.

The search consists of a dynamic composition of a boolean query, in which clauses are added depending on user needs. This process is shown in Figure 5. The user can perform a search over *parsed_text* using a Query String query (with AND operator) and a search over *hashtags* or *mentions* using a Term query. The advanced search includes filters over *created_at* field using a Range query, the synonym-based query expansion through the *custom_search_analyzer* and the keyword-based query expansion. Given the selected user for personalization, this expansion adds two Should clauses to the query: one includes a Terms query over *parsed_text* using the most significant terms from his profile, while the other includes a Terms query over *hashtags* using his most popular hashtags. To obtain better personalized results, the search over the *parsed_text* field uses a similarity based on the query likelihood model with Dirichlet smoothing ($\mu = 2000$). It is also possible to hide the tweets published by the selected user.

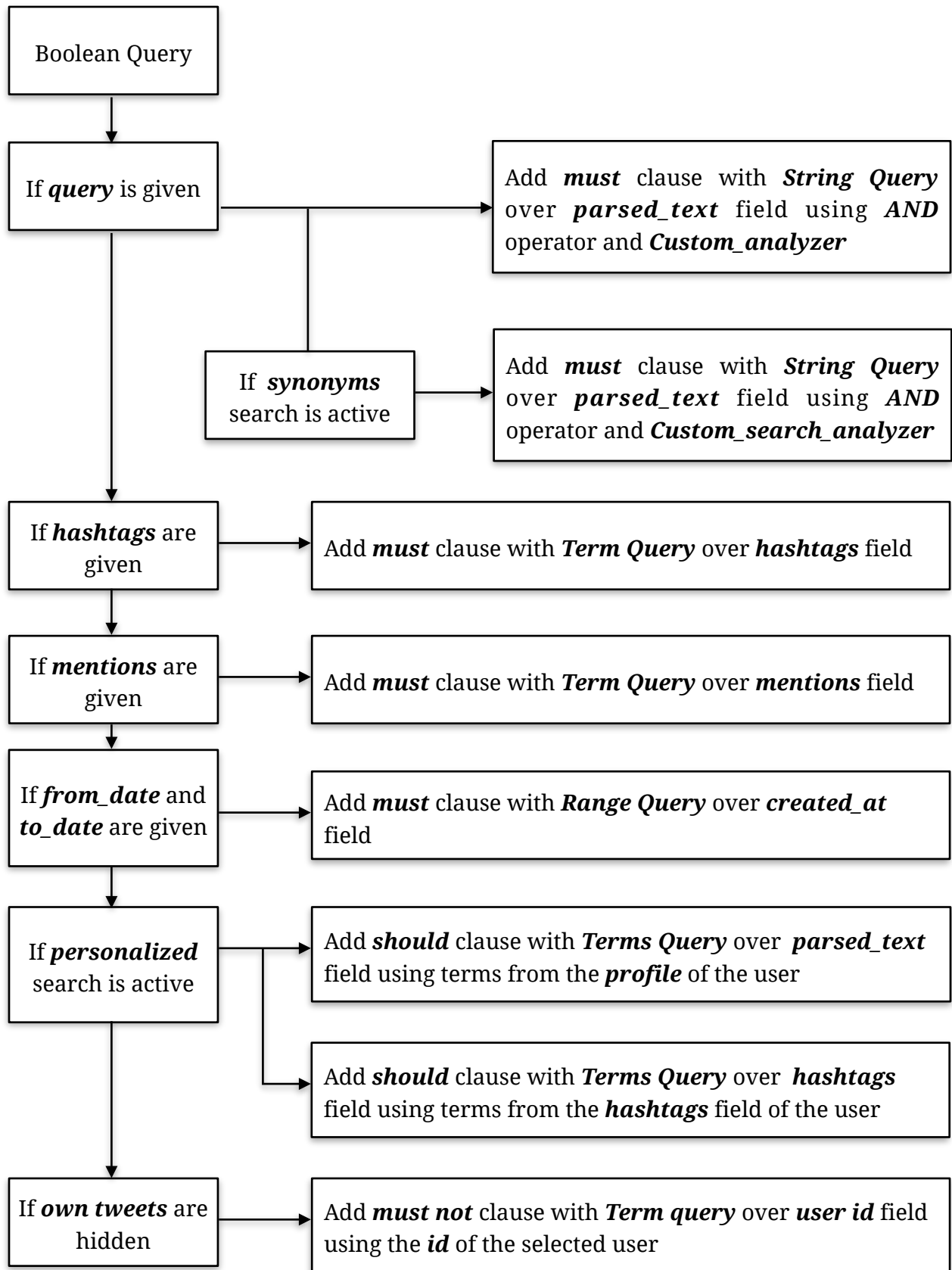


Fig. 5 Boolean query building process.

Demonstration plan

The user interface is shown in Figure 6. It includes a form made up of the fields needed to perform a simple or advanced search and the lists of the most popular hashtags and mentions among all the tweets. The terms of these lists were retrieved using two simple aggregations over their respective fields.

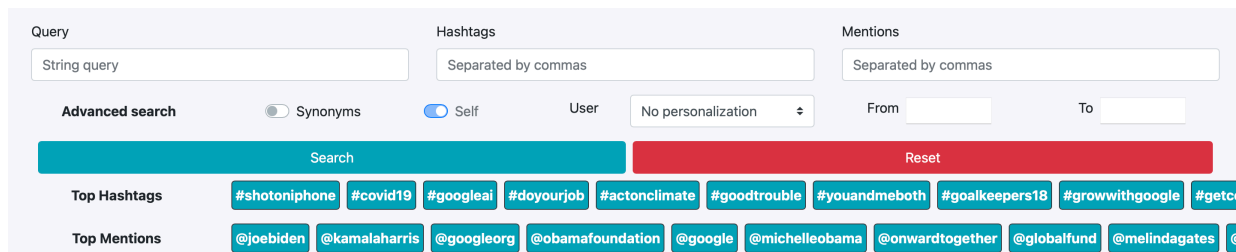


Fig. 6 User interface.

The following paragraphs show four different user cases:

1. A textual search can be performed by filling the Query field with the desired *query*. Figure 7 shows the first three results obtained by the query “earth”. It also shows the impact of the emojis in the search result.
2. A search can be performed also over a combination of fields. In this case, it is required to enter the desired *query*, *hashtags* and *mentions* in the appropriate fields, being careful of separating the *hashtags* and *mentions* using commas. Figure 8 shows the result obtained by the query “portrait” combined with the hashtag “#iphone” and the mention “@oyuelaphoto”.
