Opera Software “Discover” Service

Research on Relevancy and Freshness of the Suggested Content and Ways to Improve It

Anton Shakhov

Bachelor’s Thesis

___.___.______ ________________________________

Bachelor’s degree (UAS)

SAVONIA UNIVERSITY OF APPLIED SCIENCES **THESIS**
Abstract

Field of Study Technology, Communication and Transport			
Degree Programme Degree Programme in Information Technology			
Author(s) Anton Shakhov			
Title of Thesis Opera Software “Discover” Service. Research on Relevancy of the Suggested Content and Ways to Improve It.			
Date	17.06.2015	Pages/Appendices	52
Supervisor(s) Sami Lahti, Jussi Koistinen			
Client Organisation/Partners Opera Software ASA			

Abstract

Opera Software ASA is a Norwegian company developing Internet browsers for a variety of devices. Starting from version 15 it includes a feature “Discover”, which is a news recommendation system created to suggest news articles right in the browser window.

The purpose of this thesis work was to conduct a research on relevancy of content in Discover and suggest improvements for the service. First, a research on the main content filtering techniques in recommender systems was done. By analysing filtering methods, different advantages and drawbacks were revealed. After covering the theory on recommendation systems, currently used filtering techniques in Discover were presented.

It was concluded that news recommendation systems gained considerable popularity over the recent years, with increasing amount of people preferring news aggregators to other traditional ways to retrieve news stories. As news aggregation is a competitive market, users have a wide choice of services to use. Considering this tendency, the importance of relevant content in Discover was considered to play a vital part in user engagement.

Prior to suggesting improvements, different relevancy criteria were presented. Those factors help to determine the significance of a news story to the general type of reader. By evaluating the current situation with content relevancy in Discover, a number of disadvantages in the current system were presented.

The final chapter of this thesis includes the suggestions for improving the relevancy of content in Discover, which are based on previously stated relevancy factors. The improvements are proposed for general recommendations, as well as for personalized recommendations.

Keywords
Opera Software, Discover, Recommender System, News Aggregator, Content Filtering, Content Relevancy, News Recommendations

ACKNOWLEDGEMENTS

I would like to thank Opera Software ASA for giving me a chance to write this thesis on the topic of Discover service.

My special thanks go to Cosimo Streppone, the developer in the Discover team, who helped me a lot during the process of research in the area of recommender systems and provided me with necessary background information on technologies used in Discover.

I would like to express my deepest gratitude to Simen Kjellin, who was my internship mentor at Opera, for his guidance and support. Without his advice on content relevancy topic, this thesis would not have been completed.

I am heartily thankful to Synne Nygaard, the Discover team manager, for her immense assistance during thesis review phase. Without her help and advice, this thesis would not be possible.

I am also grateful to Christopher James Williams, the developer in the Discover team, for his help in proofreading the thesis and checking the Discover specifics stated in this paper.

Finally, I would like to thank my thesis supervisor, Sami Lahti, for his continuous support and patience. Without his constant feedback, this thesis would not have been finished in time.

CONTENTS

1 INTRODUCTION

With the immense development of the World Wide Web, a powerful online experience became available to Internet users thanks to web browsers. Browsers help users to retrieve information and to connect with people across the globe instantaneously. Because of the enormous popularity of the Web, there is a high competition between browsers in usage share. Browser developers compete with each other by trying to provide users with the best online experience. New and innovative features help to retrieve more users and keep existing users interested in the product.

Opera software released Discover feature in their browsers to provide users with fresh stories acquired throughout the Web. Discover is a news recommendation system, which is supposed to help users to receive relevant news stories right in the browser and assist them in spending their free time.

The inspiration for this thesis came from the author's personal interest in the Discover feature. While this service was found to be riveting, several drawbacks connected with content relevancy were noticed in the existing recommendations algorithm. It was concluded that by improving the relevancy of the suggested content, Discover would be able to attract more attention and engagement from the users.

The aim of this thesis is to suggest improvements that could result in better content relevancy for the end user. The project work will be started by providing general information about Opera Software and the Discover service in particular. A research in main filtering techniques will be conducted to find out which approach the Discover service uses to select the content to be shown. After that, the concept of content relevancy will be introduced by clarifying relevancy criteria to be used for suggesting suitable news pieces.

The final part of this thesis will include the research in current relevancy problems in Discover. Several suggestions for improving content relevancy will be recommended. By implementing those suggestions, it is expected that Discover would be able to amend the present situation with relevancy and freshness and to improve the general look of its recommendations.

2 OPERA SOFTWARE GENERAL INFORMATION

2.1 History

Opera Software ASA is a Norwegian software company that is best known for its family of web browsers called “Opera”. The project originated as a research work of the Norwegian's largest telecommunications corporation Telenor. Jon Stephenson von Tetzchner and Geir Ivarsøy, who were previously employed by Telenor, founded the independent company on August 30, 1995.

The first version of the browser was released in 1997. At first, Opera browser was a trial-ware product, meaning that it had to be purchased after the end of a trial period. With version 5.0, released in 2000, Opera became ad-sponsored. Browser started to display advertisement banner that was integrated in the UI (User Interface). In 2005 version 8.5 was introduced, where advertisements were removed completely. Instead, Opera Software started to receive financial support from Google, which became browser's default search engine.

2.2 Presto and Blink rendering engines

Rendering engine is a piece of software that uses marked-up content (HTML, XML, images, etc.) and formatting information (e.g. CSS, XSL, etc.) and then combines both to display the formatted content on the screen. Layout engine is a key component of every browser.

Opera was using Presto as its rendering engine up to the version 12.16, which was then switched to Blink – engine used by Google and some other developers of web-browsers.

2.2.1 Presto

Presto was a proprietary layout engine developed by Opera Software. It was first released on 28 January 2003 with Opera browser version 7 for Windows. Presto was a dynamic rendering engine, meaning it could re-render web pages both partially and completely in response to DOM events.
Besides Opera browser itself, Presto was also used by Nintendo and Nokia browsers, which were based on Opera.

Presto was created as a competitive engine to Netscape and Internet Explorer's Gecko and Trident engines. The reason behind creation of Presto was to enhance interoperability across the Web and to push the Web forward.

2.2.2 Blink

Blink is an open-source (BSD v2.0 and GNU LGPLv2.1) rendering engine created by Google as a part of the Chromium project. Blink contributors include Opera, as well as some other companies: Intel, Samsung, Yandex, etc. (Chromium authors log, 2015). Blink was first announced in April 2013 (Adam Barth, 2013).

Initially, the reason behind creation of Blink was a multi-process architecture of Chrome browser that was hard to support with the use of WebKit. Blink allowed to comprise more than 4.5 million lines of code from WebKit right from the start (Adam Barth, 2013). This meant that Blink would had much healthier codebase, which, in turn, leads to fewer bugs and better stability.

2.2.3 Presto → Blink switch

In 2013 Opera announced a switch to WebKit engine, together with introducing Opera browser version 15. In particular, Opera decided to use Blink engine, a fork of WebKit. This engine is also used by Google Chrome, a web-browser developed by Google.

Bruce Lawson (2013), web evangelist of Opera, explained the switch to another layout engine with the following statement:

“The WebKit project now has the kind of standards support that we could only dream of when our work began. Instead of tying up resources duplicating what's already implemented in WebKit, we can focus on innovation to make a better browser. […] We'll continue to advance the Web by contributing to the WebKit and Chromium projects. We have great experience in making products that work everywhere. In our internal builds, we've experimented with adding support for some new standards and enhanced some features that were lacking compared with Presto.”

With the change of rendering engine, Opera also introduced several new features in their browser: Stash (discontinued with version 25 and replaced with Bookmarks) and Discover.

