Generating “Visual Clouds” from Multiplex Networks for TV News Archive Query Visualization

Haolin Ren*†, Benjamin Renoust‡, Marie-Luce Viaud†, Guy Melançon* and Shin’ichi Satoh§
*University of Bordeaux LaBRI CNRS UMR 4800, Bordeaux, France
†National Audiovisual Institute (INA), Paris, France
‡Osaka University, Institute for Datability Science (IDS), Osaka, Japan
§National Institute of Informatics (NII), Tokyo, Japan

***Abstract*—Advances in multimedia analysis, enables access and indexing of huge and rich video content. However, delivering this heterogeneous content remains a challenge. We proposes the use of multiplex networks to combine both textual and visual semantic cues extracted from TV news videos for interactive search refinement on heterogeneous data. After preprocessing, the indexed videos can be queried, for which result space is summarized in a *visual cloud* helping query refinement with contextual information by combining visual and textual cues. We leverage on properties of multiplex networks to extract a hierarchy of layers, which not only contextualizes results, but also guides the construction of our visual cloud, and its interactions.**

I. Introduction

The size of digital news archives makes it necessary for media studies to rely on automatic processing for quantitative analysis, not limited to their textual content. Visual analytics of multimedia data proposes then to extract high-level representations and support high-end analysis of multimedia concepts (not limited to text). Advance in computer vision now allows the extraction of visual semantic concepts, which can be used in turn to index video documents in an archive [1]. Querying and retrieving relevant information still remains a difficult task, one with a relatively high cognitive cost for users, especially considering semantic ambiguity. This has stimulated information visualization to support the strategies adopted by users to find their way in the information space formed by results. In particular, tag clouds can give an overview of this semantic space and support exploratory tasks [2].

With news videos, the heterogeneous combination of both textual and visual concepts presents an interesting challenge. Modeling query results as multiplex document networks – each semantic concept being associated to a layer [3] – can turn this heterogeneity as an advantage and derive a Layer Interaction Network (hereafter *LIN*), which offers advance interaction and coordination [4].

Our contribution is a multimedia analytics system that takes advantage of this modeling to support exploration of a TV news archive. We combine textual and visual concepts extracted from news segments with a multiplex network.We derive the *LIN* and extract hierarchical relationships by optimizing group cohesion through entanglement measures [3]. From this hierarchy, we construct a “visual cloud” that brings contextualization to the search results. The visual cloud is integrated in an interactive visual analytics system that supports diverse search tasks [5].

After reviewing the literature in the next section, we describe our data and preprocessing in Section III. We introduce the multiplex abstraction in Section IV then detail our hierarchical clustering in Section V, and visual cloud generation in VI. We discuss our results in Section VII before concluding.

II. Related Work

In the search results analysis process, we are interested in the last two steps: clustering/labeling, and visualization [6]. Beyond traditional topic modeling, graph based methods [7], [8] are proposed without integrating multimedia indexing or visualization. Our work is closer to the work of Scaiella *et al.* [7], as we also create a hierarchical semantic grouping with a graph. Their graph is very close to the work inspiring our own [3], for which group cohesion is measured by association of topics based on multiplex network entanglement.

In turn, visualization addresses the display of search engine results by trying to end with the *list-of-snippets* representation and organize results spatially [9], or even by treemaps [10], [11]. Multiplex graphs have also been used [4]. While they are good for coordination and contextualization, they don’t provide space efficiency. In contrast, we propose an integrated methodology for multimedia data, with exploration, analysis, interaction and search refinement.

Tag cloud provide a good overview of the semantic space [2] and tag placement has been improved to convey semantic proximity [12], [13], [14]. However, none of these approach are adapted to heterogeneous data or multiplex networks, which would enable coordinated interactions [3]. We used the fastest tag cloud method, Wordle [15], which can also be easily constrained and coordinated (see Section VI).

Visual Analytics also proposed to explore topical analysis and news events [16], [17], [18]. OTMedia [1] explored the relationship across different sources of information. The NHK News 7 dataset has also been tackled by diverse works: one proposed linked videos [19], another, 3D timelines of clustered data [20]; summary news videos are generated in [21], and political networks explored in [22]. We combined preprocessing of the last two works, while proposing a novel semantic analysis and interactions.

978-1-5386-7021-7/18/$31.00 ©2018 IEEE

III. Data and Preprocessing

Our data consists in a 12-year archive of daily Japanese public broadcast NHK News 7 [22], from 2001 until 2013. Each program is 30 minutes long with synchronized closed captions, about 6 months of 24/7 viewing in total. A program is composed of different *news segments*. As provided by Ide [23], we obtain the segments using a sliding window of topic distribution (Fig. 1, top). The extracted topics being very noisy, we further extract textual semantic information, for each segment, with a keyword extractor trained for news documents (including named entities) [24]. The Japanese keywords are translated to English with Bing Translator (microsoft.com/translator).

