University of Waterloo
Faculty of Engineering

Narrowing Down in One EC Search

Yahoo Inc.
14F, No.66 Sanchon Rd
Nangang District, Taipei City, Taiwan

Prepared By
Tzu-Wen Wu
ID 20563585
2A Department of Systems Design Engineering
September 22, 2016.

Tzu-Wen Wu
46 Soho Crescent, Markham, ON, L3P 7H5

September 22nd, 2016

Dr. Paul Fieguth, Professor and Department Chair
Systems Design Engineering
University of Waterloo
Waterloo, ON
N2L 3G1

Dear Professor Fieguth,

I have prepared this report, "Narrowing Down in One EC Search" as my 2A Work Report for the E-Commerce Search Team at Yahoo Taiwan. This report is the first of three that I must submit as part of my degree requirements, and it has not received any previous academic credit. This report was entirely written by me and has not received any previous academic credit at this or any other institution.

Yahoo Taiwan is an autonomous business unit that has been focusing heavily in the e-commerce business. The e-commerce business at Yahoo is uniquely established for the market in Taiwan and driven by a large e-commerce engineering department. As a Software Engineering Intern, I worked at Yahoo's office in Taipei City with the E-commerce Search team led by Jeff Tang.

The E-commerce Search team at Yahoo Taiwan is responsible for improving the search experience of all the e-commerce platforms at Yahoo, from both backend and frontend. Along with another engineer and a designer, our goal was to recognize problems in the existing search platform, specifically in One EC Search, and design a solution that can be adopted to the upcoming Yahoo App. This report documents the complete journey of the project from the problem analysis to the final prototype developed in Android.

My sincere gratitude is given to my teammates, Stanley Hong and Audrey Chiang, who have contributed wholeheartedly to the project. The solution as discussed in the report was developed by all team members and should be credited as such. I would also like to thank the EC Search team for providing necessary guidance that altogether allowed the project to come together.

Sincerely,

Tzu-Wen Wu

20563585

Abstract

Yahoo’s E-commerce Search team serves as a function to improve the users search experience while using Yahoo’s e-commerce platforms. For this design, the emerging One EC Search technology at Yahoo Taiwan is further analyzed to develop a new solution, using the unfamiliar Boolean Search, to help users quickly narrow down and locate the items of their interest. The design journey starts by a background and problem analysis of One EC Search that introduces the exiting platforms and search functionalities at Yahoo. It is followed by the establishments of necessary design criteria that are transformed to a prototype implemented in Android OS. The prototype is used to conduct user tests and collect feedbacks on the user experience, to understand the importance of the newly introduced solution and its need to be adopted to Yahoo’s mainstream product, Yahoo App. The report concludes with recommendations on various potential improvements to be made by specific engineering and design teams at Yahoo Taiwan.

Contents

List of Tables

List of Figures

1

Introduction

1.1 Preface

Yahoo Taiwan has been operating outside of its global corporate framework as an autonomous business unit for the past few years, which has also been a key driving force in the APAC (Asia-Pacific) region for Yahoo. The core business of Yahoo Taiwan focuses heavily in e-commerce with heavy cultural connections to the locals' living habits, and demand for technologies that bring convenience to shopping. The e-commerce products that Yahoo Taiwan has developed include Auctions, Shopping, and Marketplace, all tailored for different user segments and business model. Over the years, these products have established a unique e-commerce market in Taiwan in replacement for globally known e-commerce sites such as Ebay and Amazon.

On top of these three products that have been running individually as web and mobile applications, Yahoo Taiwan has recently been developing a combined search platform that allows user to retrieve the result of all three e-commerce applications in one place, hence the project name of "One EC Search". Despite its functionality, the goal of the One EC Search project is not to replace the existing products but to attract light e-commerce users to the Yahoo online-shopping market by providing them an easier way to reach out to the wide range of items. Therefore the existence of current Yahoo e-commerce products is still very important to serve the specific user segments with full potential.

1.2 Situation of Concern

The One EC Search project is currently used on the Yahoo Taiwan website as a simple search interface, and the company is striving to adopt the One EC Search technology in the rising mobile market (Android and iOS). With the recent introduction of the Yahoo App, an app that combines all Yahoo's main products (finance, weather, etc.), a section will be inserted for Yahoo's online shopping where One EC Search will be used in hopes to attract more users from other Yahoo assets.

However, as useful as the One EC Search technology is, combining the search results of all three platforms increases massively the amount of items and the complexity to organize items based on category mapping. Consequently, the search quality of One EC Search is definitely in need of improvement which significantly degrades users' search experiences. For one, the large amount of item listings only overwhelms the users with unwanted items. In addition, given such large amount of information, users have no easy way of filtering, and narrowing down the search result leading to excessive time spent on manually filtering items.

This report will address this issue by outlining the problem and documenting a proposed solution that helps user narrow down and locate the best item in a much quicker timeframe. Given that the solution is designed realistically to best apply the existing e-commerce platforms, some information and design will contain texts in Chinese, which will be translated to English if relevant.

2

Problem Analysis

2.1 State of the Art

To understand the problem further, we start by looking at each platform more closely in order to understand the data that are handled behind One EC Search. As mentioned previously, Yahoo Taiwan currently offers three distinct e-commerce applications including Auctions, Shopping and Marketplace.

Auctions

Auctions is a C2C (Customer-to-Customer) product that provides a completely free and opened platform for anyone to sell their items. It features an Ebay-like environment which includes brand new and used items, as well as a biddable, and negotiable system. Given the unrestrained characteristic, Auctions currently hosts over 24 million items across a wide range of different categories. Furthermore, the quality of the items are not monitored nor standardized. As a result, Auctions tend to have cheaper items despite its riskier transaction.

Shopping

Shopping is a B2C (Business-to-Customer) product that features an Amazon-like environment where products shipped from a warehouse or sourced from a vendor are completely Yahoo branded. As such, Shopping generates the greatest revenue per sale compare to the other two e-commerce platforms, though it only has 500 thousands listings which is relatively small. The small amount of items allows each item to not only be categorized very precisely, but also to have specific cluster data based on the category the item belongs to. These cluster data are then used for filtering the search result within the Shopping platform.

Marketplace

Marketplace is a B2B2C (Business-to-Business-to-Customer) platform that allows merchants to host their own stores and list their products on the site. These merchants are typically larger-size business units, which usually guarantees a safer and higher quality

transaction. Marketplace currently hosts over 3.5 million items, which is not as massive as Auctions, but not simple enough like Shopping to be easily organized.

