



HAL
open science

Linking multimedia content for efficient news browsing

Rémi Bois, Guillaume Gravier, Eric Jamet, Emmanuel Morin, Maxime Robert, Pascale Sébillot

► **To cite this version:**

Rémi Bois, Guillaume Gravier, Eric Jamet, Emmanuel Morin, Maxime Robert, et al.. Linking multimedia content for efficient news browsing. 2017 ACM International Conference on Multimedia Retrieval (ICMR), Jun 2017, Bucharest, Romania. hal-01522413

HAL Id: hal-01522413

<https://hal.science/hal-01522413>

Submitted on 15 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Linking Multimedia Content for Efficient News Browsing

Rémi Bois
CNRS, IRISA & INRIA Rennes
263 Avenue Général Leclerc
Rennes, France 35042
remi.bois@irisa.fr

Guillaume Gravier
CNRS, IRISA & INRIA Rennes
263 Avenue Général Leclerc
Rennes, France 35042
guillaume.gravier@irisa.fr

Éric Jamet
CRPCC, Université de Rennes 2
Place du recteur Henri Le Moal
Rennes, France 35043
eric.jamet@mshb.fr

Emmanuel Morin
LS2N, Université de Nantes
2 Chemin de la Houssinière
Nantes, France 44300
emmanuel.morin@univ-nantes.fr

Maxime Robert
CRPCC, Université de Rennes 2
Place du recteur Henri Le Moal
Rennes, France 35043
maxime.robert@univ-rennes2.fr

Pascale Sébillot
INSA, IRISA & INRIA Rennes
263 Avenue Général Leclerc
Rennes, France 35042
pascale.sebillot@irisa.fr

ABSTRACT

As the amount of news information available online grows, media professionals are in need of advanced tools to explore the information surrounding specific events before writing their own piece of news, e.g., adding context and insight. While many tools exist to extract information from large datasets, they do not offer an easy way to gain insight from a news collection by browsing, going from article to article and viewing unaltered original content. Such browsing tools require the creation of rich underlying structures such as graph representations. These representations can be further enhanced by typing links that connect nodes, in order to inform the user on the nature of their relation. In this article, we introduce an efficient way to generate links between news items in order to obtain an easily navigable graph, and enrich this graph by automatically typing created links. User evaluations are conducted on real world data in order to assess for the interest of both the graph representation and link typing in a press reviewing task, showing a significant improvement compared to classical search engines.

CCS CONCEPTS

•**Information systems** → **Nearest-neighbor search**; *Recommender systems*; •**Human-centered computing** → *Hypertext / hypermedia*;

KEYWORDS

news graph; user evaluation; hyperlinking

ACM Reference format:

Rémi Bois, Guillaume Gravier, Éric Jamet, Emmanuel Morin, Maxime Robert, and Pascale Sébillot. 2017. Linking Multimedia Content for Efficient News Browsing. In *Proceedings of Identifying and Linking Interesting Content in Large Audiovisual Repositories*, Bucharest, Romania, June 2017 (ICMR'17), 6 pages.
DOI: 10.475/123.4

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ICMR'17, Bucharest, Romania

© 2016 Copyright held by the owner/author(s). 123-4567-24-567/08/06...\$15.00
DOI: 10.475/123.4

1 INTRODUCTION

With content being massively made accessible grows the need to automatically extract information and organize multimedia collections so as to help users search and explore large amounts of content to gain knowledge and insight. Entity extraction and linking, along with topic and event detection, are now widely available to describe content and help search pieces of information. While these techniques are instrumental to content description and search, they are however not sufficient to user-friendly exploration and navigation of a collection to gain insight, e.g., to summarize or to synthesize information. In the absence of a precise search intent, exploration is much more adapted than search.

An efficient way to offer exploration capabilities is to make sense of the relation between pieces of content, with solutions ranging from topic detection, exhibiting relations existing between large sets of items, to hyperlinking, creating connections between individual items in the collection. A large spectrum exists between those two solutions, e.g., clustering, threads, ... But when collections grow, some of these approaches become unreliable. Clusters tend to multiply or become too large and threads lengthen to unmanageable sizes. Hyperlinking on the opposite can be controlled, offering only a small set of suggestions to continue the exploration of the collection. Nevertheless, when using hyperlinking, one can end up in closed loops, where only a small set of items are interconnected with no way to reach other parts of the collection.