3. The search can be personalized according to the interests of one of the users listed in the *User* field. Figure 9 shows the results obtained by the query “fight” without any kind of personalization, while Figure 10, 11, 12 show the results obtained by the same query using a personalization based on different users. The results for the user “Bill Gates” (Figure 10) are related to the fight against the spread of diseases,

the results for the user “*Barack Obama*” (Figure 11) are related to the fight against the climate change, while the results for the user “*Joe Biden*” (Figure 12) are related to the political campaign for the presidential elections. The tweets published by the same user chosen for personalization can be excluded using the *Self* switch.

4. The advanced search using the synonym-based query expansion can be activated by the *Synonyms* switch. Figure 13 shows the three results obtained by the query “polyomelitis”, which could not give any results without this kind of query expansion.

Position 1 **Score 1.39**

Tim Cook
@tim_cook 22-04-2018

Wishing everyone a happy #EarthDay2018! We all share this planet and a duty to protect it. 🌍🌍🌍

Position 2 **Score 1.38**

Tim Cook
@tim_cook 22-10-2019

Companies have a responsibility to use their innovation and agility to lead on the climate crisis. Thank you to Ceres for their work and for this award — and to @LisaPJackson and the team for driving us all forward. 🌍🌍🌍

Position 3 **Score 1.38**

Tim Cook
@tim_cook 17-07-2017

🤔🌍🌍🌍📅 Happy #WorldEmojiDay! 🎉 We've got some 😎 new ones to show you, coming later this year! 🙄📌 <https://t.co/xBR9ZJ7l4g> <https://t.co/fhDrr4J5KG>

Fig. 7 The first three results given by the query “earth”. The first and last results do not contain the term “earth” but the corresponding emojis.

