Grape News

Final Report -- Team Gungnir

ucdgrapenews.com

Chang Liu 16201322
Hong Su 15211605
Lu Tong 15210751
Wenrui Shen 15210671
Xinlei Lin 15211483
Zhenyu Liu 15211283

Augest.16.2017

Table of Contents

1. User Scenarios

1.1 Problems being addressed

Due to the advancements in digital technologies, information overload, which conveys the notion of receiving too much information for an individual to process, has become one of the defining characteristics in the latest decades (Menon, Sheridan & Ferrel, 1976). This problem is particularly apparent in the field of news industry. News producers continue to increase their volume of production and delivery platforms for reaching and maintaining news consumers (Swartz J., 2011). Especially for web news sources, over thousands of news articles are produced on hundreds of online news streams every day, which has led to the vast oversupply of news information (Keim & Daniel A., 2011). The growing amount of news articles increases the obstacles in understanding the current events because old news coverages are quickly replaced by the latest information (Allan J (ed.), 2002). Additionally, news articles that are published on different platforms often have similar contents (Alonso, O., Fetterly, D. & Manasse, M, 2013). These overlapping contents might decrease the speed of acquiring information and increase news fatigue levels (Nordenson B, 2008).

Consequently, news readers tend to have a feeling of helpless when it comes to seeking out the news that is useful to them (Holton, A. E & H. I. Chyi, 2012). Given such circumstances, it is understandable that most of news consumers feel overloaded with the amount of news. In other words, comprehending how an event develops over time can be a difficult and time-consuming process, and that is where this project comes to the rescue.

1.2 Target user

Our target users are those news readers who have a preference for getting news online and feel overloaded with the amount of news that they are confronted with. They want to be well-informed with popular topics without wasting time on reading duplicated content and seeking out the important news articles from different news platforms. The following user stories can provide a more detailed description of user scenarios:

User Story 1: Jack is a software engineer working in a software company. He likes to keep updated with popular topics and chat with his colleagues on these hot events

during lunch time. However, discovering topics is a tiresome job as trending topics are updated in real time. Therefore, he wants to have a website which can offer him a range of popular topics, and these topics can be updated dynamically.

User Story 2: David is a full-time student studying Computer Science at UCD. One day he saw a discussion about the Manchester Bombing on twitter and was wondering what was going on with this event. However, searching the keywords 'Manchester Bombing' on Google would return over millions of results, and these news articles come from various news delivery platforms. Browsing news on different websites is time-consuming. As a result, David wants a news aggregator that can provide comprehensive news coverages belonging to a particular topic.

User Story 3: Chloe is a bank clerk working in AIB. She likes to read news from different news platforms to explore their full content. However, when she was reading news about one topic on these different websites, she found that a lot of similar articles were reported, and it was difficult for her to identify the important news. Therefore, she hopes to find a website to detect overlapping news and select the most representative news articles for her.

User Story 4: Philip is a coffee shop owner working in a fast-paced environment. He likes to discover events in broadcast news streams. However, major events can easily span over time, resulting in high time consumption for identifying and tracking news stories. Therefore, he would like to use a product that can provide him past and present news stories in an accessible format.

User Story 5: Mary is a professor who works in the University Administration Office. She likes to get a quick understanding of popular topics during her breakfast and subscribes to the topics she is interested in. When Mary has enough time, she will read more detailed news information about these topics. Therefore, a website that can remember favourite topics would be more suitable for her.

2. Technical Problem

2.1 Motivation

Solutions to information overload, like its causes, are multi-faceted, and there is no single tool or technique that will correct the problem. Therefore, this project intends to help online news reader deal with the ever-increasing problem of information overload by delivering a news aggregator that combines interactive timeline visualisation with text mining techniques. The timeline visualisation facilitates in providing a user-friendly environment where current events are displayed in the context of the past. This interactive approach is considered as one of the most intuitive and effective methods for tracking long-lasting news stories (Tran, Giang, M. Alrifai, & E. Herder, 2015). The most representative news articles that are collected from several mainstream news providers are filtered and selected via the clustering algorithm. As constructing and updating timelines manually is time and effort consuming, it is necessary to provide a novel approach that generates and updates timelines from a huge number of news articles automatically.

2.2 Similar Products

2.2.1 News Deeply

News Deeply provides independent digital media projects that explore a new model of storytelling around a global crisis. However, News Deeply focuses only on a fixed number of static topics, which might miss the popular topics that the public is interested in. In addition, users of News Deeply can receive weekly updates, special reports and featured insights if they sign up for newsletter, but they can't subscribe interested topics.

Figure 2.1 screenshots of News Deeply

2.2.2 Timeline App

The Timeline App aims to provide media-rich content that connects past to present by displaying the overview of related historic events, and users can subscribe topics they are interested in. Nevertheless, this website is a platform where users can write their own stories. In other words, timeline app's data are user generated content which is very different from our website.

Figure 2.2 screenshots of Timeline App

2.2.3 Google News

Google News is a news aggregator that delivers comprehensive and up-to-date news coverage and displays the most popular news stories around the world. However, Google News mainly focuses on the latest news rather than the full coverage of a story. If users want to get more detailed information, they still need to search by themselves. In addition, although the enriched content in homepage provides a wide choice for consumers, it could also aggravate the level of news fatigue at the same time.

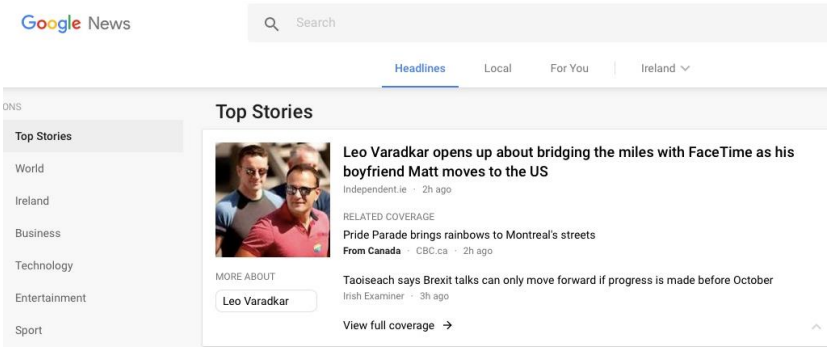


Figure 2.3 screenshots of Google News

The table below features a comparison between Grape News and the above-mentioned websites, which lists the reasons why Grape News might be a better choice for our target user.

	Topic Dynamic Update	Timeline Auto Generation	Topic Subscription
News Deeply	×	×	√
Google News	√	×	√
Grape News	√	√	√
Timeline App	√	×	√

Table 2.1 Similar products comparison

2.3 Core Technical Problems

From what have been discussed above, our project try to help users understand how news stories develop with the time passing. The system is supposed to be able to collect popular topics and relevant news articles from mainstream news providers. Meanwhile, the most representative news should be selected for users. In addition, the timeline infographic could help users read faster, understand more, and keep focused for longer. (Sadhu, 2017).

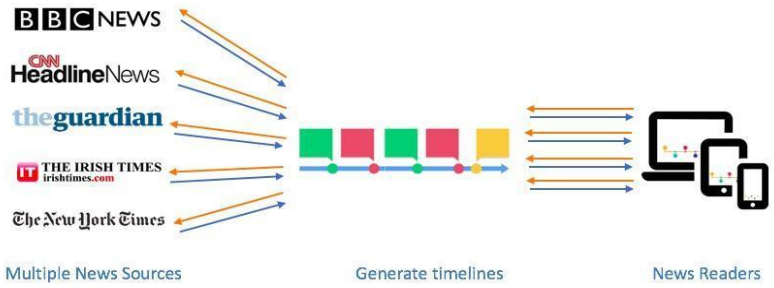


Figure 2.4 Idea of Grape News

Providing popular topics with the salient news articles in timeline format could encounter the following problems:

2.3.1 Front-end

There are three mainly challenge about front-end.

a. How to achieve responsive Web design?

The internet users are shifting their reading habit from computer to mobile or tablet (Connect, 2017). Therefore, it is necessary to build a website that satisfies different device screens. The traditional adaptive layout design is effective, yet not efficient. We are looking for a solution that can write once and suit all.

b. How to present the data in a timeline format nicely?

The way of presenting data can influence reading efficiency (Agrawal, 2017). Simply listing all news articles lacks user interaction and may lead to the interest losing. Giving each news articles a distinct timestamp can help user build a clear idea of when and how it happened. There are many ways to format news need, the challenge is how to display the data nicely in a chronological order.

c. How to design user authentication?

