

# Improving User Experiences for Wiki Systems

---

**Master's Defense**

**Advisor: Dr. Chris Pollett**

**Committee Members: Dr. Robert Chun, Dr. Philip Heller**

**Parth Patel | 12th December, 2022**

# Outline

---

- Purpose
- Background
- Preliminary work
- Recommendation System
- Yioop's Recommendation System
- Enhanced Yioop's Recommendation System
- Experiments
- Conclusion
- Future work
- References

# Purpose

---

- Improve user experiences for Yioop's wiki system
- Add emoji picker tool, UI testing and advertisement credits redeem features to Yioop
- Improve recommendation system in Yioop using Hash2Vec embedding
- Extend Yioop's recommendation system to recommend wiki resources

# Background - Wiki Systems and Yioop

---

- Wiki systems are web applications that allow users to collaboratively manage information
- Popular wiki system - Wikipedia
- Yioop is an open source web application with features of search engine and wiki system
- Developed by Dr. Chris Pollett since 2009 using PHP

# Background - Groups and Threads in Yioop

- A group is collection of threads and wiki pages
- A thread contains title and description
- Access rules for a group governs users activities

The screenshot shows the 'My Groups' page in Yioop. At the top, there is a navigation bar with the Yioop logo and the text 'My Groups'. Below this, there is a search bar with a 'Filter' button and a 'Go' button. The main content area displays a list of groups, each with a set of icons (representing different actions like post, edit, delete) and a trash can icon. The groups listed are:

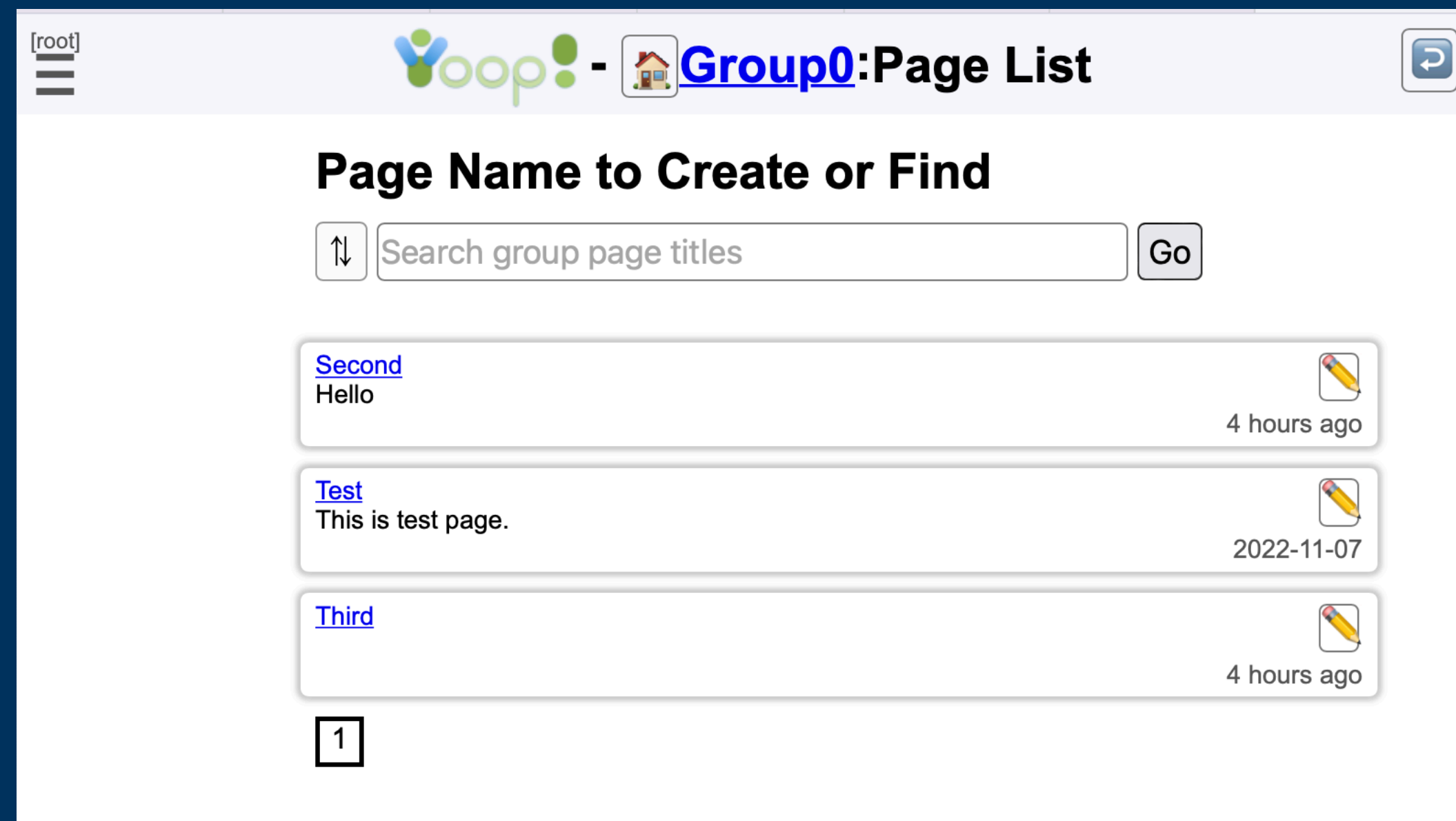
- Group0**: (7 posts, 4 threads). Last Post: [Third Wiki Page Created!](#)
- Public**: (498 posts, 498 threads). Last Post: [%D8%A3%D8%B1%D8%A8%D8%B9%D9%85%D8%A7%D8%A6%D8%A9\\_%D8%A3%D8%B1%D8%A8%D8%B9 اصفحة ويكي خلق](#)
- Help**: (3011 posts, 3011 threads). Last Post: [Subsearches Wiki Sayfasını Oluşturdu.](#)
- Search**: (0 posts, 0 threads). Last Post: No Posts Yet

At the bottom of the page, there is a footer with links: [- Blog](#) - [- Privacy](#) - [- Terms](#) - [- ThisSiteBot](#) - [- Developed at SeekQuarry](#) - (c) This Site - [- This Search Engine](#)

