

Displaying Cross-Cultural Differences in News Videos III

Luvena Huo (lh2933), Tiansheng Sun (ts3348)

Supervisor: Dr. John R. Kender

Columbia University

1. INTRODUCTION

The continuous growth of globalization has resulted in the spread of single events through multiple news sources. These sources range across different affinity groups in the world and are spread over diverse types of media. We are studying the differences and similarities in the portrayals of the same event across different affinity groups, specifically via video.

In this report, we describe how we redesigned the user interface to display the information in a more understandable and convenient way for the end user. Our goal was to be able to complete the design, complete rounds of user testing to confirm the clarity of the user experience, and begin developing this new prototype.

2. PREVIOUS STUDIES

In the previous semester, David (Dave) Dirnfeld [1] modified Jiaqi Liu's [2] user interface design by including a PatternFly timeline and building up a SQL Database to store and retrieve information. While this was a leap in improvement on the design of the user interface, we found several existing problems that we want to explore this semester, including issues with the clarity of the timeline as well as an expansion of the user interface.

2.1 Issues with the Timeline

There were several problems with clarity in the timeline. One of the glaring confusions that presented itself in Dave's original design stemmed from the three rows/categories: "USA", "Common", and "China" (See Fig. 1). The sections for "USA" and "China" are clear: each data point in the time frame on that row represents a specific piece of news associated with either "USA" or "China". However, we found the "Common" row confusing: its primary purpose is to illustrate

similarities between the two affinity groups by connecting news from the "USA" row to the "China" row, but it had nothing to do with "time" even though it was placed on the timeline. Therefore, we wanted to find additional ways to more cleverly capture the connections between two news pieces.



Fig. 1. Original UI with Dave's Modifications.

We also wanted to find ways to improve upon the design of the user interface. We wanted to make sure the site's clarity and usability was thoroughly brought out, and we touched up on the general design of the interface. Basically, we wanted to maintain a simple user interface while making changes for situations such as allowing someone with no prior background knowledge of our application to be able to easily navigate the different features. We also took into consideration color and font according to the WCAG guidelines - for example, previously, each news data point was represented with a hollow rectangle with little distinction in color. In order to make the data points more user friendly and appealing, we wanted to modify the rectangle and color of the data points to increase usability.

2.2 Expanding the User Interface

After discussing with members of the other teams, we all agreed that it would be necessary to expand the user interface in order for the website to work for more than just the information that we have currently hardcoded. Specifically, we discussed

additional features such as being able to select different news topics before selecting different affinity groups for each specific news topic. In order to implement this, we have to add new features to the existing user interface to accommodate for more data in the future.

3. DESIGN PROCESS AND STRUCTURE

3.1 User Testing

In order to ensure that each iteration of our design prototype was appropriate for the users, we designed multiple design prototypes using Figma, and conducted multiple rounds of user testing for all prototypes and design choices within the prototypes. After any features were added or whenever we created another iteration of our prototype, we would perform multiple rounds of user testing for that specific version of the prototype. Each user test was standardized in order to reduce error across tests. When the purpose of our test was more feature driven, we would generally provide some background information or guide them to pay more attention to certain aspects of the site; when the purpose of our test was a more holistic assessment of how a user might navigate the site, we would generally leave the user to explore the site on their own and make observations based on the ease of their experience, while asking questions about their experience afterwards. We also often performed A/B testing when deciding between different implementations of the same feature (e.g. color, methods of displaying background information, etc). After each user test, we usually ask the user to point out areas of improvement or features that were confusing, and take note of potential areas of improvement for our next iteration. Here is an example of a user test that we performed on an earlier test: https://drive.google.com/file/d/1YiQPmfjGQ9StDzsb6sNWET_U14wntWw/view?usp=sharing

3.2 Standardized Content

Our design follows a standard 12 column grid system in order to ensure clarity. Our final prototype and design also followed the Web Content Accessibility Guidelines (WCAG) Version

2.1^[3], which adheres to four main principles: perceivable, operable, understandable, and robust.

3.2.1 Perceivable

In our designs, we did multiple rounds of user testing in order to ensure every relationship in the diagram was clear and “can be programmatically determined” (WCAG 1.3.1) - including removing the “common” column from the original design. Through a timeline, we defined a meaningful sequence of events (WCAG 1.3.2), and added pop ups and explanations for navigating the site in order to increase ease of use (WCAG 1.3.3).