Building a user authentication system and support subscription is a way of holding user loyalty. However, there are many ways to do it, such as Django Oath toolkit, HTTP Signature Authentication. Considering front end is a single page application, which means the page will be rebooted once it gets refreshed, we are looking for a solution that not only suits for back end, but also compatible for single page application.

2.3.2 Back-end

There are three main challenges in the back-end.

a. How to define and retrieve topics dynamically?

Giving a definition of topics is not an easy task. If the topic is defined too specific, we can't find suitable news articles to construct a timeline. If the topic is too general, enormous relevant articles might be stacked on the timeline, which can also increase the levels of news fatigue. Moreover, topics need to be updated in real time for providing users the latest news information. Therefore, we are looking for a solution that can provide appropriate and dynamic trending topics.

b. How to collect past and present news articles from multiple news platforms?

To put current news in the context of past, it is crucial to extract historical and present coverages for the specific topic. The traditional approach of retrieving news is utilizing news APIs, whereas most of the news API has been officially deprecated, such as Google News Search API, and other free APIs tend to have daily request limit. When exceeding the rate limit per day, it will return a "too many requests" response code. Considering such circumstances, we are trying to find out a better solution for retrieving previous and present news articles.

c. How to select the most representative news articles?

Since our news data sources provide numerous news data, it is understandable to have a large number of similar or unimportant contents that report on the specific topic. However, displaying all collected news articles in the timeline still not well resolved the problem. It is necessary to propose a solution to pick out the most representative news articles for users.

3. Technical Solution

3.1 System Functionality and User Interface

Grape news is built to help online news readers get the big picture of a certain topic and to mitigate the information overload problem to a certain extent.

The website consists of eight pages: homepage, timeline page, search page, user setting page, login page, signup page, about us page and topic list page. Due to the length limit, only main pages and key functions are introduced in the report.

1. **Homepage**: Browse latest topics with their recent news

Navigation bar (1) Image slider (2) Topics cards (3).

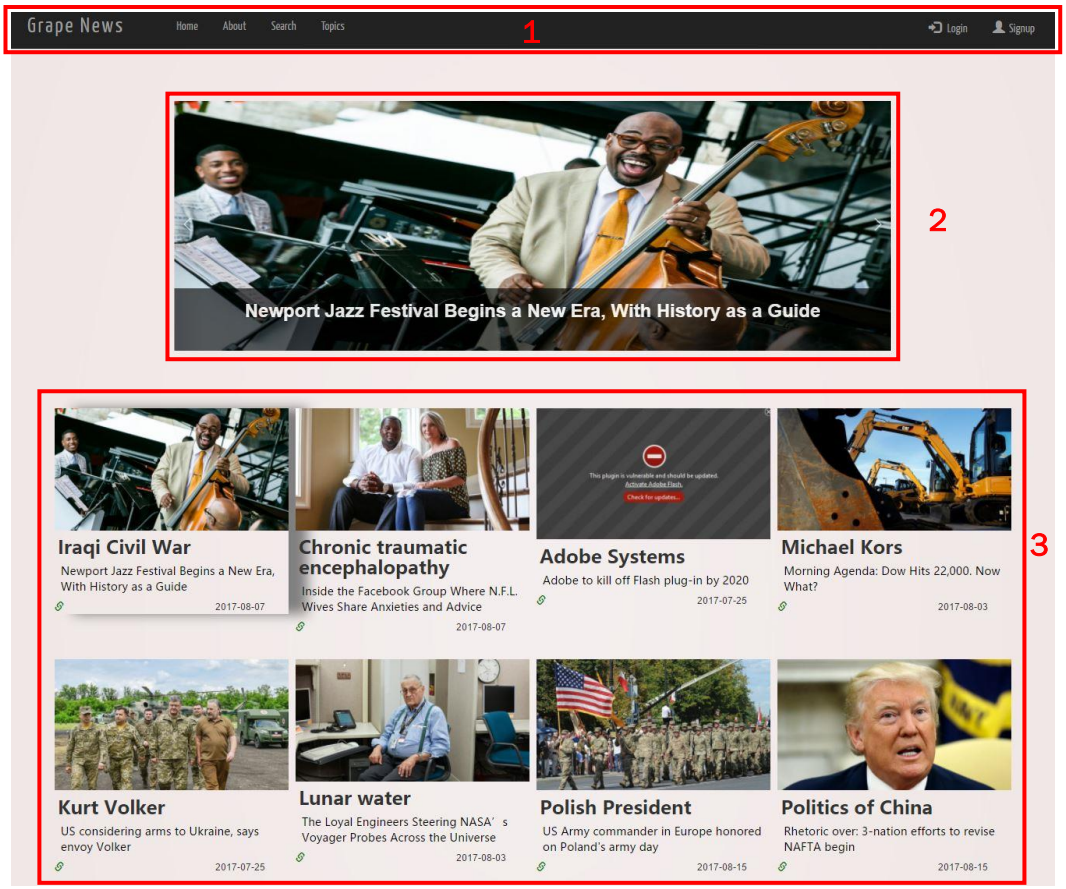


Figure 3.1 Screen shot of homepage

(1). The Navigation Bar provides main pages' link to guide users.

(2). The Image Slider provides three latest topics on the website. Clicking on the images could lead to the specified topic timeline page.

(3). The Topics Cards could be loaded automatically if the user scroll down, which provide topics' name and their latest news information. Clicking on the image could lead to the timeline page of the specified topic.

2. **Timeline page:** Reading relevant news' summarization and subscribing the topic

- Topic bar (1), Content Presenter (2), Timeline Panel (3), Buttons (4)

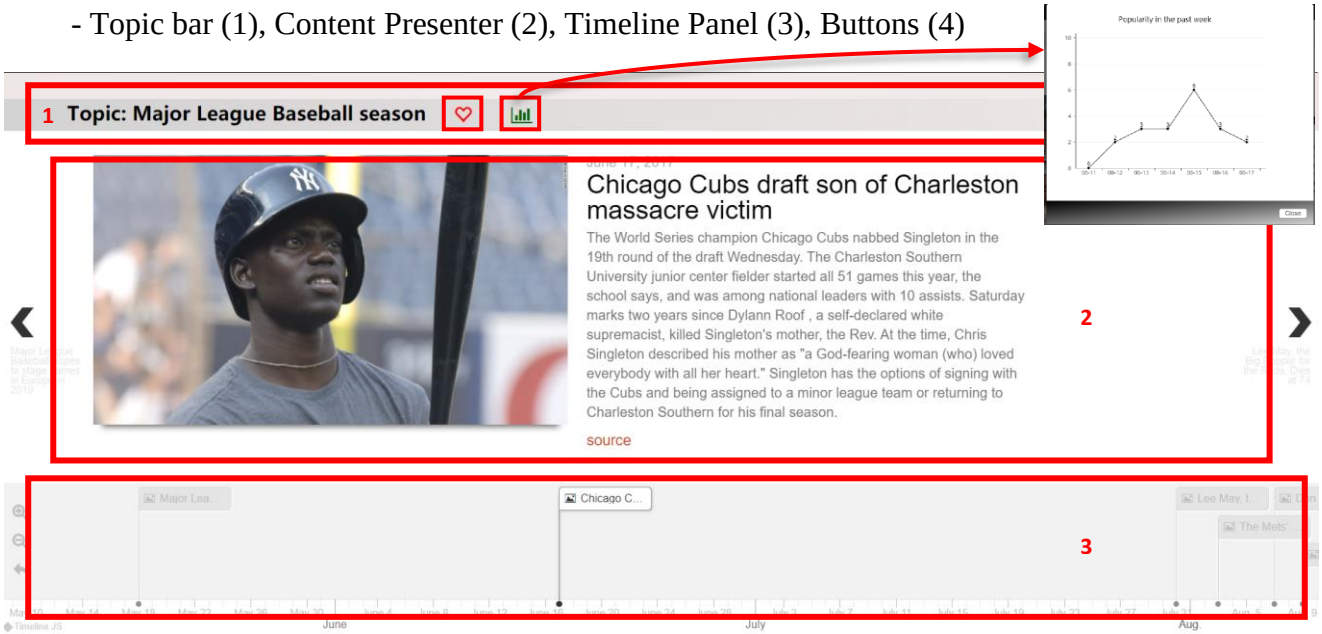


Figure 3.2 Timeline Page

(1) Topic bar, the top grey topic bar is used to display the topic name and provide the topic popularity in the past week by click green chart button. In addition, if the user login the system, a red heart button would show up to support subscription function.

(2) Content presenter, this part provides a summarization of the original news. The user could also click on the red "source" link to check original news page.