The screenshot shows a thread page in Yioop. At the top, there is a navigation bar with the Yioop logo and the text 'Group0:Talk:Thread Demo'. Below this, there is a timestamp '4:08 pm'. The main content area displays a thread titled 'Thread Demo.' with a user profile picture and the name 'root'. The thread content is: 'This is the demo of a thread functionality in Yioop'. Below the thread content, there is a 'Comment' button. At the bottom of the page, there is a footer with links: [- Blog](#) - [- Privacy](#) - [- Terms](#) - [- ThisSiteBot](#) - [- Developed at SeekQuarry](#) - (c) This Site - [- This Search Engine](#)

# Background - Wiki pages and Resource

- Wiki page contains textual data and files called as resources
- Different types of wiki pages

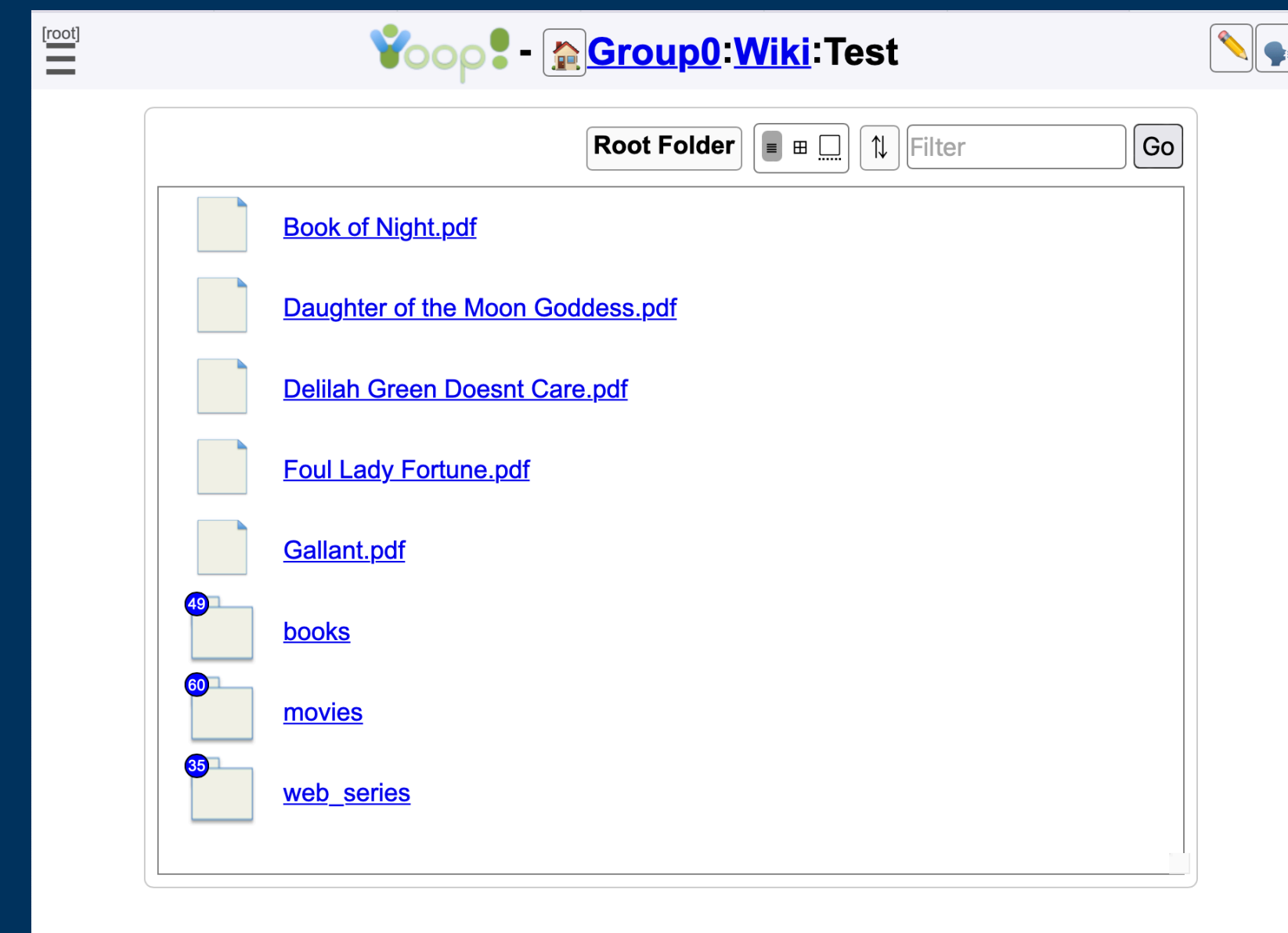


The screenshot shows the 'Group0:Page List' interface. At the top, there is a search bar with the placeholder text 'Search group page titles' and a 'Go' button. Below the search bar, there are three entries in a list:

- Second**: Hello, 4 hours ago
- Test**: This is test page., 2022-11-07
- Third**: (empty), 4 hours ago

At the bottom left, there is a small box containing the number '1'.

Collection of wiki pages in a group



The screenshot shows the 'Group0:Wiki:Test' interface. At the top, there is a search bar with the placeholder text 'Filter' and a 'Go' button. Below the search bar, there is a list of media items:

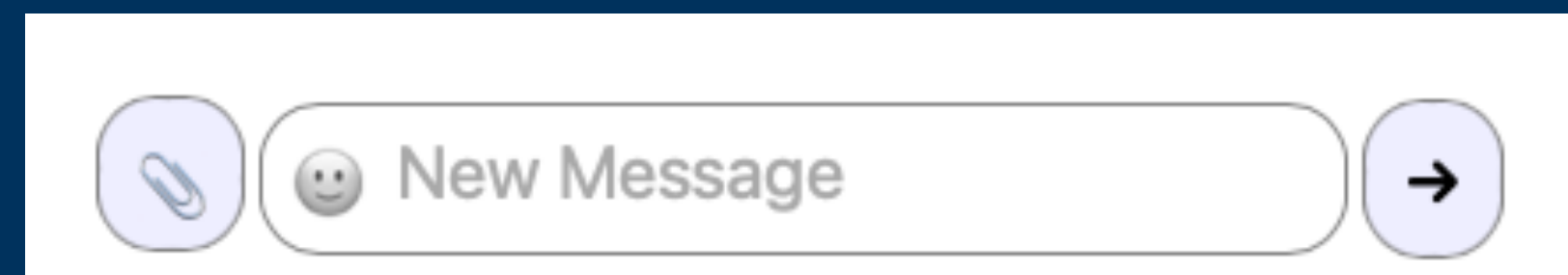
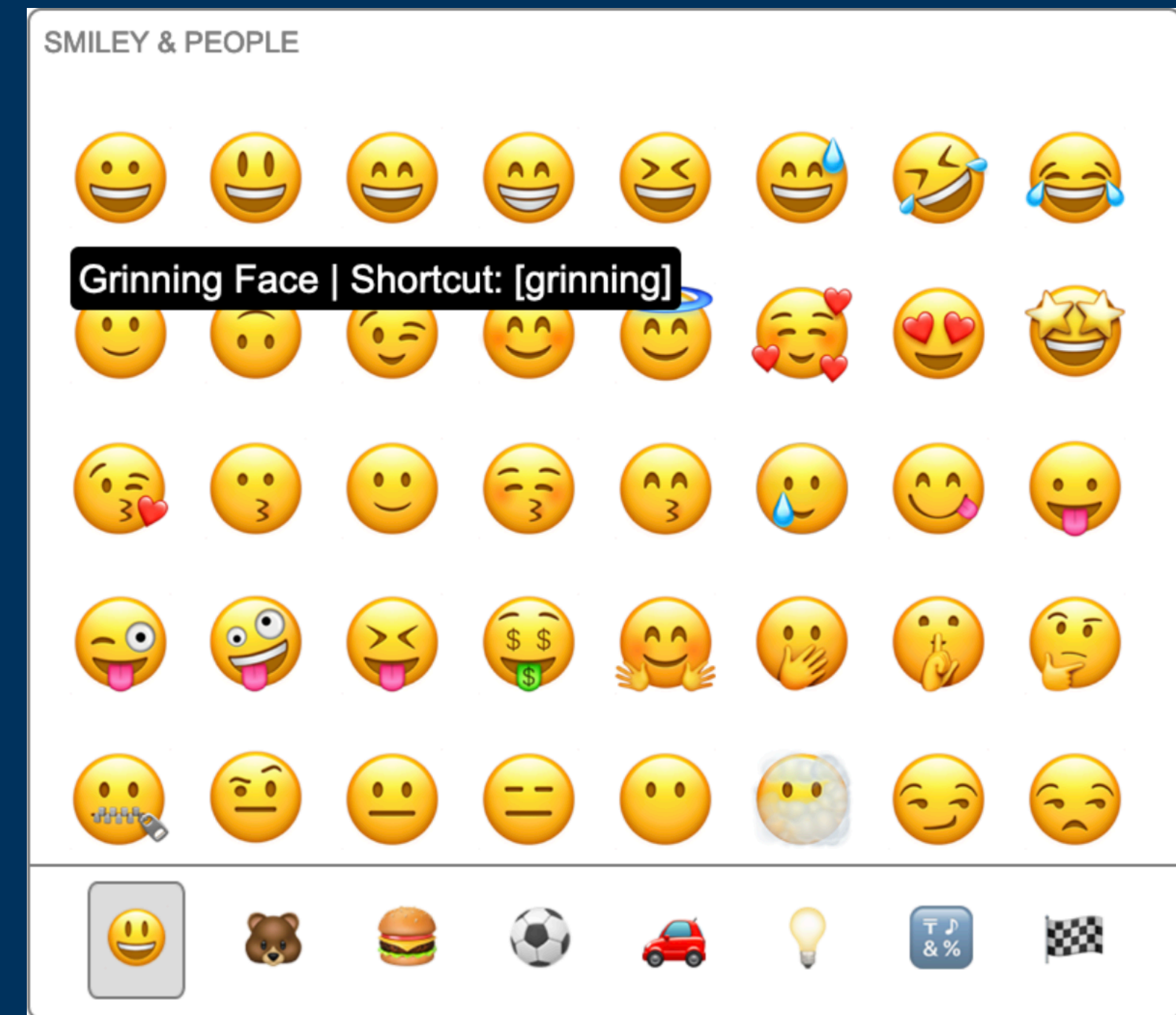
- [Book of Night.pdf](#)
- [Daughter of the Moon Goddess.pdf](#)
- [Delilah Green Doesnt Care.pdf](#)
- [Foul Lady Fortune.pdf](#)
- [Gallant.pdf](#)
- [books](#)
- [movies](#)
- [web\\_series](#)