In this article, we focus on language-based multimedia hyperlinking in the news domain, connecting elements from a multimedia collection with the goal of enabling easy exploration by analysts (journalists in our case). News data have been extensively studied, due to the relatively large accessibility and interest to both media professionals and general public, however mostly from the search angle. We rather focus on an exploration scenario without precise information need, where one typically has to get a comprehensive view on a topic or event in a limited amount of time. In this context, we consider hyperlinking and put forward the notion of *acceptable* graphs linking news pieces in such a way that users can easily find all relevant information on a topic by exploring the graph.

Departing from standard approaches, e.g., \mathcal{E} -NN graphs, topic threading or event linking, we propose a novel nearest neighbor

graph construction algorithm that creates hyperlinks in a reasonable number to avoid user overload and disorientation, yet ensuring relevance and serendipitous drift. Interestingly, the algorithm requires no threshold, making use of the intrinsic properties of the document representation space. We further propose a typology of links between pieces in the news domain along with rules for automatic link categorization. These two elements, graph construction and link categorization, result in an explorable organization of a large collection of news. The effectiveness of this organization, in particular that of link categorization, is assessed by means of user tests, where journalists were asked to write a synthesis on a particular topic in a limited amount of time.

2 NEWS BROWSING APPROACHES

In this section, we explore the approaches available to organize news collections and discuss how graph representations can help browsing news datasets.

2.1 Standard approaches

News collections structuring for search and exploration has been extensively studied and many solutions coexist, each with their pros and cons. One can distinguish between two main use-cases: the need to know "what has been happening in the last days/hours", and the need to know "how this relates to other news". The first one is typically targeting the general public who wants a quick tour of the main news, while the second one is usually more in line with professionals' needs.

Day to day news structuring answers the need of the general public to have an easy access to information. For this specific need, the clustering approach is most often used, in a Google News fashion. This approach consists in grouping together news articles that discuss the same specific event, and is manageable for small timeframes [14]. Most items in a cluster are very similar, and while some articles may bring more details or talk about a specific aspect of the topic at hand, it is assumed that users will most often read only one of the news pieces. Groups of clusters are sometimes arranged in topical categories to lower the information overload [10]. However, when considering more than a day worth of news, clustering leads to the creation of a large number of small sets of articles that are hard to comprehend.

Professionals such as journalists or press-attachés need a richer structuring of news data. To write their own piece of news or report, they try to gather as much information as possible, and need to go through most articles to unveil new details. The clustering approach described before can be reused for this task, and individual clusters can be organized as threads in order to ease intra-cluster navigation [9]. However, the same pitfalls appear when dealing with large timeframes, as the number of clusters to display becomes overwhelming [18]. Another approach consists in using timelines to represent the chronology of events [13, 19]. This allows to fuse clusters that deal with related events, e.g., the continuation of a story, thus reducing their number [1]. According to user studies, the temporal relation between news items is the most important to media professionals [7], allowing them to reveal the causes and consequences of some events. However, similarly to clustering,

when considering large timeframes, it becomes hard to display the list of news stories that are available in the dataset.

2.2 Hyperlinking and graph representations

Hyperlinking of news pieces, i.e., creating links between two documents within a collection, allow users to directly go from one piece of news to another. By following links (that can be seen as user-independent recommendations), the user is able to navigate in an informed way, choosing his next step among a limited set of links that are related to the news item he is currently viewing [4]. When carefully crafted, those links can provide a highly navigable structure, as evidenced in a number of application domains [16], including news [11].

Structures created by the hyperlinking process can be seen as graphs, in which nodes correspond to documents, and edges are links between document pairs. More formally, a graph G is defined as a set of nodes V and a set of edges E such that $\forall e \in E, e = (v_i, v_j), v_i, v_j \in V^2$. The notion of navigability translates in this case into a set of characteristics for the graph, such as an easy access to most elements of the collection, combined with a limited set of suggestions for each item [16]. Navigable graphs allow for interesting applications, such as connecting the dots between two arbitrary pieces of information [17]. In the absence of pre-defined segments acting as the source of the links—the so-called anchors in [4]—, there are two main algorithms to create graphs that connect documents with hyperlinks: K nearest neighbors (K -NN) and \mathcal{E} nearest neighbors (\mathcal{E} -NN) graphs. Both consist in creating links between related elements, called (nearest) neighbors, in a collection, relying on a content-based distance function that measures how similar two documents are. The neighbor selection criterion is either a fixed number of neighbors K for K -NN, or a distance threshold \mathcal{E} for \mathcal{E} -NN.