(3) Timeline panel, the bottom panel provides a timeline scaler which allows the user to zoom in or out to see the articles thumbnails in different granularity. Along with the switchover in the panel, the content presenter would display different articles accordingly.

3. **User setting page** – Modify personal information and unsubscribe topics

- User Profile block (1), Notification (2)

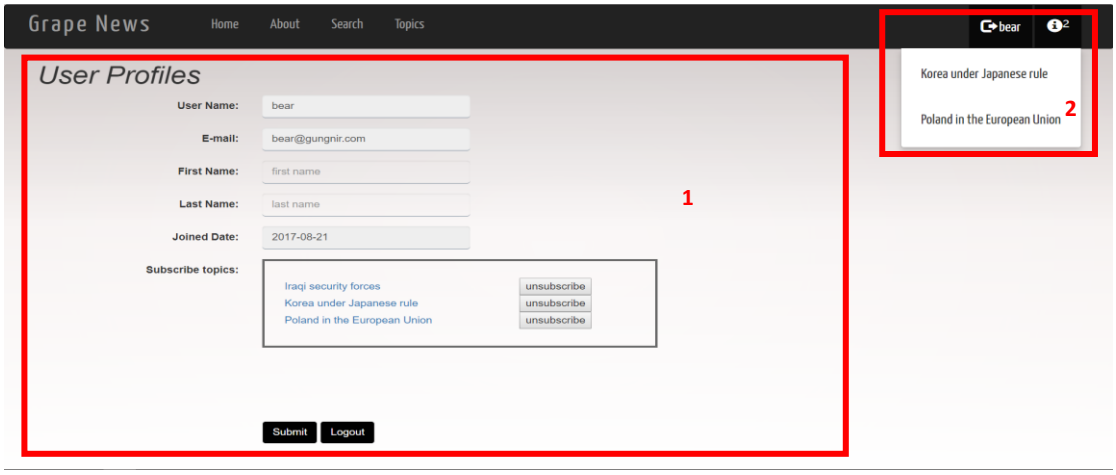


Figure 3.3 User setting Page

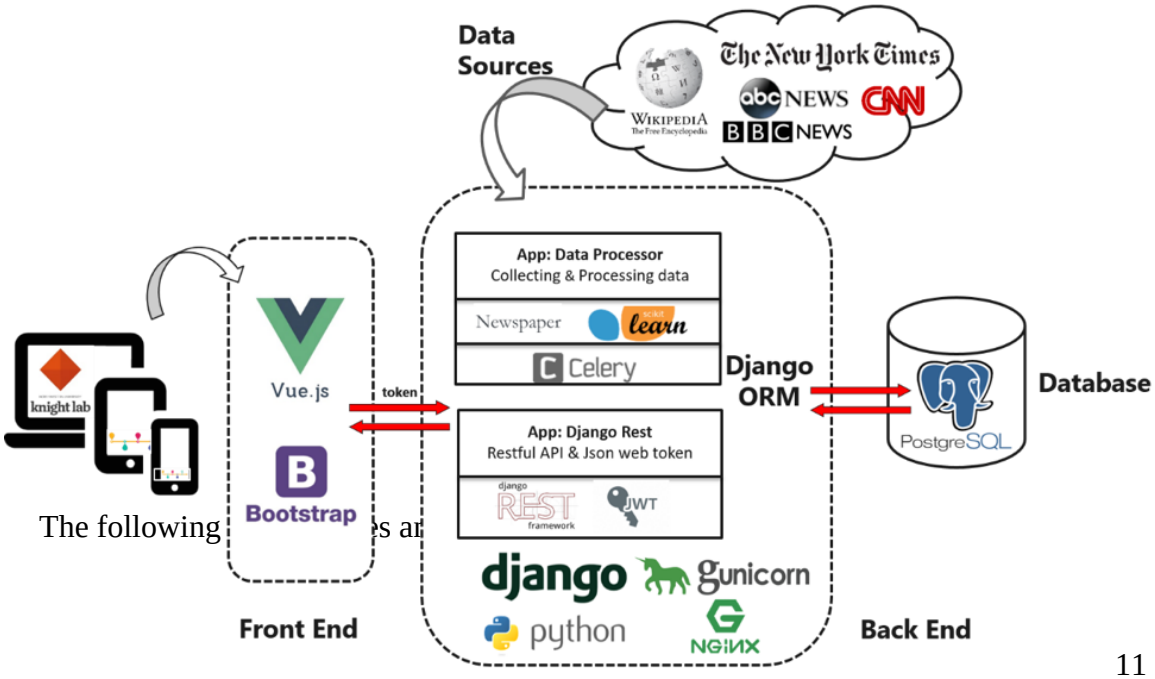
(1) Profile Block, when user login the system, they will be able to access user setting page by clicking their name in right corner of navigation pane. The user could modify their first name, last name and subscript name in this page.

(2) Notification, once the user subscribed topics have any update, the exclamation mark button would show up besides of user name. The updated topics details can be checked by clicking the exclamation mark button.

3.2 System Architecture

The graph below demonstrates the project architecture.

Figure 3.4 Overall architecture graph



The following

s ar

Front-end	
Vue.js	Vue.js is a progressive JavaScript framework that is able to build reactive web interface along with numerous features like two ways data binding and virtual DOM.
Bootstrap	Bootstrap is the most popular framework for developing responsive, mobile first projects on the web (Mark Otto, Jacob Thornton, 2017).
TimelineJS3	It is an open source, highly customizable library that can generate interactive timeline and support rich media
Back-end	
Postgres Database	The world's most advanced open source database
Django Framework	A free and open-source web framework following the model-view-template(MVT) architectural pattern
Django Rest	Representational state transfer (RESTful) web services providing interoperability between computer systems on the Internet.
Celery	Task queue that focuses on real-time processing and task scheduling
Newspaper	A Python library for summarizing and extracting news content, images and URLs
Scikit Learn	A Python library for data mining and data analysis
JWT	JSON Web Token used for token-based authentication
Deploying environment	
GUnicorn	A Python Web Server Gateway Interface (WSGI) HTTP server
Nginx	A web server used as a reverse proxy, load balancer and HTTP cache

Table 3.1 Technology stack

3.3 Frontend Components

The front-end is mainly built upon Vue.js and Bootstrap two frameworks.

3.3.1 Vue.js

In the times of rapid web app development, JavaScript web frameworks can become a silver bullet. There are various existing JavaScript frameworks such as Angular, React and Vue.js. The reason Vue.js has been chosen in this project are listed below:

- **Learning Curve**

Before using React, it is necessary to learn JSX, ES2015+, and the user even needs to be familiar with systems building. In regards of Angular, its learning curve is even steeper. Getting started with its basic functionality is easy, however, to be productive

in a slightly more complex environment, it requires a deep understanding of Angular's inner workings such as dependency injection, controller and injector (Vuejs.org, 2017).

Considering the prebuilt data binding and virtual DOM, together with the powerful scaffold: Vue-cli, Vue is much easier to set up a project compared to React and Angular (Hacker Noon, 2017).

- **Size and performance**

Frameworks	Angular 2	React	Vue
Definition	MVC Framework	JavaScript Library	MVC Framework
Size	144 kB (minified & compressed)	142 kB (minified)	26 kB (minified & gzipped)

Table 3.2 JavaScript Frameworks comparison

In regards of the performance, vue.js is slightly ahead among the three due to its light weight library (Vuejs.org, 2017).

3.3.2 Bootstrap

Another challenge that frontend development has is to ensure the web pages look good on all devices. Bootstrap is modular and includes fewer stylesheets and provides predefined components for developers to use or revise. The reasons why bootstrap has been chosen are listed below.

- **Responsive**

Features	Adaptive Web design	Responsive Web design
To execute	Easy	Hard
Flexibility	Less	More
Loading time	Slow	Fast

Table 3.3 Compare with Adaptive Web design

Responsive Web design is more flexible than Adaptive web design (Matthew Harris,2015). RWD can promise the layout runs nicely on any screen size. It also requires less loading time compared to Adaptive Web Design. In addition, Responsive Sites only need one layout to work properly across different platforms, while adaptive design needs to list all possible layouts.

- Advantages of Bootstrap comparing with Foundation (Codementor.io, 2017)
    - Pre-defined components
    - Better browsers compatibility
    - Provide less and sass CSS process

3.4 Back-end Components

3.4.1 Django Framework

Django is a high-level server-side web framework for developing web applications (Djangoproject.com, 2017). The main reason Django Framework has been chosen is that it emphasizes reusability, components pluggability and rapid development. Django also provides various third-party libraries, which can help developers build websites conveniently. Compared with another model–view–controller (MVC) framework Flask, Django has a better supported public community (Home, Development and Framework, 2017).