A media list wiki page



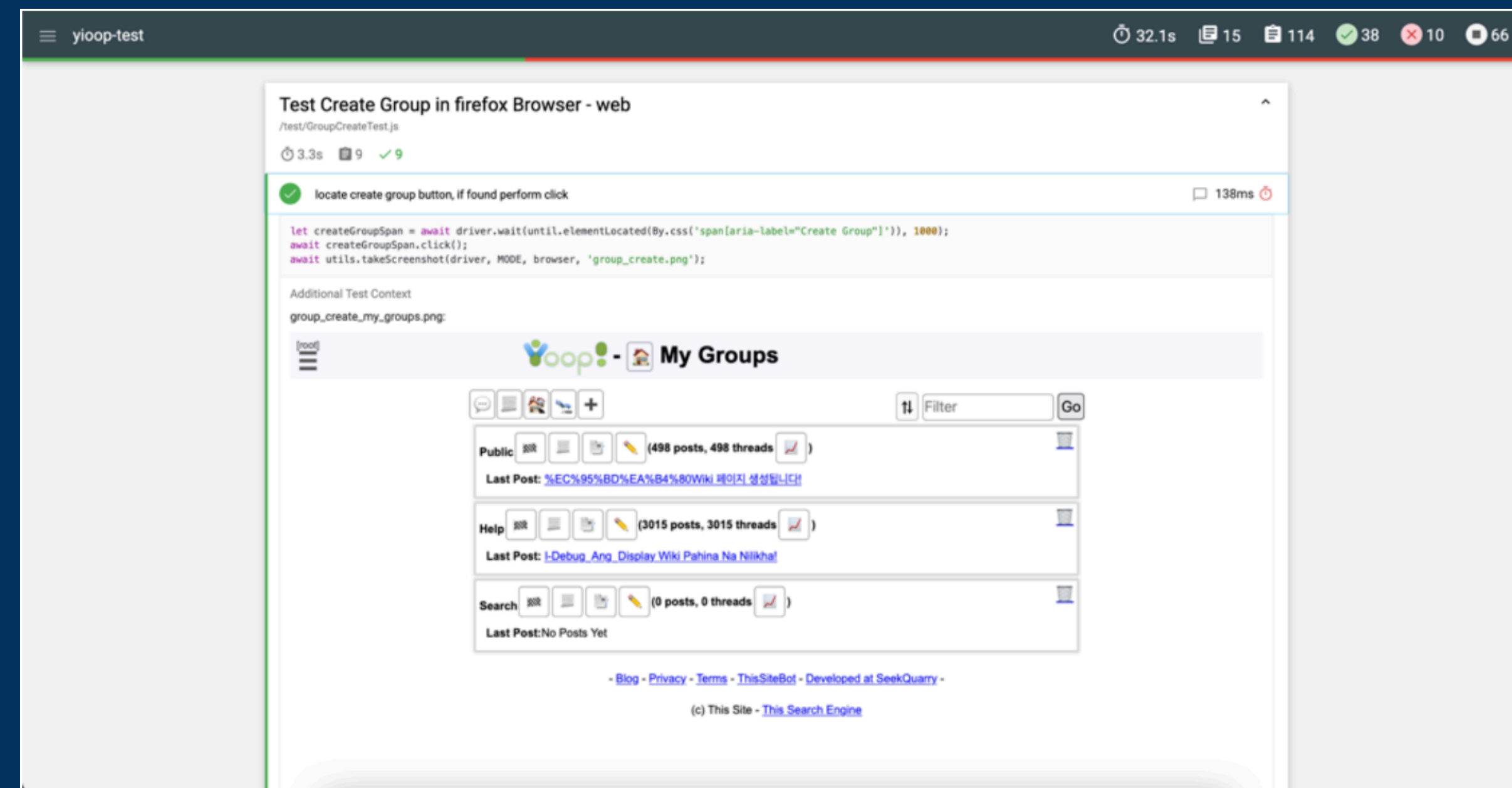
# Preliminary work - Emoji Picker Tool

- Yioop provides direct messaging feature
- First version lacked support for emojis
- Implemented emoji picker tool
- Supports around 1500 emojis
- No external code dependency
- Improved experience for direct messaging



# Preliminary work - UI Testing

- User interface affects the experience while interacting with an application
- Yioop UI testing mechanism was outdated
- A new UI testing project was created to ensure correctness of UI
- Developed using Selenium and Node.js





# Preliminary work - Credits Redeem

- Yioop supports keywords based advertisement
- Keyword advertisement bidding requires credits
- Credits are purchased using credit / debit cards
- Developed mechanism to convert unused credits back to real money
- Stripe services were used for transferring money to user bank account

**Balance: 5000 credits**

**Credit Transactions** + Row 0 to 2 of 2 Show 50

[Purchase Credits](#) [Redeem Credits](#) [Manage Details](#)

**Redeem Credits** ?

Quantity:

Debit Card Number:

CVC:

Expiration:  /

Using the Redeem button credits the above card the Quantity field's amount in US dollars and agrees to the [Program Terms](#).

| Type             | Amount | Date                            | Total |
|------------------|--------|---------------------------------|-------|
| Starting Balance | 0      | Thu, 28 Apr 2022 22:18:26 -0700 | 0     |
| Buy Credits      | 5000   | Thu, 28 Apr 2022 22:27:25 -0700 | 5000  |

# Recommendation System

---

- A recommendation system retrieves information based on user profile
- User profile is created based on information consumed by user
- Amazon, Netflix, Twitter, etc have recommendation system
- Greatly increases user experience
- Few types of recommender systems
  - Collaborative based
  - Content based
  - Hybrid

# Yioop's Recommendation System

---

- Yioop's recommendation system recommends threads and groups
- Content based recommender system
- Recommendation media job runs at regular interval to update recommendations

### Recommendations

**Threads:**

- [Trang\\_Ph%C3%A2n\\_Lo%E1%BA%A1i\\_Wiki\\_Trang\\_Tạo\\_Ra!](#)
- [Ki%E1%BB%83m\\_tra\\_Ch%E1%BB%89\\_m%E1%BB%99t\\_Trang\\_Wiki\\_Trang\\_Tạo\\_Ra!](#)
- [Cho\\_ph%C3%A9p\\_%C4%91%E1%BB%83\\_thu\\_th%E1%BA%ADp\\_th%C3%B4ng\\_tin\\_trang\\_Web\\_Wiki\\_Trang\\_Tạo\\_Ra!](#)

**Groups:**

- [Group1](#)

# Yioop's Recommendation System

---

- Leverages term frequency - inverse term frequency (TF - IDF) technique
- TF measure represent frequency of a term within a document
- IDF measure represent number of documents containing the term

$$tf(t, d) = \log(1 + freq(t, d))$$

$$idf(t, D) = \log\left(\frac{N}{count(d \in D: t \in d)}\right)$$

TF and IDF equation [12]

# Yioop's Recommendation System

- TF-IDF applied on BoW created from titles and description of threads
- TF-IDF scores calculated for each user based on their history
- Users view history captured in ITEM\_IMPRESSION table
- Cosine similarity measured between users and threads TF-IDF

$$\text{cosine similarity} = S_C(A, B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}},$$

|   | USER_ID | ITEM_ID | ITEM_TYPE | VIEW_DATE  |
|---|---------|---------|-----------|------------|
|   | Filter  | Filter  | Filter    | Filter     |
| 1 | 1       | 3512    | 2         | 1670705619 |
| 2 | 2       | 3512    | 2         | 1670705619 |
| 3 | 1       | 10      | 3         | 1670705619 |
| 4 | 2       | 10      | 3         | 1670705619 |
| 5 | 1       | 3512    | 2         | 1670705621 |
| 6 | 2       | 3512    | 2         | 1670705621 |
| 7 | 1       | 10      | 3         | 1670705621 |
| 8 | 2       | 10      | 3         | 1670705621 |

# Enhanced Yioop's Recommendation System

---

- Existing Yioop's recommendation system required more time for processing data
- TF-IDF technique does not capture semantic meaning of words
- Recommendations generated were not accurate
- This project aims to enhance Yioop's recommendation system
- Primarily Hash2Vec embedding technique will be used to replace TF-IDF
- Further extend recommendation functionality for wiki resources