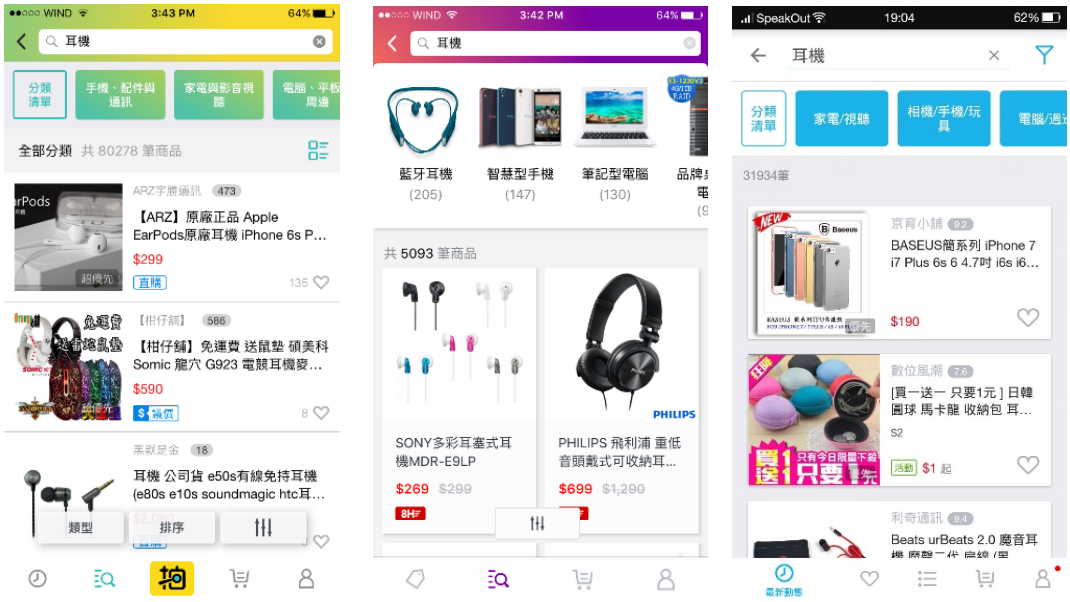


Figure 1: Screenshots of Auctions, Shopping and Marketplace respectively

2.2 Limitations

Each platform is specially designed for its targeted users. As One EC Search merges all three platforms into one, many problems arise even though they all ultimately serve the same purpose. By adding the three platforms, the total amount of item exceeds more than 28 million resulting in a much diluted search result, where it becomes challenging for a user to find the most fitting item of interest.

Having 28 million items would be nothing if it existed a good filtering mechanism. But due to the differences in size and restriction across the three platforms, the ability for One EC Search to categorize and organize all the items become very limited. Each platform has its own defined categories that best fit the quantity and variation of items it encompasses. For example, Auctions have a very detailed, multi-layers categorizing system since it needs to cover over 24 million items. Every generic category such as electronics, clothing, furniture, etc. can further be broken down to the specific type of item. Table 1 demonstrates few examples of similar layer-one categories across the three platforms.

Table 1: Sample of similar categories (first layer) across the three platforms

Platform	Auctions	Shopping	Marketplace
Categories	女裝與服飾配件 Women's Clothing and Accessories 女包精品與女鞋 Women's Accessories and Shoes 美容保養與彩妝 Skin care and cosmetics 男性精品與服飾 Men's Clothing and Accessories 手錶與飾品配件 Watches and Accessories	女鞋/男鞋/運動鞋 Shoes / Sneakers 女包/男包/皮夾 Bags / Wallets and Purses 服裝 / 飾品 / 配件 Clothing / Accessories 牛仔服飾 Denim Clothing 內睡衣 Underwear and Pajamas 精品/手錶/珠寶飾品 Gifts / Watches / Accessories	服飾 Clothing 美妝 Makeups 鞋包/精品/配飾 Shoes, Bags/ Gifts/ Accessories

As different categories are used in each platform, One EC Search merely provides a simplified, one-layer categorizing system that tries to cover as many items as possible. However, not only does it lack accuracy in categorizing items, having such generic categories is far from being useful as it does not help narrowing down the search result. Consequently, One EC Search requires a new solution to assist users with narrowing down and finding the desired items.

2.3 Opportunities

As part of Yahoo's EC Search team, our goal is to enhance users' search experiences and One EC Search clearly shows problems in need of resolutions. Regarding the problem of dealing with massive amount of items without an effective filtering mechanism, there are two main opportunities that would tackle the problem.

New Category Mappings

Since the problem is ultimately sourced from the inability to properly categorize the items, the most straightforward solution would be the development of a new category system that unifies all three platforms. If the mapping of categories to each platform can be precise and non-generic, it would certainly be very powerful. However, the complexity to map all the categories from all the platforms intelligently is extremely high and has actually been attempted previously by the team at Yahoo. The best result that has been achieved only allows mapping of Shopping and Marketplace's items to Auction's second category layer, out of four to five layers. In another word, the result still remains too generic to be helpful. The best solution would require a fourth custom categorizing system that are defined

manually by Yahoo's editors. Unfortunately, it cannot be done due to the lack of resources. A new category system is therefore not likely to be achieved in the short term.

Boolean Search

The most important element in any search task is the keyword that is entered for query. The more precise the keywords are, the more accurate the search result will be. Thus, in the case for One EC Search, choosing the right keywords can significantly impact the relevancy of the search result. In addition to the right keyword, One EC Search also provides the ability to conduct Boolean searches. As the Boolean logic is composed of AND, OR, and NOT, the One EC Search technology can be used in a similar fashion to search multiple terms together and filter out the unwanted terms by using + or – in the query string. An example would be "Speakers +2.1 +Bluetooth +SONY –TV –Bar". Therefore, it provides a very compelling way of narrowing down the search result without using any categorical filters. However, the Boolean search is not commonly used by online shoppers as they are not aware of such functionality or they do not know the right keywords to insert to the Boolean logic. Consequently, it is a completely new opportunity that requires careful user consideration in designing the solution. For this report, the Boolean Search will be drawn for further design.