3.4.2 RESTful APIs

RESTful API is applied in this project because it can serialize data into standard JSON format, which is convenient for frontend interaction (Anon, 2017). Besides, Django REST framework provides a highly integrated Viewset model that provides handy features such as GET List and Retrieve object. Apart from that, RESTful APIs make it possible to develop backend and frontend independently.

3.4.3 User Authentication

The user authentication is utilized Json Web Token, so users who sign in with correct username and password are able to hold a unique token to access protected service, such as personal information modification (Jwt.io, 2017). Compared to other user authentication methods, JWT authentication is fast and can help maintain user status even for a single page application.

3.4.4 Celery

The project requires a role that can schedule tasks periodically, and that is where Celery comes into play. Celery provides full-featured task scheduling function and is easy to

use and maintain. If the connection is lost or failed, the worker and client will retry automatically. A single Celery process can handle millions of tasks per minute, while the round-trip latency is maintained in sub-millisecond. Therefore, Celery is treated as task scheduler in this project (Docs.celeryproject.org, 2017).

3.4.5 Database

PostgreSQL is a powerful open source object-relational database system, which can store text data without considering its length. This feature is particularly useful in this project as the length of each news article is not same. Besides, PostgreSQL supports the storage type with an array and JSON formats (Postgresql.org, 2017).

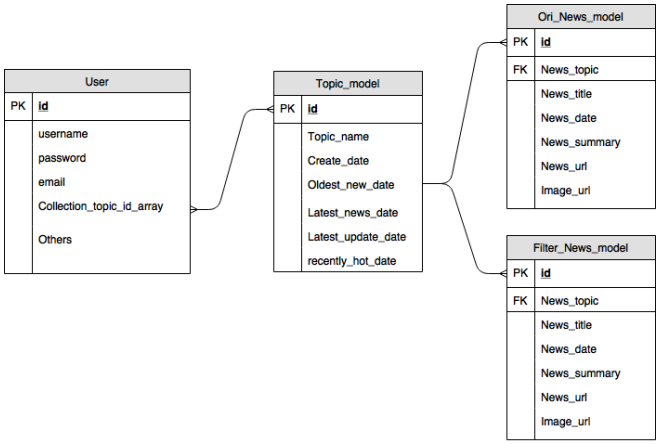


Figure 3.5 Database tables

There are 4 tables used for storing information in the database: User_info_model, Topic_model, Ori_News_model, Filter_News_model, which are used for storing users' personal information, topics details and news details, such as links and contents.

3.5 Data Stack

The main steps of data stack are shown in below Figure 3.6:

1. Extracting topics from MediaWiki API
2. Processing topics and storing processed topics into Topic Table
3. Extracting processed topics and using as keywords to generate search queries
4. Retrieving relevant news articles from four web sources
5. Pre-filtering irrelevant news using Keywords Matching strategy and storing pre-processed news data into Original News Table

6. Getting original news data from database and processing news to keep the most representative articles via K-Means Clustering algorithm
7. Inserting processed news data into Processed News Table
8. Retrieving requested data using Django ORM
9. Using Django Rest Framework to provide corresponding data through Restful API

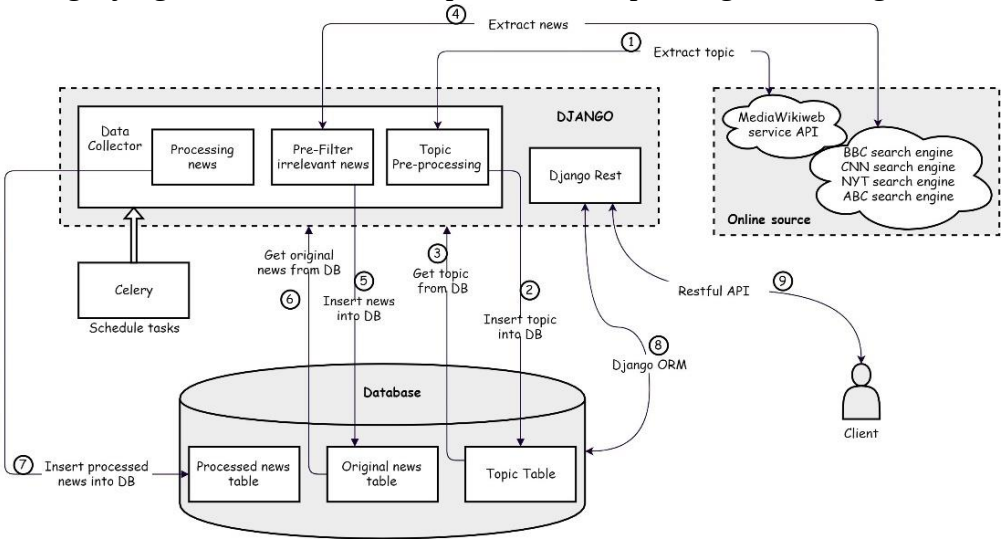


Figure 3.6 Data Flow Diagram

3.5.1 Topic Collection

MediaWiki web service API is utilized for retrieving daily topics from Wikipedia's Current Events portal (WCEP). WCEP is a platform for archiving main events and trends which are manually provided by website editors on a daily basis (Hienert, Daniel, & F. Luciano, 2012). By looking at the categories of the events, we found out that there is a wide coverage of topics although most of the reported news events belong to the armed conflicts, politics, crimes and natural disasters categories, while less events have been reported in the science and arts categories. These topics are just good fits for forming news timelines. Furthermore, an obvious advantage of MediaWiki API is that it organizes major events information in a combine of topics and time units (i.e. days), and the results are stored in JSON format which can be easily processed further.

In fact, Google trends was tried to retrieve daily hot topics at first, but the problem is that the trends provided by Google are too specific. Using these trends as keywords for retrieving news can only get few relevant articles. Therefore, we finally decide to retrieve trending topics from MediaWiki web API.

3.5.2 Topic Processing

Figure 3.7 is an example of four topics got from WCEP on 12 August. To improve the retrieving accuracy, we remove "2017", useless content in brackets and punctuations, such as "-" and "/". The word number of a topic is also limited to improve the quality of topics. If a topic only contains one word, we would not adopt it as a usable topic. These processed topics are stored in the database (Topic Table), which can then be used as keywords for searching relevant news articles. In addition, stop-words are removed before retrieving news articles, such as "in", "of", "the", etc. As topics are updated in real time, we crawl and process topics four times a day.

Figure 3.7 Screen shot of Wiki current events

3.5.3 News Collection

a. Retrieving news data

After acquiring topics, relevant news articles that are connected to a certain topic should be collected. As most of available APIs have rate limits, we decide to write own web Scraper for extracting news data from original news websites. Four mainstream news providers (BBC News, CNN News, ABC News and NYT News) are used as main data sources. The topics are treated as keywords to generate search query for retrieving news articles, and these search queries are written separately for four different news websites. As a news story tends to span a long period, we retrieve 60 days' news articles. Meanwhile, the Scraper can parse the HTML or JSON response and extract corresponding news publish time and news URLs for later use.

Since we want to collect news data incrementally, previous and present news data is retrieved only for the first time and the updated data will be grabbed four times a day.

In our case, we compare the publish time of updated news and last update time under the same topic to identify whether the new data is valid updates. News URLs are also extracted and stored for the newspaper Python library to generate summaries.

b. Pre-processing news data

One central issue of our information retrieval approach is the relevancy between retrieved articles and the topic used as search query. Traditionally, relevancy of web pages can be calculated with the use of similarity measures (Usharani, J., & D. K. Iyakutti, 2013). However, we tried to calculate cosine similarity, but the result doesn't go well. To address this issue, we came up with a strategy called "Keywords Matching". The main steps of this strategy are shown in Figure 3.8.