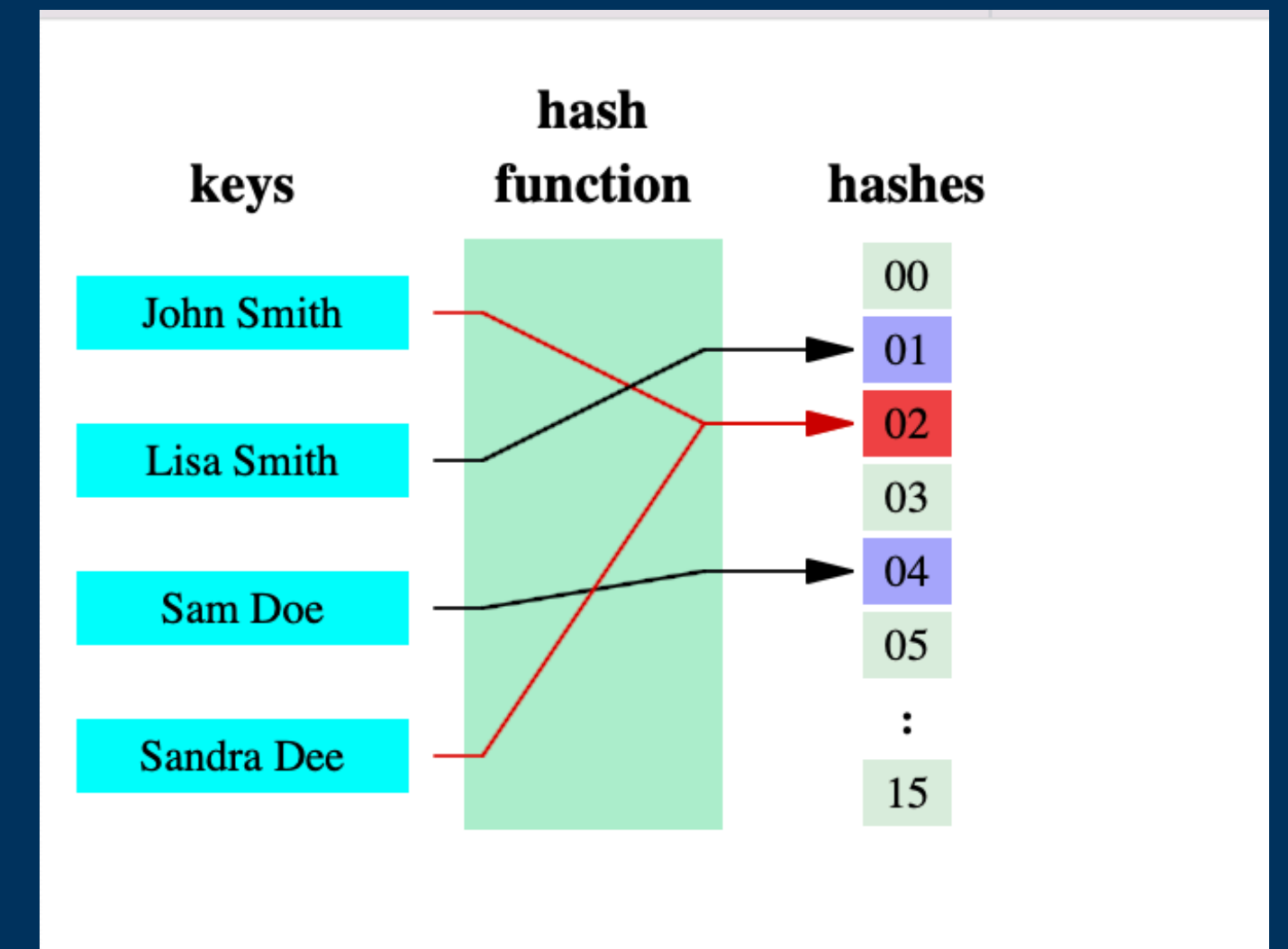
# Word Embeddings

---

- Word embedding is process of assigning vector to a word
- Simple embedding like CBoW or skip-grams are inefficient
- Advanced embedding techniques like Glove or Word2Vec utilize neural networks to compress CBoW embeddings to lower dimensions
- Neural networks require high end computing hardware, often taking longer time to process
- None of the above techniques can be used for Yioop

# Hash2Vec Word Embedding

- Hash2Vec technique leverages hashing operation to calculate embeddings that can preserve semantic meaning [8]
- Hashing operation converts a word into a numerical value
- The space complexity is  $O(nk)$ ,  $n$  = number of unique words and  $k$  = size of embedding vector
- The space complexity and processing time makes Hash2Vec ideal for Yioop



# Hash2Vec Implementation for Yioop

---

- Example
  - this is an example text corpus
  - $k = 2$
- Parameters value used
  - $n = 200$
  - $k = 5$
  - $h = \text{md5}$ ,  $\xi = \text{crc32}$
  - $f = (e^{-x})^2$ ,  $x = d/\sigma$

---

## Algorithm 1 Hash2Vec

---

Parameters:  $n$  the embedding size,  $k$  the context size,  $h$  hash function,  $\xi$  hash sign-function and  $f$  aging function.

```
1: words  $\leftarrow$  Dictionary()
2: for every word  $w$  in text do
3:   if  $w \notin \text{keys}(\text{words})$  then words[ $w$ ]  $\leftarrow$  Array( $n$ )
4:   for every context word  $cw$  with distance  $d$  do
5:     weight  $\leftarrow f(d)$ 
6:     sign  $\leftarrow \xi(cw)$ 
7:     words[ $w$ ][ $h(cw)$ ]  $\leftarrow$  words[ $w$ ][ $h(cw)$ ] + sign  $\times$  weight
```