3 DISCOVER SERVICE

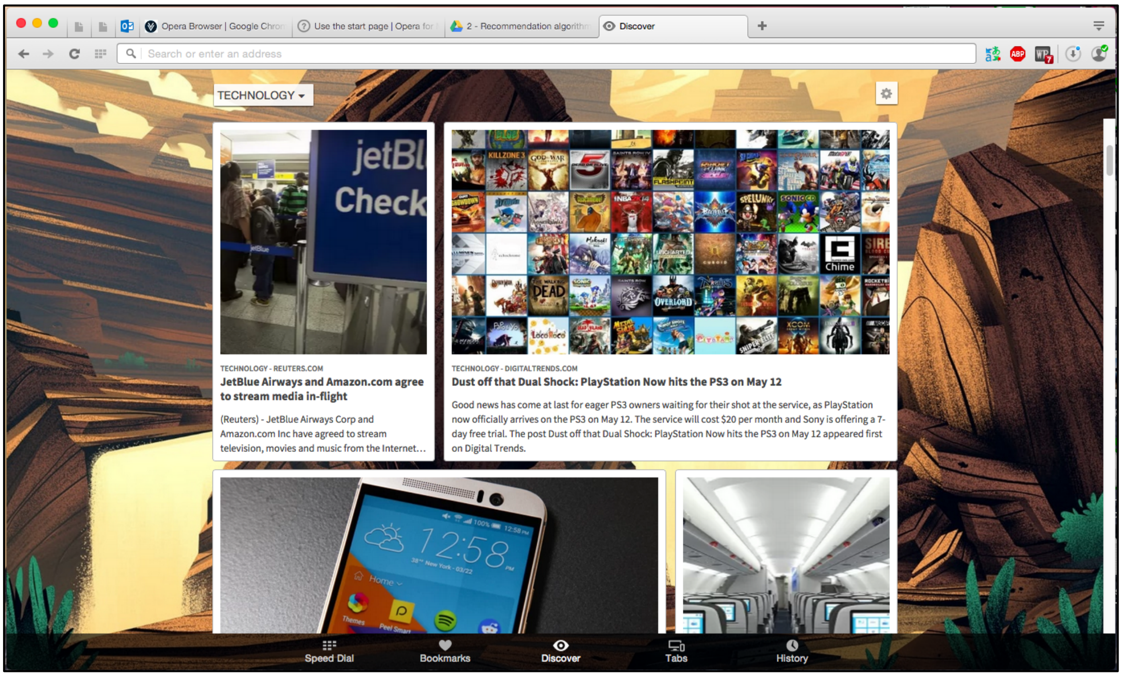
3.1 Overview

"Get hot, new content, with no browsing necessary. The new Discover feature allows you to lean back and get fed with new articles from your country, or whatever region you want to get inspiration from, right in your browser – all in one place. Pick and choose your category: news, food, technology or something else you are more interested in. Opera brings you a selection of relevant global and regional sources to discover web content more easily." (Opera Software press release, 2013)

Discover is a feature in Opera browser introduced with version 15. It allows users to get news categorized in various topics right in the browser window. The content is crawled from a variety of websites from all over the world. Discover supports 37 different countries and languages, and the content comes from more than 9000 sources (Streppone, 12 December 2014).

3.2 User interface & functionality

Below is a picture of Discover interface in Opera browser for Mac OS (version 29).


Opera Browser | Google Chro
Use the start page | Opera for
2 - Recommendation algorithm
Discover
Search or enter an address
TECHNOLOGY
Speed Dial
Bookmarks
Discover
Tabs
History

TECHNOLOGY - REUTERS.COM

JetBlue Airways and Amazon.com agree to stream media in-flight

(Reuters) - JetBlue Airways Corp and Amazon.com Inc have agreed to stream television, movies and music from the Internet…

TECHNOLOGY - DIGITALTRENDS.COM

Dust off that Dual Shock: PlayStation Now hits the PS3 on May 12

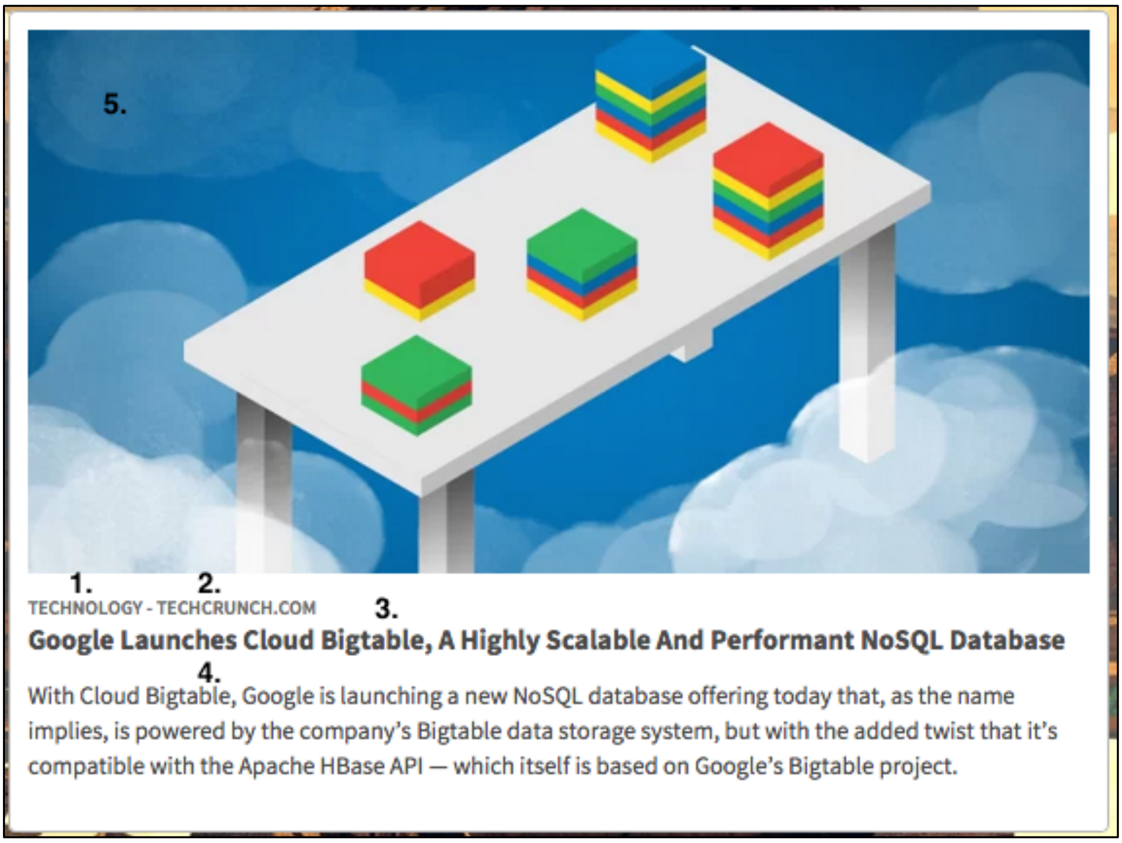
Good news has come at last for eager PS3 owners waiting for their shot at the service, as PlayStation now officially arrives on the PS3 on May 12. The service will cost $20 per month and Sony is offering a 7-day free trial. The post Dust off that Dual Shock: PlayStation Now hits the PS3 on May 12 appeared first on Digital Trends.

IMAGE 1. Discover UI in Opera browser for Mac OS.

In the center of the screen article previews, with pictures from the sources the article was crawled from, can be seen. This list supports infinite scrolling, meaning that when user gets to the bottom of the list, new content is loaded automatically.

Each preview contains the following information (see image 2):

1. Category the article is published in
2. The name of the website article was crawled from
3. The title of the article
4. Introductory text (up to 2 paragraphs)
5. Picture from the article (if present)



TECHNOLOGY - TECHCRUNCH.COM

Google Launches Cloud Bigtable, A Highly Scalable And Performant NoSQL Database

With Cloud Bigtable, Google is launching a new NoSQL database offering today that, as the name implies, is powered by the company's Bigtable data storage system, but with the added twist that it's compatible with the Apache HBase API — which itself is based on Google's Bigtable project.

IMAGE 2. Article preview card in Discover.

At the top of the Discover tab, two drop-down lists can be seen. The list to the left suggests 13 or, for some countries, 14 different categories of articles. In the list to the right user can select a country and language he/she wants to get the articles in.