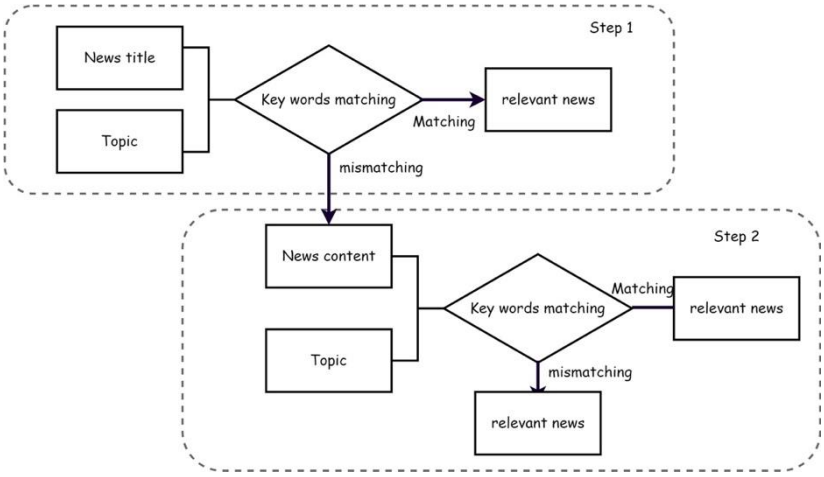


Figure 3.8 Keywords Matching steps

Step 1:

News title and topics are compared after tokenization and removing stop-words. If the topic tokens all appear in the news title, the news is considered as relevant news. If not, we will move on to step 2.

Step 2:

Similarly, we compare the news content with the topic tokens. If the word number of topic tokens is equal or less than 4, all topics tokens should appear in the news content. If topic tokens number is more than 4, only 4 topics token that all appear in news text would be considered as relevant news. Plus, this threshold is set according to the empirical analysis.

3.5.4 News Processing

For each topic, the amount of news might be tremendous, thus how to pick out the essential information has been the most tough challenge for us. We tried SimHash algorithm which is applied for identifying near-duplicate web pages, but this approach doesn't perform well when the number of news articles increases. Therefore, we utilize the K-Means clustering algorithm to extract the representative news, which is defined as the document whose feature vector is the closet to the centroid of all documents in the feature vector space (Kimura J, Yoshitomi Y & Tabuse M, 2015).

K-Means clustering is the common unsupervised learning algorithm, where data is organized into clusters in the absence of any external information, only relying on the data itself (Kanungo T, Mount D M, Netanyahu N S, 2002). In our case, the clustering algorithm is implemented using the Scikit-learn library. The steps of K-Means clustering are summarized as follows:

1. **Initialization**: Select k initial cluster centroids at random.
2. **Assignment step**: Assign every item to its nearest cluster centroid using Euclidean distance.
3. **Update step**: Recompute the centroids of the clusters based on the new cluster assignments, where a centroid is the mean point of its cluster.
4. Go back to Step 2, until a maximum number of iterations (set as 10) is reached.

The biggest challenge of applying K-Mean clustering algorithm is to determine proper key input parameter k – the number of clusters. If k is too low, clusters that should not be merged might get smeared. If k is too high, the data would be divided into many small and similar clusters because of over clustering.

Our solution is to apply Elbow method to find an optimum k value. We tried to use silhouette coefficient, but the performance is not good as expected. Elbow method is an approach which looks at the percentage of variance explained as a function of the number of clusters (Bholowalia P & Kumar A, 2014). The traditional way of utilizing Elbow method is to increment the value of k and find out the appropriate k value manually, which requires a huge amount of human effort. Therefore, we utilize a method to apply Elbow method automatically with the help of linear regression (Rencher and Christensen, 2013). The furthest point towards the linear regression line

and laying above the line is defined as the elbow point. Additionally, Principal component analysis (PCA) is also used to reduce the dimensionality for better visuals of distribution. Figure 3.9 provides three examples of the performance of Elbow method.

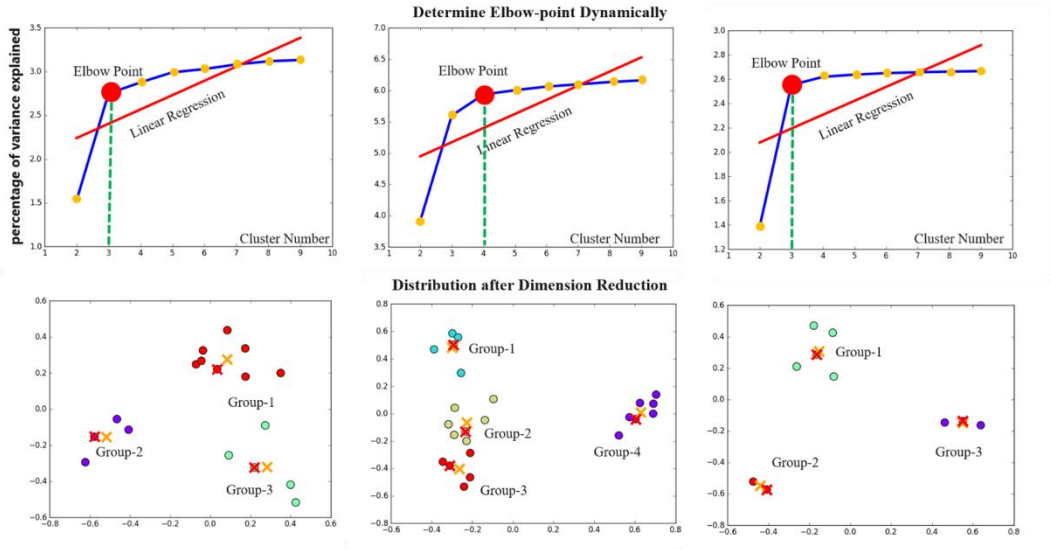


Figure 3.9 Performance of Elbow method

It can be shown from the figure that, the chart at the top left shows that the optimum k value determined by Elbow method is 3, and the distribution figure below demonstrates that 3 is the most appropriate k value. The remaining figures also illustrate that Elbow method has a good performance in finding out the proper k.

After dealing with the k value challenge, we apply K-Means clustering for daily news under the same topic. Since we found out that in most cases, similar news coverages from different news providers tend to appear only on the first publish day, so only the news articles shared with the same topic and published on the same day are clustered. As discussed above, only one news that is the closet to the centroid in each cluster is selected as the most representative article.

Additionally, there might be a small number of the daily news articles under some topics. Based on our experiments, implementing K-Means clustering to few number of news is no longer effective. In this case, we calculate cosine similarity when daily news number is less than 5. If the cosine similarity is less than the threshold that is empirically set, the latter news would be removed, if not, the news should be keep. All

the processed news data are stored into Processed News Table. The news processing steps are shown in Figure 3.10.

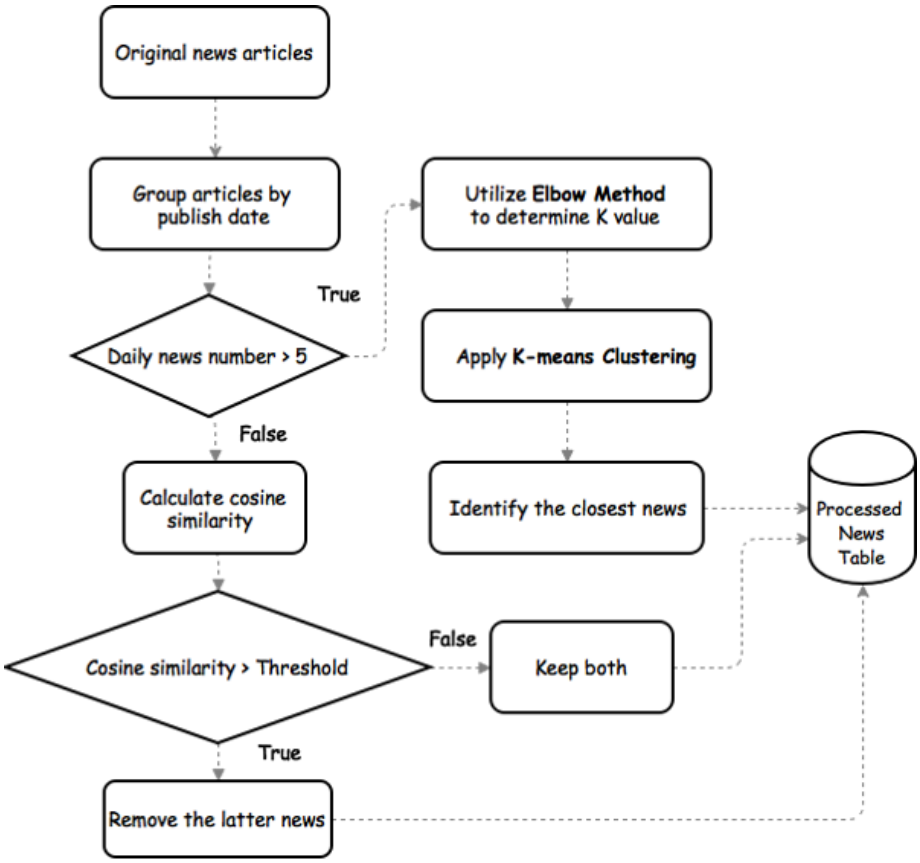
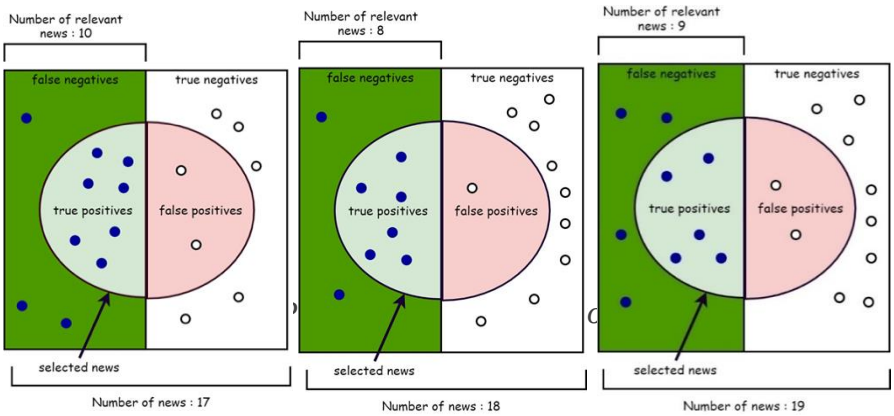