In practice, finding the optimal threshold, K or \mathcal{E} , is difficult and requires some annotation to estimate the ratio of irrelevant links, a process that is often complex and subjective [6]. Moreover, graphs created with these methods exhibit some strong limitations in terms of navigability. K -NN graphs do not discriminate between news that are heavily discussed, and that could thus rightfully be linked to many other news pieces, and news that are reported by only a few medias, with few connections to other items. Using the same threshold K for the whole collection thus leads to links that are too few for some news items, or too numerous for others. The use of a distance threshold in \mathcal{E} -NN graphs skirts this issue by creating only relevant links. However, \mathcal{E} -NN graphs tend to create very large hubs [15], with a few nodes being connected to hundreds of others, causing navigation in such structures to be cumbersome. In experiments not reported in this paper, we also observed that combining a fixed number of neighbors and a distance threshold do not alleviate these issues.

3 EXPLORABLE NEWS GRAPH

Nearest neighbors news graphs are attractive because of their practicality for navigation purposes but, as we just highlighted, current nearest neighbor graph construction algorithms fail at providing structures that are easily navigable by humans. In this section, we introduce an explorable news graph structure building on two key

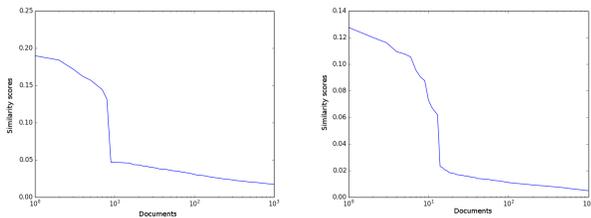


Figure 1: Illustration of similarity drops between close neighbors and far ones on two real-world examples.

steps: an adaptive nearest neighbors algorithm that creates news graphs with good explorability properties and hyperlink categorization according to a typology adapted to the news domain.

3.1 Adaptive nearest neighbors

The main reason why \mathcal{E} -NN and K -NN graphs exhibit bad properties regarding navigability comes from the fact that they rely on a threshold that is the same for every item in the collection. This results in a difficult trade-off between the relevance of the hyperlinks and their number, with high fluctuations of the trade-off between items. We thus propose to adapt the threshold on a per node basis, automatically deciding on the appropriate number of near neighbors by detecting a large gap in the representation space between close neighbors and far ones. Such gaps are known to happen naturally in large collections such as social graphs [3] and are linked to the variations of the density of points in the representation space. For an item i corresponding to node v_i , the gap corresponds to a drop in the similarity between item i and other items sorted in descending order of similarity. This is illustrated in Fig. 1 for two examples taken from a real-world dataset. Only items appearing before the gap are linked to item i .

Revealing those drops depends on two factors: a tailored representation space and an efficient drop detection algorithm. Preliminary experiments on embedded representations, e.g., averaged word2vec [12], revealed that such representations are good at ranking but tend to homogenize the representation space, situating most documents close one from another and thus flattening the similarity drop. We thus use a direct lexical similarity with tf-idf weighting for which large drops do exist and can be accurately detected based on a similarity ratio between consecutive documents after sorting documents in descending order of their similarity to a given node. For node v_i , drop detection is based on $\Delta_i(v_j, v_k) = 1 - \frac{\text{Sim}(v_i, v_k)}{\text{Sim}(v_i, v_j)}$ where $\text{rank}_i(j) = \text{rank}_i(k) + 1$ in the list of documents sorted with respect to their similarities to item i of the collection. The adaptive nearest neighbors (ANN) approach can finally be written as

$$\text{ANN} = \{(v_i, v_j) \mid d(v_i, v_j) < d_i, \forall v_i, v_j \in V^2, v_i \neq v_j\}$$

with $d_i = d(v_i, \text{argmax}(\Delta_i(v_k, v_l)))$, where the maximization is taken over all $v_k \in V, v_l \in V$ s.t. $\text{rank}_i(l) = \text{rank}_i(k) + 1$.

