Project Number: **770299**

NewsEye:

A Digital Investigator for Historical Newspapers

Research and Innovation Action
Call H2020-SC-CULT-COOP-2016-2017

D7.8: NewsEye Demonstrator (c) (final)

Due date of deliverable: M45 (31 January 2022)

Actual submission date: 31 January 2022

Start date of project: 1 May 2018 **Duration:** 45 months

Partner organization name in charge of deliverable: ULR

Project co-funded by the European Commission within Horizon 2020		
Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	-
RE	Restricted to a group specified by the Consortium (including the Commission Services)	-
CO	Confidential, only for members of the Consortium (including the Commission Services)	-

Revision History

Document administrative information		
Project acronym:	NewsEye	
Project number:	770299	
Deliverable number:	D7.8	
Deliverable full title:	NewsEye Demonstrator (c) (final)	
Deliverable short title:	NewsEye Demonstrator (final)	
Document identifier:	NewsEye-T71-D78-NewsEyeDemonstrator-c-final-Submitted-v6.0	
Lead partner short name:	ULR	
Report version:	V6.0	
Report preparation date:	31.01.2022	
Dissemination level:	PU	
Nature:	Report	
Lead author:	Axel Jean-Caurant (ULR)	
Co-authors:	Antoine Doucet (ULR)	
Internal reviewers:	Lidia Pivarova (UH-CS), Günter Hackl (UIBK-IUI)	
Status:		Draft
		Final
	x	Submitted

The NewsEye Consortium partner responsible for this deliverable has addressed all comments received, making changes as necessary. Changes to this document are detailed in the change log table below.

Change Log

Date	Version	Editor	Summary of changes made
16/03/2021	1.0	Axel Jean-Caurant (ULR)	Final draft submitted to reviewers
26/03/2021	1.1	Axel Jean-Caurant (ULR)	Internal reviews taken into account
09/04/2021	2.0	Axel Jean-Caurant (ULR)	Final version after further minor modifications
17/04/2021	3.0	Antoine Doucet (ULR)	Minor adjustments towards submission
10/01/2022	4.0	Axel Jean-Caurant (ULR)	Draft update covering the work done during the extension period
17/01/2022	5.0	Axel Jean-Caurant (ULR)	Final update taking reviews into account
31/01/2022	6.0	Antoine Doucet (ULR)	Minor adjustments and submission

Executive summary

The objectives of WP7 of the NewsEye project are to disseminate and exploit the research results produced during the project. The role of the NewsEye Demonstrator showcased in this report is to prove the practicality of these tools in a well-structured environment, similar to most digital library websites already available online. The Demonstrator makes use of the various APIs developed in WP3, WP4 and WP5 to allow users to answer precise research questions such as the contextualised case studies described in public Deliverable D6.10. While the present deliverable provides a general description of the capabilities of the Demonstrator, the public Deliverables of WP6 provide feedback on it (D6.9) as well as more information on how it can be used by researchers in real use-cases (D6.12), and how this and other project results can be used by the general public (D6.11).

It is to be noted that, even though the Demonstrator is technically a part of Task T7.1, it is very strongly linked to Task T3.4 (Tool to query the initial and enriched data set, as described as of April 2019 in public Deliverable D3.4). In reality, the Demonstrator provides the updated results of both of these tasks.

Contents

1 Introduction

The NewsEye platform presented in this document has been developed to showcase the various outputs of the project. It gathers elements from all technical work packages (WP), from WP2 to WP5 (see Figure 1). The data and tools have been produced in WP2 to WP5 in the following steps:

- **WP2** is responsible of processing the raw images provisioned by the partner national libraries. There work consists on identifying the textual content of these images. To do so, they first identified meaningful lines in the page images, before using an ATR model to extract the text. WP2 is also in charge of gathering the lines identified in consistent groups, to recreate the different articles that can be found in a newspaper page.
- **WP3** then enriches these articles with semantic annotations. In particular, named entities are identified and qualified in various ways.
- **WP4** makes use of the articles to train and apply topic models, which goal is to help classify articles with similar semantic content.
- **WP5** is in charge of developing the Personal Research Assistant (PRA) which uses the data produced in the previous work packages to offer various analysis methods.

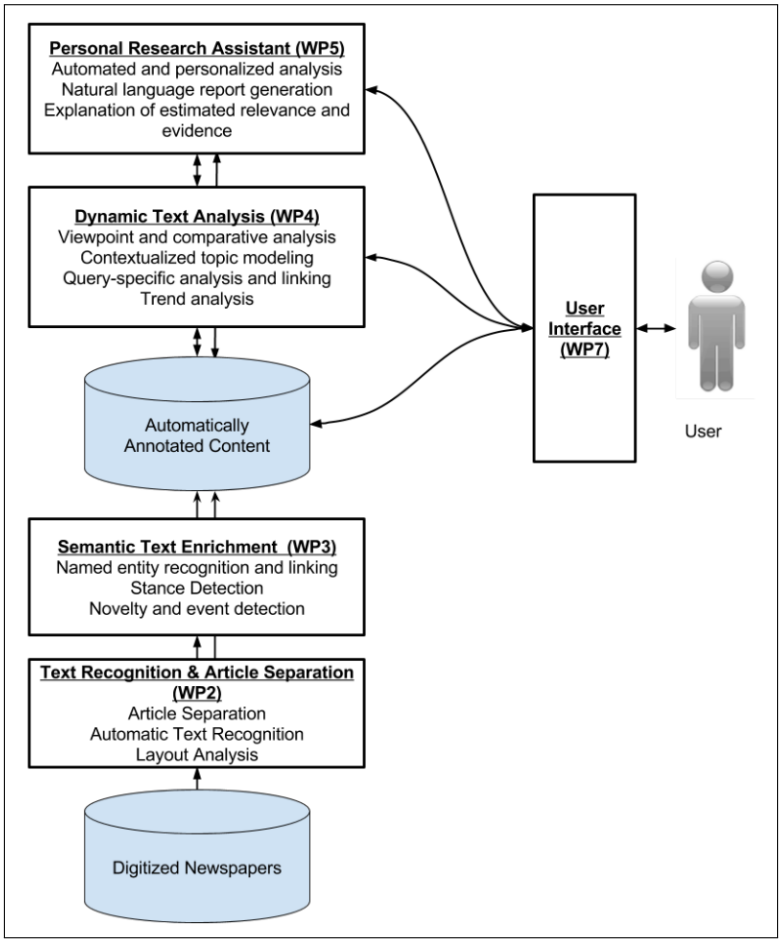

Figure 1: The role of the different work packages of the NewsEye project in processing and using the newspaper data.