---

# Hash2Vec Implement for Yioop

---

- Preprocessing of terms in title and description of threads is done
- Applying Hash2Vec technique produced vectors for each unique term
- Threads vector are calculated using vectors of terms in its title and description
  - $V(T) = V(T1) + V(T2) + \dots + V(TN)$
- Threads vectors are normalized
  - $V(T) = V(T) / |V(T)|$ 
    - $|V(T)| = \text{sqrt}(V(T) * V(T))$

| ID     | ITEM_TYPE | VECTOR                                           |
|--------|-----------|--------------------------------------------------|
| Filter | Filter    | Filter                                           |
| 3367   | 2         | vziKBM+EqIY/TXvOTRmxeD8y1yvGljCLvyKHUq4JXeC/...  |
| 3368   | 2         | vziM254152I/TX83jj+6/T8y2VnMPWjAvyKJd3cuNXa/...  |
| 3369   | 2         | vziNU2GRGcw/TX/Hc0WKZj8y2bW/sqm/vyKJ0eTw0k6/...  |
| 3370   | 2         | vziNITolQv4/TX+LMKbukD8y2Y891iXOvyKJrAZG73G/...  |
| 3372   | 2         | vzMVLPQzc/0/TX/5nFWiEz8y2dXNWK4ivyKJ8Wq+dRS/...  |
| 3373   | 2         | vziNKA3Emd8/TX+TZlukHz8y2ZR7s99XvyKJsS3tsZK/...  |
| 3374   | 2         | vziNTgec2cw/TX/BBUfLnD8y2bGj4fh4vyKJzdzqJ1HS/... |
| 3375   | 2         | vziMjZmaeGc/TX7Z0UmnEj8y2R3lylqqvyKJPI6YIDC/...  |

# Hash2Vec Implement for Yioop

---

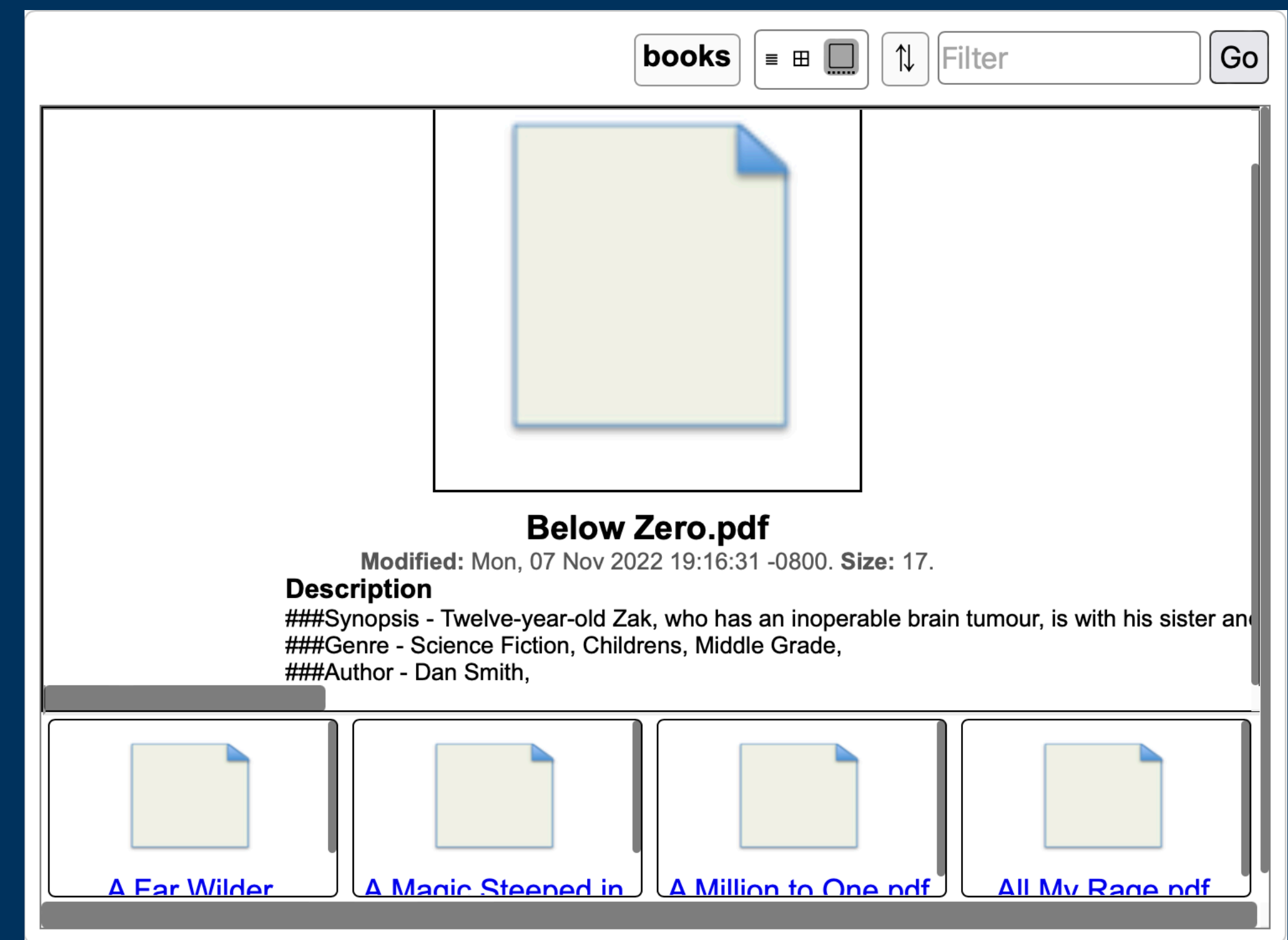
- Different from existing system, user profile embedding vectors are calculated
- Embeddings of threads viewed by user are added and normalized
- Cosine similarities between user profile embeddings and threads embeddings are calculated
- Groups recommendation are calculated using embeddings of threads in it

|   | USER_ID | ITEM_ID | ITEM_TYPE | SCORE              |
|---|---------|---------|-----------|--------------------|
|   | Filter  | Filter  | Filter    | Filter             |
| 1 | 1       | 3516    | 2         | -0.012321225221818 |
| 2 | 1       | 3233    | 2         | 0.011862890698645  |
| 3 | 1       | 3234    | 2         | 0.011990854943252  |
| 4 | 1       | 3235    | 2         | 0.011699824674205  |
| 5 | 1       | 3236    | 2         | 0.011828947946366  |
| 6 | 1       | 3237    | 2         | 0.011834296149311  |
| 7 | 1       | 3238    | 2         | 0.011865462202601  |
| 8 | 1       | 3239    | 2         | 0.011865892181212  |



# Recommendation for Wiki Resources

- Recommendation functionality extended to recommend wiki resources
- A resource has a name and an optional description
- Apply Hash2Vec technique on the description of resources
- How to get descriptions for resources?