The colors used in the site (mainly on the data points) are not essential to understanding the differences between affinity groups but helped to enhance the user’s experience (WCAG 1.4.1). The color contrast ratios where colors were used as “images” (data points) were greater than three to one: blue #2981BC - 4.23:1 and red #E85454 - 3:60:1 (WCAG 1.4.11), and text color contrast ratios were above 7:1 (WCAG 1.4.6). After performing several user tests, we found that users generally preferred contrasting color palettes across different affinity groups as opposed to similar color palettes (red and blue was generally preferred to blue and green). Width of text blocks was never over 80 characters wide, and paragraph/line spacing was large enough for clarity (WCAG 1.4.8). Hover content, reflow, and text size were already in accordance with the accessibility standards.

3.2.2 Operable

Each one of our web pages are titled with headings where necessary, animations are essential and simplified, and content focus is meaningful.

3.2.3 Understandable

For each feature in our prototype, context is provided through titles, highlighted data points, or linear connections, regardless of which section of the user flow one might be in. We ensured that throughout the project, navigation is consistent throughout the section of flow the user might be in (page does not shift unnecessarily). We created back buttons to initiate error prevention in case

incorrect events or affinity groups are selected, and a user’s selection can be changed at any time (WCAG 3.3.4). Help or additional information needed to navigate the site is provided through pop up buttons (WCAG 3.3.5).

3.2.4 Robust

Content is robust across multiple user groups (some aspects of being “robust” are currently more irrelevant to our project - e.g., assistive technology compatibility).

4. DESIGN IMPROVEMENTS

4.1 Flow Clarity

We wanted to work to increase the clarity of the user interface and make the user flow self-explanatory. We considered multiple options: including background information on the landing page of the site, adding clarifying information through a pop-up information modal on the homepage where the diagram is, or including information above the diagram on the homepage. After a couple rounds of A/B user testing, we found that the pop-up information modal was the most effective way to communicate the information, as it simplified the user interface for the homepage, and it is conventional to use pop up modals for clarification. We noticed that when the information was on the landing page instead of the homepage, people tended to not absorb the information or not read it through, and would not have the option to access that information unless they left the homepage.

4.2 Showcasing Relationships Between News

One of the major purposes of our user tests was to find better ways to showcase relationships between news. To achieve this, we initially created two different prototypes with different designs in both. The first design focuses on finding a similarity pair to another piece of news across affinity groups given a specific news. The second design focuses on giving users the opportunity to blindly look into all pairs of news. Both seem to be useful features to include in our project.

In one of the original designs, a user would have the option to click on a “Show Commonalities” button once a news data point has been selected (see Fig. 2). When a user clicks the button, any “similarity pairs”, or data points in the other affinity group that are reporting on the same event, will display itself on the user interface if any exists (See Fig. 3). The advantage of this design is its simplicity and clarity in displaying commonalities; however, unlike the other design that we will talk about in further detail, there is no option to view “all similarity pairs”. Despite this, user test results were generally positive, and indicated that the design was clear and able to communicate information concisely.



Fig. 2. Design with “Show Commonalities” button before button is clicked.

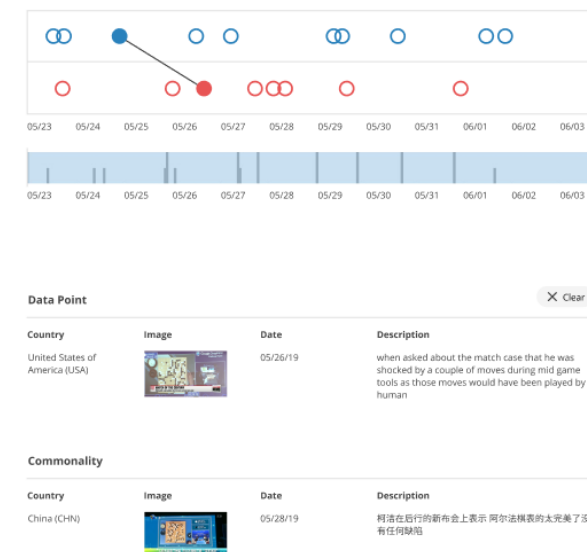


Fig. 3. Design with “Show Commonalities” button after button is clicked.

In the second design, a user has the option of clicking on a “similarity pairs” tab, which takes the user to a page that lists all pairs of related news across affinity groups (see Fig. 4). To check the similarities between a pair listed, the user can simply click into the pair to display the information. If the user instead clicks on a specific news data point from the diagram, they would still have the option to view similarity pairs through the “similarity pairs” tab, but this would only display similarity pairs with commonalities to the selected data point. During the user tests and A/B testing, all users thought the design made sense and was understandable. However, some users, especially during the A/B testing round, were a bit puzzled at the functionality of the *similar pairs tab*. Many think that the first design is clearer without this specific feature.