Table 2: SWOT analysis of new category mapping versus Boolean search

Strengths		**Weaknesses**	
New Category Mapping	Boolean Search	New Category Mapping	Boolean Search
Provides a similar system as the existing search functionality in terms of filtering mechanism, and user flow.	Does not interfering with exiting categories and does not require the use of categories to narrow down. It is also easy to implement the solution.	Manual human work, required in order to maintain the quality of the categories. More resources are required.	Introduces a new user flow that users will have to learn and adapt to.
Opportunities		**Threats**	
New Category Mapping	Boolean Search	New Category Mapping	Boolean Search
Once created, the other categories can potentially be merged all into one as a master categorizing system	The solution can be adopted on top of the category system without conflicts. They can potentially co-exist.	Potentially creates more complication on top of the three currently existed categories from the three platform.	Users may not appreciate the new feature and thus may not be used after all.

3

Design Process

3.1 Targeted User

The design process at Yahoo starts by analyzing the user group in which a new solution would like to target. This will affect how the solution is designed. There are two general groups of potential users: the major user group and one of the minorities. Choosing the current majority as targeted user would indicate that the new features to be provided is served as an enhancement to retain these users. On the other hand, choosing the minorities would set a goal for the new feature to attract more of these users and increase the percentage among the distribution. Based on an internal data provided by Flurry Analytics in Table 3, the current majority of users for all Yahoo apps in Taiwan are between the ages of 35 to 45. The user group with ages of 25 to 34 trails behind as second, while the rest: 13-17, 18-24, and 45 and above are the minorities.

Table 3: Age group percentage across Yahoo News, Shopping, Marketplace, and Auctions for June and July 2016. (Retrieved from Flurry Analytics)

News			**Shopping**			**Marketplace**			**Auctions**		
Age	Users	%	Age	Users	%	Age	Users	%	Age	Users	%
13-17	3053	2.2%	13-17	7266	3.8%	13-17	14651	8.0%	13-17	7637	3.6%
18-24	12107	8.5%	18-24	24338	12.8%	18-24	35111	19.2%	18-24	29524	13.9%
25-34	37137	26.2%	25-34	51677	27.3%	25-34	45079	24.6%	25-34	68094	32.0%
35-54	78282	55.2%	35-54	97665	51.5%	35-54	79859	43.6%	35-54	101493	47.7%
55+	11253	7.9%	55+	8573	4.5%	55+	8550	4.7%	55+	6244	2.9%

The distribution of smartphone users across different age group in Taiwan can be graphed as Figure 2, which interestingly displays a similar trend as the curve of adoption proposed by Roger [1]. In fact, the user group of ages from 18 to 24 can be compared to the early adopters due to the fact that they are more aware and appreciative of changes [1]. Therefore, they play a very important role in a mobile product's life span. However, after

graphing Yahoo's usage rate of different age groups, as shown in Figure 3, Yahoo's products only attract the second half of the parabolic arc in Figure 2, and misses the most important target group of the market. Consequently, this solution chooses to target the age group of 18 to 24, the minorities, in hopes to capture the first half of the arc, and creating a more balanced distribution.

Figure 2: Smartphone ownership rate by age in Taiwan [2]

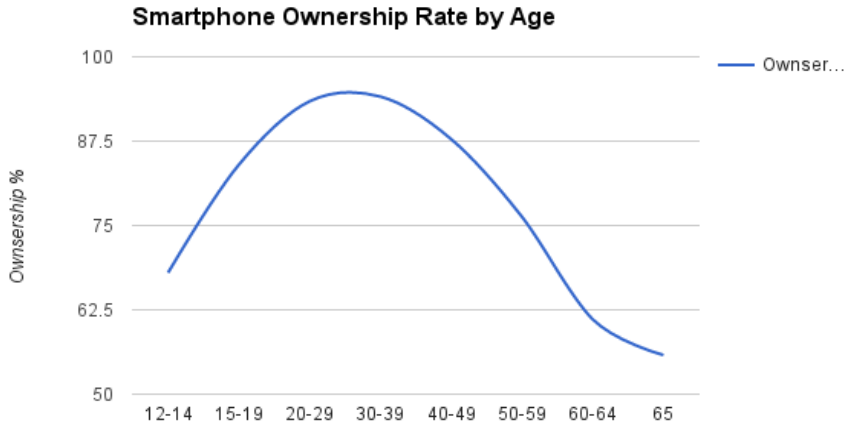
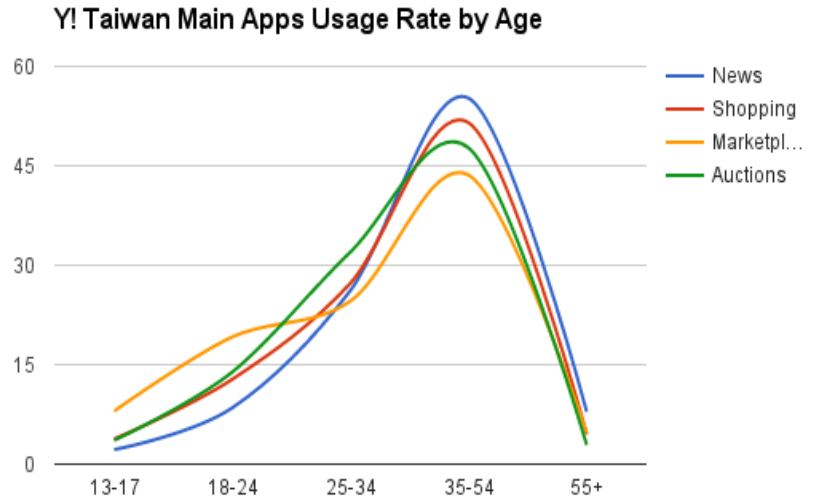


Figure 3: Yahoo e-commerce apps' usage rate by age in 2016 provided by Flurry



3.2 Design Requirements and Constraints

The design requirements and constraints are driven by the selected targeted user group as well as the opportunity statement of Boolean Search previously developed during the problem analysis. They will be referenced in the next section for justification of the features. These criteria are also evaluated against Nielson's design principles to maximize the usability of the feature [3].

Below are the main design requirements for the solution:

- R1. The solution must allow the user to utilize the existing Boolean Search functionality in One EC Search. This translates to the ability to insert wanted and unwanted terms into the query. It should also be integrated to the existing user interface to provide the most familiar user journey.
- R2. The solution must advise users with the most relevant and appropriate terms that can be inserted to the Boolean Search. These suggested keywords must be expectable and relevant to the searched query term in order to establish a predictable mental model in users [3].
- R3. The solution must eliminate as much manual work from the user as possible, and automate the process of typing. Any typing done by the user usually takes up the most time and effort, thus should be minimized as much as possible.
- R4. The solution must follow up and provide feedback on users' actions such that users are aware of their doings [3]. This helps the users to easily understand the functionality of Boolean Search and be able to adapt to it.
- R5. The user interaction must be as simple and intuitive as possible, building on top of the existing user flow. It should not degrade user's existing search experience.
- R6. The solution must provide quick ways to undo actions or go back to a previous state in order to compare the search result with different query terms inserted [3].