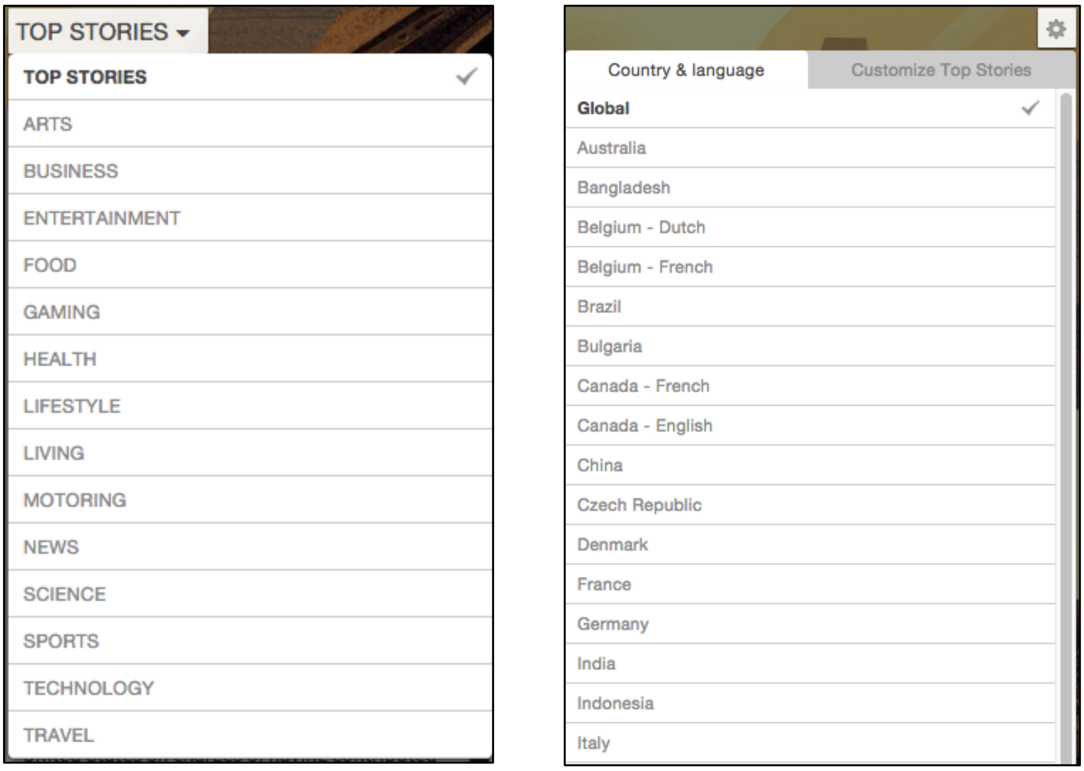


IMAGE 3. Top Stories and Country & Language drop-down lists

User also has an option to customize the Top Stories screen by selecting only the categories he/she is interested in (see image 4).

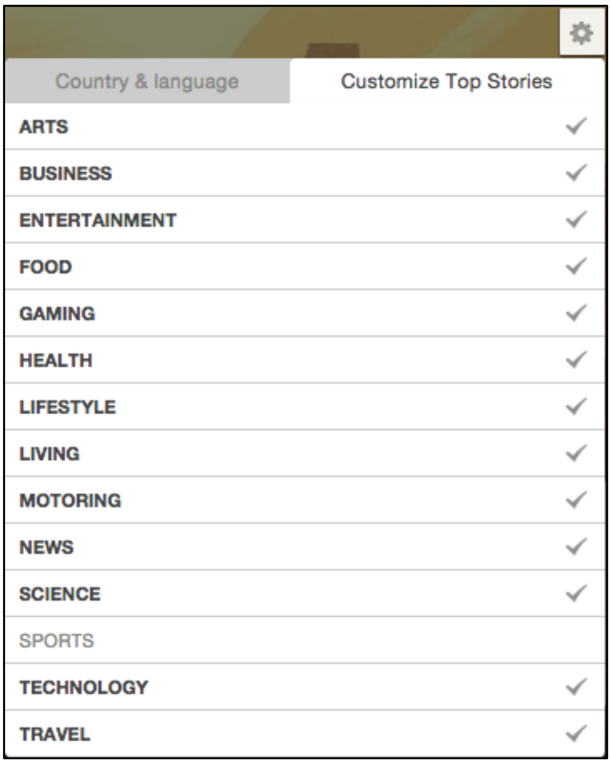


IMAGE 4. Top Stories customization screen.

After selecting a story to read, a new browser tab is opened with the URL of the website article originated from.

3.3 Discover support across various Opera products

Discover feature is supported in various products of Opera Software. These include:

- Opera browser for Windows, Mac OS and Linux
- Opera browser for Android
- Opera Mini for Android, iOS and Windows Phone (coming soon)
- Opera Coast for iOS

All of the products described above support the same functionality as a desktop version of the browser (see chapter 3.2).

User interface differs from product to product in order to represent the specifics of a particular browser. To support customization of the UI, Discover provides the API that is used by browser teams to create their own look of the Discover feature.

4 RECOMMENDER SYSTEMS

Discover feature is, in other words, a content recommendation system built in Opera browsers on various platforms. It gives users news recommendations that can be adjusted in country, language and category settings.
Before describing the specifics of Discover, the general behaviour of recommender systems will be introduced in the following subchapters.

4.1 What is a recommender system?

Recommender systems are software tools and techniques providing suggestions for items to be of use to a user (Ricci, Rokach, Shapira, 2011). RSs (recommender systems) have become extremely popular in recent years and are used in a variety of applications. Recommendations can be made for books, movies, music, news and even persons. RSs are used in order to increase sales and sell items that are more diverse, increase users' engagement and retention, and get more information on users' needs and wishes.

In order to produce recommendations, typically two different filtering techniques can be used:

- Collaborative approach
- Content-based approach

Depending on the approach, recommendations are created either by analysing user's past behaviour and the behaviour of users with similar tastes (collaborative filtering), or by looking into the discrete characteristics of the chosen item and recommending an item with similar attributes (content-based filtering). It is also a common practise to combine these approaches, creating hybrid technique of content filtering (Burke, 2002).

The above-described recommendation algorithms can often be found on websites that deal with a huge amount of information. Some examples can include:

- Spotify (www.spotify.com), music streaming platform
- Netflix (www.netflix.com), video streaming platform
- Amazon (www.amazon.com), Internet-based retailer
- Google News (news.google.com), news aggregator by Google

As can be seen, the diversity of recommendation algorithms appliance is immense. Almost every Internet service that deals with users has some kind of recommendation feature in order to prevent users from information overload and provide the most desired (according to artificial intelligence estimation algorithm) product/service/piece of information.

Each of the filtering approaches has both its advantages and disadvantages. In the following chapters the difference between collaborative and content-based filtering is described.

4.2 Collaborative filtering

Collaborative filtering (CF) is the technique used for information filtering by some recommender systems (Ricci et al., 2011). It uses the recommendations of other people in order to suggest content to the user. The idea behind collaborative filtering is that users who had similar preferences in the past are likely to have them again in the future. Considering that, CF systems collect preferences and taste information from a user base and then use retrieved information to suggest content to new users. This is the main difference of CF systems from the other systems, as it applies users' opinion to categorize information instead of analysing the information itself.

Until recently, most of the recommender systems were using CF as a main algorithm to generate recommendations to the users. User-based CF and item-based CF were the most popular algorithms in use. The model of nearest neighbourhood is applied in both approaches to find similar users/items to those already recommended. E-commerce applications and different commercial sites are the ones using neighbourhood models most of the time (Linden, Smith, York, 2003).

CF technique is not perfect and has some drawbacks that are worth mentioning:

- The Cold Start problem – before system can recommend something, user has to rate some items.
- First Rater problem – item cannot be recommended until it has been rated.
- Rare Taste problem – users with rare tastes are likely to get bad recommendations, as they are likely to not have close neighbours (users with similar tastes).
- Scalability – often recommender systems have to deal with large amount of information, which requires a lot of computation power for recommendations calculation.
- Sparsity – even when a platform has millions of users only few of them actually rate content, which can lead to popular items still having a low amount of ratings.

Nevertheless, CF approach is considered to be the most popular and widely implemented in RSs (Ricci et al., 2011).

4.2.1 User-based collaborative filtering

The original approach used in CF was user-based. The RS had to compute a weighted average of similar users' ratings for items that have not been suggested yet to the active user. After that, RS could produce some recommendations. Similarity is determined based on historical ratings behaviours.
When translated to pseudocode, user-based CF could be presented as follows:

```
user = activeUser()
sim_users = usersSimilarTo(user)
for i in itemsRatedBy(sim_users) and
    i not in itemsRatedBy(user):
         s = score(i, user, sim_users)
         recommendations += (i, s)
sortByScore(recommendations)
return recommendations
```

This algorithm will work well if RS will be able to find a large data set of similar users who rated the item. In general, the less information an algorithm can get about the interest of a particular user, the more data will be required to produce a solid list of recommendations.