Figure 3.10 News processing steps

We reviewed and assessed the performance of our news processing strategy, and the results are shown in graphs and tables below. This method successfully removed some unrelated news and stored most useful news into database. As shown in the table, the rates of precision and accuracy are relatively high, all their statistics are higher than 70%. However, the performance of recall was not as good as others, which reminds more effort needed in next release.



Topic	A	B	C
Name	North Korea and the United Nations	Syrian civil war	G20 Hamburg summit
TP	7	6	5
TN	5	9	8
FP	2	1	2
FN	3	2	4
Precision	77.78%	85.71%	71.43%
Recall	70.00%	75.00%	55.56%
True negative rate	71.43%	90.00%	80.00%
Accuracy	70.59%	83.33%	68.42%
F1-measure	73.68%	80.00%	62.50%

Table 3.4 The statistic of new processing performance

4. Evaluation

4.1 Proposed Hypothesis

The proposed hypothesis is "Is there a statistically difference in the average completion time and mean accuracy levels between news readers who use Grape News and users using other news delivery platforms (BBC news, CNN news, ABC news and New York Times)".

We have an alternative hypothesis (H_1) and null-hypothesis (H_0).

H_0: Users using Grape News have the same average completion time and accuracy levels as users using other news sources.

H_1: Users using Grape News have higher average completion time and accuracy levels than users using other news sources.

We will test whether the null-hypothesis (H_0) can be rejected by conducting a comparison experiment. The results can show that whether using Grape News could help users gain efficiency and accuracy when acquiring news information.

4.2 Experimental Methods

4.2.1 Overall experimental design

To evaluate the system performance as the whole as well as in many certain aspects, two types of experiments are going to be carried out: verification evaluation experiment and user experience evaluation experiment.

In the verification evaluation section, participants will be given a questionnaire containing a list of multiple choice questions which are related to two different topics. Each topic contains 8 quizzes and answers can be found in the content of both Grape News website and other news sources. Participants should answer these questions and task completion time would be recorded. The ground truth for each quiz is manually provided by the question designers for calculating answering correct number.

The user experience evaluation mainly focuses on whether the designed UI is user friendly or not and acquiring subjective feelings from users. The UI feedbacks are

collected via a set of Likert scaled questions which utilize the traditional 5-level scale. The open question is designed to collect any other suggestions or feedbacks from participates.

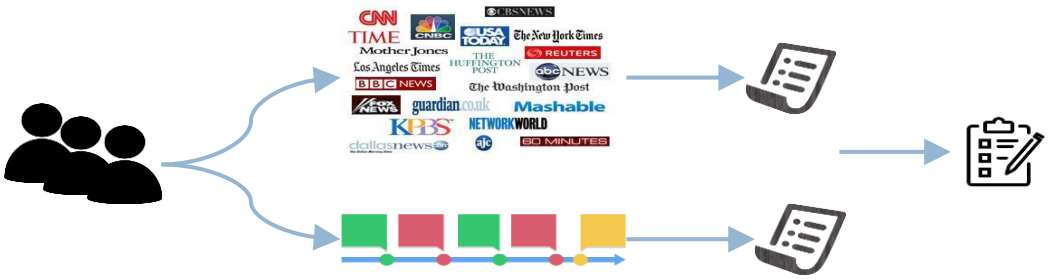


Figure 4.1 The flow of evaluation

Two groups are required to conduct the experiment, one is the experimental group and the other one is the control group. The experimental group uses Grape News website to complete the tasks while the control group uses other news sources. These two groups should finish the tasks under same experimental conditions except whether using Grape News.

In addition, a 4-tester pilot evaluation is conducted before the final evaluation to test the feasibility of overall evaluation plan, designed questions and statistical analysis approach which then can be adjusted.

4.2.2 Measured Variables

In the verification evaluation experiment, the independent variable is defined as whether using Grape News or not, and the dependent variables are the speed of question answering and the correct number of question answering. In addition, there are some other variables that should be kept be the same throughout the experiment, such as room lighting, background noise, temperature, etc. Moreover, to avoid the influence of confounding variables like prior experience, it is necessary to ask these participants first that whether they have known about the topics before. If they have a background about the topics, they would not participate in the evaluation.

Therefore, the evaluation metrics are accuracy and efficiency of tasks answering in the verification evaluation section, and that for user experience evaluation are ratings for user satisfaction, ease of use, intuitiveness, etc.

4.2.3 Selected Subjects

The subjects in this evaluation can be regarded as general news readers. 20 volunteers are planned to be recruited. These participants are divided into two groups by randomization with the same number in each group. These two groups are then randomly decided that one of the groups is experimental group and the other as the control group. Plus, in order to reduce any bias gained by participants' personalities and reading abilities, each group will be treated as the experimental group for one topic. For example, if group A is decided as the experimental group for topic 1, then they should act as the control group for topic 2.

4.2.4 Data Collection

For verification evaluation section, performance will be objectively measured using two metrics: completion time (efficiency) and the correctness of response answers (effectiveness). As there are 8 questions in the questionnaire, to simplify data analysis process, answering one question correctly would earns one point. The perfect score is 8, and the completion time are recorded in seconds.

Subjective measurements are utilized in the user experience evaluation section. The scores of user satisfaction, ease of use and intuitiveness can be also converted to numerical values as follows:

1. Strongly Disagree
2. Disagree
3. Neutral
4. Agree
5. Strongly Agree

As there are 8 Likert scaled questions, we will collect 128 ratings in total.

4.2.5 Data Analysis

After acquiring the raw data, the responses of all participants are treated statistically. Considering the sample size is manageable, mean scores of the correct responses, average completion time and average user experience ratings for each topic in each group are calculated and analyzed manually.

In the first part, the average completion time and correct answering numbers of two groups would be analyzed. It is noted that the data from the unfinished experiments or inaccessible outcomes (the participants don't obey the rules) should be cleaned first. As

two types of data are recorded, the one-dimension data, the grade per second, can be integrated for analyzing the performance of users.

The experimental results are carried out using a paired t-test in R. This test is used when the samples are dependent and can provide an exact test for the equality of means of two normal populations with unknown, but equal, variances (Edgell, Stephen E., & Noon, Sheila M, 1984). The equation of t-test is shown as follows.

$$t = \frac{\overline{X_D} - u_0}{\frac{S_D}{\sqrt{n}}}$$

For this equation, the differences between all pairs can be calculated. The average (X_D) and standard deviation (S_D) of those differences are used in the equation. The constant μ_0 is non-zero because we want to test whether the average of the difference is significantly different from μ_0. This t-test would give an empirical p-value. If the p-value is less than 0.05, which means there is a statistical difference in completion time and response accuracy between two groups, the null-hypothesis would be rejected. These objective results would show that whether our website can have a positive effect on mitigating the problem of information overload.

For the user experience section, the average ratings of each task are calculated. The average scores can illustrate the attitudes of users in many certain aspects of the website. It also can identify potential issues with the feasibility of our product as well as its benefits. These subjective feedbacks can help to improve the user experience in the next release.