We use the face detection and tracking proposed in [22] (Fig. 1). Faces instances are detected in each frame [25], then regrouped with point tracking [26] creating *face-tracks*, sampled with k-faces [27], and represented using the average of its 128-dim OpenFace embedding vectors [28]. Face-tracks are clustered using GreedyRSC [29]. About 3,000 clusters are manually annotated, resulting in over 15,000 face-tracks of 139 public figures. We index each video segment with date-time of broadcast, keywords, and face-tracks. A query can be placed upon these criteria, returning a subset of news segments, including their associated semantic concepts, *i.e.* keywords and detected faces.

IV. Multiplex Abstraction

A. Layer Interaction Network

Our input are the search results, which news segments associated with their bag of features (*i.e.* semantic concepts). "Graph of topics" [7] (or *LIN*[3]) are used to group results of a web search query. As pointed out, *"a graph of topics provides a contextualization of snippets"* [7], and enables us to measure group cohesion through multiplex *entanglement* [3].

Let S be the set of results returned by a query (see Fig. 1 (a)). Each segment in the results $s \in S$ is indexed by a set of concepts $t \in T$. We first consider a graph $G = (V, E)$ connecting segments to concepts. Two segments s, s' may share concepts t, t', ...We build a graph where nodes correspond to segments $s, s' \in V$ and edges $e = (s, s') \in E$ are created if s and s' share at least one concept, e is *labeled* by the concepts $(t, t', \dots)$ shared by s and s' (see Fig. 1(b)). This graph corresponds to a multiplex graph [3] $G' = (V, E')$ in which $E' = \bigcup_{t \in T} E_t$, *i.e.* each concept t forms a layer.

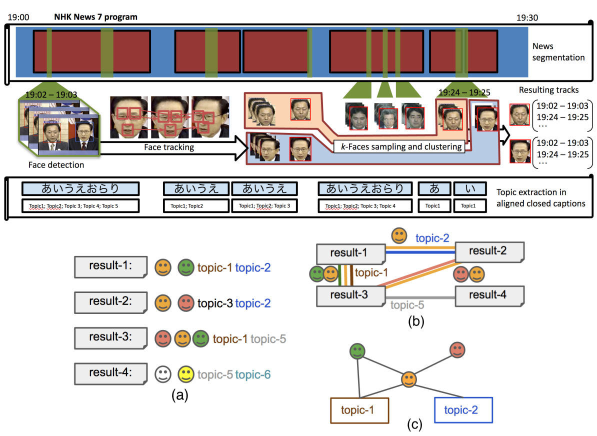

Fig. 1. (Top) Indexing each video segment: closed captions-based segmentation, face tracking, and keyword extraction. (Bottom) Abstraction of the search results. (a) indexed video segments. (b) the multiplex network of results. (c) the associated *LIN*.

This formalism allows us to derive the *LIN*[3] $G_T(T, F)$, together with entanglement measures. The nodes of G_T are thus concepts $t \in T$, connected by an edge $f \in F$ if they overlap in G' (hence index at least two distinct segments). An edge f is weighted by $n_{tt'}$ the size of $|E_t \cap E_{t'}|$ which is the number of edges in G labeled by both t and t'. Following the same definition, $n_{tt} = |E_t|$ is the number of edges of G labeled by a concept t. Fig. 1(c) is a toy example of a LIN.

From there, Renoust *et al.* [3] compute an entanglement index γ_t for each concept: $\gamma_t.\lambda = \sum_{t' \in T} \frac{n_{tt'}}{n_t} \gamma'_t$. It measures how much concept t is entangled with other concepts in the network G'. In other words, it measures the share of concept t in mixing with other concepts that brings results close one another. The more the concepts interact together, the more the group of results will be cohesive, and the more $\gamma_t \forall t \in T$ values will be similar. This is captured by the *entanglement homogeneity*[3] $\mathcal{H}$ which is then defined as the cosine similarity $\mathcal{H} = \frac{<1_T, \gamma>}{|1_T||\gamma|} \in [0, 1]$.

B. Concept and Group Similarity

We now extend the work of Renoust *et al.* [3] by introducing a notion of similarity between concepts and groups of concepts: the *Co-occurrence Possibility* (CP). In G_T, concepts are connected when they co-occur through at least two distinct segments. The CP quantifies this association such as the higher the CP, the closer two concepts can be considered.