Below are the main design constraints for the solution:

- C1. The solution must not conflict with existing user interaction. It should be a feature that can be adopted easily on top of the current interface.
- C2. The Boolean Search must not be confused with normal search by the users. A clear distinction of Boolean Search and normal search must exists visually.

- C3. The users should not have to type out any provided keywords manually. Only the keywords thought by the users should be manually entered.
- C4. The users must not feel lost using Boolean Search at One EC Search. There must be an evident connection between the keywords inserted to Boolean Search, and the displayed search result.
- C5. The users does not ideally have to rely on filters anymore, as the solution aims to provide an alternative to using filter. Thus, the user journey of using our solution should not require the use of any filtering mechanism.

3.3 Features and Functional Analysis

Despite the heavy emphasis on Boolean Search, the main vision of the solution still focuses on helping user narrow down within One EC Search in general. Therefore, the features that are developed on the basis of the requirements and constraints involve additional functionalities that all together speeds up the process of narrowing down. The new features that are covered by the solution include automatic keywords generation handled behind the scene and finally the new user interface that will utilize Boolean Search.

Prior to designing the feature, it is important to understand the behaviour of the current users. For the scope of this solution, it covers the journey from user searching for an item of interest to narrowing down to a few choices. In this case, the common behaviours are scrolling, comparing items and narrowing down with new searches. Despite the scrolling gesture is a natural instinct for smartphone users due to the influence of social media apps, e-commerce apps should not be considered equally. Unlike social media apps, the listings on an e-commerce app are normally sorted by relevance. As users are constantly having to scroll down, it indicates that the search result presented to them is not relevant or appealing, and requires an improved result. Therefore, the most concrete way of narrowing down is based on the traits of items that are originally presented to them. This allows users to take immediate action when they see specific keywords that are either wanted or unwanted, from items that they would otherwise scroll pass. In short, the design encourages users to take actions to help them filter down to more appealing items, rather than scrolling down endlessly in a cluttered search result.

Keywords Generation

Before a Boolean search is conducted, the most important factor is the keyword that are chosen. However, the best available word to enter as a keyword in the Boolean logic is not normally intuitive to light e-commerce users. Therefore, the first feature of the solution is to provide users with the right keywords automatically (R2). With the suggested keywords available, users can then select keywords as wanted or unwanted through the user interface as they will be added to the Boolean Search without requiring any additional input from the users (R1).

The source of the keywords can vary, although they are most commonly derived from the titles of the item listings. This intuitively makes sense as the search algorithm executed by the search engine is heavily dependent on the comparison of the title and the queried string. Thus, many potential useful and expectable keywords can be extracted from a listing's title (R3). Table 4 demonstrates such idea using examples from Amazon and the keywords that can be extracted from their original titles

Table 4: Few example of titles from Amazon and their manually extracted keywords

Listing Title	Possible Keywords
"Cooler Master Storm QuickFire TK - Compact Mechanical Gaming Keyboard with CHERRY MX RED Switches and Fully LED Backlit"	Cooler Master, Storm QuickFire TK, Mechanical, Gaming, CHERRY MX RED, LED, Backlit
"Parasom A1 Magnetic Bluetooth Headphones, V4.1 Wireless Stereo Bluetooth Earphones Sport Headset In-Ear Noise Cancelling Headphone Earbuds for Gym Running -Sweatproof, Microphone (Black/red)"	Parasom, A1, Bluetooth, Wireless, Stereo, In-Ear, Noise Cancelling, Earbuds, Sport, Microphone
"ASUS F555UA-EH71 15.6 Inch, Intel Core i7, 8GB, 1TB HDD Laptop, Windows 10 (64bit)"	ASUS, F555UA-EH71, 15.6, Intel Core i7, 8GB, 1TB, Windows 10

Apart from the title, the keywords can also be obtained from the cluster data provided by Shopping. As briefly mentioned earlier, the cluster data are available to Shopping due to its small amount of items and organized structure. Each item posted on Shopping's site

are required to provide details of the item based on the cluster data for the specific category the item belongs to. Thus, more potential keywords can be obtained by retrieving the cluster data of the category. For instance, the category of laptop would contain data such as the brand, the processor, the RAM and HDD size, the operating system etc. All of these information can be used as keywords during a search. Although such cluster data is exclusive to items in Shopping and is used as a filter, it can be leveraged to improve the quality of the keywords when analyzing the listings' titles in One EC Search.

Finally, it can help tremendously in narrowing down by having an item's model number or product id as a keyword. It allows the users to search for a specific item in One EC Search, saving time in finding the correct model number and typing it in the search box (R3, C3). This use case generally comes near the end of the user journey where users start comparing the prices and sellers across the three platforms in hopes to find the best choice for purchase. Even though not all items have a product id, most standardized products in the market will have one, thus making it practical. Items belong to Shopping will have information on their product id as part of item's information. However, items within Auctions and Marketplace will be required to extract the model number through the title of the item's listing.

Boolean Search in UI

The current use of Boolean Search requires users' manual inputs to the search box. However, on top of not knowing the appropriate keywords to enter, it already takes significant amount of time to enter the keywords manually. Therefore, the second part of the solution is the new UI design that would allow users to easily leverage the Boolean Search feature in a timely manner (R1). Only with a smooth user interaction, can the solution be appreciated.

As the feature is driven by each individual items, the design and functionality must be adopted to each item's listing accordingly. The UI design of the solution starts by looking at the easiest hand gesture without interfering with the existing user interaction (C1). From the current Auctions, Shopping and Marketplace apps, there are three main unused gesture that can be used for the new feature: long press, swipe, and press on icon. Long press is a gesture that is commonly used to retrieve hidden information of the item which in this case connects well to the solution. However it requires additional time to trigger the event,

which can significantly slow down the process. Swipe is a gesture that is often connected to other actions on the item, such as deleting, hiding etc. It is very fast in movement, but less intuitive for users. Lastly, pressing on an icon is the most explicit gesture although it would require more visual space on the existing UI. Adding extra icon in the existing interface is a very costly action as too many visible information and functionality contradicts with a "simple and aesthetically pleasing" design [3]. Table 5 summarizes the decision based on the decision matrix developed by Pugh [4]. The swipe gesture is proven to be the better choice of all.