To illustrate how user-based CF system makes recommendations, consider the simplified example below. A data set with some ratings for different items is presented in Table 1, where it is required to predict whether *User 5* will like *Item 3* or not. When using CF, the system will first decide on users that have similar tastes to *User 5*. That will be *User 2* and *User 3* (ratings for *Items 1*, *2* and *4* for those users are the same or not stated). Next, rankings for *Item 3* will be compared and the prediction will be made. As both *User 2* and *User 3* don't like *Item 3*, it is likely that *User 5* will not like *Item 3* as well.

TABLE 1. Data set of item ratings for different users

User/item	Item 1	Item 2	Item 3	Item 4
User 1	+	–	+	+
User 2		+	–	–
User 3	+	+	–	
User 4	-		+	
User 5	+	+	? → –	–

Despite the popularity of user-based CF, it has a number of drawbacks related to scalability and real-time performance. The complexity of computations to be done in order to recommend content grows linearly with the amount of users. In addition to that, it is also hard to explain recommendations done by user-based filtering, meaning that this approach can sometimes produce unpredicted results.

4.2.2 Item-based collaborative filtering

Item-based collaborative filtering was originally developed by Amazon in order to address the problems of new user ramp-up (cold start) and scalability (Linden et. al, 2003) present in user-based CF approach.

As opposed to user-based CF, item-based technique starts to determine a recommendation by looking up the items that active user has already rated. After that, RS computes a score for the other items present in the data set based on the similarities and ratings to already rated items. Rated items that are more similar to the candidate item have a stronger influence on the final score of this item.

When translated to pseudocode, item-based CF could be presented as follows:

```
user = activeUser()
userItems = itemsRatedBy(user)
for i in itemsSimilarTo(userItems):
    s = score(i, user, userItems)
    recommendations += (i, s)
sortByScore(recommendations)
return recommendations
```

The respective user-rating vectors are used in order to compute the similarity between two items. Thus, when a user gives similar or same ratings to two items it will be interpreted as a similarity. The more similar the co-ratings are, the more alike are the items.

This approach has some major advantages when compared to user-based filtering. Firstly, the number of items to compare is usually much smaller than the number of users. It means that comparing items is a less expensive task in terms of computation. Moreover, items usually have more ratings than users do. This means two things: better recommendations and less scalability problems, as new ratings will not greatly affect the item-item similarity. Thus, similarities do not need to be updated in real time and can be computed every *i* minutes/hours/days.

Although item-based CF overcomes some disadvantages of user-based approach, it still cannot provide novelty and personalized recommendations. Moreover, some experiments show that item-based filtering provides poorer recommendations than user-based CF (Mild, Natter).

4.3 Content-based filtering

Another approach to recommending items is content-based (CB) filtering, which uses features associated with the chosen item in order to find similar items that user might like as well (Ricci et al., 2011). In other words, this filtering technique creates associations between the items in a data set. When a user choses one item from the collection, the system compares its characteristics to the others. Items with the highest score of similarity are then presented as recommendations. RS that are based purely on CB filtering ignore the preferences of other users.

Genre, subject matter or keywords can be used in CB systems in order to describe the item. In general, the more terms a RS can consider before it produces recommendation, the better the recommendation will be.
For user profile creation, the system analyses preferences of an active user as well as the history of user's previous interaction with it.

Nowadays, information that consists only of text can be easily interpreted and categorized automatically. For other types of information (i.e. images, music, movies), more complex operations are required to perform categorization. In order to automate the process of describing the features of a text-based item, an item presentation algorithm can be applied. Term frequency-inverse document frequency algorithm (tf-idf), for example, is one of the widely used methods helping to reflect how important a word is to a document in a collection (Rajaraman, Ullman, 2011). As to multimedia items, those are usually categorized manually by humans.

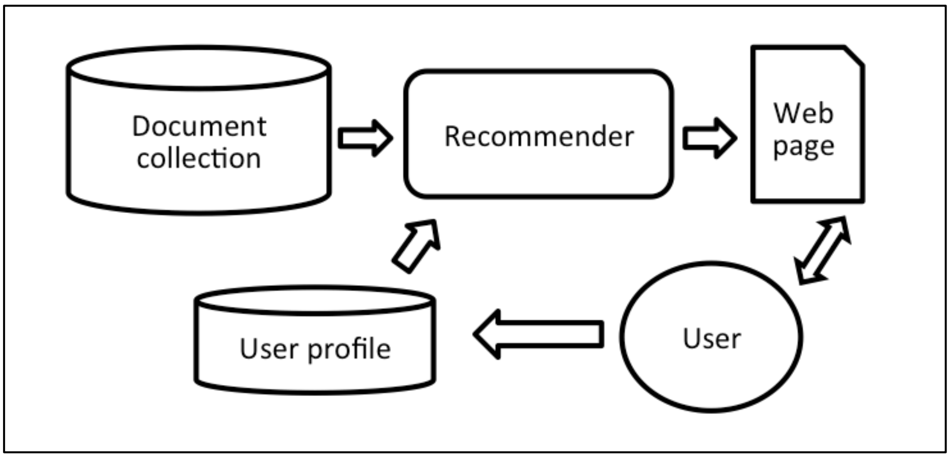


IMAGE 5. Content-based RS architecture

Image 5 shows an example of a simplified architecture of a content-based recommender system implemented on some web page. *User profile* is created by analysing the feedback from the *user*. *Recommender system* compares *user profile* with the *document collection*. After that, documents are ranked using some chosen/present criteria and returned to a *web page*, which contains the results -- final recommendations for the user.

Content-based filtering algorithms have their disadvantages. Some of them include:

- Limited content analyses – a problem of small amount of attributes being assigned to an item. This can especially happen with multimedia items, where attributes are usually assigned by humans.
- Overspecialization – a problem that happens when a system makes its decision taking into account a limited number of features of an item. This can result in recommendations that are too alike to what the user has already seen or recommendations that are almost identical to each other. However, sometimes this issue can be resolved by introducing the randomness factor.
- New user problem – newly registered users are problematic for the RS, as it will not have adequate user profile information and therefore will not be able to produce reliable recommendations (Adomavicius, Tuzhilin, 2005).

4.4 Hybrid filtering

Hybrid recommendation system is the system that combines Collaborative and Content-based filtering or other types of RS (i.e. Demographic, Knowledge-based, etc.) (Burke, 2002). Depending on the requirements, hybrid filtering system combines the advantages of other filtering techniques in order to compensate the drawbacks present when using only one filtering technique (Murat, Şule, 2010). Most commonly, collaborative filtering is combined with some other technique in an attempt to avoid the ramp-up problem. Hybrid filtering RSs also show good results when dealing with multiple-interests (one user having many different interests) and multiple-content (one item having different content) problems.

Several methods can be used in order to create hybrid filtering system (Burke, 2002):

- CF and CB filtering can be implemented separately, but the results of both filtering systems are then combined in order to provide better recommendations
- CB filtering characteristics can be integrated into CF technique
- CF characteristics can be integrated into CB filtering

- A new filtering model is built by combining CF and CB filtering advantages

4.5 Filtering techniques used in Discover feature

Discover feature uses several filtering techniques when suggesting content to the users. Filtering technique setting depends on the category users have chosen and on the settings users have provided to the system in order to receive recommendations. It is significant to note that at the moment Discover feature does not have any sophisticated recommendation algorithm and uses only basic filtering techniques. Therefore, recommendations done by Discover can be improved a lot by applying better filtering approaches.

4.5.1 Knowledge-based filtering in Discover

The main filtering technique used for recommending content is knowledge-based filtering. It uses knowledge about users and available content in order to produce recommendations that are supposed to meet user's requirements. Discover offers a choice of settings to the active user in order to produce recommendations that will be relevant to what user wants to know. Those settings were briefly described in chapter 2.2:

- Categories drop-down menu
- Country & language drop-down menu
- Top stories customization drop-down menu

These three settings affect the final list of recommendations that the user will get. In addition to the settings on the frontend side, some adjustments to the recommendations can be made on the backend with the use of Discover API. These settings are not available for the final users, but are used by the Opera products, where Discover is present.