As a reminder, WP6 then uses all this data and tools to try to answer concrete historical research problems. The NewsEye platform is developed within WP7, as a key asset to increase the impact of the

project results. A few research papers were published to disseminate aspects of the demonstrator as it stood at the time of publication [1, 2, 3].

2 Available data

Inside the NewsEye platform, users have the possibility to access and search through various issues and articles from twenty different newspapers, in four languages: German, French, Finnish and Swedish. It is to be noted that an English newspaper is currently being processed to be added to the available collection, as a more widely accessible showcase. These have been provided by the three partner libraries of the project, namely the *"Bibliothèque nationale de France"* (BNF, France), the *"Österreichische Nationalbibliothek"* (ONB, Austria) and the *"Kansalliskirjasto"* (or *"Nationalbibliotek"*) (NLF, Finland). These documents were processed according to the pipeline described in the introduction of this deliverable. These documents span a period of time from around 1850 to 1950. Several newspaper titles are represented in the different languages the project focuses on (French, German, Finnish and Swedish).

2.1 Documents

Two types of documents are available in the NewsEye platform. Entire issues or individual articles can be searched for, using a variety of keywords and search features. Both individual articles and entire issues are indexed in the platform. This indexing process allows for quick retrieval of information, and by comparing the keywords of the query to the content of documents, the system can find relevant documents. These documents were indexed in two different ways to allow for various search criteria. First, the documents were indexed "as is", where the words of documents were left unchanged to allow for precise search. The documents are also indexed in their stemmed version, where each word is transformed into its stem. For example, the terms "fishing", "fisher" or "fishes" were all transformed into the stem "fish".

2.2 Metadata

Different types of metadata are available for every document in the NewsEye platform. The metadata can either be used to help users during a search by filtering a list of results, or to further analyse a collection of documents using the Personal Research Assistant described in Section 4.3.

2.2.1 Bibliographic metadata

The metadata were directly obtained from the three national libraries, partners of the project. These are typical metadata that can be used to refine a search or filter a list of results, such as the publication date, the language of the documents and the newspaper they come from.

2.2.2 Named entities

The named entities are important information that can be found in the full text of issues and articles. They can be of different types according to the material and what is wanted to be analysed. In our case,

we focused on typical named entities that can be useful for a large range of users and that are very well represented in the collection that was processed by the NewsEye project (see public deliverable D3.5). Namely, the named entities that were extracted from the collection are:

- **Persons:** the name of a character, whether real or fictive.
- **Locations:** the name of a place
- **Organisations:** the name of an administrative entity
- **Human productions:** the name of a content creator (typically the name of a newspaper for example)

Named entity mentions were identified and classified. Furthermore, they were linked to an external and language-independent knowledge base and enriched with stance mark-up. The different steps of the process are summarised as follows:

1. **Identifying mentions:** given a text in natural language, the first step is to identify which words or which list of consecutive words represent in fact a named entity.
2. **Classifying mentions:** once we identified various mentions in a text, it is necessary to try to classify them according to the types defined previously.
3. **Linking mentions:** this next step is necessary to try to avoid ambiguity. The goal here is to link, when possible, the various mentions to a knowledge base (we used Wikidata, but this process is agnostic to the knowledge base and could very well be executed on another). This gives further information on the detected mentions by mapping them to a unique identifier. This process has two advantages. First, if two people have the same name, it provide a way to distinguish them. Then, because the NewsEye collection is multi-lingual, it can efficiently link the mentions in different languages to a single entity.
4. **Detecting the stance:** this process tries to identify what was the stance of a text towards a particular mention of a named entity. At a large scale, this can be useful to identify the point of view of a particular newspaper towards a certain entity, and can allow for interesting analysis such as the evolution of an entity's stance across time and/or the comparative evolution between entities and/or sources (stance detection is described in public deliverable D3.6).

2.2.3 Topics distribution

Topic analysis has been handled by WP4 (more details can be found in public deliverables D4.5, D4.6 and D4.7). The idea behind topic analysis is to classify documents according to their semantic content. Various topic models were trained on the whole collection, language by language. Three latent Dirichlet allocation (LDA) models were trained in French, German and Finnish with 100 topics each, and three dynamic topic models (DTM) were trained for the same languages, with 50 topics and an average of 34 time slices (more details are available in public deliverable D4.5).

This process results in a distribution of topics for most documents (it is meaningless to compute topics distribution for too small articles because of the lack of information and context). Topics are represented by a set of words that often co-occurred and are thus semantically related. Using these distributions, we can classify documents, or identify documents that are similar to one another depending on their semantic content.

3 Search interface

Because researchers are already used to the way digital libraries present information, it was natural to create something similar. Most of digital libraries have a similar way to present information: a search area where users can enter a query, a list of results with detailed information, a set of facets used to filter the results according to various metadata. The interface of the NewsEye platform is presented in Figure 2.