Table 5: Decision Matrix of choosing the best hand gesture

Issue: Choosing the best gesture to be used for the feature		**Long Press**	**Swipe**	**Explicit Icon on listing**
Criteria	Weight			
Speed	25	-1	1	1
Space Required	30	1	1	-1
Ease of Use	20	0	1	1
Intuition	25	1	0	1
	Total	1	3	2
	Weight Total	**30**	**75**	**40**

The concept of adding a wanted and unwanted keywords are triggered by the user's liking of an item. If a user likes a trait from an item, the user can add that trait as a keyword to the new search. On the other hand, if a user dislikes a trait from the item, the user can add that trait to be filtered out in the new search. This liking and disliking system is relatable to a popular dating and matchmaking app, Tinder, which also uses a very smooth swiping mechanism. In Tinder, users use the basic left-right-swiping interaction to show interest to online profiles for potential matches, as shown in Figure 4.

Figure 4: Tinder's UI of swiping left (L) and swiping right (R)

Considering that Tinder is the most popular dating app for the age group of 16-34, encompassing our targeted users, it is favorable to adopt a similar swiping system [4]. By establishing a connection to Tinder, it develops an "effective mental model" in the users and allowing for the "transfer of knowledge" [3]. Therefore, the main user interaction for the feature would involve swiping right when user likes an item, and swiping left when user dislikes an item. Below is the concept of adopting Tinder's UI to the listings.

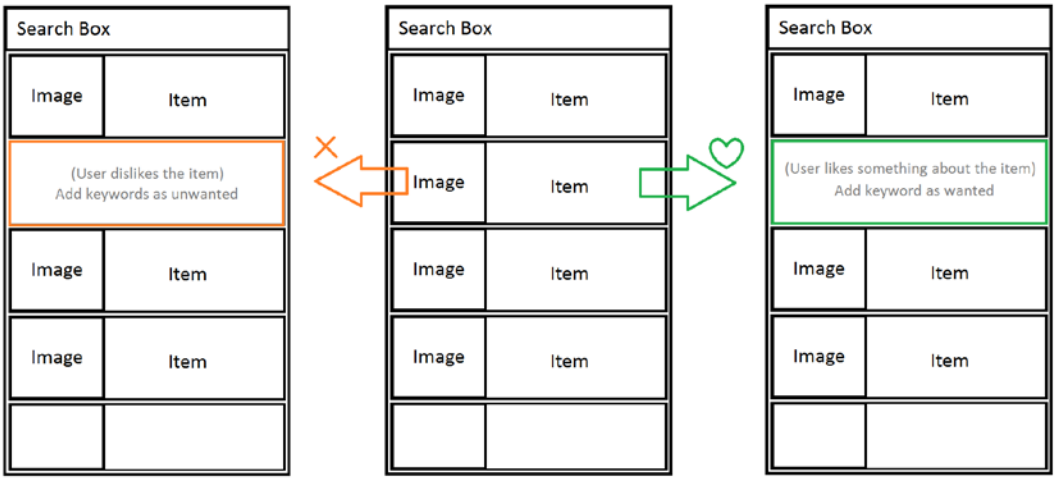


Figure 5: Concept of adopting Tinder's swiping mechanism on each item

Having the ability to swipe left and right on each listing implies three times the space to display information. This is very dangerous as it can easily overwhelm the users with information and inconsistency. Therefore, the design must be as simple as possible where swiping right and left will have identical layout and logically grouped functions (C4) [3]. When an item is swiped, the keywords that are associated with the item will be displayed.

Furthermore, to automate the process of typing the keyword into the search box, every keyword can be directly pressed which will be added to the search box automatically. However, if the keywords are directly added in the search box, not only will the search box overflows easily, it will be hard to differentiate the actions done by the user and the actions done by the app. This does not satisfy C2 as users are easily confused from the poor visibility of the system status [3]. Thus to separate the actions done by both parties, a tagging system is introduced and used in conjunction with the keywords. Upon pressing the keyword, it will be added as a tag below the search box as conceptualized in Figure 6.

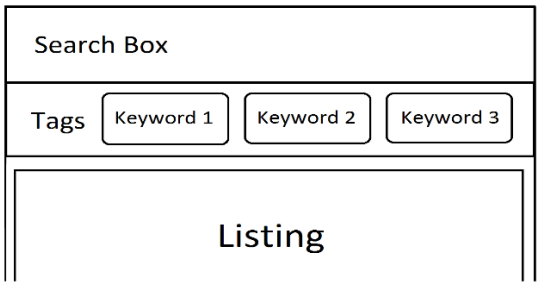


Figure 6: Visual concept of the tagging system

This allows the keywords to be added without interfering with the original queried string entered by the user in the search box, in addition to the ability of removing any tag immediately (R6). Finally, when a tag, or keyword, is added, the search result page will be refreshed and narrowed down to provide immediate feedback to the users (R4).

The last part of the solution addresses C5 by integrating some relevant and existing narrowing down functionalities into the swiping system. These existing functionalities include the previously mentioned category system, as well as the ability to search similar items from one. Despite their needed improvements, they are added to retain the previous filtering functions while unifying all into the swiping system. If the user swipes right, or likes the item, the user can narrow down to its specific category or search for similar items. Likewise, if the user swipes left, or dislikes the item, the user can filter out the belonging category or other similar items from the search result. It is important to understand that these small functionalities are added in the assumption that they will continue to be improved. They will be implemented but not addressed further in the solution.

4

Prototype Implementations

4.1 Backend API

As keywords generation is the first part of the solution, we start by implementing the backend structure of the feature. The implementations include title segmentation, data comparison and model extraction. Since the front-end UI will be built in Android, the back-end APIs are written in Java. In addition, most data queried at Yahoo are done through YQL (Yahoo! Query Language), including all the e-commerce platforms, which will be used in our implementation.

Title Segmentation

Most of the important keywords come from the title of the item since it maximizes the probability of being searched by a user. To extract associated keywords, text segmentation must be performed on the title. Academia Sinica, Taiwan's central research academy, has developed and open sourced a text segmentation API specifically used for strings in Chinese characters. On top of segmenting the texts, it also defines the word class for each word it has segmented [6]. Table 6 shows a couple examples of Chinese texts, the results of the text segmentation, as well as their English translation. For this prototype, the API will be used to extract keywords from the title.