CP of two connected concepts t and t' in G_T is defined as follows:

$$CP\left(t, t'\right) = \frac{n_{tt'}}{n_{t't'}} \cdot \frac{n_{tt'}}{n_{tt}} \in [0, 1] \tag{1}$$

With n_{tt} and $n_{tt'}$ the number of (co)occurrences of the concept layers t,t' in the multiplex network. The higher the value of CP, the closer the two corresponding concepts will be. CP is maximized ($CP\left(t, t'\right) = 1$) if and only if two concepts always appear together within results.

When t and t' are disconnected in G_T, we define a dissimilarity as follows:

$$CP\left(t, t'\right) = -\frac{n_{tt} + n_{t't'}}{|E|} \in [-1, 0] \tag{2}$$

With $|E|$ the number of connected pairs of nodes in the multiplex network. The CP value is then negative, with minimal value $CP\left(t, t'\right) = -1$ when the two concepts t and t' cover all the links between segments but never co-occur together, hence representing two separated topics.

Following the same idea, the CP between a concept t and a group of concepts S is formulated as:

$$\sum_{t \in S} CP\left(t, S\right) = \sum_{t' \in S} CP\left(t, t'\right) \text{ , } t \neq t' \tag{3}$$

Finally, the CP between two groups of concepts S and S' ($S \neq S'$) of concepts is then defined as:

$$CP\left(S, S'\right) = \sum_{t \in S} CP\left(t, S'\right) \tag{4}$$

We may verify that $CP(S,S') = CP(S',S)$. This formulation of similarity could be seen as analogous to forces of a force directed layout algorithm[30]. Similar concepts should tend to "attract" each other, while dissimilar concepts will "repulse" one another, placing them far apart: high negative avoid unconnected concepts to be assigned to the same group.

V. LAYER HIERARCHIZATION

In Detangler [4], the *LIN* captures the semantic context with *"a sense of hierarchy"*. However, the network drawn "as is" tends to quickly become a fur-ball as the number of concepts increases. Hierarchical relationships can provide a meaningful navigational mechanism by organizing information into a small number of hierarchical clusters [31]. To extract this hierarchy, our strategy is to identify concept and segment subsets with optimal cohesion (entanglement). Since it is an NP-hard combinatorial optimization problem, Renoust *et al.* [3] did not offer any solution. Keeping in mind that we have a subgraph of segments corresponding to each (group of) concept(s), we propose a heuristic solution to maximize cohesion in groups of segments.

This algorithm aims at building trees of concepts from the *LIN* G_T. Each concept starts labeled with its own group l_t and we aggregate concepts from the topology of G_T such as two concepts will be connected if they are linked by an edge in G_T. Orientation of links are decided from $max(\gamma_t, \gamma_{t'})$ (rooting on the most entangled concepts). The algorithm runs in three steps (Fig. 2). The first step aims at initializing parent-children links. The second step associates all children nodes to their closest concepts (or group of concepts). The last step iteratively assembles groups of concepts.

Step 1: Generate "parent-children" pairs for all concepts. In this step (Fig. 2(a)), each concept t_1 , that is not a parent already, will be associated to a concept t_2 in its neighborhood $t_2 \in \mathcal{N}_{G_T}(t_1)$ such as $CP(t_1, t_2)$ is maximal. All the pairs form now C groups, each node will then be labeled with its parent's group l_p.

Algorithm 1 Generate "parent-children" pairs for all concepts

```
procedure INITIALIZATION
    for t in T do
        assign to t a unique label l_t
procedure STEP 1
    for t in T do
        if t is not parent then
            t_2 ← n  \  max_{n∈N(t)}(CP(t, n))
            if t_2 is not parent then
                x ← parent  \  max_{x∈{t,t_2}}(γ_x)
            l_y ← l_x  \  x is parent of y,  x, y ∈ {t, t_2}
```

Step 2: Link children to their closest concept or group of concepts. We need now to associate concepts to the group in which they can maximize entanglement. We now compare the group $c \in C$ in which a concept t maximize $CP(t,c)$ to the neighboring child concepts t_2 that also maximizes $CP(t,t_2)$. A pair of nodes may maximize together entanglement in comparison of those of a group, so we create in this case a new parent node. In other words, this avoids too coarse aggregations often induced by very occurring concepts, such as queried criteria. We continue updating associations until no change occur anymore. Results are illustrated by in Fig. 2(b).