# Description Search Source

- Yioop provides functionality to configure search sources
- Search sources are used by media jobs to fetch information from web
- Introduced Description search source to fetch description for resources
- Use XPath to fetch relevant information on given website

**Type:** Description Source

**Name:** IMDB

**URL:** https://www.imdb.com/find?q=

**Language:** English

**Path Terms:** TV Shows, video

**Info XPath**  
Year/Rating | //ul[contains(@data-testid,'hero-title-block\_\_metadata')]/li/a  
IMDB | (.//div[contains(@data-testid,'hero-rating-bar\_\_aggregate-rating\_\_score')]/span)[1]  
Plot | //span[contains(@data-testid,'plot-!')]  
Genres | //a[contains(@class,'ipc-chip')]  
Credits | //div[contains(@data-testid,'title-pc-expanded-section')]/ul[contains(@data-testid,'title-']

**Item XPath** //li[contains(@class,'find-resul

**Title XPath** //a[contains(@class,'pc-metad

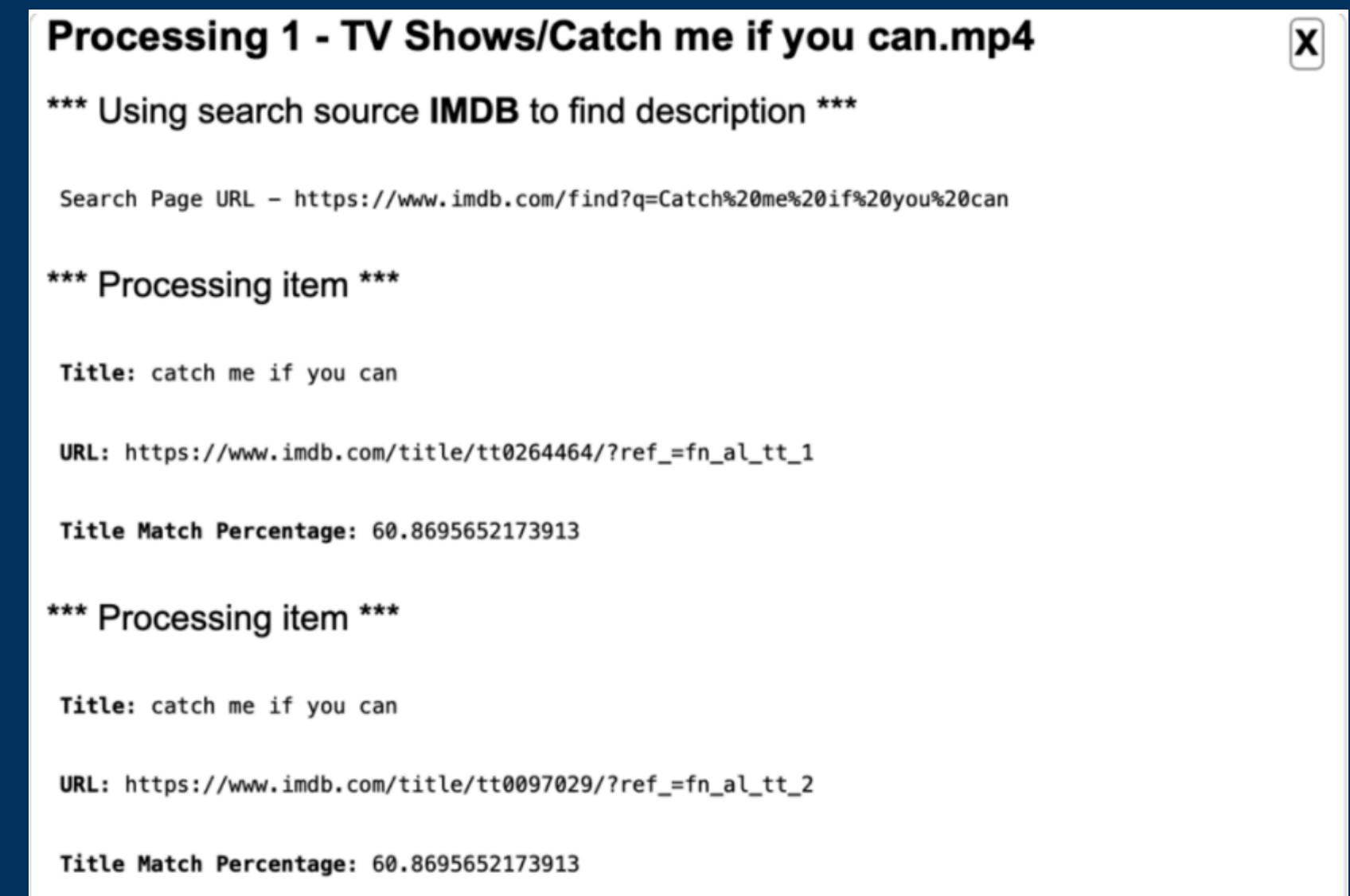
**Url XPath** //a[contains(@class,'ipc-metac

**Test Values:**  
TV Shows/Catch me if you can.mp4  
TV Shows/Inception.mp4  
TV Shows/Wrong Turn.mp4

Save Test

# Description Update Media Job

- Developed DescriptionUpdate media job to process description search sources
- Implemented an efficient mechanism to track resources without description
- Able to fetch descriptions using just the name of resources
- Selects the result with highest title match score to fetch required details



# Extended Yioop's Recommendation System

- Extended recommendation media job to apply Hash2Vec on fetched description of resources
- Similar mechanism leveraged to keep track of resources frequently viewed by users and have descriptions
- Resources with top 3 scores are recommended on home page

## Page Resources:

- [The Stardust Thief.pdf](#)
- [Nettle & Bone.pdf](#)
- [Kaikeyi.pdf](#)

|   | USER_ID | GROUP_ID | PAGE_ID | RESOURCE_PATH                                           | SCORE             | TIMESTAMP  | RESOURCE_ID |
|---|---------|----------|---------|---------------------------------------------------------|-------------------|------------|-------------|
|   | Filter  | Filter   | Filter  | Filter                                                  | Filter            | Filter     | Filter      |
| 1 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.059151130484822 | 1670705810 | 9620        |
| 2 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.05242068775456  | 1670705810 | 21721       |
| 3 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.066277429720664 | 1670705810 | 40566       |
| 4 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.060397784821666 | 1670705810 | 30022       |
| 5 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.046821552006818 | 1670705810 | 2153        |
| 6 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.058398633599215 | 1670705810 | 45640       |
| 7 | 1       | 10       | 3510    | /Users/admin/Desktop/CS298/yioop/work_directory/app/... | 0.037390870194849 | 1670705810 | 2692        |