Table 6: Text segmentation results from Academia Sinica's API

Listing Title	Segmentation Result by API
鐵三角 ATH-CKR3 高音質密閉型耳塞式耳機 *(English: Audio Technica ATH-CKR3 High Quality Noise Cancelling In-Ear Earbuds)*	鐵三角(Na) ATH-CKR(FW) ３(Neu) 高音質(Na) 密閉型(Na) 耳塞式(Na) 耳機(Na) *English: Audio Technica, ATH-CKR, 3, High Quality, Noise Cancelling, In-Ear, Earbuds*
雷蛇 2014 黑寡婦機械式鍵盤 專業版 綠軸 中文 *(English: Razer 2014 Blackwidow Mechnical Keyword Professional Cherry-MX-Green Chinese)*	雷蛇(Nb) ２０１４(Neu) 黑寡婦(Na) 機械式(A) 鍵盤(Na) 專業版(Na) 綠(VH) 軸(Na) 中文(Na) *English: Razer, 2014, Blackwidow, Mechanical, Keyboard, Professional, Cherry-MX, Green, Chinese*

With every call to One EC Search's YQL, a JSON-formatted search result that contains each listing's information is returned and stored in the program. Rather than displaying to

the users right away, each title of the listings is extracted from the JSON Object and segmented by Academia Sinica's API. Only the nouns and adjectives will be returned to the program as potential keywords that will be further analyzed in the next stage.

Data Comparison

The second stage of the implementation involves data comparison with the keywords that are segmented by the API in order to improve the quality of the keywords. To begin with, every listing from Shopping will be searched against Shopping's YQL to retrieve the cluster data behind the item. As the cluster data contains possible keywords, these data will be compared to the keywords previously segmented. If a keyword exists in the cluster data, it will be recognized as a quality-keyword. The keywords that have yet to be qualified will be further analyzed using another algorithm that runs through the segmented keywords of the first hundred listings and look for the most repeated keywords that are not in the cluster data. If the segmented keyword is not in Shopping's cluster data but repeats multiple times throughout the search result, it will be recognized as a quality-keyword as well. In the end, only the qualified keywords will be returned as the final list.

Model Extraction

The last stage of the back-end development is independent to the previous stages and involves the extraction of items' model numbers. For listings from Shopping, the model number can be retrieved easily using Shopping's YQL in a similar fashion to previous step. However, for listings from Auctions and Marketplace, each title must once again be dealt intelligently using regular expression. A few regular expression rules are developed for this case based on various possible formats of model numbers in general. These rules are also prioritized such that the most common format of model numbers are matched first. Figure 7 shows the four rules that are implemented. Based on these rules, each title will be matched against the regex and return the most probable model number.

```
//Rule 1 (Most Common): Ex. MDR-Q38LW
Pattern pattern1 = Pattern.compile("([a-zA-Z]+\\-*[a-zA-Z]*[0-9][a-zA-Z0-9\\-]*)|([0-9]+\\-*[0-9]*[a-zA-Z][a-zA-Z0-9\\-]*)");
//Rule 2: Ex. Beats urBeats
Pattern pattern2 = Pattern.compile("[A-Za-z]+[ ]+[0-9a-zA-z]+");
//Rule 3: Ex. SHS3200
Pattern pattern3 = Pattern.compile( "[A-Za-z][A-Za-z0-9]{2,}" );
//Rule 4: Ex. GTX 980
Pattern pattern4 = Pattern.compile("[A-Za-z][A-Za-z0-9]{2,}[ ][A-Za-z0-9]{2,}");
```

Figure 7: Regular expression patterns that are implemented for model extraction

4.2 Frontend UI

The two main UI components that are newly implemented for the solution are the swipe layout and the tagging system. As the One EC Search App is technically non-existent, a search interface connected to the YQL is built, prior to the implementation of the solution, to stimulate an e-commerce search environment. Figure 8 demonstrates a normal search result page that users will be interacting with.

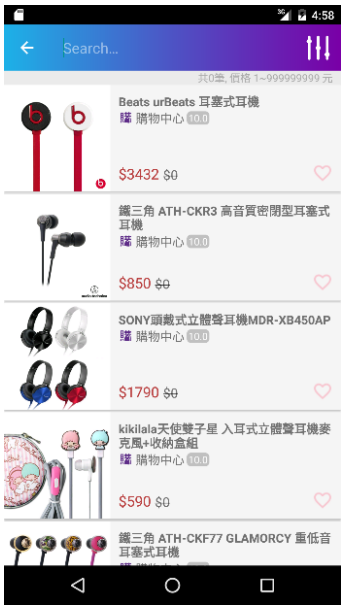


Figure 8: Temporary search result page implemented for the solution

Swipe Layout

For this prototype, an open sourced AndroidSwipeLayout developed by Daimajia is used to enable swiping in both directions [7]. The right and left layout have identical structures with two buttons for the existing functionalities, and the remaining space for the keywords. The two layouts are differentiated in a few ways to make the users aware of the difference. To start off, they have different background colors with the right layout having a blue background, and the left layout having a purple background. These colors are chosen to best fit the temporary design of the search result page. Moreover, when a user starts swiping the layout, an icon will appear based on the direction the user is swiping to indicate the actions about to be taken. For swiping right, a “+” icon will appear. For swiping left, a “–” icon will appear as demonstrated by Figure 9. These icons are inspired by the logics behind Boolean Search and provides more intuition to the design.