Algorithm 2 Link children to their closest concept or group

```
procedure STEP 2
    C = {C_1, C_2, ··· , C_k} (groups formed from step 1)
    repeat
        changed ← False
        for t in nodes \ t is not parent do
            c \ max_{c∈C}(CP(a, c))
            t_2 \ max_{t_2∈N(t)}(CP(t, t_2)), t_2 is not parent
            if CP(t, c) > CP(t, t_2) then
                if l_t ≠ l_c then
                    l_t ← l_c
                    changed ← True
            else
                x ←parent, x ∈ {a, b} \ max_{a,b}(λ_a, λ_b)
                l'_x, new label
                l_y ← l_x ← l'_x
                changed ← True
    until ¬changed
```

Step 3: Regroup all groups in a forest. We now compute the hierarchy between group, by simply comparing CP of groups together, starting with the pair of neighboring group displaying the highest CP in the graph (Fig. 2(c)) (and if there is a connection between them). We stop when there are only negative distances (*i.e.* $\forall i,j \ CP(c_i, c_j) \leq 0$) or only one q groups ($q = 1$ or user specified).

Algorithm 3 Regroup all groups in a forest

```
procedure STEP 3
    repeat
        c_x, c_y \ max_{c_i,c_j∈C}(CP(c_i, c_j))
        u ← max_{k∈{x,y}}(γ_{Parent(c_k)})
        v ← min_{k∈{x,y}}(γ_{Parent(c_k)})
        l_{c_v} ← l_{c_u}
        Parent(c_v) ← Parent(c_u)
    until |C| = 1 or max_{C_i,C_j∈C}(CP(C_i, C_j)) <= 0
```

VI. MULTIMEDIA ANALYTICS

A. Visual Cloud Generation

To best support users, the visual cloud should be compact, aesthetic and expressly map the thematic grouping resulting from the hierarchical structure extracted. Inspired by usual tag cloud, it should also integrate images seamlessly. We start by embedding the hierarchical relationships.

Pack Layout initialization: The Pack Layout algorithm [32] uses enclosed diagrams to represent containment (nesting) as the hierarchy (similarly to tree map algorithms). The size of each leaf node reveals a quantitative dimension associated to data points and the enclosing circles show the approximate cumulative size of each subset. Circle packing does not use space efficiently, however it can indicate relative positions of nodes following their hierarchical relationship. Each concept is assigned to the center of the enclosing circle it is represented by (Fig. 3(b,c)). Pack Layout creates a leaf-node per node in the tree, instead, we assign parent nodes to the center position of the higher order circles. This results in a more even distribution of position in the plan, giving a relative position for each concept guided by the hierarchy (Fig. 3, left).

Visual cloud layout: Wordle algorithm [15] is arguably the fastest tag cloud algorithm. Words initial position can not be strictly specified, and size depends on words relative frequency. Words are introduced one by one to some random

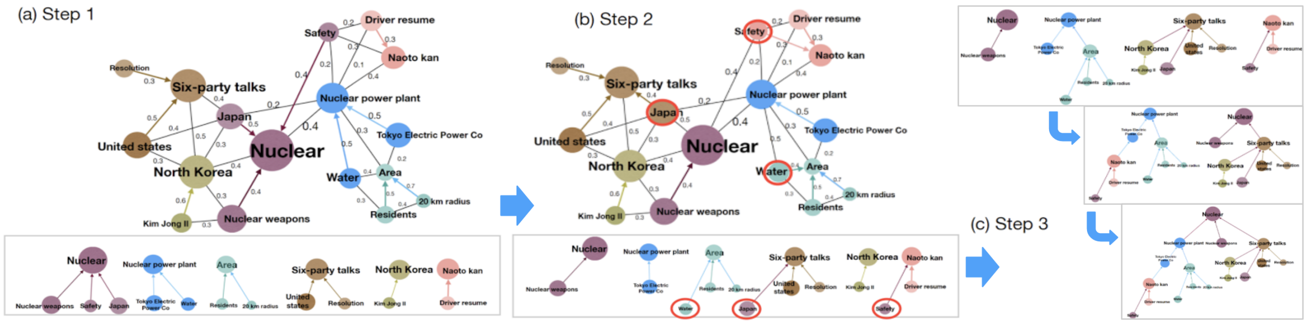

Fig. 2. The construction of the hierarchy. Step 1: constructing the first trees by pair association. Step 2: moving nodes to optimize group assignation (highlighted in red). Step 3:aggregating groups to construct the final hierarchy.

position close to the center of the canvas and iteratively placed in the order of frequency. A word is then displaced if it intersects with any previously positioned words. This displacement is made following an increasing Archimedean spiral until no more intersection is found. We use this to our advantage by constraining the spirals with the Pack Layout's circles (center and separation distance), see Fig. 3 (c,d). We extended the algorithm to take into account any rectangle shape. As a result, concepts are placed in the proximity of their previously calculated position, relatively reflecting their hierarchical structure (Fig. 3).