Position 1 Score 14.28

Tim Cook
@tim_cook 23-09-2017

Portrait Lighting on the new iPhone 8 Plus by @oyuelaphoto is amazing. Can't wait to see what our customers do with the new #iPhone! <https://t.co/5CdfeUDHCU>

Fig. 8 The result given by the query “portrait” combined with the hashtag “#iphone” and the mention “@oyuelaphoto”.

Position 1 Score .65

Hillary Clinton
@HillaryClinton 18-09-2020

The single most important fight of our times, the fight that makes it possible to wage every other fight, is the fight to protect the right to vote.

Position 2 Score .37

Joe Biden
@JoeBiden 03-11-2020

Donald Trump will always fight for his wealthy and well-connected friends. I'll always fight for you.

Position 3 Score .37

Hillary Clinton
@HillaryClinton 11-09-2020

One of the best ways to honor those we lost 19 years ago is by fighting for those who survived. It's outrageous that first and second responders are still fighting for the health care they deserve.

Fig. 9 The first three results given by the query “fight” without personalization.

Position 1
Score 2.20

Bill Gates
@BillGates 17-05-2018

Glad to see this story told. I'm always inspired by the local leaders and health workers involved in the fight to #endpolio. <https://t.co/WJuWGzcOr5>

Position 2
Score 2.20

Bill Gates
@BillGates 29-07-2019

It's hard to overstate how important finding a better diagnostic is for fighting #Alzheimers. That's why I'm investing in new ideas for easier and more accurate diagnosis of the disease. <https://t.co/WM9phNmyJo>

Position 3
Score 2.19

Bill Gates
@BillGates 20-11-2019

Some of our most vital partners in the fight to #EndPolio are philanthropies like @alwaleed_philan, @bloombergdotorg, Tahir Foundation, Dalio Philanthropies & Ningxia Yanbao Charity Foundation. Their support has helped deliver vaccines and protect millions from polio paralysis. <https://t.co/w6yUIN1CwX>

Fig. 10 The first three results given by the query “fight” with user personalization set on “Bill Gates”. All of them are about the fight against the spread of diseases.

Position 1 Score 2.20

Barack Obama
@BarackObama 06-10-2016

ICYMI: Read about the historic #ParisAgreement and what it means for the fight to #ActOnClimate. <https://t.co/eMefgFZk53>

Position 2 Score 2.20

Barack Obama
@BarackObama 05-10-2016

This historic step in the fight to #ActOnClimate came faster than anyone predicted. <https://t.co/W2rtcNXkl7>

Position 3 Score 2.20

Barack Obama
@BarackObama 05-10-2016

"Today is a historic day in the fight to protect our planet for future generations." —President Obama #ActOnClimate <https://t.co/x3dJSCYUcj>

Fig. 11 The first three results given by the query “fight” with user personalization set on “Barack Obama”. All of them are about climate change.