4.5.1.1 Settings available on the frontend side

In the categories menu user can restrict the stories he/she wants to read about by selecting one of the 14 categories articles are divided into:

- Arts
- Business
- News
- Technology
- Food
- Travel
- Sports
- Living
- Gaming
- Entertainment
- Health
- Lifestyle
- Top stories
- Motoring

Depending on the setting, user will get articles on the specific topic only.

In the country & language menu user can select the country and/or language to get the articles in. Discover has 37 different countries and languages, which gives a wide variety of news stories to read. Moreover, categories filtering can be applied to any of the countries/languages.

The last setting that helps user to receive relevant recommendations, is the top stories customization menu. Initially, when user selects the Top stories category, articles from various categories are displayed on one screen, giving a variety of information to look into. However, if user is not interested in some of the categories (e.g. does not want to read sports news), he/she can opt-out those categories in the Top stories menu. After that, only articles on the remaining topics will be suggested.

4.5.1.2 Settings available on the backend side

In addition to the settings available to the end users, Discover also provides an API for clients (Desktop and mobile versions of the browser) making it possible to adjust the final feed presented to the user. Those setting are not available for the users, but are used by browser teams in order to specify which content they want to get from the Discover servers.

- Image filtering – clients have the possibility to receive articles both with and without a picture, or with a picture only.
- Sources filtering – clients can receive content from the specified sources only. Other sources available in Discover will be ignored if this parameter is used.
- Countries and categories mixing – clients have an option to mix content from several countries and/or languages. Also, different categories can be mixed for the Top Stories screen, which allows excluding some categories that are of no use to the client.
- Format filtering – clients may receive content with or without a rich summary. This means that article summaries can be omitted. In this case, only the title will be displayed to the users.
- Date and time filtering – clients have the possibility to receive articles for the specified time frame. If this parameter is present, Discover servers will return articles that were crawled in a specific time frame.
- Stories count adjustment – it is possible to receive various amounts of articles in a single request to the Discover servers. The default value is 40, meaning that the server will return 40 articles at a time. Clients can request a different value if they want to receive a bigger or smaller set of articles to display to the users.

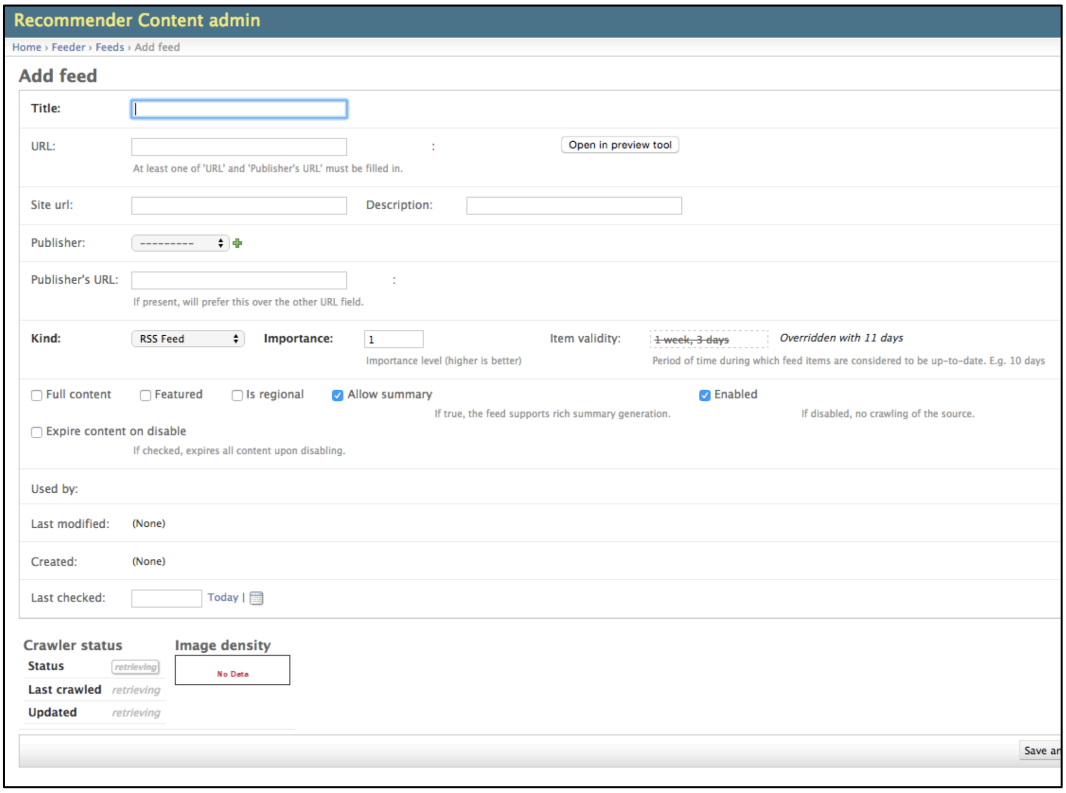
4.5.2 Content classification in Discover

Discover suggests a list of categories that users can receive articles from. There, the content is already divided into topics. The question that arises here is how those categories are created.

Initially, Discover uses RSS feeds (Rich Site Summary) in order to get a base of articles to suggest. Before providing those suggestions to the user, some content classification has to be done. In this step, content is filtered on the basis of news topics. With the current setup of Discover, this step is performed manually with the use of a CMS (Content Management System) by content managers.

Content management system is a system that allows publishing, editing, deleting and maintaining the content with the use of a central interface. The main advantage of such systems is that a person with limited programming skills can use it in order to add content with the help of a simple and understandable interface.
CMS is used by content managers in the Discover team in order to add and manage the news sources that are later presented to the users. It is also used to add and maintain categories, countries and languages to Discover. In image 6, a part of the user interface of the new source addition window is presented.

Before adding the source to Discover, a content manager first selects a website that is to be added. After that, it is needed to divide the content from the selected website to correspond with the available categories. Most of the time this is done by using separate RSS feeds provided by the website. There, the news pieces are usually grouped by topics (i.e. news, sports, entertainment). Nevertheless, sometimes it can happen that a website provides categories that are not used in Discover (i.e. parenting, housing, music). In this case, content manager has to decide whether those topics can be categorised into Discover categories or if they should be disregarded. For example, parenting and housing news can be summed up in Living category, while music can be added either to Arts or to Entertainment, depending on the content. With that done, content manager can add RSS feeds to the country the content belongs to.



Recommender Content admin

Home › Feeder › Feeds › Add feed

Add feed

Title:

URL: : Open in preview tool

At least one of 'URL' and 'Publisher's URL' must be filled in.

Site url: Description:

Publisher: ---------

Publisher's URL: :

If present, will prefer this over the other URL field.

Kind: RSS Feed Importance: 1 Item validity: ~~1 week, 3 days~~ *Overridden with 11 days*

Importance level (higher is better)

Period of time during which feed items are considered to be up-to-date. E.g. 10 days

□ Full content □ Featured □ Is regional ☑ Allow summary ☑ Enabled

If true, the feed supports rich summary generation.

If disabled, no crawling of the source.

□ Expire content on disable

If checked, expires all content upon disabling.

Used by:

Last modified: (None)

Created: (None)

Last checked: Today |

Crawler status

Status retrieving

Last crawled retrieving

Updated retrieving

Image density

No Data

IMAGE 6. New source addition screen

With the help of CMS UI, new content can be easily added to Discover. Content manager has to fill in the required fields (meaning classify the content according to the topics) and then add a source membership to the country, where the articles from the source should be shown. With all these steps done, new content, already filtered, appears in Discover.

4.5.3 User-based collaborative filtering in Discover

Another filtering technique used in Discover is user-based collaborative filtering. CF is performed for the Top stories category in order to suggest a better article selection for the active user. In the past, Top stories screen was only showing a mix of recent stories from the most popular categories. With recent changes, the most clicked stories are pushed higher to be of a better visibility to the user. It is assumed that the articles that were clicked (and, most likely, read) by other Discover users might also be interesting for the active user. Because of lack of user preferences information, the algorithm uses country and language preferences as a measure of similarity.

The algorithm for the most clicked articles analyses so called feedback events that are gathered when user uses Discover. There are two types of events that are taken into consideration by the algorithm: article preview views and full article views. Those events have different interest weight, which is used to determine article popularity, with full article views getting more weight than just article previews.