Regions of Interest: To highlight regions of interest, we offer multiple visual encoding. Textual concepts and image borders are colored upon the group to which they belong. The number of groups can be interactively set, since it corresponds to a different cut of the hierarchy, it simply updates colors, while keeping the layout stable. The size of a concept encodes its frequency on the edges of the multiplex network. Because frequency is only one aspect of the significance of a concept in its group, we introduce a new highlighting that displays how concepts mix with others: an optional background heatmap displays the entanglement index of concepts (Fig. 3(d,e)) and contrasts enough with the categorical scale. Higher entanglement indices assigned to warm color attract user attention. A max blending function as the heat value diffuses from the outer box of concepts maintains text readability.

B. Integration in the Search Engine

Users can place advanced queries on the three criteria of time-frame, face and keywords (Fig. 4 (a)). Above the list of results and the visual cloud (Fig. 4 (d, c)), the system shows a brush-able time bar chart that positions the query results in time (the timeline background is tuned to our usage scenario to

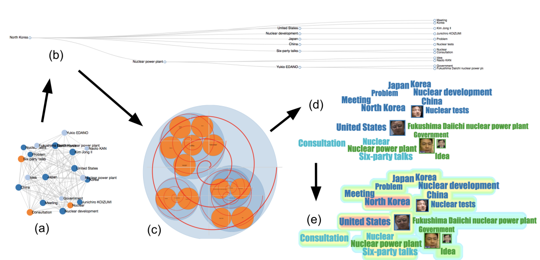

Fig. 3. Visual cloud layout: (a) *LIN* (b) extracted hierarchy (c) Pack Layout embedding (with spirals) (d) visual cloud (e) with heat map.

show periods of interest in color). Similar to traditional search engines, a list of results is presented (Fig. 4 (d)). It is ordered by time, titled by date/time/segment. A snippet composed of the first lines of captions can be expanded. Clicking on a result launches a video player at the segment position.

Interactions: Extracting a hierarchy from the *LIN* implies that we loose its topology but the neighborhood of a concept is key to exploration. It is restored through hovering interaction: all concepts are dimmed except for those neighboring the concept in the *LIN* with timeline highlighting (Fig. 4(c,e)). When the heat map is active, new heat values are computed on-the-fly which map the *local* entanglement indices of concepts in the multiplex subgraph induced by the concept hovered (similar to [4], Fig. 4(e)). Filtering results is achieved by clicking a concept or brushing the timeline, however, *leapfrog* interaction [4] by double-clicking a concept or the timeline brush will result in a new query corresponding to this filter, recomputing a new visual cloud for a finer grain analysis. This corresponds to *Search* tasks often formulated by Shneiderman's mantra *"Overview first, zoom and filter, then details-on-demand"* [33].

VII. Results

Usage Scenario, Abe and North Korea: In a previous study, the authors investigated appearances of Japanese Prime Ministers on NHK[22]. One interesting conclusion was the growth in screen appearances of Abe during Prime Minister Koizumi's ruling, before becoming himself Prime Minister in the following elections. Our system allows to refine this study by placing a complex query to search all news segments during Koizumi mentioning Abe by name or face (Fig.4). A demonstration video is available at https://youtu.be/VfGwa6T94t8.

The timeline (which is augmented with Prime Minister's rulings on the background) confirms the growing mention of Abe. This is no surprise knowing that Abe was chief cabinet secretary during Koizumi's third term. The visual cloud proposes 5 groups: about elections, about Yasukuni Shrine, about the newspaper Asahi Shimbun, about Japan/Korea/China, and about North Korea. But turning on the heat map, the most prominent word becomes North Korea by far. Indeed, Abe was chief negotiator on issues related to abductions of Japanese citizens by North Korea, managing to free 5 of them. Leapfrogging on the keyword "North Korea" (Fig.4(e)) makes a new search of Abe associated with North Korea during Koizumi