Fig. 4. Design with “Similarity Pairs” tab feature.

4.3 Similarity Pairs Filter

However, without the second design, we are losing the functionality of being able to view commonality connections that Dave had in his original design and that we had in the second design. Since we were primarily going with the first design, we wanted to find a way to retain this functionality while still removing the commonality row from our final design to improve clarity. We explored multiple design options, including using tag names to filter all similarity pairs that a user might choose to select and a filter. We’ve tried other designs that considered listing similarity pairs out but realized that it was difficult to list “similarity pairs” out without being confusing and too wordy in our user interface.

We then explored the possibility of sorting these commonalities by tag name in order to simplify the user interface and for users to be able to more easily find the pair that they are looking for. Borrowing upon the suggestions from user results for our second design, we realized that there could be a method to display all similarity pairs in a way that makes information both clear and easy to find. Part of the issue with listing all similarity pairs in our second design was that a “similarity pair” was labeled by numbers (e.g., “Similarity Pair 1”), with no way to create identifying information. The labels with numbers are confusing as they are hardcoded without any specific meaning. An alternative solution, formulated with the help of the professor, is to find an algorithm to sort all the pairs based on keywords and how common the two news pieces are. For this option, we would still list all similarity pairs on the homepage, but would have a descriptive phrase as a “tag name” that identifies each similarity pair (e.g., “AlphaGo’s First Win”). While this would simplify the search for a specific similarity pair, this design would still look slightly more cluttered on the homepage user interface and create confusion due to the “similarity pairs” tab, and requires more work to implement, both from the perspective of the user interface in terms of organization as well as the natural language processing team. However, this option could be something to consider implementing or refining in the future.

The option that we are currently considering and that holds the most process is a filtering button, which allows a user to be able to filter out news sources that do not have a similarity pair. When the filter button is not selected (the default option), all news data points would be shown (see Fig. 5). However, when the filtering button is selected, the only data points that would remain have a similarity pair in the other affinity group (see Fig. 6). The pro of this design choice would be that it maintains the clarity and simplicity of the user experience; however, the issue still remains that it is difficult for users to find the exact similarity pair that they might be searching for. This issue persists even

beyond finding similarity pairs - even if a user was just looking for a specific news article (one data point), there is currently no way for them to immediately find it; they would have to go through a trial-and-error process where they click on all the data points in order to search for the piece of news that they are looking for. This process could potentially be simplified by both tag names and utilizing the information that we have stored in the current tags that do appear when a user hovers over data points.

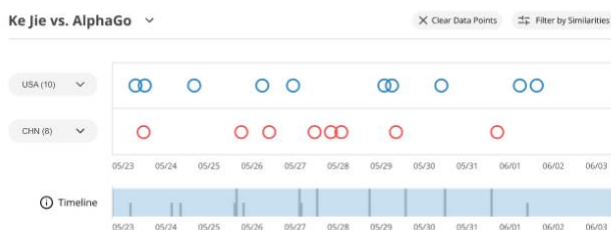


Fig. 5. Diagram showing all data points (default) before filtering by similarities.



Fig. 6. Diagram showing only the data points in a similarity pair with a point in the other affinity group (after selecting “Filter by Similarities”).

4.4 Timeline Axis

Neither of us changed anything with the timeline axis in our prototype. During user tests, some users mentioned that the timeline axis was confusing because both days and times were shown as tick values under some zoom levels. Since the level of distinction for time does not have to be so high in our project, we have decided to show at least day level information as tick value to minimize confusion.

4.5 Confusion Over Two Sets of Dates

In the current timeline, there are two sets of dates used to present the information. The lower set of dates stays stagnant and will always represent the

entire start to end time period. The upper set of dates changes as the user zooms in on a section of data and displays the start to end period of the zoomed in data.

However, we found that users were generally confused over the two sets of timelines during user testing (only about 1/5 users figured out the functionality without context). We looked into alternative timelines that could show time-related information in a clear way, including TimeLine (a timeline that displays patient records with visual information), multi-scale timeline sliders, and forensics timelines. However, we concluded that it would be easiest to find a way to increase the clarity of the timeline by simply providing a brief explanation for why the timeline was set up the way that it was. Therefore, in our design, we added a pop-up button next to the timeline explaining the “zoom in” feature of the timeline as well as the two sets of dates in the timeline, which proved to be effective in an additional round of user testing.

4.6 Displaying News Data Point Information

Once a news data point is clicked, the information about the news will be shown underneath the timeline. We tried multiple different designs to display this information during the first few rounds of the user tests. One of the designs we explored divided and displayed all information into four column sections: country, image, date and description. Another design focused on the image and description, placing the image to the left and the description to the right (see Fig. 7). Since some users mentioned that the second design is much clearer and the image and text information of the first design is way too small, we have decided to stick with our second design.