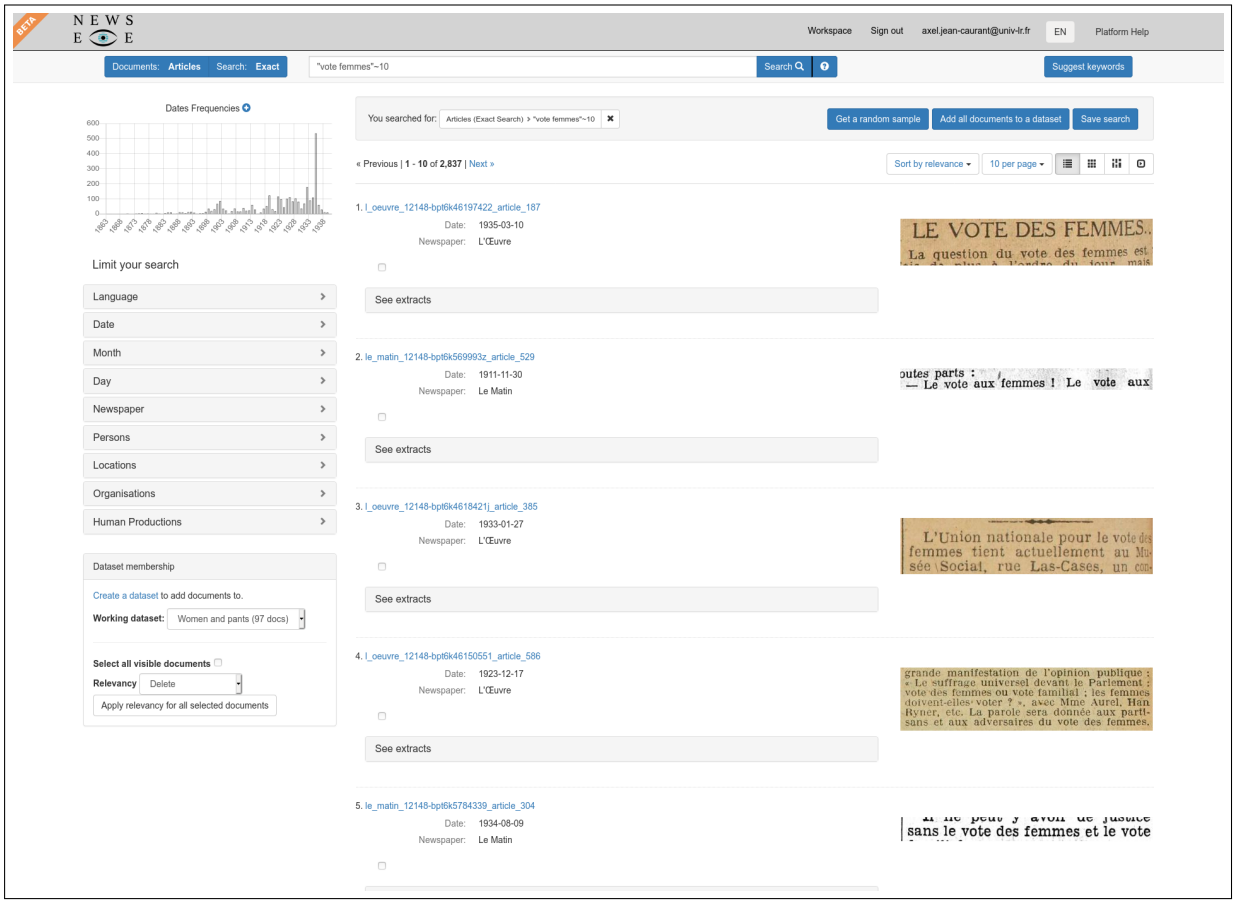

Figure 2: The search interface of the NewsEye platform. It is similar to most digital libraries websites. The search bar is located on the upper part of the page. A list of facets used to filter the results can be selected from the left side of the page. The list of results presents documents relevant towards the search. Here, the query is `"vote femmes" 10`.

3.1 Full text search and metadata filtering

As stated previously, the NewsEye platform offers a way for users to query the available collections to find relevant documents. The queries are composed of keywords and various filters to identify articles or issues relevant to the research question.

3.1.1 Search features

The users can express queries in different ways, thanks to the search bar displayed in the upper part of the NewsEye user platform. The users' first choice they have to make when expressing queries is what kind of documents they are interested in (issues or articles) and how the keywords should be interpreted by the system. Both these parameters can be selected using the two buttons next to the search bar (see Figure 2). Since all the documents in the platform have been indexed in two different ways (see Section 2.1), two types of searches are available:

- **Exact search:** as its title suggests, this type of search will identify documents that contain the keywords exactly the way they were expressed in the search bar.
- **Stemmed search:** in this kind of search, the keywords used in the query are replaced by their stemmed form, i.e. their grammatical root (for example, "presidential" will be transformed into "president"). This kind of search allows for a larger number of results returned, containing words in a form the user was not necessarily aware of. However, it can also create more noise, by returning documents not relevant to the research subject.

The exact and stemmed searches are one of the possible search feature the user can use. More complex search features are nonetheless available, to make a query more precise in order to find documents as relevant as possible.

- **Phrase search:** When you want to query a document for multiple terms appearing next to each other, you can add quotes. For example, the query `"washington dc"` will match documents containing these two terms next to each other, in the same order. The phrase search can be combined with the following search functions.
- **Wildcards:** `'?'` matches a single character. For example, `wom?n` will match documents containing "women" or "woman". `'*'` matches one or several characters. For example, `balti*` will match documents containing "baltic", "baltique" or "baltikum". This tool may or may not work as it is very expensive to run.
- **Fuzzy search:** It is based on the Damereau-Levenshtein distance and allows the querying of syntactically similar terms. The maximum default distance is set to 2. For example, the query `roam~` will match documents containing terms like "roam" and "foam" but not "roast". The distance between "roast" and "roam" is 3 (two deletions and one addition). It is possible to set the maximum distance to 1 using the following syntax : `roam~1`.
- **Proximity search:** It is sometimes useful to query documents for words appearing close to one another. For example, the query `"woman vote"~10` will match documents containing these two terms, separated by 10 words at most. Results of such a search are presented in Figure 2.
- **Term boosting:** You can affect the way the pertinence of documents is computed (and thus the order of the results) by specifying the importance of the terms in the query. For example, the query `president roosevelt^2` will affect the score by giving the term "roosevelt" twice as much importance as the term "president". It is also possible to negatively affect the importance of a term by specifying a number between 0 and 1 (for example `president^0.5 roosevelt` is the equivalent to the previous query).
- **Boolean NOT:** To query for documents that should not contain a particular term or phrase, you can prepend it with the `'-'` symbol. For example, the query `president -roosevelt` will return documents containing the term "president" but not the term "roosevelt".

All these features can be combined to produce complex queries.

3.1.2 Metadata filtering

After a query is entered in the search bar of the platform, users have the possibility to further refine their results using various filters based on the metadata described in Section 2.2. These filters are called "facets" in the context of the user interface (see Figure 2). First, users have the possibility to filter the returned documents by language. Because the collection available on the platform is multi-lingual, it is often the case that some query terms are associated with documents in various language. If this is especially true for searches containing keywords such as person names or place names, some OCR errors can also be responsible for false positives in the list of documents returned after a search.

Then, from the date metadata associated with documents of the collection, three facets are offered to the user. Users can filter results according to a range of dates, using a date picker. It is also possible to select documents that were produced during a particular month of the year (for example get all documents that were published in March of any year), or during a particular day of the week (for example get all documents that were published on a Saturday).

It is also possible to select documents coming from a particular newspaper. This facet can be useful if the research question the user is interested in is link to the specific view of a given newspaper, or to compare how different newspapers treat a particular subject.

Finally, it is possible to filter documents on the fact that they contain named entities mentions that were linked to a knowledge base (see Section 2.2.2).