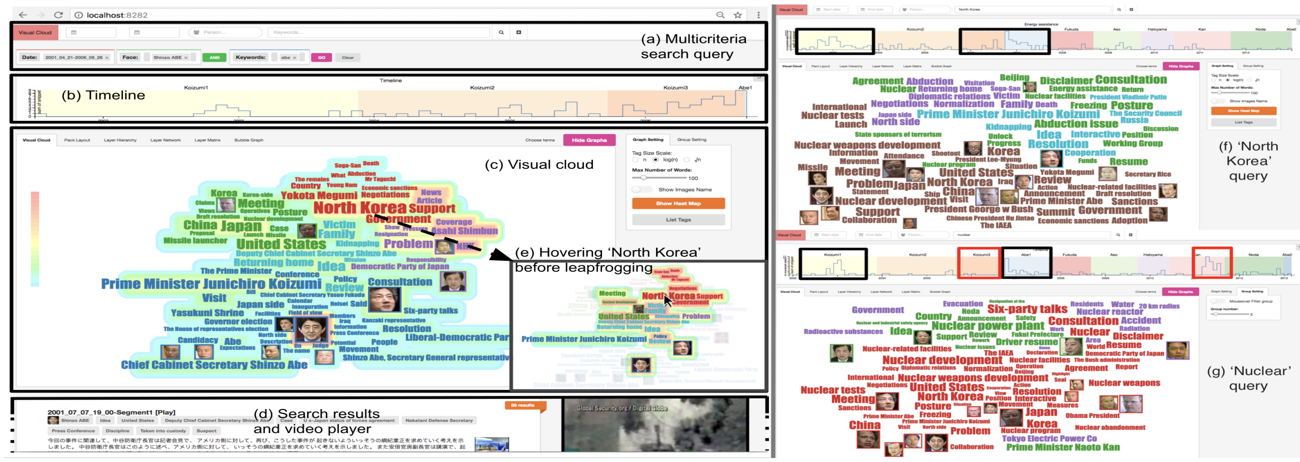

Fig. 4. Left. Interface overview, search query over Abe during Koizumi terms (a) - very active during the end of Koizumi3 (b), mention of 'North Korea' stands out (c). Hovering over 'North Korea' shows that it strongly relates to 'abduction' (e). Right. Comparison of the queries on *North Korea* (f), and *Nuclear* (g): two periods coincide (in black) Koizumi1 and Abe1 but we can notice two major differences (in red). Koizumi3 period did not associate much *North Korea* and *Nuclear*, and Kan period associate *Nuclear* with *Accident*. Demonstration video available at https://youtu.be/VfGwa6T94t8.

terms. Browsing the timeline among the three terms highlights different subtopics at each terms. We may mention a meeting during Koizumi's first term, associated with the three faces of Koizumi, Kim Jong Il, and Abe; mentions of sanctions, missile launch and draft resolutions during Koizumi's third term.

Now a new search on the keyword *North Korea* highlights that it is most active during first Koizumi ruling and especially during first Abe ruling (Fig.4, left). North Korea related abductions was excessively reported on the media and Abe's administration has put pressure on NHK to "pay attention" [34]. A last search on the keyword *nuclear* gives 3 spikes in the time-line (Fig.4, right). Two of the spikes coincide with *North Korea* previously described (Fig. 4). One big difference comes with Koizumi's third term, when no mention of nuclear issue is made. The last in 2011 after the Great East Japan Earthquake about the nuclear power-plant accident.

Validation of the hierarchical algorithm: We offer a heuristic that optimizes entanglement homogeneity $\mathcal{H}$in networks formed by association of results and concepts that captures cohesion of a group of documents[3]. To the best of our knowledge no other work attempt create a hierarchy of concepts to maximize cohesion between documents related to concepts on one hand, while preserving the relationships in the *LIN* on the other hand. However, we can still compare $\mathcal{H}$with that of groups of documents corresponding from concept clusters induced by a Louvain segmentation [35] on the *LIN*, and with random segmentation to serve us as a baseline. The random segmentation only agglomerates nodes randomly from the network's topology until reaching a given number of clusters. Based on 8 queries of 50/100/150/200 concepts, we randomized 10 generations of $k$clusters ($k$reused from the Louvain segmentation [35] to equally compare between the three types of segmentation). We then average $\mathcal{H}$for each group, across queries and segmentation. Results in Table I confirms that our segmentation results in more cohesive subgroups.

Implementation and complexity: The system is implemented in HTML5 with the popular semantic-UI and d3 libraries. Database indexing and access is implemented in python. The hierarchy algorithm complexity is bounded by the computation of entanglement which requires an Eigen decomposition of a matrix of dimension the number of concepts.The construction of the *LIN* and $n_{t,t'}$are made while constructing the multiplex network of results. Both depend on the number of results $|V|$and number of concepts $|T|$, $O(\frac{1}{2}(|V| * (|V| - 1) + |T| * (|T| - 1))$. The computation of the hierarchy is a greedy optimization similar to Louvain [35] estimated in $O(|T|log|T|)$. No complexity is discussed for the Pack-Layout algorithm [32] but runs in milliseconds for a thousand circles. The number of circles depends on $|T|$, which is at best a few hundreds. The word cloud generation is based on Davies' heavily optimized implementation (see www.jasondavies.com/wordcloud/about/, which is bounded by computation of bounding boxes and collisions (not impacted by our modifications). Including the Pack-Layout initialization, the word cloud generation can be considered instantaneous. The heat-map generation is done in one pass through each concept, with a static Canvas implementation faster and more memory efficient than DOM population of SVG elements.