Table 1 reports the characteristics of ANN graphs computed over the News Aggregator dataset¹, which contains ~420k urls to news articles from a large number of publishers. This dataset comes from

Category	#nodes	nodes	degree	P (in %)	R (in %)
Science	16k	93 %	11	73.2	55.8
Business	15k	95.3 %	8	69.7	45.8
Health	14k	94.6 %	13	66.4	56.5
Entertain.	16k	96.5 %	17	74.3	59.2

Table 1: Number of nodes, amount of reachable nodes, median degree, precision and recall for adaptive nearest neighbors depending on the categories.

a five months monitoring of four Google News macro-categories—business, health, science and technology, and entertainment—within which articles on the same story form a cluster [5]. After retrieving all possible documents and filtering out clusters with less than 4 documents, we constructed ANN graphs over each category. For each category, we report the number of nodes, the ratio of nodes included in the largest component of the graph, the median number of links per node, as well as hyperlink recall and precision computed from the story cluster labels (a link is relevant if the two documents are within the same cluster). Note that the use of story-based cluster labels is only indicative of the overall objective of creating an easily navigable graph, where links between stories should appear to enable exploring the collection without searching for many entry points and to favor serendipitous drift. It is however encouraging to see that ANN graphs exhibit large precision values, indicating that few inter-cluster hyperlinks are created, however in sufficient number for most nodes to be part of a single connected component or, in other words, to enable a path between most pairs of nodes (>93% overall). Results not reported within the scope of this paper show that ANN graphs offer much better trade-offs between precision and connectivity than K -NN and \mathcal{E} -NN graphs.

3.2 Link characterization

In spite of the limitation of the number of hyperlinks offered by the ANN approach, there still might be a significant number of hyperlinks per document. This calls for further organization of the collection to enhance navigation in the ANN graph and help users find their way by selecting appropriate hyperlinks to follow on. To this end, we propose to characterize links according to a typology arising from the needs in relation with large-scale news collection exploration.

News data depend a lot on chronology, which resulted in many approaches organizing collections as timelines so as to be able to follow the evolution of specific stories. The temporal relation is clearly the most important type of relations according to media professionals [7]. But it is insufficient alone, in particular when exploring large news datasets that include articles with very similar content from different newswires that tends to clutter timelines. Extending temporal relations, we designed a typology consisting of 7 types of oriented links [2] defined as follows:

Near duplicate identifies a link between two nodes discussing the same event, where the target node provides little to no additional information compared to the source node.

Anterior/Posterior indicates a target node reporting on an event related to the source that occurred before (resp. after) the event of the source node.