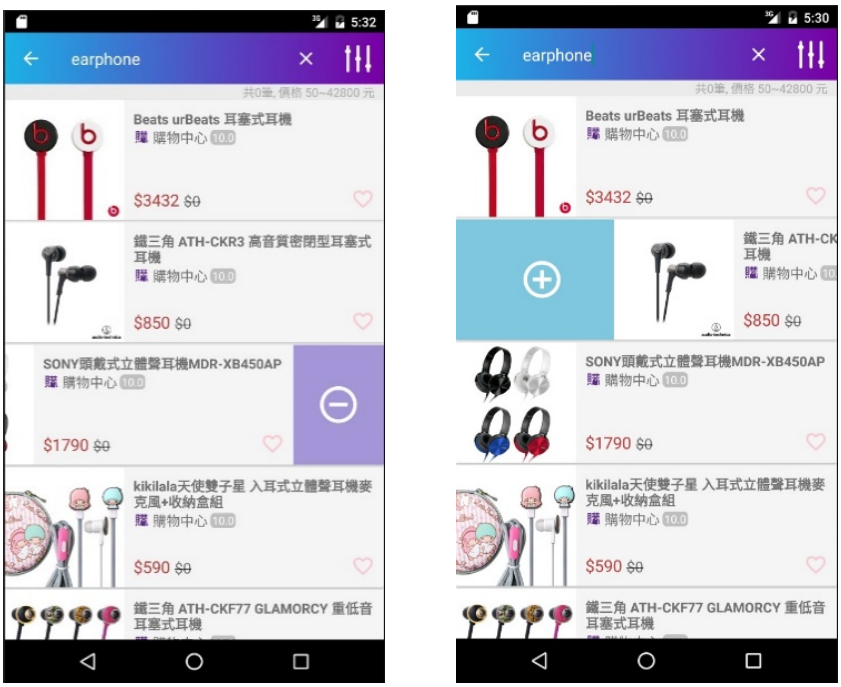


Figure 9: Icon displayed when swiping left (L) vs swiping right (R)

Every keyword extracted from each listing will be displayed chronologically in a Flexbox, developed by Google, within its swiped layout. The Flexbox enables the keywords to be added continuously and be wrapped automatically in a given space [8]. However since the number of keywords varies for any listing, the case must be handled when too many keywords exist resulting in an overflown Flexbox. To overcome the problem, a custom implementation of the Flexbox is created to dynamically change the size of each keyword until all keywords can fit and be displayed within the Flexbox. Figure 10 consists of the detailed explanation of the Swipe layout.

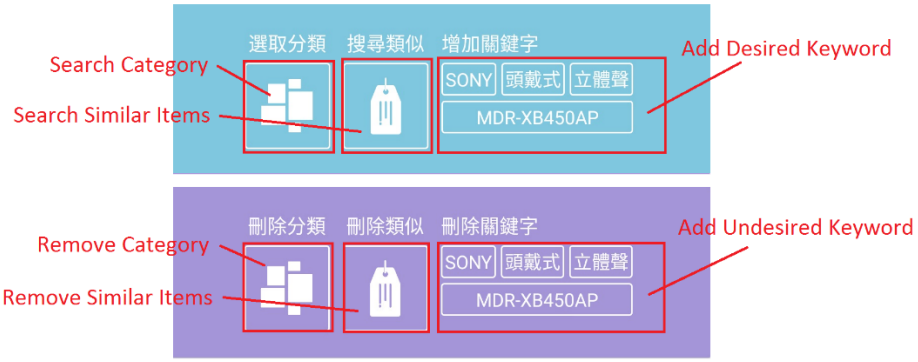


Figure 10: Diagram of the swiping layouts

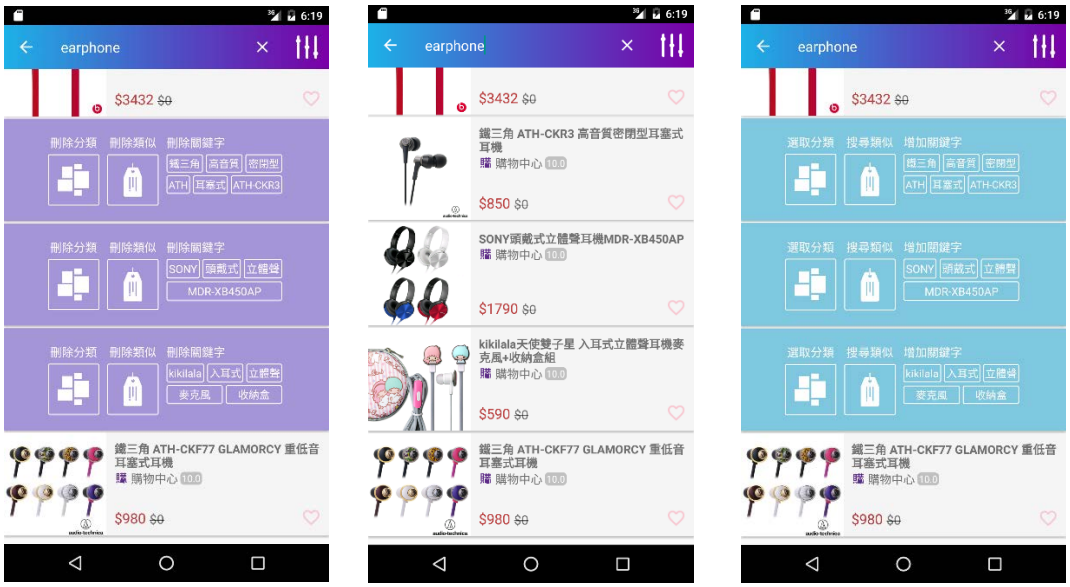


Figure 11: UI overview of after swiping left (L) and after swiping right (R), compared to the original search result (C)

Tagging System

A tagging system is needed to support the keywords. It is implemented below the search bar as a horizontal scrollable RecyclerView such that the amount of tags is not limited. When the tags overflows, the users can scroll horizontally to see the rest of the tags. Furthermore, the tags are arranged such that the wanted tags are always at the front, followed by the unwanted tags. This is done based on a quick user study that suggests the desired keywords are more likely to be viewed than the undesired keyword. Each tags are also differentiated by the same icons and colors as the layouts to visually connect to the swiping layout, as displayed in Figure 12. Finally, the tags are collapsed when users are scrolling down the search result in order to save the space taken by the tags.