VIII. Discussion and Conclusion

Although tag clouds do not offer much room for visual encoding except for layout, size, and color, users have positively welcomed the heat map. It helps to correct the information overload when too many concepts are displayed, by bringing initial focus on the highlighted concepts. However, it reduces the perception of group differences. Future work will explore the design space offered by joining heat-maps and visual clouds, especially for interaction (used here to reintroduce the *LIN* topology). We also improve traditional tag clouds with thumbnails as a supplementary information, bringing new information in form of visual cues. We only use faces this time but we plan to use other cues, such as objects or logos. We compared here word occurrence and tracking time in terms of a rougher segment occurrence, but the comparison of these heterogeneous measurements remains open.

The design of our multimedia system completely falls into the *search* task as described by Brehmer *et al.*'s typology [5] (in their Fig. 1) consisting of two parameters: *search location* and *search target*, either being *known* or *unknown*. Each situation maps to a subtasks: *lookup, locate, browse* and *explore*. Usual search engines are often used for *lookup* tasks (with *location* and *target* both *known*). Video broadcasters such as Youtube link videos together to support *browsing* tasks (when the *location* is *known* but not the *target*). *Locating* tasks consists in *knowing* the *target* but not the *location*, made successful by keyword search. The visual cloud supports *exploration* and *browsing* tasks, our keyword search is too strict to provide proper *lookup* task support. It should be improved, together with video linking, to better support *lookup* and *locate* (beyond time location with the timeline).

One last important future work concerns *comparison* tasks. We currently refine information through visual cloud hovering, timeline browsing, and leapfrogging. However, beyond side-by-side comparison of two queries tabs, we do not have explored other means of comparison. This need quickly rises as we would like to compare periods of time.

Finally, we have presented a system designed for exploration of the NHK News 7 archive with a visual cloud, that improves from tag clouds in several ways. It takes roots in a multiplex network formulation, and uses group entanglement of search result to build a hierarchy with stable grouping, that results in a very fast and interactive drawing. The cloud is coordinated with search query, results, and timeline to allow further browsing, exploration, and query refinement. We illustrated our system with the case of Abe Shinzo and North Korea, studied the ability of our hierarchy to optimize group entanglement, and presented implementation and complexity.

TABLE I
COMPARING $\mathcal{H}$ AMONG GROUPS OF RESULTS OF VARYING SIZES

Group Size	*Random*	*Louvain*	*Multiplex*
All sizes	0.43874	0.53359	**0.59600**
50 concepts	0.58213	0.66137	**0.68647**
100 concepts	0.38544	0.46707	**0.55022**
150 concepts	0.38665	0.52803	**0.60585**
200 concepts	0.40075	0.47792	**0.54149**

REFERENCES

[1] N. Hervé, M.-L. Viaud, J. Thièvre, A. Saulnier, J. Champ, P. Letessier, O. Buisson, and A. Joly, "Otmedia: the french transmedia news observatory," in *Proc. ACM Multimedia*. ACM, 2013, pp. 441–442.

[2] J. Sinclair and M. Cardew-Hall, "The folksonomy tag cloud: when is it useful?" *Journal of Information Science*, vol. 34, no. 1, pp. 15–29, 2008.

[3] B. Renoust, G. Melançon, and M.-L. Viaud, "Entanglement in multiplex networks: understanding group cohesion in homophily networks," in *Social Network Analysis*. Springer, 2014, pp. 89–117.

[4] B. Renoust, G. Melancon, and T. Munzner, "Detangler: Visual analytics for multiplex networks," in *Computer Graphics Forum*, vol. 34, no. 3. Wiley Online Library, 2015, pp. 321–330.

[5] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE TVCG*, vol. 19, no. 12, pp. 2376–2385, 2013.

[6] C. Carpineto and G. Romano, "A survey of automatic query expansion in information retrieval," *Computing Surveys*, vol. 44, no. 1, 2012.

[7] U. Scaiella, P. Ferragina, A. Marino, and M. Ciaramita, "Topical clustering of search results," in *Proc. WSDM*. ACM, 2012, pp. 223–232.

[8] A. Di Marco and R. Navigli, "Clustering and diversifying web search results with graph-based word sense induction," *Computational Linguistics*, vol. 39, no. 3, pp. 709–754, 2013.

[9] E. Gomez-Nieto, F. San Roman, and et al., "Similarity preserving snippet-based visualization of web search results," *IEEE TVCG*, vol. 20, no. 3, pp. 457–470, 2014.