3.2 Search results page

After a search, users are presented with a list of results relevant to the query expressed in the search bar (see Figure 2). By default, these results are ordered according to a relevancy score, computed by the system. This score depends on the size of documents, and the number of matches with keywords of the query. If the computation of the score is quite complex, we can sum it up by saying that short documents with a lot of matches with query keywords are deemed the most relevant by the system. Users have the possibility to not only select the number of documents displayed by page, but can also modify their sorting order from the default one to an order by publication date instead. For each result in the list, the associated metadata are displayed, along with a snippet image. Also, the keywords of the query can be seen in the context of the document by clicking on the "See extracts" button.

Another information displayed on the search results page is the date histogram located in the upper-left corner of the page. This histogram presents an overview of the distribution of the publication dates of the resulting documents. Such a visualisation can be useful to get a quick overview of the documents returned by the system, allowing users to identify interesting period of times towards their query regarding the documents available in the collection.

To get a quick glimpse on the results returned, users can click on the "Get random sample" button. A pop-up will then appear showing ten documents chosen at random from the results list. This feature is useful in trying to evaluate if the query entered matches the requirements of the search problem to solve.

From this page, users have the possibility to manage the content of their personal datasets. These will

be described in further details in Section 4.1. Finally, clicking on the title of a document or its snippet image opens a dedicated page that is described in the next section.

3.3 Documents show page

The show page allows users to access the detailed image of an article, as well as the complete issue it is coming from. This page is composed of several panels presenting various types of information (see Figure 3).

Figure 3: The show page of individual documents allows users to access individual articles, as well as the entire issue. Some metadata associated with the issue are presented in the upper-right corner of the page. When an article is selected, its textual content is visible on the right side of the page. Named entities of the entire issue or the selected article are displayed underneath.

3.3.1 User interface

First, on the upper-right corner of the page, the users can see the metadata associated with the current document. These include the title of the issue, its publication date, the newspaper it is coming from, but also the total number of pages of the issue.

The main panel of this page is the document viewer. It offers a high resolution view of the issue, along with the location of the articles that were automatically extracted. When an article is clicked on, it is highlighted in the viewer and its textual content generated by the automatic text recognition process is displayed in a panel on the right side of the window.

If the document contains named entities, they are displayed on this page. When an article is selected, the list of named entities is updated to reflect the named entities present in this article. All entities

detected are associated with a particular class (persons, locations, organisations or human productions, see Section 2.2.2). When possible, some of these mentions are linked to an entry in the knowledge base. The associated page is accessible by clicking on the information button. Clicking on an entity reveals the mentions that were extracted from the text of the article or the issue.

Figure 4: After activating the compound mode, the users can click on article parts to add them to a list visible on the right of the page. These article parts are also highlighted in the viewer during this process.

3.3.2 Compound articles

The article separation is a difficult process and the results are not always satisfying depending on the user needs. In order to counteract this and give users more control over the data, the ability to create compound articles has been set up. We can identify at least two problems that can come up with article separation. The first one is a case of under-segmentation, where multiple articles where identified as one. This case is more difficult to process currently, as it would imply to modify the underlying available article segmentation. The second case in that of over-segmentation. In this case, a single article is segmented in multiple parts. The creation of compound articles has been set up to counteract this particular problem. In the document show page, user can activate the compound article creation mode (see Figure 4). Then, once in this mode, the user can select several article parts from the viewer that are then added to a list. Once the selection is finished, users can validate the creation of the compound article by giving it a title (see Figure 5). If an issue contains previously created compound articles, they can be seen on the top-right panel of the page. Clicking on it will highlight its article parts in the viewer and a new panel will appear in the dataset membership section of the page (see Figure 6).

4 Personal workspace

While the search interface is similar to the ones we can find on most digital libraries, the personal workspace is a novelty rarely seen on platforms similar to that of NewsEye. It gathers different elements specific to each user, as well as some generic information about the whole NewsEye collection (see Figure 7).

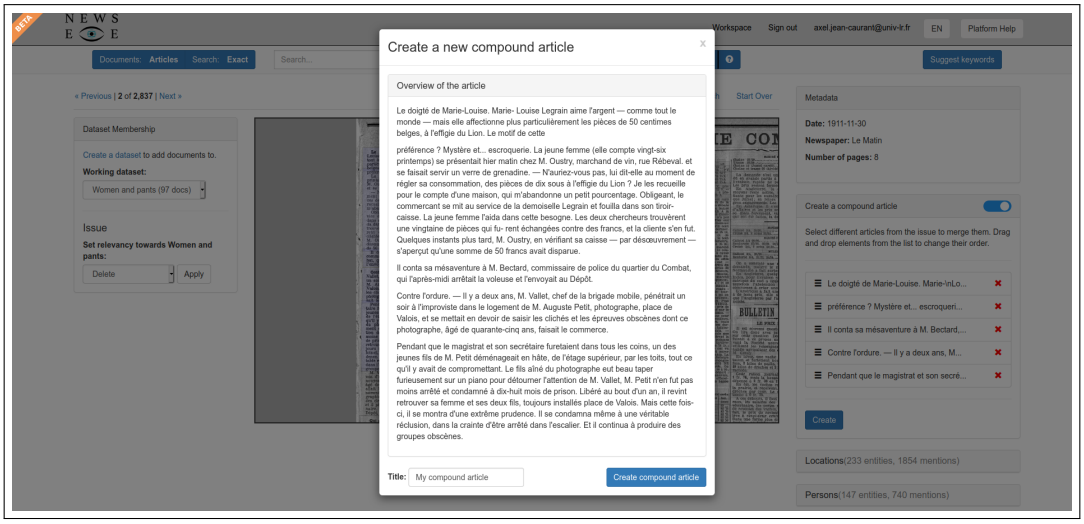

Figure 5: After validating the content of a compound article, users can have a look at the entire text and give a title to their compound article.

Figure 6: Once the compound article is created, it is added to the list of compound articles on the left side of the page. Clicking on it will highlight the associated article parts in the viewer.

4.1 Datasets

Working on a particular research topic, users have the need to find documents relevant to the questions they are asking. These documents can be found after performing various searches in the NewsEye platform. One very important aspect of this process is to be able to save these documents (or a reference to these documents). This can be done in the platform in the form of "datasets". Users can create as many datasets as they need, to gather documents that are meaningful to answer a particular research question, or documents that belong together according to the user's need.