¹www.archive.ics.uci.edu/ml/datasets.html

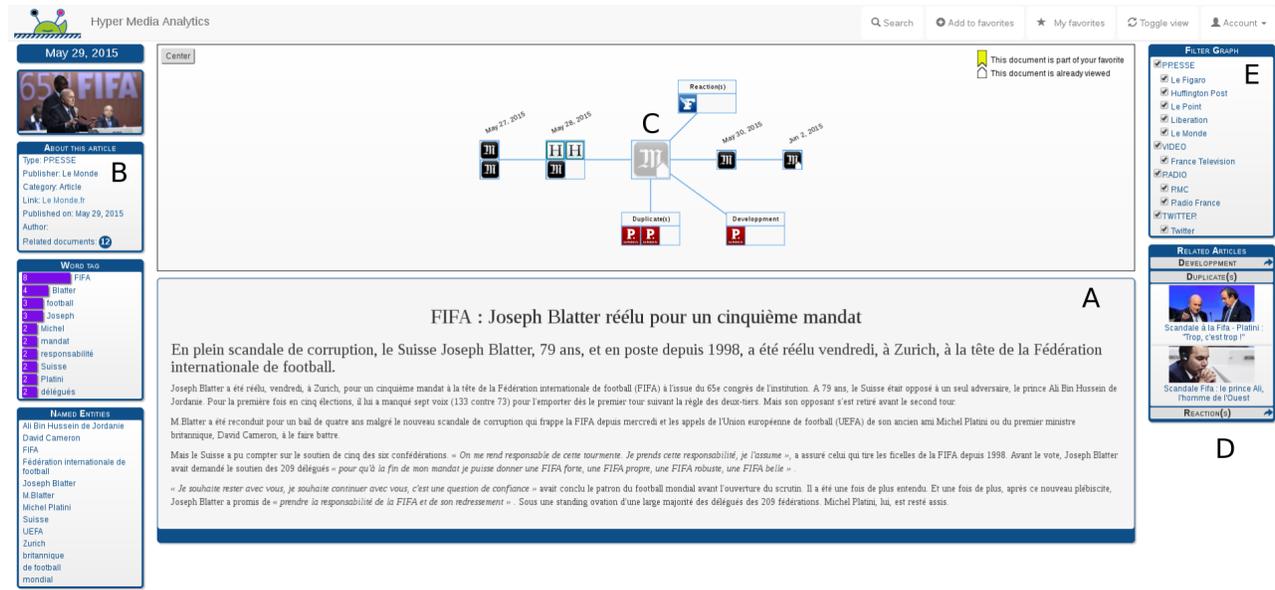


Figure 2: The LIMAH news exploration and analytics interface

Summary/Development corresponds to a link providing a subset (resp. superset) of information with respect to the source. **Reacts/In reaction** to designates a reference (resp. followup) to another news piece in the collection.

These types of links are dual, e.g., a link with the *anterior* type between documents A and B necessarily coexists with an inverse link *posterior* between B and A. The anterior/posterior type, which is the most commonly found, allows to keep a sense of the chronology of news stories, and allows to decide whether one wants to know about past or future events. The near duplicate type allows a user to ignore some news pieces if he decides to, while still making them available. This is not only a way to discard articles, but also a way for professionals to decide which source of information they prefer when different newswires published similar information. A global filtering allowing only a few sources would not solve this issue as journalists reported that their preferred source changed according to the topic, e.g., changing source when going from the political aspects of a news story to its financial implications. The summary/development type serves a similar purpose and is a way to get the gist of an information, or on the opposite to further read about the very same event. Summaries often report short news from national press agencies and are only a few lines long. Developments are obviously longer and add context. They might correspond to editorials in which journalists comment the news in depth and often provide a variety of links to the various aspects of a story. Finally, the reaction links provide a specific type of followup, either by quoting a source, or by reporting the reaction of people of interest to a specific news piece, e.g., a politician answering an opponent's declaration. We chose to use this larger notion of reaction rather than to divide it into citation and response to avoid defining too many types whose semantics would be difficult to grasp.

Automatically categorizing links established within the ANN graph relies on a set of handcrafted rules. Near duplicates are detected first based on a cosine similarity over tf-idf weighted terms. Summaries and developments are then detected by comparing the length of document pairs. We finally assign the reaction type by detecting cue phrases such as "réagit" (reacted) or "répondu" (answered). Remaining links are considered as temporal relations and given the anterior/posterior type depending on publication dates.

4 EXPERIMENTS

Adaptive nearest neighbors news graphs with typed links offer significant explorability features whose benefit we evaluated by means of user studies, comparing search-only approaches and ANN graphs with and without link categorization on an information gathering task. We first describe the dataset and interface before discussing experiments and results.

4.1 Dataset and interface

Documents were extracted over a 3 week period (May 20–Jun 8, 2015) from a number of French newswire websites and include press articles, videos, and radio podcasts. Some of the topics discussed during the time the corpus was gathered had worldwide repercussions, such as the FIFA scandal or Lybian refugees crossing the Mediterranean sea. Others were dealing with French politics with the renaming of one of the main French political parties, or with local news. Podcasts and videos underwent speech transcription so as to enable indexing and segmentation. To deal with possibly long audio or video recordings, topic segmentation based on automatic transcripts [8] was used, each segment being treated as a document per se. In total, the resulting collection contains 4,966 news articles, 1,556 radio segments and 290 video segments.