[10] E. Clarkson, K. Desai, and J. Foley, "Resultmaps: Visualization for search interfaces," *IEEE TVCG*, vol. 15, no. 6, 2009.

[11] A. Nocaj and U. Brandes, "Organizing search results with a reference map," *IEEE TVCG*, vol. 18, no. 12, pp. 2546–2555, 2012.

[12] L. Barth, S. G. Kobourov, and S. Pupyrev, "An experimental study of algorithms for semantics-preserving word cloud layout," *University of Arizona Report*, 2013.

[13] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma, "Semantic-preserving word clouds by seam carving," *Computer Graphics Forum*, vol. 30, no. 3, pp. 741–750, 2011.

[14] J. Xu, Y. Tao, and H. Lin, "Semantic word cloud generation based on word embeddings," in *PacifVis,*, 2016, pp. 239–243.

[15] F. B. Viegas, M. Wattenberg, and J. Feinberg, "Participatory visualization with wordle," *IEEE TVCG*, vol. 15, no. 6, pp. 1137–1144, 2009.

[16] M. Krstajic, M. Najm-Araghi, F. Mansmann, and D. A. Keim, "Incremental visual text analytics of news story development." in *Visualization and Data Analysis*, 2012, p. 829407.

[17] H. Li, B. Jou, J. G. Ellis, D. Morozoff, and S.-F. Chang, "News Rover: Exploring topical structures and serendipity in heterogeneous multimedia news," in *Proc. Multimedia*. ACM, 2013, pp. 449–450.

[18] A. Kochtchi, T. v. Landesberger, and C. Biemann, "Networks of names: Visual exploration and semi-automatic tagging of social networks from newspaper articles," in *Computer Graphics Forum*, vol. 33, no. 3. Wiley Online Library, 2014, pp. 211–220.

[19] H. Luo, J. Fan, J. Yang, W. Ribarsky, and S. I. Satoh, "Exploring large-scale video news via interactive visualization," in *IEEE VAST*. IEEE, 2006, pp. 75–82.

[20] M. Itoh, M. Toyoda, C. Z. Zhu, S. Satoh, and M. Kitsuregawa, "Image flows visualization for inter-media comparison," in *IEEE Pacific Visualization Symposium, 2014*. IEEE, 2014, pp. 129–136.

[21] I. Ide and F. Nack, "Explain this to me!" *ITE Trans. on Media Technology and Applications*, vol. 1, no. 2, pp. 101–117, 2013.

[22] B. Renoust, T. Kobayashi, T. D. Ngo, D.-D. Le, and S. Satoh, "When face-tracking meets social networks: a story of politics in news videos," *Applied Network Science*, vol. 1, no. 1, p. 4, 2016.

[23] I. Ide *et al.*, "Topic threading for structuring a large-scale news video archive," *Image and Video Retrieval*, vol. 1, no. 1, pp. 123–131, 2004.

[24] K. Fujimura, H. Toda, T. Inoue, N. Hiroshima, R. Kataoka, and M. Sugizaki, "Blograngera multi-faceted blog search engine," in *Proc. WWW, Weblogging Ecosystem*, 2006.

[25] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[26] J. Shi and C. Tomasi, "Good features to track," in *Proceedings CVPR'94*. IEEE, 1994, pp. 593–600.

[27] T. D. Ngo *et al.*, "Face retrieval in large-scale news video datasets," *IEICE Trans. on Information and Systems*, vol. 96, no. 8, pp. 1811–1825, 2013.

[28] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, "Openface: A general-purpose face recognition library with mobile applications," Technical report, CMU-CS-16-118, CMU, Tech. Rep., 2016.

[29] D. D. Le and S. Satoh, "Indexing faces in broadcast news video archives," in *2011 IEEE 11th ICDM Workshops*, 2011, pp. 519–526.

[30] A. Noack, "Modularity clustering is force-directed layout," *Physical Review E*, vol. 79, no. 2, p. 026102, 2009.

[31] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proc. CIKM*, ser. CIKM '02. New York, NY, USA: ACM, 2002, pp. 515–524.

[32] W. Wang, H. Wang, G. Dai, and H. Wang, "Visualization of large hierarchical data by circle packing," in *Proc. SIGCHI*, ser. CHI '06. New York, NY, USA: ACM, 2006, pp. 517–520.

[33] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Visual Languages*. IEEE, 1996, pp. 336–343.

[34] G. McCormack, "Japan and north korea: The long and twisted path toward normalcy," Working Paper. US-Korea Inst. at SAIS, Tech. Rep., 2008.

[35] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 10, p. 8, Oct. 2008.