4.1.1 Interface

On the NewsEye platform, a new dataset can be created from the personal workspace (see Figure 8). Users only need to provide a unique name for the dataset (users cannot have two datasets with the same name). Once this is done, documents can easily be added to this dataset from various pages of

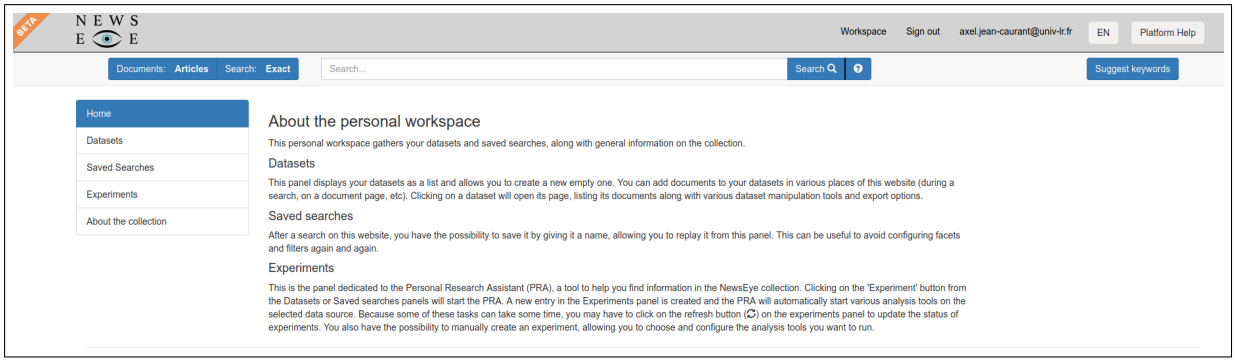

Figure 7: The personal workspace can be accessed by clicking on the link in the upper part of every page.

the platform.

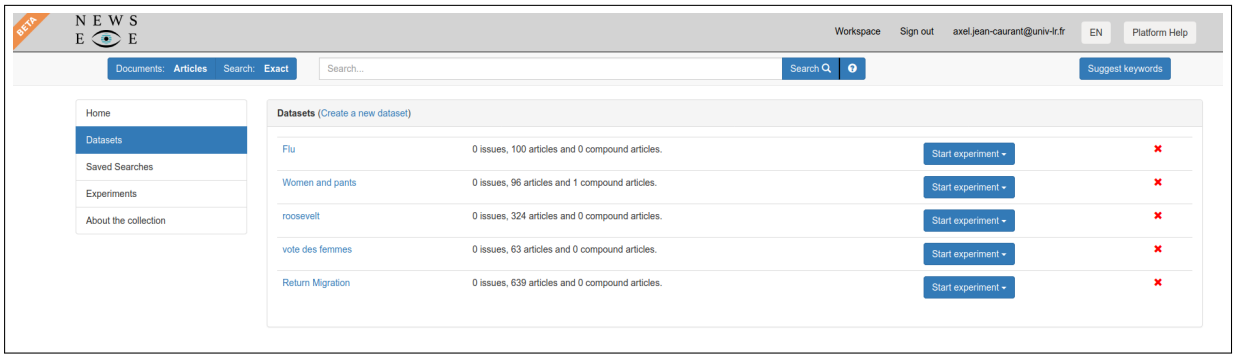

Figure 8: All the datasets created by the user are listed on this page. Users can also create a new empty dataset from there.

On the search engine results page, after a query, users have the possibility to select one or several documents to add to the current dataset they are working on using the "Dataset membership" panel (see Figure 2). They can either individually select documents with a checkbox, or choose to select all visible documents. In this context, "visible documents" refer to the documents displayed on a page of results. To be able to select more documents, users have to modify the parameter that sets the number of results per page. Along with the documents to add, users have to select the relevancy of these documents towards the research question the dataset represents. There are 4 different levels of relevancy: "not relevant", "somewhat relevant", "relevant" and "very relevant". While the "not relevant" option can seem a little out of place, it may actually be required in some particular cases. A document may contain keywords linked to a particular research question without being relevant to it. Having this information could prove very useful in the future, especially with the possibility of training various machine learning models in the context of WP4 and WP5. This is especially interesting for supervised learning, as this will allow models to be trained on positive and negative examples. After a search, users can quickly view if documents are already part of some datasets thanks to a small tag next to the document in the results list.

Documents (issues, articles and compound articles) can also be added to a dataset in the same way from individual document show page (see Figure 3).

4.1.2 Tools

After clicking on a dataset name from the personal workspace (see Figure 8), users are redirected to the dataset show page where they can view documents and access some tools (see Figure 9).

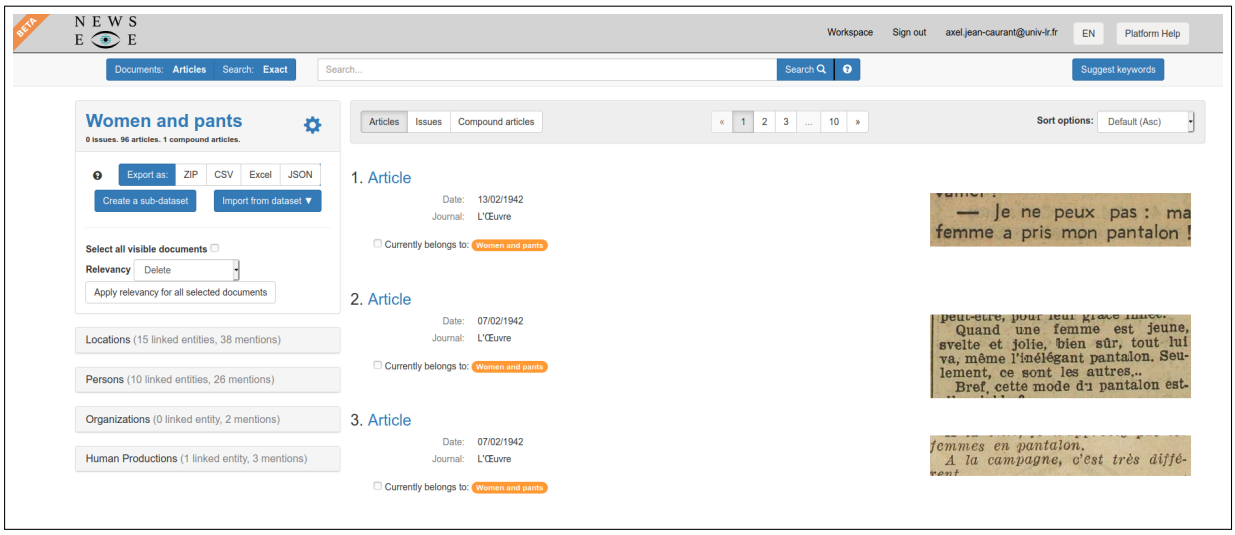

Figure 9: This page gathers information on a particular dataset, allowing users to see its content. From there, users can manage the content of this dataset, view which named entities are represented and export its content.