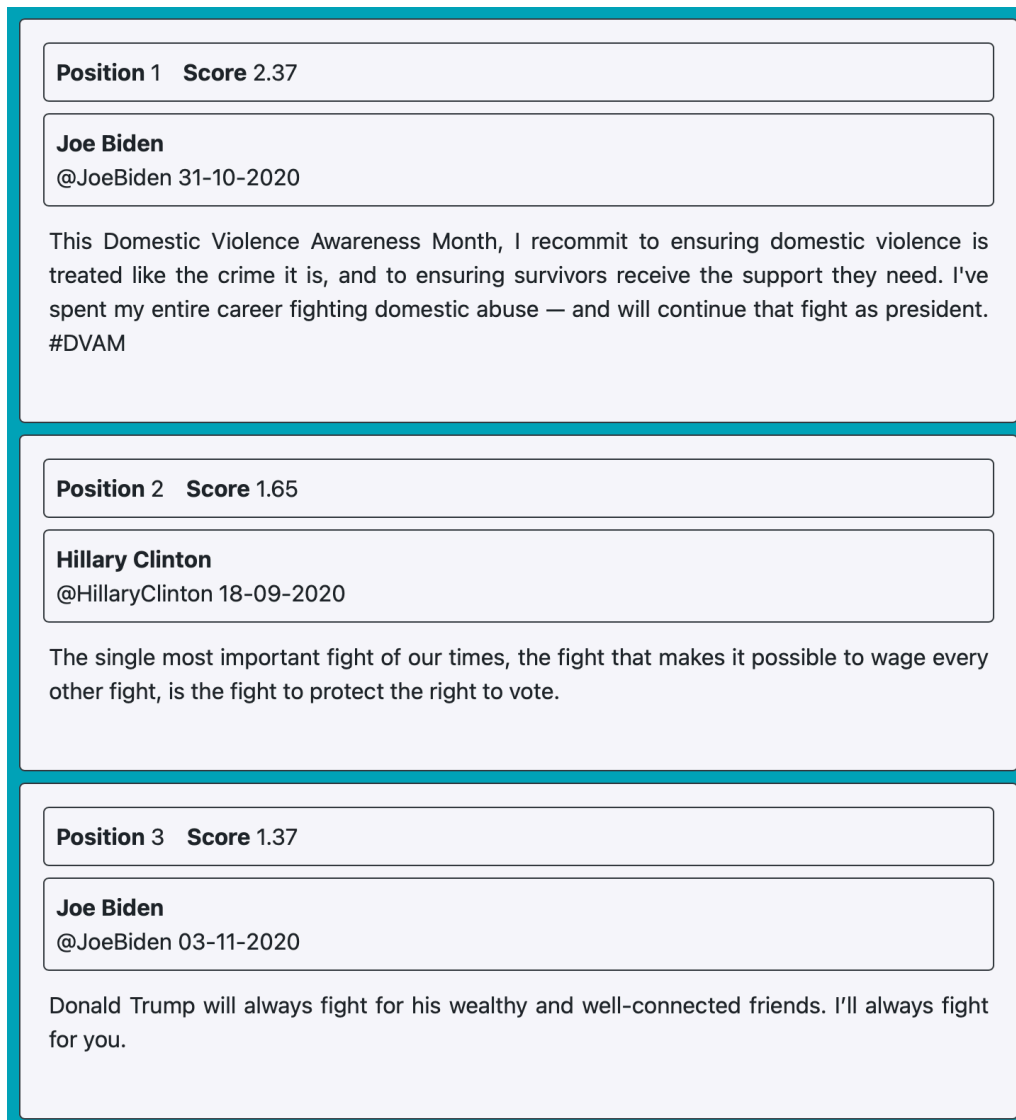


Fig. 12 The first three results given by the query “fight” with user personalization set on “Joe Biden”. All of them are about his political campaign for the presidential elections.

Position 1 Score 2.79
Bill Gates @BillGates 30-12-2018
<p>Almost a decade ago, I predicted that the world would soon be polio free. But we had more polio cases in 2018 than in 2017. Even so, I remain optimistic that we could eradicate polio soon.</p>
Position 2 Score 2.65
Bill Gates @BillGates 16-01-2020
<p>These global health heroes have helped Bangladesh become a model for other countries for how to respond to infectious disease outbreaks: https://t.co/5CEHpFCKlp https://t.co/KI6ADN43ux</p>
Position 3 Score 2.64
Bill Gates @BillGates 07-10-2020
<p>For the last 25 years, Dr. Firdausi Qadri, an immunologist and infectious disease researcher in Bangladesh, has been working to protect entire communities from cholera epidemics. https://t.co/F2pQilYqry</p>

Fig. 13 The first three results given by the query “polyomelitis” using synonym-based query expansion.