After deciding on the design, we did an A/B user test to determine how to showcase the date, tag and country information of each piece of news. One design, specifically suggested by one user, is to include a pop-up tab displaying the additional information. However, since the pop-up button would be an additional and unnecessary click for

the end user, we ended up going with a second design that directly listed the country, date, and tag information underneath the image.



Fig. 7. Finalized Information Display for a Data Point / Commonality Pair

5. CODE

While finalizing the last bits of our prototype, we spent the last few weeks modifying the existing user interface. We have decided to stick with using Spark for the project with the bootstrap library for web components. Here are the specific changes that we made to the user interface.

5.1 Information Pop-Up Modal

A button next to the title pops up information about our project (see Fig. 8.). This is part of the header and can be seen regardless of the page that the user navigates to.

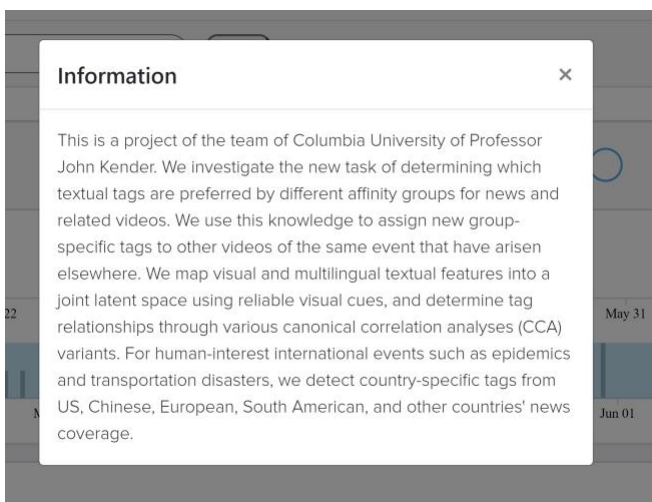


Fig. 8. Information pop-up that contains background information about our project.

5.2 Introduction Page and Topic Selection

We have separated the landing page and the home page by creating an introduction page (see Fig. 9.), which includes a drop-down menu listing all possible event topics. Once the user selects an event topic, the user will be directed to the homepage page. The dropdown menu that allows the user to change the event topic persists through the homepage as well, allowing the user to have the option to switch events at any time.

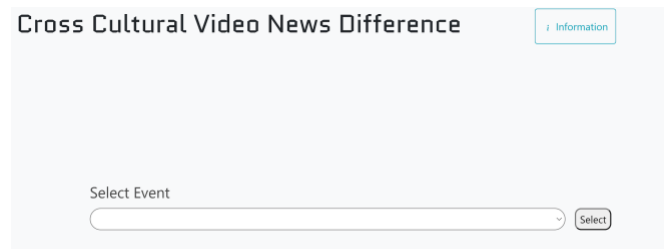


Fig. 9. Introduction and Landing Page.

5.3 Timeline Design

We modified the timeline and data points according to the results of our user tests and final prototype. We have changed the shape of each data point from a hollow square to a hollow circle, and the colors from green and blue to blue and red. When a user clicks a data point, the hollow circle would then be filled.

5.4 Mapping Relationships Between News

We have made several design modifications to how relationships are mapped between news articles according to our prototype. First of all, we deleted the “Common” entry in the timeline. For patternfly timelines, the input for each entry has to be an array of JSON objects and in the original design. To remove the *Common* entry, we simply deleted the JSON object related to the *Common* entry. In order to show the related news once by clicking the “Show Commonality” button, we also had to modify the JSON object for each affinity group by checking if there is a related news. If there was, we appended the index and information of that news to the JSON object.

```

if c[1].name == 'United States':
    tags =
Tags.query.filter_by(commit_id =
c[0].id).all()
    similarity =
c.Commits.get_common().all()
    if not similarity:
        data = {
            "date":
c[0].video_date.strftime(date_format)
        ,
            "details": {
                "name": c[0].index,
"message": c[0].message,
                "tags": [t.tag for t
in tags],
                "isCommon": 0
            }
        }
    else:
        for a in similarity:
            data = {
                "date":
c[0].video_date.strftime(date_format)
            ,
                "details": {
                    "name":
c[0].index, "message": c[0].message,
                    "tags": [t.tag for
t in tags],
                    "isCommon": 1,
                    "common": a.index,
                    "other_message":
a.message
                }
            }
        commit_data[0]["data"].append(data)

elif c[1].name == 'China':
    similarity =
c.Commits.get_common().all()
    if not similarity:

commit_data[1]["data"].append({
    "date":
c[0].video_date.strftime("%Y-%m-%d %H