- **Merging datasets:** the ability to merge datasets together helps users easily build meaningful datasets. One simple use case is to first create various datasets to gather documents relevant to different aspects of a main research question. If needed, users can then gather all these documents in a more general dataset. In the platform interface, this feature is available on each dataset page ith a click on the button "Import from dataset". After the selection of another dataset, a modal window summarising the changes is presented to users. After validation, the current dataset is updated with the documents of another. To actually implement the previous use case, users first have to create a new empty dataset. From the page of this new dataset, users have to select each of the other dataset they want to merge and import documents from it, to the newly created dataset. It is to be noted that the merged datasets are not destroyed automatically, as they may still be useful to analyse separately.
- **Splitting datasets:** if the ability to merge datasets together can support a bottom-up research approach (as in the previous use case), some users may prefer to work in a top-down way. This translates in the fact of users searching for general information about a particular research question, and then trying to identify documents relevant to a specific aspect of this question. This can be done in the platform interface using the "Create a sub-dataset" button. Although this new dataset is meant to include documents from the current dataset, there is no hierarchy relationship between the two. Once this button is clicked on, a modal window is open for users to select which documents they want to add to a new dataset. Documents to add to the new dataset are to be selected by their relevancy within the current dataset. After providing a name for the sub-dataset, it is created on the fly and selected documents are added to it. This newly created dataset then becomes available in the personal workspace as any other. It is to be noted that the documents are not removed from the original dataset. Users will still get the information that these documents are now part of two different datasets thanks to the colored tags alongside documents.

- **Modifying the relevancy of documents in a dataset:** after a dataset has been created and documents added to it, users still have the possibility to modify the relevancy of said documents in the dataset. This can of course be done from the dataset page, by selecting which documents to modify and setting the new relevancy. This can also be done directly from the search engine results page. After a query has been computed, results are presented to users. If some of the documents in the results list are already part of an existing dataset, a small tag will be visible next to these documents with the name of the dataset they are part of (see Figure 9).
- **Exporting datasets:** the NewsEye platform aims not only at helping users find relevant information in a large amount of historical newspapers, but also analyse their content in order to answer precise research questions. However, not every use case can be taken into account. This is the reason why it is important to allow users to use external tools as they see fit. To allow this, it is necessary for them to be able to export their datasets in various formats. Two main use cases have been identified and implemented, one for regular users and the other for more advanced users, comfortable with using parsing tools. First, a ZIP export is available. It gathers the documents of the dataset as a list of files containing the text for each document. The filenames are in the following form: `relevancy_date_id.txt`. Then, datasets can be exported as a JSON or a CSV file. These files contain a bit more information than the previous export. Not only do they contain the text, date and relevancy of documents, they also contain a IIIF link to the original image. It would be technically possible to also export original files like ALTO and/or PageXML which describe the OCR and ATR processes. However, this was not considered to be useful by the DH researchers of the project. This option can still be implemented in the future, is reasons arise. Named entities of a dataset can be exported if the JSON export is used (see Figure 10).


```
{
  "id": "innsbrucker_nachrichten_ibn19181003_article_21",
  "type": "article",
  "language": "de",
  "date": "1918-10-03T00:00:00Z",
  "newspaper_id": "innsbrucker_nachrichten",
  "iiif_url": "https://platform.newseye.eu/iiif/innsbrucker_nachrichten_ibn19181003_page_1/2363,3490,1119,542/!400,200/0/default.jpg",
  "relevancy": 2,
  "text": "KB. Lugano 2. Oktober. Der „Epoca“ zufolge äußerte\neine hochgestellte Persönlichkeit, daß der Mangel an Fleisch.\nMilch und besond
  "named_entities": [
    {
      "mention": "Mailand",
      "indexStart": 441,
      "indexEnd": 448,
      "stance": "neutral",
      "linked_entity_url": "https://www.wikidata.org/wiki/Q490"
    },
    {
      "mention": "Italien",
      "indexStart": 615,
      "indexEnd": 622,
      "stance": "neutral",
      "linked_entity_url": "https://www.wikidata.org/wiki/Q38"
    },
    {
      "mention": "Kalabrien",
      "indexStart": 275,
      "indexEnd": 284,
      "stance": "neutral",
      "linked_entity_url": "https://www.wikidata.org/wiki/Q1458"
    },
    {
      "mention": "Epoca",
      "indexStart": 28,
      "indexEnd": 33,
      "stance": "neutral",
      "linked_entity_url": "https://www.wikidata.org/wiki/Q226610"
    }
  ]
},
```


Figure 10: Sample of a JSON export. Metadata for each article are available, along with the list of named entities it contains.

4.2 Saved searches

Along with the creation of datasets, users may want to save particular searches for later use. This can be done from the search engine results page thanks to a simple button (see Figure 2). Once clicked on,

a modal window opens to ask users to provide a description to the search to save. After validating, this particular search will be available in the personal workspace page. Users can then click on this search to return to the associated search engine results page (see Figure 11).

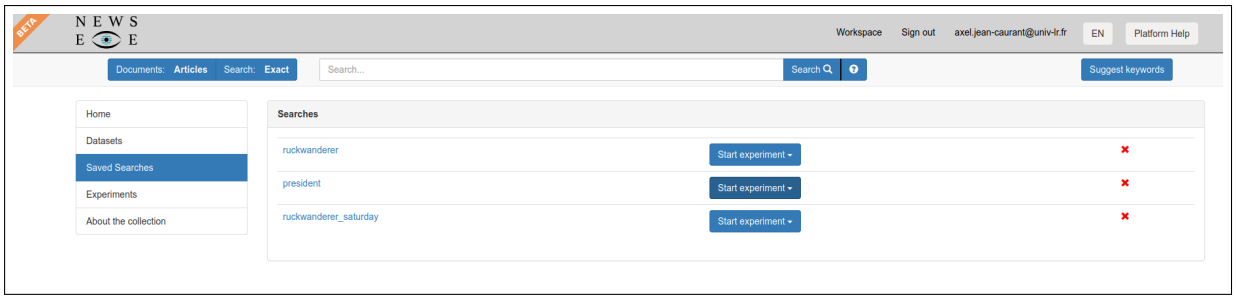

Figure 11: This panel available in the personal workspace shows the searches previously saved by the user. Clicking on the title of a search will replay it and send the user to the associated search results page.