Data on these events is gathered every hour for each of the articles that are not older than 2 days. Popularity score is computed for two days, one day, 12, 8 and 4 hours. The fresher an article is, the more weight it gets. Scores over time are combined in a linear fashion with some additional down-weighting applied, if required. Down-weighting is used for the articles that are not that fresh to make sure that fresh content is always more visible to the user. This way the algorithm assures that popular stories always stay fresh. (Streppone, 1 June 2015)

4.5.4 Hybrid filtering

In the previous chapters, filtering techniques used in Discover were described. As already stated in the paragraph 3.4, if a recommender system uses more than one filtering technique, it is considered to be using hybrid filtering approach.

To summarise filtering methods used in Discover, image 7 was created (below):

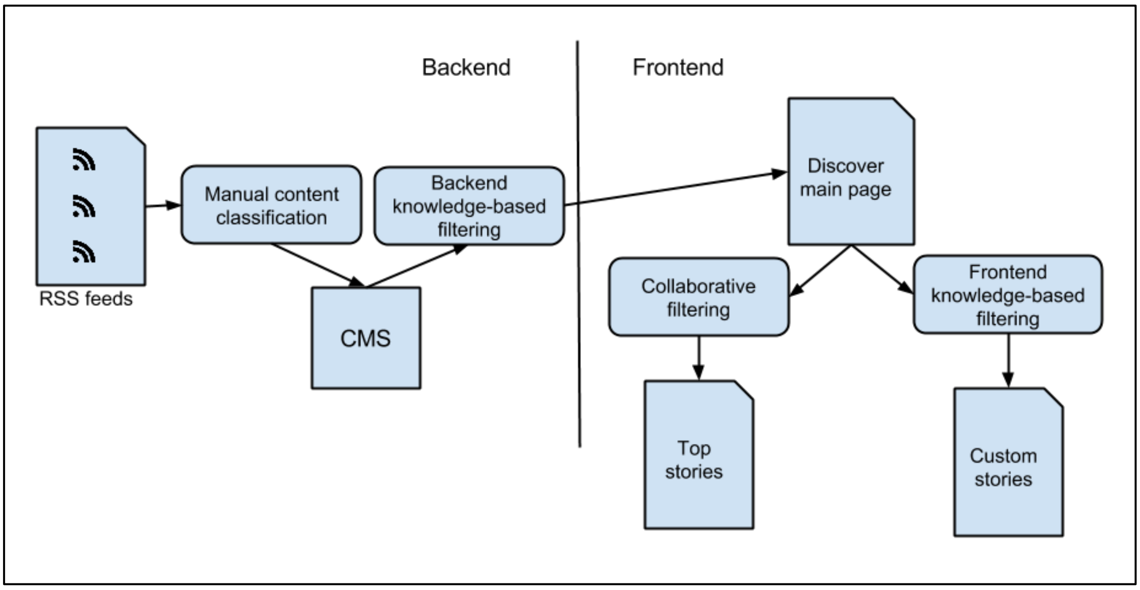

Backend
Frontend
RSS feeds
Manual content classification
Backend knowledge-based filtering
CMS
Discover main page
Collaborative filtering
Frontend knowledge-based filtering
Top stories
Custom stories

IMAGE 7. Filtering techniques used in Discover

As can be seen, each of the filtering techniques is implemented separately, but their results are combined in order to produce better content recommendations.

5 CONTENT RELEVANCY AND FRESHNESS IN NEWS RECOMMENDATIONS

The popularity of digital media has been constantly increasing over the past few years. More and more consumers prefer digital media to TV, radio and printed newspapers. This happens because of some characteristics online media has, like ease of access to the desired content, a huge variety of content, portability (news can be always found on consumers laptop/mobile device), etc. The annual research conducted by The State of the News Media (see image 8) has shown a constant increase of interest in online media and a decrease of interest in any other news sources in the US. At the end of 2010 online media has left behind printed newspapers and radio and became second most popular source of information (TV still having a lead).

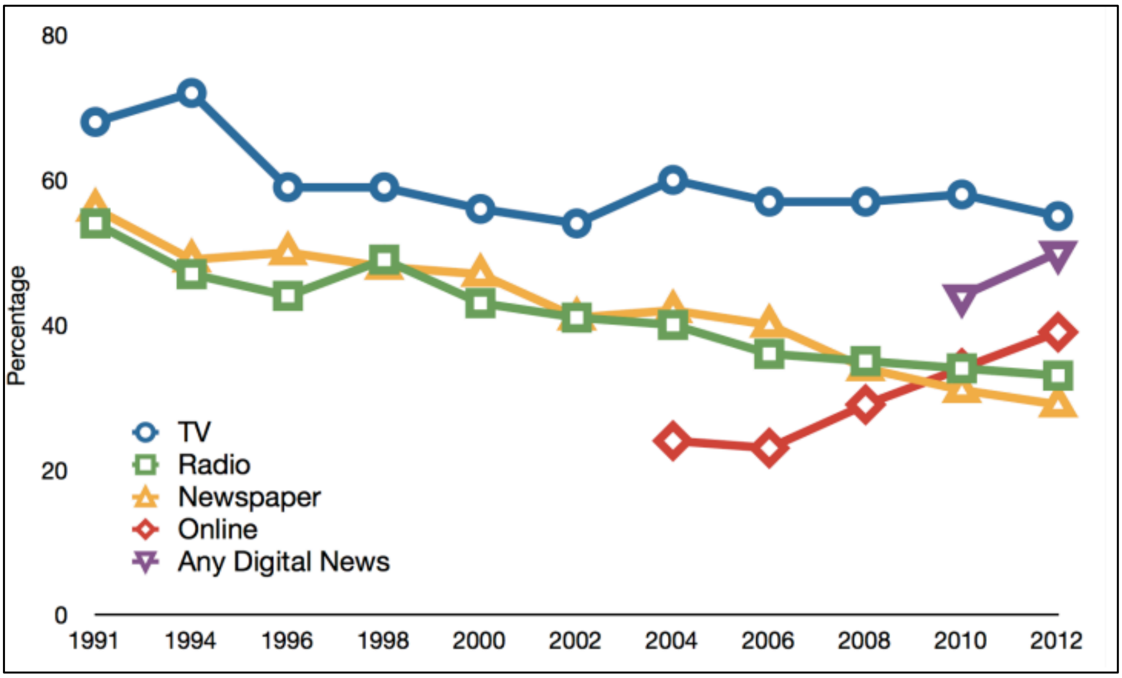


IMAGE 8. Where do people get news from (The State of the News Media, 2013)

According to Outsell report (2009), 57 per cent of news media users now go to digital sources, and they are also more likely to turn to an aggregator (31 per cent) than to a newspaper site (8 per cent) or other news site (18 per cent).

News recommendations systems are created in order to provide users with fresh and relevant news stories. At the moment the significance of content relevancy for recommendation services is very high: news aggregation is a competitive market, and users have a wide variety of choices when it comes to selecting a service for news recommendations. On the other hand, services overload is a problem that the user faces when selecting the news channels to read. It is often difficult for the users to determine which of the various channels of information can suit their needs in a most useful and efficient way.

Generally speaking, there are two kinds of news aggregators (Chowdhury, Landoni, 2006):

- Aggregators that simply present material from various sources in their UI
- Aggregators that gather and distribute content to suit their customers' needs by completing the appropriate organizing and processing

News has some characteristics that make it difficult for the users to keep track of the latest news items on a chosen topic. The reason behind that is that news is produced and distributed by a number of news publishers in a different format, language and form. Therefore news aggregators are very useful, as they can save time user would spend on searching and retrieving news information from a variety of channels and agencies.

Opera tried to face the problem of sources diversity by integrating news aggregator – Discover – into their products. This allows users to get news recommendations right in the browser without a need of searching for any other recommender. Despite this convenience, various tests performed by the Discover team have showed that content recommended by Discover is not always relevant and fresh. It leads to the problem that the recommendations provided by the service might not be interesting for Opera users, meaning they have to find some other news aggregator that suits their needs more. There is no reason why user would use a service where he/she has to search for something that is relevant over a service that provides this relevant content right away.

5.1 What news is considered to be relevant?

Generally speaking, something is considered to be relevant to a task, if it increases the likelihood of accomplishing the goal, which is implied by that task (Hjørland, Christensen, 2002). In terms of news, relevant content is the content that satisfies users needs. Those needs can vary: some users want to read the latest news or catch up with latest technology trends, others want to be entertained, etc. For a news recommendation service it is vital to identify the most important needs of a majority of users and decide, which of those needs are to be satisfied. It is hard to provide relevant content for every user, and because of that the needs of a majority are to be considered of a higher priority.