4.3 Practical Setup

4.3.1 Pilot Evaluation

The pilot evaluation is an offline setup. 4 volunteer participants were invited to complete a list of tasks in a specific location. Three of them are international students from China, and the other one is an Irish native student. They were given a printed paper contained the task questions and a laptop showing news web pages. Times were recorded using stopwatches. From the process and results got from the pilot evaluation, we found there are much remains to be improved.

a. Selected subjects

There is an obvious difference in reading speed between native speakers and non-native speakers of English. Although we have already considered this factor and take the two groups as an experimental group in turn, non-native speakers still find it a little difficult in answering some certain quizzes. Therefore, we decide to invite more native speakers to participate the final evaluation and delete the questions that are too difficult to find answers for non-native speakers. Moreover, considering the current subjects are all UCD students, it is necessary to invite variant people who have different backgrounds to participate the evaluation. Respondents who mix all kinds of backgrounds could be a better sample of our target user.

b. Evaluation plan

The overall evaluation plan is reasonable and feasible, but it is worth reminding that users should be given enough time to get familiar with Grape News website and other news platforms. It was the first time for those volunteers using our product, so providing a brief introduction might be more helpful. Besides, participates will have less pressure without supervision.

4.3.2 Final Evaluation

The offline experiment was conducted at 2 pm on August 4 in the room B002, CS Building. The room should be quiet and have the comfortable temperature with bright lighting. 20 volunteer participants were invited from other project groups and teammates' friends in advance, but only 17 came and one of them have heard of topic 2 before. Therefore, 16 people finally took part in the final evaluation. The majority of them are UCD students (62.5%) and local residents (18.8%) with age between 22 and 46. Some of them are native speakers of English (37.5%), and the rest of them are Chinese (31.25%), Indians (12.5%), Italy (6.25%), Romania (6.25%) and Slovak (6.25%). Each of the participant were given a printed question sheet that contained task questions and a 13-inch laptop with the same standard screen shown Grape News or other news sources. A brief introduction of our product was given to participants, and they were also given the opportunity to familiarize with other news sources.

To make sure that every participant has an isolated experience, they were sitting in different corners of the classroom and are not disturbed by others. The instruction is

same for both groups except whether using Grape News. The detailed evaluation questionnaire can be found in Appendix. All the questionnaires were collected for further data analysis.

4.4 Experimental Results

In the evaluation verification section, the experimental results are analyzed via R. The figures below show the detailed comparisons between experimental group and control group in grades, time cost and grade per second used for t test.

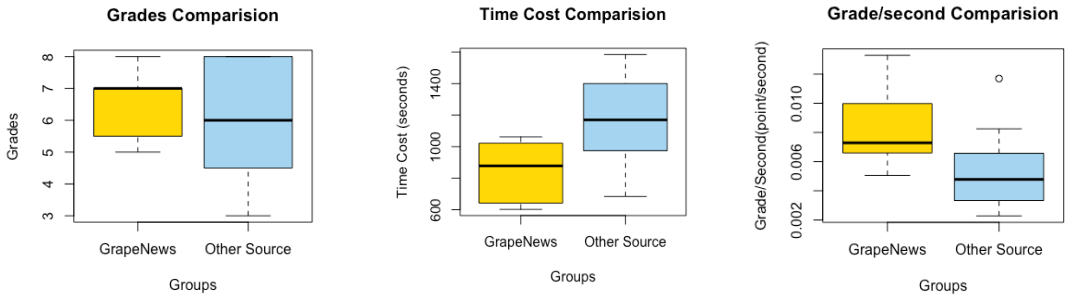


Figure 4.2 Statistics of evaluation

The first two figure illustrate that users who use this system can earn higher grades with less time cost than using other news source websites. The third chart shows experimental group has better performance than control group from the perspective of average grade per second, which also proves the previous results.

After calculating the collected data, the t value is 2.71 which is between 3.106 and 2.201 (the 95% confidence interval). This result explains that the two groups are statistical different although the difference is not significant. Therefore, we reject the hypothesis H_0 and accept H_1: users who use Grape News have higher average completion time and accuracy levels than users using other news sources.

In the user experience evaluation section, answers to each question are calculated to average values. The results are shown in the Radar Plot below, which describes the rating levels of user satisfaction is scaled from Neutral to Agree. Most of participants are trending to agree with that our system is well performing.

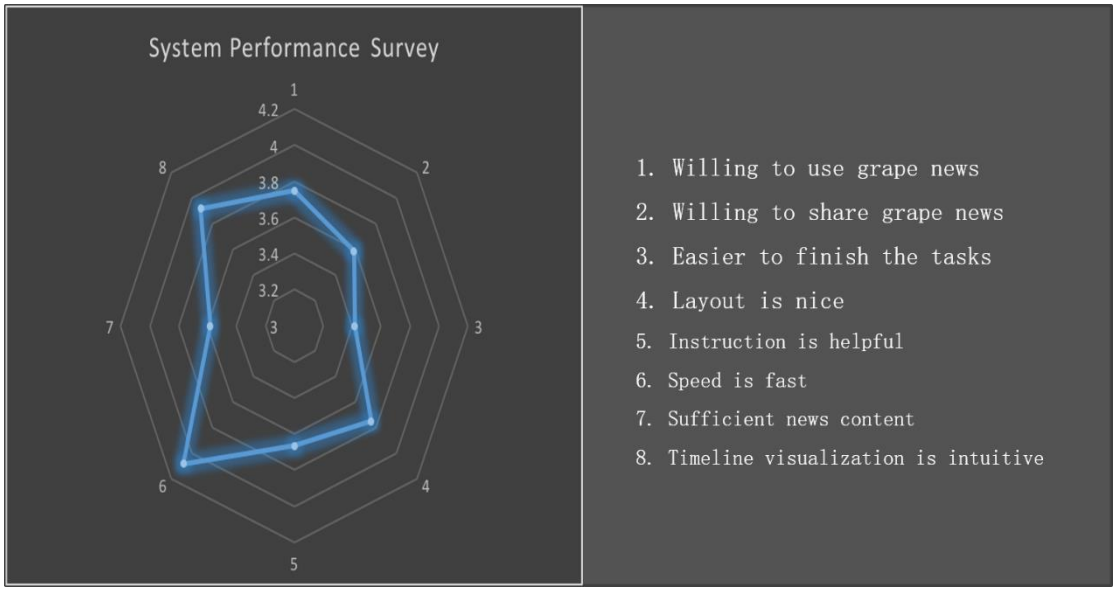


Figure 4.3 Radar graph about users feedback

4.5 Learning from the evaluation

The evaluation results in the verification section demonstrated that the website is useful in making sense of a topic compared with using other platforms. However, from the feedback of piolet and final evaluation, we still found places where needs further improvement:

- Increase the news cards on the homepage
- Button “Show More Topics” could be replaced by auto loading
- Cancel email notification
- The irrelevant news still exits in the topic timeline page
- Add Frequency Chart to show the topic popularity
- Original News could be replaced by a summarization

5. Conclusion

5.1 Project Management Strategy

The project is developed with mixing Scrum and Kanban methods, which also known as Scrumban. The visual features of Kanban method are utilized, as well as the sanctioned nature of Scrum. Trello, a handy project management tool, is represented for Kanban to organize the plan and keep track of the processes.

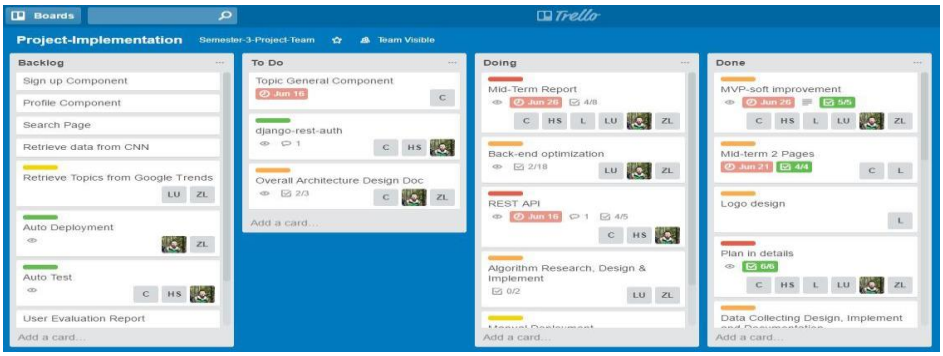


Figure 5.1 Trello screenshot

Some practices that the project has benefited from are listed below

Pair coding - The group members are divided into three pairs: front end, data collection & processes and database & interaction**.**

Story-driven - Front-end is developed, driven by previously defined user stories.

Test-driven - Database development has followed TTD methodology, which defines acceptance test first and then develops the functions to meet the acceptance.

Extreme programming - The rest of team are using this methodology to ensure that the quality of the system is maintained and to continuous improve the website.