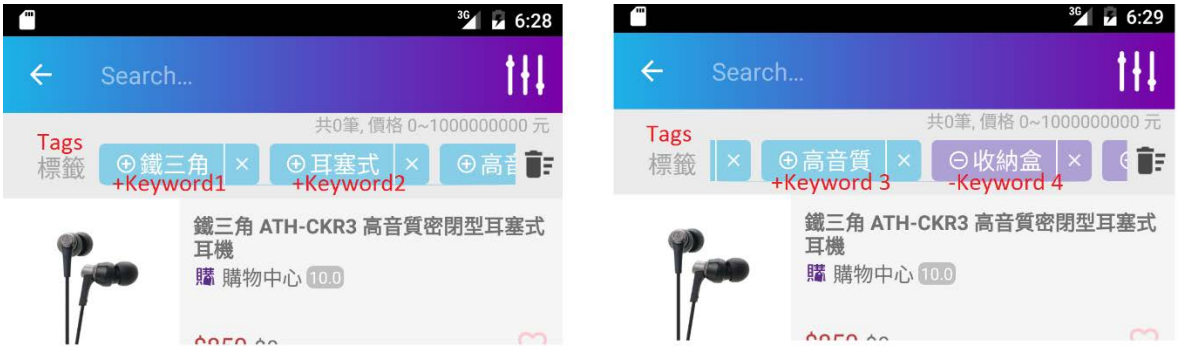


Figure 12: UI of the tagging system

5

Discussion

Once the prototype is completed with all the desired functionalities, it is evaluated against its performance and impact on the One EC Search journey. There are three main focuses of evaluation for this solution: the quality of the generated keywords, the overall speed, and the user experience when using the new feature. The keywords quality are assessed by the accuracy and usefulness in comparison to its title, while the speed and user experience are measured through user testing.

5.1 Keywords Quality

The quality of the keywords extracted through the designed algorithm overall appears to be positive. It is especially excellent for standardized products such as electronics that have very specific keywords powered by Shopping's cluster data. Even listings from Auctions and Marketplace are able to generate rich keywords that can help users narrow down. In addition, the model numbers are extracted accurately from all three platforms, if they are contained within the titles. In some rare cases where the model number is not included in the title, the algorithm returns an incorrect model number, which should be handled to avoid incorrect information. Furthermore, the keywords are more limited for non-standardized products such as clothing and accessories that are too flexible and descriptive in characteristic. To improve the quality of the keywords for non-standardized products, it is recommended to manually provide a list of custom keywords associated with this category. It establishes a standard in the quality of keywords returned by these products.

5.2 Search Speed

To analyze the impact on the speed, a few user test are conducted and timed as recorded in Table 7. The test is conducted by giving the users a specific listing that the users must look for starting by searching its generic keyword, to stimulate a more realistic search case. The users must then look for the cheapest available listing of the same item. The task must be completed with and without the new feature, and searching the listing directly is not allowed. The users are only allowed to search a term if it exists within the search result.

Table 7: Recorded time from the user tests

Generic Keyword	Time without using the new feature	Time while using the new feature
Headphone	128s	75s
Keyboard	111s	78s
Monitor	154s	93s
Camera	163s	112s
Lipstick	144s	129s

Based on the data recorded, there is an evident reduction of time, specifically for electronic products. In the old fashion way, most users are forced to rely on the existing category system and scroll extensively deep into the search result. However, with the new feature, users can easily retrieve keywords from a listing and right away narrow down with the implicit Boolean Search feature without requiring any manual inputs.

5.3 User Experience

Among those who have tested the solution in the previous stage, most of them have also provided feedbacks on their experiences while using the app. Due to the fact that it is a completely new feature, it appears to require a significant amount of time to adapt to the feature, and leverage the most out of it. A tutorial is definitely recommended to have for first-time users. Furthermore, some users who are used to the top-down filtering approach are less appreciative of the new bottom-up solution where the filtering selections happen within each listing. By having the keywords outside, as a global property, it can potentially be more effective and straightforward to those users. However, as the solution aims to introduce a new user flow in narrowing down, it should not directly be concluded that a new method of displaying the keywords is necessary.

The user interaction are well received and have met the design criteria. The icons and colors that are displayed while swiping are great help of differentiating users' actions, making it more intuitive for the users than what was expected. The complementing tagging system also works very cohesively with the adding of keywords, allowing the users to be in complete control of the Boolean Search and not lost while using the feature.

6

Conclusion and Recommendation

One EC Search is an essential and powerful tool for Yahoo's e-commerce market with many potential applications such as the upcoming Yahoo App. It allows users to view the listings from all Yahoo's e-commerce platforms, rather than three different applications. While One EC Search provides great convenience to online shopping, the massiveness and disorganization of the data have significantly degraded the overall user experience. Therefore, the vision of this solution is to provide an alternative filtering mechanism that can allow users to narrow down effectively in One EC Search.

Despites some improvements in need for the solution judging from the prototype, the goals are definitely met by leveraging Yahoo's existing technology of Boolean Search. Boolean Search has always been an advanced feature implicitly embedded to the search functionality, and not exposed to normal users until the intent of this solution. Consequently, it is certainly a new user experience required to be adapted by the users as observed from the user tests. However once the users are familiar with the feature, the Boolean Search, driven by the automatic keywords generation and swiping layout, has shown its ability to greatly reduce the time and effort for the users to find the most desired items. Therefore, the core components of the solution are important for One EC Search.

Unquestionably, there are aspects of the solution to mature and improve on. Every specific functionalities should be extracted and further developed by the relevant engineering teams at Yahoo Taiwan. The keywords generation should be integrated into the backend of YQL, and managed periodically by e-commerce content editors. The user interface can be improved greatly by designers through a deeper user research and understanding of the most fitting interaction that best utilizes the Boolean Search and the automatic keywords generation. All the components in the solution shows great potential and scalability, which proves its capability in becoming a game-changing solution for One EC Search. As a result, the project has proven its success in highlighting and demonstrating its importance to be adopted into the e-commerce platforms at Yahoo.

References

[1] A. Quan-Haase, Technology and Society: Social Networks, Power, and Inequality. 2nd Edition, Oxford University Press, 2015.

[2] "Global Habit: Use of Smartphones in 14 Asian Cities," Hakuhodo, 2013.

[3] C. D. Wickens, An Introduction To Human Factors Engineering. 2nd Edition, New Jersey: Pearson Eductation Inc., 2004.

[4] F. McGrath, "What to Know About Tinder in 5 Charts," 24 April 2015. [Online]. Available: http://www.globalwebindex.net/blog/what-to-know-about-tinder-in-5-charts.

[5] Daimajia, "Android Swipe Layout," 5 Jan 2016. [Online]. Available: https://github.com/daimajia/AndroidSwipeLayout.

[6] D. G. Ullman, The Mechanical Design Process. 4th Edition, New York: The McGraw-Hill Companies, Inc., 2010.

[7] A. Sinica, "中文斷詞系統 (Chinesse Text Segmentation System)," National Digital Archives Program, Taiwan, [Online]. Available: http://ckipsvr.iis.sinica.edu.tw/.

[8] "FlexboxLayout," Google Inc., 2016. [Online]. Available: https://github.com/google/flexbox-layout.