There are several factors that can help to identify if a story is relevant to the general type of reader. These are to be discussed in the following chapters.

5.1.1 Relevancy criteria: freshness

The etymology provided by the Online Etymology Dictionary suggests that the word “news” can be literally interpreted as “new things”. Topics that are currently happening/developing are always considered to be newsworthy. Consumers of online news are used to receive the latest updates on the topics of their interest, and that is why old news is to be quickly disregarded.

The importance of freshness in relevancy, however, depends on topic and type of news story. For example, a breaking news article becomes outdated faster than an in-depth feature article on the same topic.

There are several factors that determine why freshness is important for a news story:

- For the developing news, new articles are more up-to-date and, therefore, more relevant than older ones. As stories tend to develop over time, the facts can change, leading to older articles becoming inaccurate. As an example, an article about some hockey match published after the first period will have very low relevancy after the match has ended.
- Current news and events, as well as breaking stories, are more relevant for the users than old ones. Those stories affect readers right at the moment of reading and their outcome might be uncertain. These kinds of stories also make users return to the news source, because they would want to receive updates.
- Life span of an online article is short, especially considering news articles. Non-fresh content will not be relevant to the users, as it will pass its lifespan.
- There are a variety of sources users can consume news from. If the content is not updated frequently, there is a high chance that the user has already read/heard about it somewhere else.
- Even when the topic of a news story is not of a general interest for the user, it can be considered important and relevant if the content is fresh and the story is breaking. Usually users want to be up-to-date on the important news that is relevant for a large amount of people.

5.1.2 Relevancy criteria: importance

An important story will always be more interesting for the users, as it impacts a large amount of people and, therefore, is more relevant than the story having a low importance factor. For news recommendation systems, it is even more vital to detect important topics than stories, because even when an article ceases to be important, the topic might still be of a high interest to the users. Among the articles covering the same important topic, the freshest and high quality articles should be presented to the readers first in order to catch their attention. Newsworthiness factors can usually be considered as importance factors. Nevertheless, news RSs should not only focus on news content, as users should have a variety of topics to choose from. This will keep users that have a different aim in mind when using news RS interested in the provided content as well. With all that said, important topics should be identified not only for news stories, but for the other categories of news as well.

The most common factors to consider when choosing important stories are presented below:

- Impact or consequences – the greater the impact for the readers is, the more relevant the story will be. Events that impact users can have consequences on their lives, and thus will be relevant for them.
- Timeliness – events that are happening at the moment user is reading the story are more relevant than stories covering the events that have happened in the past.
- Currency – some stories may have an on-going interest for the users, even though they are not happening at the moment of reading. For example, the changes in exchange ratio of Euro to Rouble have been happening for several years, but are still important for Russian readers. This factor is related to timeliness.
- Proximity – the closer the event the story is about is to the reader, the more relevant it will be. Local events will be interesting for the local users, but most likely will be of no interest to the users located somewhere else.
- Prominence – someone/something prominent or famous attracts readers' attention, as they recognize it easily. Prominence can be applied to politicians, actors, athletes, companies, brands, countries, etc.
- Usefulness – if the article has some practical advice or helps resolve some problem, it becomes relevant for the users. For example, articles on how to protect from ticks or how to lower the amount of taxes to pay will most likely be useful for a large group of readers.

5.1.3 Relevancy criteria: quality

As already mentioned before, well-written articles tend to attract more attention from the users. This is why high quality articles are more relevant for the readers, than the low quality ones.

Some sources may have greater authority on specific topics and some authors are considered to be better experts in certain fields. This is also true about the publishers:

some might cover the topic only superficially, while others may decide to write a feature article on that topic.

Criteria for determining the quality are the following:

- Source authority – if the source is known as an expert on the topic of the article, it will be more trusted by the users. For example, BBC news may be an expert on news, while Wired has much better articles on technology topics. In general, there are sources that are known for their quality, while there also are sources that publish more tabloid and gossip-oriented content.
- Author authority – same as source authority, but in regards to the authors. Some journalists are considered to be experts in specific fields, which gives them a higher trust factor from the readers.
- Article length – generally, the longer the article is, the better it will cover the topic. In-depth articles are considered to be of a better quality, as the writer has spent a lot of time writing it.
- Originality – an original article has better quality than the article that was written by copying the content. Re-writing is very popular nowadays, so it is important to find the original source of information. Originality is also applied to the media used in the article: images, videos, graphs, etc.
- First to publish – the source that has published the article on some topic first should get an advantage over other sources. Nevertheless, it is important not to forget about an overall quality of the article. If the article has a low quality, but was published first, it is better to disregard the criterion of “first to publish”.
- Quality of writing – an appropriate language style should be used in the stories to maintain the quality.
- Media enrichment – media helps to make an article more appealing to the user. The presence of media pieces also indicates that the article is of high importance to the publisher.
- Complete articles – articles that are hidden behind paywalls should be omitted, as most likely the user will not have a subscripting to the source the article was published in. Articles that are referring to a different source have a low quality as well.

5.1.4 Relevancy criteria: popularity

If an article or topic is popular, it is natural to assume that it will be relevant to most users. Popularity is one of the easiest factors to measure, and that is why it is usually used for ranking the content. However, there are some risks associated with using popularity factor only in an attempt to rank the content. If used, it can happen that articles containing inappropriate context, lowbrow humour, and vulgarity will receive a very high rank. Therefore, popularity should be considered as a lowest ranking factor, as opposed to importance, quality, variation and freshness.

Popularity factor can be applied well on static content (i.e. for topics). Topics have a higher lifespan (currency factor) than articles and can be updated with the latest content. Giving users the freshest content of high quality on a popular subject will ensure the application of several relevance criteria at once.

On the other hand, using popularity factors for articles can affect freshness, as an article builds popularity over some period of time. Therefore, popularity factors can have a higher importance for feature articles and entertaining content, while it should be of a low importance for current news and developing stories. It is also important to avoid a popularity loop, because boosted popular articles that appear on the main page will get even more interest and thus become even more popular.

Numerous factors can be used to determine popularity. Below are some of the suggestions:

- Popularity of articles in a news aggregator – popularity of single articles can be measured in a news RS. This factor should be used with caution because of the consequences described above in this chapter.
- Popularity of articles in other sources – other sources of information can be used to determine which articles are popular at the moment. For example, in case of Discover, browser history could be analyzed to find what articles users have recently read. This gives a larger database of users and a quicker popularity calculation.
- Popularity of news source in a news aggregator – popular sources are likely to be more important for the users. Knowing the most popular sources can help to determine which articles should be the first ones to be showed.
- Popularity of news source in other sources – other sources can provide more diverse information on popular sources within each category, country, language, etc.
- Popularity of topics in a news aggregator – this factor is of better use than article popularity. By focusing on topics rather than articles, new RS can always have new content that will still be interesting for the user.
- Popularity of topics in other sources – knowing trending topics outside the news recommender could be beneficial, as more data on what is popular for the users will be available.
- Article engagement – it is important to fully understand if an article was of an interest to the user. Article clicks cannot be used as a reliable measure of popularity. Instead, there are other ways to measure engagement, i.e. check-

ing how much time did the user spend reading the article, if he/she shared it with some friends, commented on it, etc.

- Social media popularity – different social media channels can say a lot about what is interesting and important for the users. Article shares, comments and likes on various social platforms (e.g. Facebook, Twitter) can be used to determine popularity. Nevertheless, it is important not to forget about freshness, as popular social media items are likely to have been seen by a large amount of users already.

5.1.5 Relevancy criteria: variation

Variation of topics and articles helps to achieve better user engagement. By providing users with a variety of sources, news aggregator will give an element of discovery and ensure a better relevancy of content. Very similar articles should be considered the least relevant, as user will not be interested in reading similar articles on the same topic several times. It is important to keep a balance between the most popular topics and sources, and the topics and sources that are more specific. This will help news aggregators reach a wider audience and keep more users interested in the provided content. Most of the people share at least some common interests, and articles on those topics are likely to become most popular. At the same time interests of users can vary, and if news aggregators can provide articles on those topics as well, readers will find it more relevant.