4.3 Personal Research Assistant and Experiments

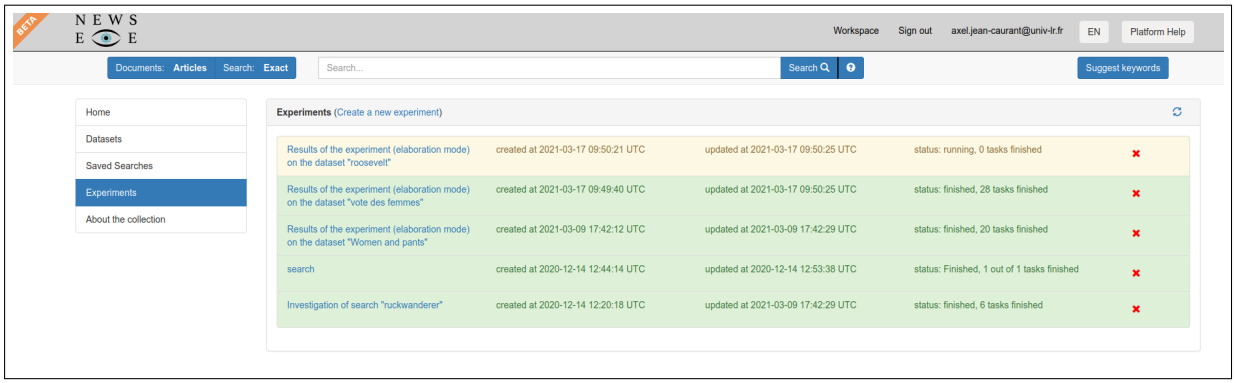

Figure 12: List of the experiments started by the user. Clicking on an experiment title will open its dedicated page. It is also possible to start a manual experiment by clicking on "Create an experiment" in the panel header. A yellow background indicates that no results are available yet, while a green background indicates that the experiments has started or is finished.

Experiments are the main way users can analyse documents in the NewsEye platform. The different analysis tools are created as part of WP5 by the computer science team located in Helsinki (their work is described in public deliverables D5.6, D5.7 and D5.8). The goal of this set of tools, called the Personal Research Assistant, is to offer users a way of executing various processes to get statistical information on a set of documents, get insight on the content of documents and help them identifying new relevant documents. Most of the features available are executed on an external server, managed by the same team. The data exchange between the platform and the tools is handled via a set of APIs. There are two ways of using this Personal Research assistant. First, users can request an automatic analysis, where the assistant identifies automatically a set of analysis tools to be executed. A manual way is also accessible where users can select which tool they want to use. In both cases, created experiments are listed in the "Experiments" panel in the personal workspace, where some details are displayed such as the title of the experiment, the creation date, the last-updated date and the current status (see Figure 12). Because some of the analysis tools can take some time to complete, users may have to update experiments by clicking on the refresh button on the top-right of the panel. When the

Figure 13: After starting an automatic experiment using the investigation capabilities of the Personal Research Assistant, users can access the generated results in this page.

title of an experiment is clicked on, the experiment page is displayed.

This page looks similar, regardless of whether the automatic or manual mode is chosen. A small panel on the top-left side of the window gathers information on the status of the experiment. The title of the current experiment is displayed on top of the page. In the main window, the experiment is represented as a tree, where each node corresponds to a set of documents (blue nodes) or a particular analysis tool (green nodes). When a node is clicked on, some information on it are displayed in a small panel on the left of the window. If this node does not represent a set of document but an analysis tool, its results are displayed below the viewer along an automatically generated report (the generation of these reports is described in public deliverable D5.7).

4.3.1 Automatic investigation

Users have the possibility to start an experiment from a dataset (see Figure 8) or a saved search (see Figure 11). A button is accessible in the personal workspace for each saved search or dataset.

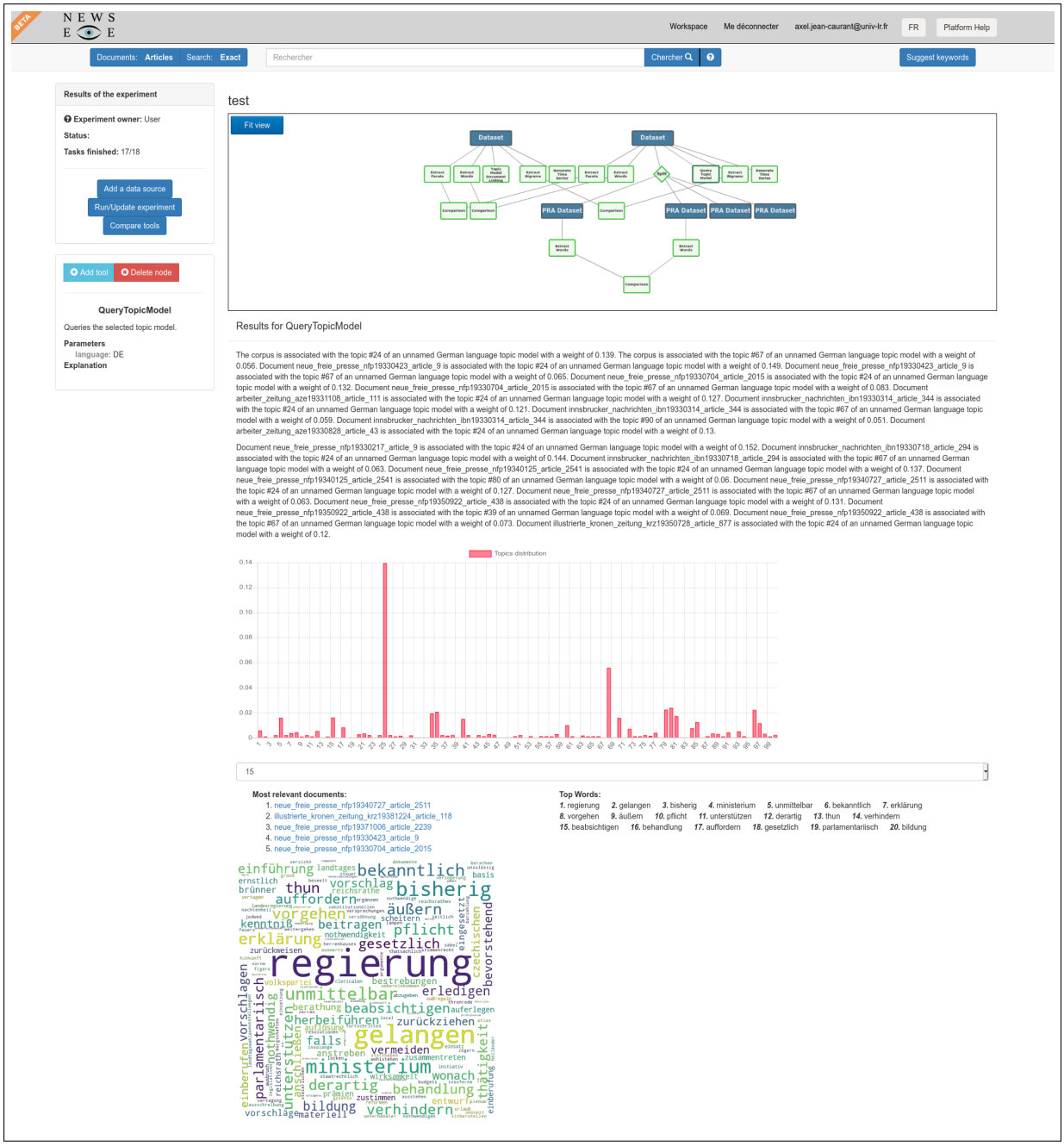

Figure 14: The page of a manual experiment is similar to the automatic experiments, but provides tools to edit it and run it.