Communication – According to Agilemanifesto (2017), "the most efficient and effective method of conveying information to and within a development team is face-to-face conversation". The stand-up meetings are conducted every 2 or 3 days through online video conference (Google Hangouts) and offline group meetings.

5.2 Reflections

5.2.1 Challenges

This team has encountered three main technical challenges throughout the website development.

Determining and collecting data are is the first challenge. Historical news data is a protected by most news organization, so it was difficult for the group to collect data efficiently in the beginning. This challenge has been overcome by shifting collecting method from API to Scraper. Scrapping news data from different news websites was one of the difficulties, particularly website structures are different from each other. Though it was a complex process, the system now is able to decipher searching result, generate HTTP request and extract useful information for each website.

We hope that the website would provide salient articles so that the user could spend less time on searching news and reading redundant content. Therefore, the second technical challenge was how to select an effective algorithm that would most represent most of the news data. K-means is one of the simplest and most efficient algorithms to cluster a large data set. Determining k value dynamically is another problem that troubled the team for a while. After research and discussion, the problem was resolved by utilizing the elbow method.

Another challenge the group encountered was how to present data efficiently to the user. The user's attention span is getting shorter under information overload. With so many different types of infographics available, it is important to ensure that an appropriate type is chosen to effectively present and communicate the information to the user. After research and trials, the website currently is presenting data with a distinct timestamp in a chronological order by using a JavaScript library timelineJS3.

5.2.2 System Review

According to the feedback of user evaluation and the group members' self-evaluation, the strength and weakness of this website are listed below:

Strength:

- Retrieve topics and related news automatically
- Timeline Visualization
- Tracking topic updates
- Generally friendly user interface

Weakness:

- Time consuming data collection

- Irrelevant news existence

- Incomplete user validation system

5.3 Lessons Learned and Future Works

We have learned a number of lessons through developing this website:

Soft skills

- Team working skills

- Agile development methodology

- Presentation skills

- Respect and learn from each other

Hard skills

- Programming Languages: Python, JavaScript

- Frameworks/Libraries: Django, Django Rest Framework, Vue.js, Bootstrap, Celery, Scikit Learn, TimelineJS3, etc.

- Tools: Git, Pycharm, Trello

In regards to potential directions for future work, firstly, we hope to refine current algorithms to ensure the articles' relatedness as well as its representativeness along with automatic timeline generation based on customized topics.

In next phase, considering the limits of linear presentation, we hope to create metro map of information. Metro map is able to demonstrate the relations among retrieved data as well as recording story development, which could further help users cope with information overload (Shahaf, D., Guestrin, C. and Horvitz, E., 2012).

6. Reference

Allan J (ed.). Topic detection and tracking: event-based information organization. Norwell, MA: Kluwer Aca- demic Publishers, 2002.

Alonso, O., Fetterly, D. and Manasse, M. (2013). Duplicate News Story Detection Revisited. Information Retrieval Technology, pp.203-214.

A. E. Holton, and H. I. Chyi. "News and the overloaded consumer: factors influencing information overload among news consumers." *Cyberpsychology Behavior & Social Networking* 15.11(2012):619.

Agilemanifesto.org. (2017). Principles behind the Agile Manifesto. [online] Available at: http://agilemanifesto.org/principles.html [Accessed 25 Jun. 2017].

Agrawal, V. and Agrawal, V. (2017). The Seven Different Types of Infographics and When to Use Them. [online] Maximize Social Business. Available at: https://maximizesocialbusiness.com/the-different-types-of-infographics-and-when-to-use-them-25068/ [Accessed 21 Aug. 2017].

Anon, (2017). [online] Available at: (http://www.django-rest-framework.org [Accessed 21 Aug. 2017].

Bholowalia P, Kumar A. EBK-Means: A Clustering Technique based on Elbow Method and K-Means in WSN[J]. International Journal of Computer Applications, 2014.

Connect. (2017). The importance of responsive design - Connect. [online] Available at: https://www.connectinternetsolutions.com/responsive-design/ [Accessed 21 Aug. 2017].

Djangoproject.com. (2017). *Django overview | Django*. [online] Available at: https://www.djangoproject.com/start/overview/ [Accessed 21 Aug. 2017].

Edgell, Stephen E., & Noon, Sheila M (1984). "Effect of violation of normality on the t test of the correlation coefficient". Psychological Bulletin. 95 (3): 576–583.

Goutte, C., Toft, P., Rostrup, E., Nielsen, F. and Hansen, L. (1999). On Clustering fMRI Time Series. NeuroImage, 9(3), pp.298-310.

Holton, A. E., and H. I. Chyi. "News and the overloaded consumer: factors influencing information overload among news consumers." Cyberpsychology Behavior & Social Networking 15.11(2012):619.

Hienert, Daniel, and F. Luciano. "Extraction of Historical Events from Wikipedia." 7540(2012):16-28.

Hacker Noon. (2017). Angular vs React—the DEAL BREAKER – Hacker Noon. [online] Available at: https://hackernoon.com/angular-vs-react-the-deal-breaker-7d76c04496bc [Accessed 2 Jul. 2017].

Home, H., Development, W. and Framework, D. (2017). What is Django? Python's Framework -Django Programming, Django CMS. [online] Hiring | Upwork. Available at: https://www.upwork.com/hiring/development/django-programming/ [Accessed 21 Aug. 2017].

Jwt.io. (2017). JWT.IO. [online] Available at: https://jwt.io [Accessed 21 Aug. 2017].

Kanungo T, Mount D M, Netanyahu N S, et al. An Efficient k-Means Clustering Algorithm: Analysis and Implementation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2002, 24(7):881-892.

Keim, Daniel A. "Incremental visual text analytics of news story development." Information Visualization 12.3-4(2012):308-323.

KETCHEN Jr., D. and SHOOK, C. (1996). THE APPLICATION OF CLUSTER ANALYSIS IN STRATEGIC MANAGEMENT RESEARCH: AN ANALYSIS AND CRITIQUE. Strategic Management Journal, 17(6), pp.441-458.

Kimura J, Yoshitomi Y, Tabuse M. Classification of Japanese Documents and Ranking of Representative Documents by Using the Characteristic of the Frequencies of Words[J]. 2015, 2(3):182.

Matthew Harris (2015). The Medium Well. Available athttp://mediumwell.com/responsive-adaptive-mobile/(Accessed at 25 June 2017)

Mark Otto, Jacob Thornton (2017). Bootstrap. Available at http://getbootstrap.com/(Accessed at 25 June 2017)

M. Najm-Araghi. "Story Tracker: Incremental visual text analytics of news story development." Inf*ormation Visualization* 12.3-4(2013):308-323.

Nordenson B. Overload! Journalism's battle for relevance in an age of too much information. Columbia Journalism Re- view (November/December) 2008; 47:30–42.

Postgresql.org. (2017). PostgreSQL: About. [online] Available at: https://www.postgresql.org/about/ [Accessed 2 Jul. 2017].

Rousseeuw, P. (1984). Silhouettes : a graphical aid to the interpretation and validation of cluster analysis. Delft: Delft University of Technology

Rencher, A. and Christensen, W. (2013). Methods of multivariate analysis. Hoboken, N.J.: Wiley.

Sadhu, R. (2017). 8 Project Timeline Tools To Create Visual Project Reports. [online] Plan Academy. Available at: https://www.planacademy.com/8-project-management-timeline-tools/ [Accessed 21 Aug. 2017].

Swartz J. (2011) Social media users grapple with information overload. USA Today. http://usatoday.com/tech/news/ 2011-02-01-tech-overload_N.htm (accessed Feb. 2, 2012).

Shahaf, D., Guestrin, C. and Horvitz, E. (2012). Trains of thought: Generating information maps. In Proceedings of the 21st international conference on World Wide Web (pp. 899-908). ACM.

Tran, Giang, M. Alrifai, and E. Herder. Timeline Summarization from Relevant Headlines. Advances in Information Retrieval. Springer International Publishing, 2015:245-256.

Thorndike, R. (1953). Who belongs in the family? Psychometrika, 18(4), pp.267-276.

Usharani, J., and D. K. Iyakutti. "A Genetic Algorithm based on Cosine Similarity for Relevant Document Retrieval." International Journal of Engineering Research and Technology ESRSA Publications, 2013.

U. Menon. "Man-Machine Systems: Information, Control and Decision Models of Human Performance." Journal of the Operational Research Society 27.1(1976):281-282

Vuejs.org. (2017). Comparison with Other Frameworks — Vue.js. [online] Available at: https://vuejs.org/v2/guide/comparison.html [Accessed 2 Jul. 2017].