We ran hyperlinking on the whole collection with adaptive nearest neighbor graphs and link categorization, creating 17,468 links

in total: 10,980 temporal, 3,878 quasi-duplicates, 725 reactions, and 575 summaries/developments. Most links unsurprisingly depict temporal relations, leading us to primarily adopt a linear chronological view of the temporal relations related to a source document. Other types of relations appear either above or under the timeline, as illustrated in Fig. 2. Note that using a linear chronological view is only possible because adaptive nearest neighbors strongly limits the number of hyperlinks departing from the source. Furthermore, link categorization enables removing from the timeline links of a different nature, facilitating visualization and comprehension.

The starting point of the end-user interface, called LIMAH for Linking Media in Acceptable Hypergraphs, is a full-fledged search bar using keywords. Search classically returns a list of documents ranked by relevance, from which the user can choose an entry point for navigation. To facilitate the choice of a good entry point, in addition to a thumbnail of each relevant document, we display the number of links departing from each document, assuming that a document with a significant number of connections is deemed to be a better starting point for exploration than one with very few. Selecting an entry point brings the user to the content visualization and navigation part of the interface, illustrated in Fig. 2. In this view, the user can initially see the entry point document itself (A) and the links that departs from it. In addition to the original content, meta-data and keywords are displayed (B), as both were judged crucial in the preliminary usefulness studies. Links appear in one of two ways. The graph view (C) quickly shows how related documents appear on a navigable timeline, facilitating the comprehension of the development of a story. Users can navigate the timeline: a mouse-over on a node highlights the keywords in common with the entry point document; a click on a node enables viewing the content in zones A and C. Note that contrary to what is usually done in graph-based navigation interfaces, the entry point does not change while navigating and the graph displayed remains the graph depicting the documents related to the current entry point (indicated as a square on the graph). This avoids user disorientation, as it is difficult for users to apprehend the underlying graph structure if the subgraph displayed changes at every click. To enable further exploration, a double click on a node defines the node as the new entry point and changes the graph and recommendations displayed. For convenience, on the right side (D), hyperlinks are also provided as a list of recommendations organized by link types, omitting chronological links (i.e., anterior and posterior developments of the story) that only appear on the timeline. This recommendation view follows a standard web setting and is deemed as easier to apprehend than the graph view, though not as rich. At any time, filters listed in the top right section (E) allow selecting specific sources and a new entry point can be found from the search bar.

4.2 Experimental protocol

In order to evaluate the interest of the graph structure and link typing, we compare three versions of the LIMAH interface. Version 1 only provides the search engine, allowing for comparison with today's usage and with a technology that users are very familiar with. In this case, areas C, D, and E in Fig. 2 are hidden. Version 2 adds the recommendation and graph structure but converts all link types to either anterior or posterior, organizing data in a linear

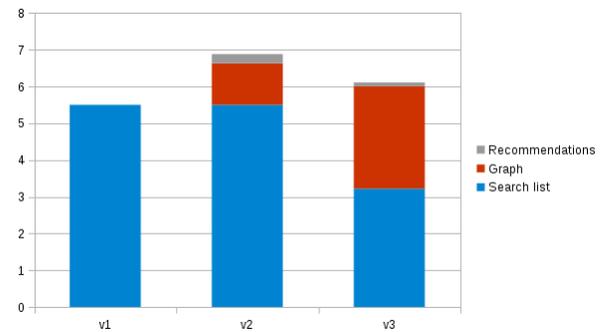


Figure 3: Number and origin of the articles viewed for the 3 versions of the LIMAH interface.

fashion. Recommendations in zone D are thus uncategorized and every link is shown on a timeline. Version 3 corresponds to the whole LIMAH interface, as presented before.

The study involved 25 journalism students in their 3rd, 4th, and 5th year of studies, split in three test pools: 8 users for versions 1 and 2 of the interface, and 9 of them for version 3. The user test involved a pre questionnaire, an information gathering task, a post questionnaire, and a final open discussion in which users could provide feedback on their use of the tool. Users were shown a short video explaining how to use the interface, and received no additional support during tests.