# Experiments - Description Update Job

---

- Experiments conducted on 2 collections of resources
- First collection had total of 137 movies, TV shows and books released in 2022
- Second collection had 78 TV shows dating as back as 1950
- Configured 2 description search sources for IMDb and Goodreads website

|              | Correct | Wrong | Missing |
|--------------|---------|-------|---------|
| Collection 1 | 130     | None  | 43      |
| Collection 2 | 45      | 33    | None    |

# Experiments - Resource recommendation

---

- Experiment was done on 137 resources in collection 1
- 5 test users were created
- 4 users interacted with 15 resources each and fifth user did not interact with any resource
- A profile label was attached to a user based on the resources interacted
- For example user interacting with only action movies is given Action profile label



# Experiments - Resource recommendation

---

| User         | Profile                               | Relevant Recommendations | Irrelevant Recommendations | Accuracy |
|--------------|---------------------------------------|--------------------------|----------------------------|----------|
| 1            | Action                                | 2                        | 1                          | 0.67     |
| 2            | Science, Fiction, History and Romance | 1                        | 2                          | 0.33     |
| 3            | Drama                                 | 3                        | 0                          | 1        |
| 4            | Thriller, Comedy, Family, Anime       | 1                        | 2                          | 0.33     |
| <b>Total</b> |                                       | 7                        | 5                          | 0.58     |



# Experiments - Threads and Groups Recommendation

---

- Experiments were conducted to compare accuracy of enhanced and existing recommendation system
- Two instances of Yioop server installed on same machine
- 5 groups with 10 threads each were created on both instances
- Thread contained descriptions on 5 categories
  - Software development
  - Political affairs
  - Sports
  - Stock market
  - Gaming

# Experiments - Threads and Groups Recommendation

---

| Group | Software Development | Gaming | Political affairs | Stock market | Sports | Profile                  |
|-------|----------------------|--------|-------------------|--------------|--------|--------------------------|
| 1     | 3                    | 3      | 1                 | 2            | 1      | Mixed                    |
| 2     | 1                    | 5      | 0                 | 0            | 4      | Gaming / Sports          |
| 3     | 0                    | 0      | 6                 | 4            | 0      | Political / Stock market |
| 4     | 7                    | 0      | 1                 | 1            | 1      | Software development     |
| 5     | 0                    | 3      | 3                 | 1            | 3      | Mixed                    |

# Experiments - Threads and Groups Recommendation

---

- 5 test users were created on both instances
- User was member of same group and viewed same 5 threads
- Recommendation job completed processing in 8 seconds on instance with enhanced recommendation system
- Recommendation job took 24 seconds to finish processing on another instance with existing recommendation system
- Results summarized on next slide

# Experiments - Threads and Groups Recommendation

---

| User           | Enhanced Recommendation System Accuracy- Thread | Existing Recommendation System Accuracy - Thread | Enhanced Recommendation System Accuracy - Group | Existing Recommendation System Accuracy- Group |
|----------------|-------------------------------------------------|--------------------------------------------------|-------------------------------------------------|------------------------------------------------|
| 1              | 0.67                                            | 0.33                                             | 1                                               | 0.33                                           |
| 2              | 1                                               | 0.67                                             | 1                                               | 1                                              |
| 3              | 0.33                                            | 0                                                | 0.67                                            | 0                                              |
| 4              | 0.67                                            | 0.67                                             | 1                                               | 0.67                                           |
| 5              | 1                                               | 0.67                                             | 1                                               | 1                                              |
| <b>Average</b> | 0.73                                            | 0.47                                             | 0.93                                            | 0.6                                            |

# Conclusion

---

- Implemented new features which contributed to improving user experiences for Yioop
- Hash2Vec embedding technique was studied and implemented for Yioop recommendation system
- Enhanced recommendation system using Hash2Vec performs better than existing recommendation system
- Extended recommendation system for wiki resources using novel approach of extracting descriptions from web
- Processing time for recommendation job is decreased

# Future work

---

- Multiple areas of future work building upon this project
- Description update job can be improved
  - Compare results from multiple sources to select best match
  - Incorporate other information of resource to improve matching process
- A weight function can be designed to weight term embeddings while calculating threads and resource embeddings
- Introduce mechanism to take feedback from user for recommendations



# References

---

- [1]. B. Corinna, G. Bela, Langer and Stefan, “Research-paper recommender systems: a literature survey”, *International Journal on Digital Libraries*. 17 (4): 305–338. doi:10.1007/s00799-015-0156-0
- [2]. J. Ramos, et al., “Using TF-IDF to determine word relevance in document queries.” *Proc. of the First Instructional Conf. on Machine Learning*, 2003, vol. 242, pp. 133–142
- [3]. M. Anirudh, “An open source direct messaging and enhanced recommendation system for Yioop”, San José State University, Dec 2021
- [4]. *Full Emoji List, v15.0*. (n.d.). <https://unicode.org/emoji/charts/full-emoji-list.html>
- [5]. *Selenium*. (n.d.). <https://www.selenium.dev>
- [6]. *Stripe | A platform for processing online payments*. (n.d.). <https://stripe.com>

# References

---

- [7]. T. Gerth, “A Comparison of Word Embedding Techniques for Similarity Analysis.”, Computer Science and Computer Engineering Undergraduate Honors Theses, 2021. Available: <https://scholarworks.uark.edu/csceuht/85>
- [8]. L. Argerich, M. J. Cano and J. T. Zaffaroni, “Hash2Vec: Feature Hashing for Word Embeddings”, 2016
- [9]. Q. Shi, J. Petterson, G. Dror, J. Langford, A. Strehl, L. Smola, et al., “Hash kernels.” *International Conference on Artificial Intelligence and Statistics*, 2009, 496–503
- [10]. *IMDb | Ratings, Reviews, and Where to Watch the Best Movies & TV Shows*. (n.d.). <https://www.imdb.com>
- [11]. *Goodreads | Meet your next favorite book*. (n.d.). <https://www.goodreads.com>
- [12]. *Wikipedia contributors*. (2022, October 5). *Main Page*. [https://en.wikipedia.org/wiki/Main\\_Page](https://en.wikipedia.org/wiki/Main_Page)

**Thank You!**