```

```

:%M:%S"),
        "details": {
            "name": c[0].index,
"message": c[0].message,
            "isCommon": 0
        })
    else:
        for a in similarity:

commit_data[1]["data"].append({
    "date":
c[0].video_date.strftime("%Y-%m-%d %H
:%M:%S"),
        "details": {
            "name":
c[0].index, "message": c[0].message,
            "isCommon": 1,
            "common":
a.index,
"other_message": a.message
        })

```

We then made the *Show Commonality* button. If a selected news does not have a related news or “similarity pair”, the button is disabled. Otherwise, once the user clicks the button, a line would be drawn connecting the two related news data points (Fig. 10).

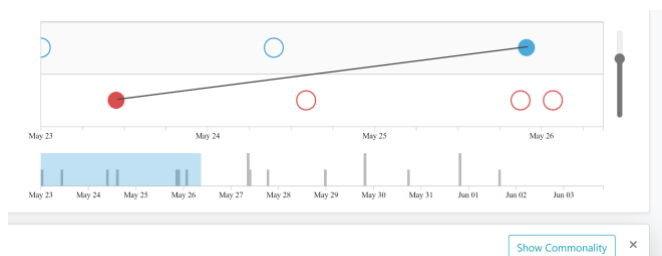


Fig. 10. Timeline display when the “Show Commonality” button is clicked.

5.5 Add Chinese Tags

This semester, the NLP team worked on extracting keywords from Chinese news and sent their progress to us. We are working on adding the tags extracted by the NLP group into the user interface, and it should be incorporated before the start of next semester.

5.6 Other styling changes

We also made some other styling changes to the project and changed the css file extensively. For example, we imported several fonts and changed the timeline so that only dates, not times, are shown.

6. FUTURE WORK

6.1 User Interface Improvements

Due to the limited amount of time, we did not finish modifying the user interface to match it with our exact finalized version of prototype. Here are some improvements that still need to be made:

- Change the information display of each news data point to match our prototype
- Implement drop down menus to select affinity groups for the timeline
- Change the line between two related news data points when they are connected to dynamically update when a user zooms in or out. To implement this, the locations of the two news data points also need to be updated dynamically.
- Implement the filter from our prototype in order to allow users to filter out data points without similarity pairs, and only view data points with similarity pairs

6.2 Problem with Timeline

Upon a closer look into the dataset, we found cases where two news data points are extremely close together on the timeline, making it almost impossible to distinguish between them and select a specific one. Therefore, to make the program more user-friendly, there needs to be a way to better distinguish between them.

6.3 Informed Titles for Similarity Pairs

In our finalized prototype, we have decided to replace the *similarity pairs* tab that lists out all similarity pairs with a filter. However, as the research moves forward, it is possible that this idea of displaying and sorting related pairs based on keywords and tags will become more relevant. If so,

this could be a very interesting topic to look into in the future.

7. PROTOTYPE & GITHUB

Link to the current GitHub:

<https://github.com/luvenahuo/CrossCulturalMultiMediaAnalysis>

Link to the current prototype:

<https://www.figma.com/file/UKiz0IaKuDXjRSGAxMcLsc/CrossCulturalNews?node-id=569%3A1217>

The most updated version of the prototype, with enlarged descriptions and photos, is in the page Final Prototype 2, and the earlier prototype we were considering is in the page Final Prototype 1. The page Explorations contains much earlier design drafts and prototypes we were considering and that we conducted user tests on.

7.1 Navigating Figma

Pages and different layers in the prototype can be viewed on the top left. To change the design or to view the HTML/CSS code for a specific component, click “Design” on the right to view color codes, positioning, border radius, etc. To test the page out as if it were a real website, click on the “play” triangle button at the very top right (next to the share button). The pages are prototyped, and the **third blue data point** on the timeline diagram is prototyped. To change or view the prototypes, click on “Prototype” next to “Design” on the top right.

8. REFERENCES

[1] David Dirnfeld. Displaying Cross Cultural Differences in News Videos II. http://www.cs.columbia.edu/~jrk/NSFgrants/videoaffinity/Interim/20y_dave.pdf

[2] Jiaqi Liu. Displaying Cross Cultural Differences in News Videos. http://www.cs.columbia.edu/~jrk/NSFgrants/videoaffinity/Interim/19x_jiaqi.pdf

[3] Web Content Accessibility Guidelines 2.0,
W3C World Wide Web Consortium (Latest
version at <https://www.w3.org/TR/WCAG21/>)