Project Proposal

Team Name: Big News
Members:
Yuanji Huang (yh3059), Siyang Yin (sy2820), Mengyao Li (ml4042), Vivien Ngo (vn2260)
Language: Python, JavaScript
Platform: Mac

Project Summary

We will use customer data to create personalized newspapers. They can choose from a default newspaper that we make or pick their own interest categories. We will show them pages that might be of interest to them and also how much time each article would take to read. This is because people these days are crunched for time, and we want to give them as much relevant news for the limited time that they have.

If we have time, we would also like to extend our application to provide customized statistics (like most-viewed topics or publications) and other information, so users can better understand their news consumption.

The programming language we are going use include Python, JavaScript, the environment and platform is Mac.

We plan to use the News API to grab headlines and articles relevant to user interests. We will use MongoDB to implement our persistent database, which will store user information. For NLP, we plan to look into scikit-learn. For data visualization, we are considering JavaScript libraries like D3.js and p5.js.

We are going to use a built-in model.

User Stories

Minimum Viable Product:

Time: As a student, I am interesting in the topic about tech news. I want to spend 30 minutes every morning to find out the latest news in the industry. My conditions of satisfaction are: I can pick out relevant articles that will take at most 30 minutes to read. I can add the articles into a reading list. The app can give an estimated time for finishing all this reading.

Topic I: As a entrepreneur, I am interested in industry dynamics. I want to make sure that I am looking at news from all the businesses that might produce new products from competitors that change the industry. My conditions of satisfaction are: based on a list of company names that I give, I want the application to show me articles about those companies. For example, this means that if I give it the name “Apple,” it will show me articles about the company and not the food.

Analytics: As a busy person, I want to know what to focus on because I have too many news articles to read. I want the app to show me top key words, so that I can pay attention to the most trending ones. This may take the form of, for example, a word cloud that highlights key words from a group of articles. My conditions of satisfaction are: I can get a rough idea of what is most popular in the news right now from a quick glance in real time (similar to Twitter trending topics maybe), so that I can dig deeper later.

Stretch [if we have extra time]:

Sources: As a journalist, I want to make sure I’m reading news from lots of different types of publications (e.g. CNN and Fox News). My conditions of satisfaction are: I can give a list of sources that are interesting to me, and the app will give me articles from all those sources. [Additionally, it may also give me stats about what sources I’m reading, times of being read and the number of comments below this article. So I can make sure that I am actually spending my time/effort well on several publications.]

Topic II: As a businessman, I am interested in industry dynamics. I want the application to give me new recommendations based on things that I’ve already read. My conditions for satisfaction are: if I read articles about 3 car companies, it may be able to recommend me articles about a 4th car company that I didn’t explicitly mention.

Duplicate Detection: As someone who reads a lot of news, I want to make sure that I don’t read the same article twice. For example, lots of local outlets might publish the same article from Associated Press. My conditions are satisfaction are: there are not two separate links to the same content and similar headings (even if they are from different URLs).

Personal Analytics: As someone who may look at a lot of different topics, I want to know what I am spending my own time on (which may not exactly be what is popular for the general population). My conditions of satisfaction are: be able to find out behavior about my news consumption, e.g. what is my most-viewed topic or publications, how much time I spend reading daily, how many articles I’m reading daily, etc. Maybe I can have a year-end report about my statistics.

Acceptance Testing Plan

Minimum Viable Product:

Time

Inputs: Articles (URLs) that I want to read

Outputs: An estimation for how much time (# minutes) it will take to read them all.

Examples:

Input	Output
New York Times articles (How Connected Is Your Community to Everywhere Else in America?	7 mins ,
blog posts like Medium (Microsoft's Panos Panay Isn't Chasing Change	7 mins
possibly news videos? (Watch Banksy's $1.4 million painting 'self-destruct'	5.5 mins
No articles	0 mins

Topic I

Inputs: A list of company names

Outputs: Latest articles about those companies

Examples:

Input	Output
Keyword: Apple	Apple October 2018 Event: What to Expect from MacBook, iPad and other n related articles.

Keyword: LaCroix	LaCroix is facing a lawsuit over the mysterious ingredient that has made it a huge hit — here's what we know about it and other m related articles.

Analytics

Inputs: Current date

Outputs: Most common keywords related to articles that are “trending” today

Examples:

Input (dd/mm/yyyy)	Output
08/10/2018	Nobel Prize in economics Global warming: ● Major Climate Report Describes a Strong Risk of Crisis as Early as 2040 ● Overwhelmed by climate change? Here's what you can do
07/10/2018	Supreme Court Saudi Arabia Kavanaugh nomination

Test guideline:

- Importantly, we want to make sure that the keywords are all interesting and insightful. This means making sure that common words like “and” or “the” are not displayed, and also words like the current date.

Stretch [if we have extra time]:

Sources

Inputs: a list of sources

Outputs: articles from the sources

Examples:

Input	Output
Associated Press	2 Americans win econ Nobel for work on climate and growth ; After flap, Trump says he has no plans to fire Rosenstein and other articles from Associated Press.
The Wall Street Journal	Google Exposed User Data, Feared Repercussions of Disclosing to Public ; Facebook Launches Portal Video-Chat Devices for the Home and other articles from Associated Press.

Topic II

Input: History of articles read

Output: Recommended articles that are relevant

Examples:

Input	Output
History: ● LaCroix is facing a lawsuit over the mysterious ingredient that has made it a huge hit — here's what we know about it ● Why you probably shouldn't worry about that LaCroix lawsuit	Recommendation: ● Don't Freak Out About the LaCroix Lawsuit

Duplicate Detection

Inputs: A list of articles

Outputs: A list of articles that doesn't have duplicated items (in terms of the body text and possibly heading)

Examples:

Input	Output
Preprocessed article list: Article A from source A Article A from source B Article B from source A Article C from source A Article D from source B	Article list after detection: Article A from source A Article B from source A Article C from source A Article D from source B in which the duplicated articles are removed.

Personal Analytics

Inputs: History of articles read

Outputs: Statistics about articles read

Examples:

Input	Output
Reading history: ● 2 Americans win econ Nobel for work on climate and growth (AP) ● After flap, Trump says he has no plans to fire Rosenstein (AP) ● Facebook Launches Portal Video-Chat Devices for the Home (WSJ)	Sources: ● 66% Associated Press ● 33% Wall Street Journal Topic: ● 33% Technology ● 33% Politics ● 33% World News
Reading history: ● 2 Americans win econ Nobel for work on climate and growth (AP) ● After flap, Trump says he has no plans to fire Rosenstein (AP)	Sources ● 100% Associated Press Topic: ● 50% World News ● 50% Politics