The information gathering task consisted in writing a synthesis about a particular subject in a limited amount of time, using the LIMAH interface to find as much relevant information on the topic as possible. The topic chosen was Solar Impulse 2, a solar-powered aircraft that circumnavigated the globe from March 2015 to July 2016. Bad weather conditions necessitating the plane to land and consequences of this unexpected halt are reported in 17 articles in the dataset, representing a total of 68 distinct information pieces over a long timespan. As the dataset comes from a large set of newswires, some pieces of information are repeated, while others are mentioned by only one or two sources. Users had to complete this task in 20 minutes, a time long enough to fully read a few articles, but short enough to forbid reading totally most of them.

4.3 Results

Fig. 3 first shows the average number of articles viewed by each user pool, as computed from the navigation logs, along with what feature of the interface was used to select an article (search, graph or recommendation). Clearly, the graph representation available in versions 2 and 3 allowed for more documents to be viewed when preparing the synthesis. Interestingly, comparing the number of articles viewed by users of versions 2 and 3 shows that typing links encouraged the use of the graph representation. For version 3, users clearly preferred navigating using the graph representation rather than the recommendation list, demonstrating the interest of this mean of visualization with a controlled-size neighborhood.

Apart from the number of articles considered for the synthesis, a measure of its exhaustiveness is also reported in Fig. 4. Exhaustiveness was measured by coding each synthesis according to the

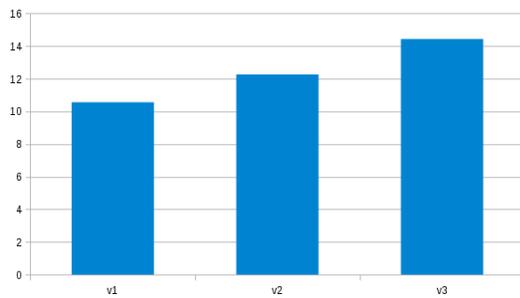


Figure 4: Knowledge extracted from the dataset depending on the version of the LIMAH interface.

proportion of the 68 information pieces contained in the synthesis. Results clearly show that users of version 3 gathered significantly more information pieces than their counterparts, even though they viewed less documents than users of version 2. A detailed analysis of the information gathered by users revealed that versions 2 and 3 lead them to find a greater number of rare pieces of information that were available in less than 4 news items, as well as a greater number of important information, than with version 1, the difference being larger for version 3.

During the open discussion following the tests, users from version 3 were mostly positive about their experience with the tool, calling it “useful”, with a “good accessibility”, and an “interesting take on recommendation”. A few users mentioned a difficulty to handle the back and forth between the graph representation and the search interface. They also had trouble drawing the line between articles relevant to the topic and articles that were not.

5 CONCLUSION

Appropriate graph representations of news articles can help professionals gather information more efficiently, as evidenced by the study presented in this paper. In particular, we experimentally demonstrated that categorizing automatically hyperlinks established between articles further improves the amount and quality of the information retrieved while exploring to gain insight on a particular topic. We also proposed an adaptive nearest neighbors algorithm that was shown to offer a better trade-off between relevance of the links and their number than standard nearest neighbors graph construction algorithms. The latter were found inadequate for navigation in preliminary studies not reported in the paper. The typology of links in the news domain is finally another contribution. While users did not comment the number of categories that were offered in the LIMAH interface, probably indicating that 7 link types was not an overwhelming amount, further studies on the ideal number of types would be of interest, in order to either increase this count, further refining the choices offered to users, or decrease this count, helping to alleviate the information overload. Automatic link categorization, currently based on handcrafted decision trees, was found accurate but could most likely be improved by either enriching the set of rules or using learning algorithms. Finally, the LIMAH interface could be improved by enriching the keywords and named entities currently extracted from each article. Users declared that linking those entities to external resources, such as

the institutions’ open data, would allow them to verify the relayed information. Interestingly, this can be seen as extending the graph to include links to data and knowledge sources rather than limiting hyperlinks to articles.

6 ACKNOWLEDGMENTS

Work funded by the CominLabs excellence laboratory, financed by the National Research Agency under reference ANR-10-LABX-07-01. We are very grateful to Arnaud Touboulic for his key contribution in the design and implementation of the LIMAH interface.