There is currently two different ways of using the Personal Research Assistant. The first one is called "Elaboration mode" with the goal to compute various statistics to give users information on a particular dataset. The second mode, "Extension mode", allows users to find new relevant documents that they may have missed. Figure 13 shows an automatic experiment page.

The status box in the top-left corner of the page indicates that this experiment belongs to the Personal Research Assistant. This means that users do not have the possibility to modify the experiment content. Under the title of the experiment, users are presented with an automatically generated report on the findings of the experiment. When a node in the tree is selected, a panel appear on the left of the page with

information about it. If the node is an analysis tool (the green nodes in Figure 13), the parameters of this particular analysis tool are displayed, along with an explanation on why this tool was executed. The results of this particular tool are displayed under the main experiment tree. An automatic generated text is presented, along with various graphs summarizing the results of the tool. Also available is an explanation on why this particular tool was run by the assistant A more exhaustive description of this work can be found in public deliverable D5.8) and an example is provided in Figure 13. If the selected node is a dataset node (blue nodes in Figure 13), two cases are possible; If the dataset is part of the user's collection, a link to access its content is presented. Else, if the dataset was automatically generated by the Personal Research Assistant, users have the possibility to convert it to make it accessible in their own personal workspace.

4.3.2 Manual experiments

To start an experiment from scratch, it must first be created using the "Start a new experiment" button on top of the "Experiments" panel (see Figure 12). Users must give it a name and a new empty experiment is added to the list. While the interface of such a manual experiment is very similar to automatically generated experiment (see previous Section), there are still some differences (see Figure 14), which we will detail below.

The first step after starting a manual experiment is to add one or several data sources to analyse. This can be achieved by clicking on the "Add a data source" button in the status box located in the top-left corner of the page. A pop-up will appear, giving users the choice to add a dataset or a saved search. It is possible to add multiple data sources in the same experiment by repeating this step.

The next step for the user is to start adding various analysis tools. To do so, it is necessary to click on the "Add tool" button after selecting a data source previously added (whether a dataset or a saved search). This opens a pop-up with the available tools that can be run on this particular node (for example extracting bigrams or named entities). Most of the tools take as input directly a data source node while others have to wait for the result of another tool to get started.

Once the tools have been set up, users can click on the "Run/Update experiment" button to execute the tools described in the tree. Some of them may take some time to run and will thus appear in a different color. Green is for tools that have finished running, orange is for tools that are still running (thus requiring the user to click on the same button, until the execution has ended) and red is for failed tasks.

5 Second version of the platform

Most of the work during the extension period of the project regarding the development of the platform focused on creating a more mature product. While the first version of the platform (presented in details in this deliverable) will remain online at the same address (`https://platform.newseye.eu`), the latest version is available at `https://platform2.newseye.eu`.

The possibilities offered by the platform became more clear during the project and a lot of features were added after numerous discussions with the researchers and stakeholders of NewsEye. Features were

thus added on top of one another, which resulted in a messy code base. This was the main motivation for the creation of a new version of the platform.

Furthermore, it appeared during the various tool testing sessions that the personal research assistant (PRA) was not production-ready due to infrastructure reasons. The second version of the platform thus contains an implementation of manual experiments as described in Section 4.3.2. As of the end of the NewsEye project, only a few of the tools are available. More will be added in the future but it was more important to focus on the technical aspects of this feature. To improve user satisfaction, the choice was made to use web sockets as a mean of communication between the server and the user. They are used in multiple places in the platform. Anytime an operation is susceptible to take a bit of time (exporting a dataset or running an experiment for example), its execution is launched asynchronously. When the task is done, the server can send a notification to the user, even if he/she continued browsing the platform, making new searches, etc.

6 Conclusion

The platform is in active development and will continue to be as subsequent new projects come up, in the context of NewsEye's exploitation and sustainability plans. As explained in the previous section, creating and executing experiments manually in the second version of the platform is already possible, and the technical infrastructure is in place and ready to integrate more tools to analyse datasets.

The platform has been developed to showcase the tools and data produced during NewsEye, providing new ways of interacting with newspaper data, which was the goal of the project. Thus, the platform serves as a proof of concept of what could be done in future developments of analysis platforms specific to historical newspapers. Although this platform is not integrated as is in libraries, it provides clear insights on new possible interactions between historical newspapers and their users.

References

[1] Antoine Doucet, Martin Gasteiner, Mark Granroth-Wilding, Max Kaiser, Minna Kaukonen, Roger Labahn, Jean-Philippe Moreux, Guenter Muehlberger, Eva Pfanzelter, Marie-Eve Therenty, Hannu Toivonen, and Mikko Tolonen. "NewsEye: A digital investigator for historical newspapers". In: *15th Annual International Conference of the Alliance of Digital Humanities Organizations, DH 2020*. Ottawa, Canada, July 2020. URL: https://hal.archives-ouvertes.fr/hal-03029072.

[2] Lidia Pivovarova, Axel Jean-Caurant, Jari Avikainen, Khalid Alnajjar, Mark Granroth-Wilding, Leo Leppänen, Elaine Zosa, and Hannu Toivonen. "Personal Research Assistant for Online Exploration of Historical News". In: *Advances in Information Retrieval*. Ed. by Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins. Cham: Springer International Publishing, 2020, pp. 481–485. ISBN: 978-3-030-45442-5.

[3] Axel Jean-Caurant and Antoine Doucet. "Accessing and Investigating Large Collections of Historical Newspapers with the NewsEye Platform". In: *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*. JCDL '20. Virtual Event, China: Association for Computing Machinery, 2020, pp. 531–532. ISBN: 9781450375856. DOI: 10.1145/3383583.3398627. URL: https://doi.org/10.1145/3383583.3398627.