REFERENCES

- [1] Amr Ahmed, Qirong Ho, Jacob Eisenstein, Eric Xing, Alexander J Smola, and Choon Hui Teo. 2011. Unified analysis of streaming news. In *20th Int. Conf. on World Wide Web*. 267–276.
- [2] Rémi Bois, Guillaume Gravier, Pascale Sébillot, and Emmanuel Morin. 2015. Vers une typologie de liens entre contenus journalistiques. In *22e Conférence Traitement Automatique des Langues Naturelles*. 515–521.
- [3] Maximilien Danisch, Jean-Loup Guillaume, and Bénédicte Le Grand. 2013. Towards multi-ego-centred communities: A node similarity approach. *Int. Journal of Web Based Communities* 9, 3 (2013), 299–322.
- [4] Maria Eskevich, Gareth J.F. Jones, Shu Chen, Robin Aly, Roeland Ordelman, Danish Nadeem, Camille Guinaudeau, Guillaume Gravier, Pascale Sébillot, Tom De Nies, Pedro Debevere, Rik Van de Walle, Petra Galušckov, Pavel Pecina, and Martha Larson. 2013. Multimedia information seeking through search and hyperlinking. In *ACM Int. Conf. on Multimedia Retrieval*. 287–294.
- [5] Fabio Gasparrini. 2016. Modeling user interests from web browsing activities. *Data Mining and Knowledge Discovery* (2016), 1–46.
- [6] Mouzhi Ge, Carla Delgado-Battenfeld, and Dietmar Jannach. 2010. Beyond accuracy: evaluating recommender systems by coverage and serendipity. In *4th Conf. on Recommender Systems*. 257–260.
- [7] Guillaume Gravier, Martin Ragot, Laurent Amsaleg, Rémi Bois, Grégoire Jadi, Éric Jamet, Laura Monceaux, and Pascale Sébillot. 2016. Shaping-up multimedia analytics: Needs and expectations of media professionals. In *22nd MMM Conference, Perspectives on Multimedia Analytics*. 303–314.
- [8] Camille Guinaudeau, Guillaume Gravier, and Pascale Sébillot. 2012. Enhancing lexical cohesion measure with confidence measures, semantic relations and language model interpolation for multimedia spoken content topic segmentation. *Computer Speech and Language* 26, 2 (2012), 90–104.
- [9] Ichiro Ide, Hiroshi Mo, Norio Katayama, and Shinfichi Satoh. 2004. Topic threading for structuring a large-scale news video archive. In *Int. Conf. on Image and Video Retrieval*. 123–131.
- [10] G. Krishnalal, S. Babu Rengarajan, and K.G. Srinivasagan. 2010. A new text mining approach based on HMM-SVM for web news classification. *International Journal of Computer Applications* 1, 19 (2010), 98–104.
- [11] Hyowon Lee, Alan F. Smeaton, Colin O’Toole, Noel Murphy, Seán Marlow, and Noel E. O’Connor. 2000. The Fischlár digital video recording, analysis, and browsing system. In *Int. Conf. on Content-based Multimedia Information Access*. 1390–1399.
- [12] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*. 3111–3119.
- [13] Philippe Muller and Xavier Tannier. 2004. Annotating and measuring temporal relations in texts. In *20th Int. Conf. on Computational Linguistics*. 50–56.
- [14] Dragomir R Radev, Sasha Blair-Goldensohn, Zhu Zhang, and Revathi Sundara Raghavan. 2001. Newsinessence: A system for domain-independent, real-time news clustering and multi-document summarization. In *1st Int. Conf. on Human Language Technology Research*. 1–4.
- [15] Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research* 11, Sep (2010), 2487–2531.
- [16] Klaus Seyerlehner, Peter Knees, Dominik Schnitzer, and Gerhard Widmer. 2009. Browsing music recommendation networks. In *10th Int. Society for Music Information Retrieval Conf.* 129–134.
- [17] Dafna Shahaf and Carlos Guestrin. 2010. Connecting the dots between news articles. In *16th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. 623–632.
- [18] Amanda Sturgill, Ryan Pierce, and Yiliu Wang. 2010. Online news websites: How much content do young adults want. *Journal of Magazine & New Media Research* 11, 2 (2010), 1–18.
- [19] Russell Swan and James Allan. 2000. Automatic generation of overview timelines. In *23rd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*. 49–56.