To summarize it all, there are several factors that make variation important:

- If news aggregators provide a wide choice of sources and topics instead of focusing on the most popular ones, more users can be reached and engaged.
- Most popular topics are likely to be interesting to a large group of users, but popularity also often means that the user is already updated on the topic. There is a big chance that the user has already gained information on the most popular matters and will not be interested in reading the stories on that matter again.
- By providing the variation in content, news aggregators can also ensure the more democratic information delivery. This will also save news aggregators from giving a story from one perspective only and reduce the chance of misinformation.
- Variation leads to more discovery from the users' side and makes the service look richer and deeper. Users will not get a good impression, if a news RS only provides stories from a handful of sources. Discovery factor will also help users to find out something new and develop their interests.

5.2 Relevancy of content in Discover

In the previous chapters, some of the most important factors for news stories relevancy were defined. As can be seen, relevancy of content is very important for news RSs. The more factors of relevancy are supported by the RS, the more regular users that news recommender will have.

In order to check if the Discover feature corresponds to those factors, various tests can be performed. As Discover is not only a news aggregator, but more of a news recommender, relevancy should be maintained on a high level. This will help to achieve higher user engagement and will improve the general appearance for new users. The longer a user has to look for relevant content, the bigger the risk of loosing this user.

When speaking about **freshness**, Discover is already behind news websites, as it takes time to receive the update, crawl it and display to the user.

Below is a simplified representation of a news lifecycle – starting from an article being published by the content provider and finishing by the news piece being displayed in Discover.

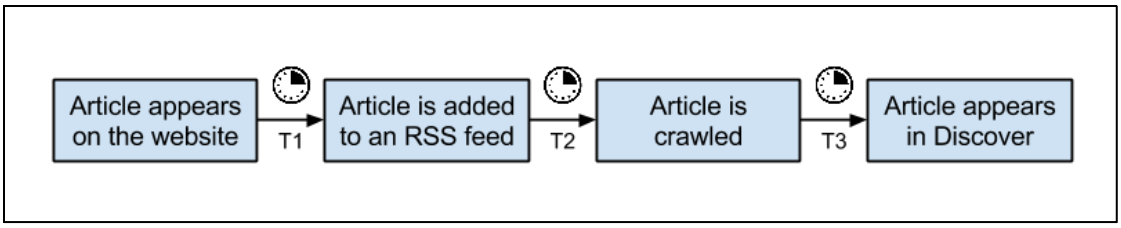

Article appears on the website
T1
Article is added to an RSS feed
T2
Article is crawled
T3
Article appears in Discover

IMAGE 9. Timespan for an article to appear in Discover

As can be seen, there are at least three delays (T1, T2, T3) that take place after the article is published and before it is shown in Discover. According to the test performed by the Discover team, timeframe T1 can vary from 5 minutes to indefinite time for some of the news websites:

TABLE 2. Average & longest update times in RSS feeds (Mitovska 2014)

	Source Name	Average update time	Longest update time
1.	BBC	~ 10 min	18 min
2.	USA Today	No defined update time noticed	3 h 56 min
3.	Reuters	~ 10 min	12 min
4.	Guardian	~ 15 min	14 min
5.	NYT	~ 30 min	28 min
6.	NBC news	~ 30 min	25 min

7.	Huffington post	~ 10 min	11 min
8.	Aljazeera	~ 10 min	10 min
9.	Telegraph	No defined update time noticed	70 min
10.	The Hindu	~ 20 min	21 min

Timeframes T2 and T3 also vary from 2-3 minutes up to about 30 minutes. These time periods depend a lot on the time crawling was started: it can happen that crawling starts right after the RSS feed is updated, and up to 30 minutes after the RSS feed is updated. The regularity of crawling procedure set in Discover is 30 minutes, meaning that all the sources get crawled once every 30 minutes.

To sum up, if ~17 minutes is taken as an average for the RSS feed to be updated, it can take up to ~47 minutes for a new article to show up in Discover.

Importance and **popularity** are two other major factors to consider when publishing content in Discover, especially in the Top Stories screen. As the name suggests, Top Stories should provide the most important/popular articles from the variety of categories present in the service. By accomplishing that, user will stay up-to-date with the current situation in the world and will become interested in checking Discover more frequently.

In the test, done within the Discover team, an attempt to identify the importance of stories in the Top Stories screen was made. The test consisted of the following steps:

1. 20 top articles in the Top Stories screen were checked.
2. Importance was reviewed by analyzing the source's main page.
3. Articles in Top Stories were scored on a three-grade scale: (1) if article was one of the top stories on the source's main page; (2) if article was on the main page, but was not a top story/ top story in some other section of a website; (3) if article was not on the main page.

The results of this experiment for content for USA are shown below.

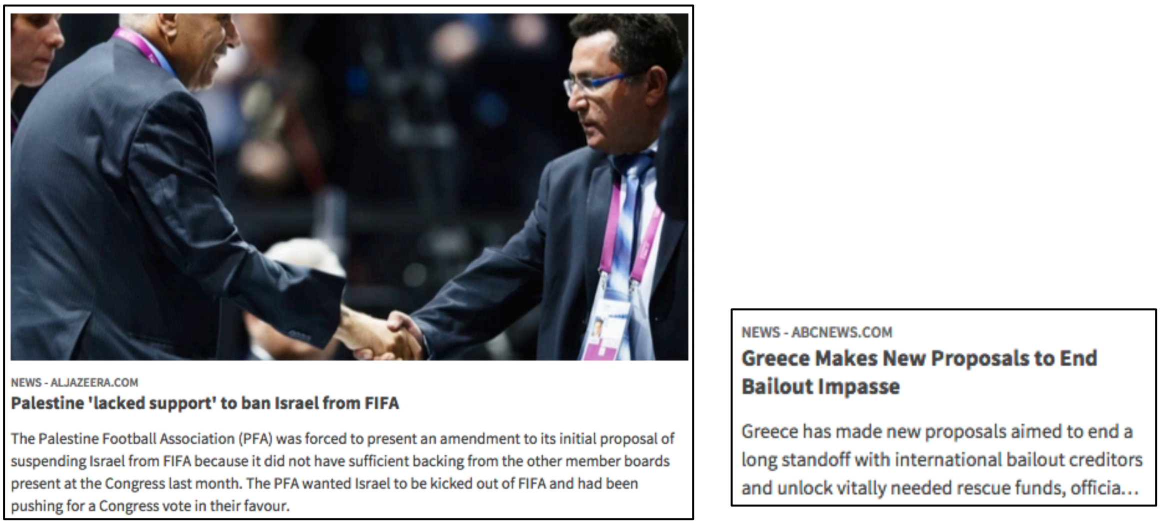
TABLE 3. Importance testing for the Top Stories screen (Kjellin 2014)

Number of sources	Average score	Category distribution
13	2.15	News – 5, Sports – 7, Entertainment – 8
16	2.25	News – 6, Sports – 7, Entertainment – 6, Arts – 1
12	2.20	News – 5, Sports – 8, Entertainment - 7
15	2.10	News – 5, Sports – 8, Entertainment – 6, Technology – 1

It can be concluded that Discover's Top Stories screen is not good enough in providing the most important and popular stories. In over 50 per cent of cases, the story in Top Stories was not present on the main page of the article's source. This experiment revealed a problem of important/popular stories detection in Discover. Moreover, by looking at category distribution and the number of sources articles were taken from, it is clear that **variation** of content needs improvement as well: only some categories appear in Top Stories (News, Sports and Entertainment taking the lead), and out of 20 articles only about 14 were coming from different sources.

Quality of the content in Discover depends a lot on the original source. Discover uses a base of more than 9000 sources that vary in quality of the content they provide. Even though sources are selected manually, and content managers evaluate source's overall look, authority of the publisher, quality of writing and content, it is still possible that Discover gets articles of poor quality or even stories containing inappropriate content.

One of the simplest but yet important characteristics in determining quality is the use of media content. Article previews containing images are always better looking than just plain text stories.



NEWS - ALJAZEERA.COM

Palestine 'lacked support' to ban Israel from FIFA

The Palestine Football Association (PFA) was forced to present an amendment to its initial proposal of suspending Israel from FIFA because it did not have sufficient backing from the other member boards present at the Congress last month. The PFA wanted Israel to be kicked out of FIFA and had been pushing for a Congress vote in their favour.

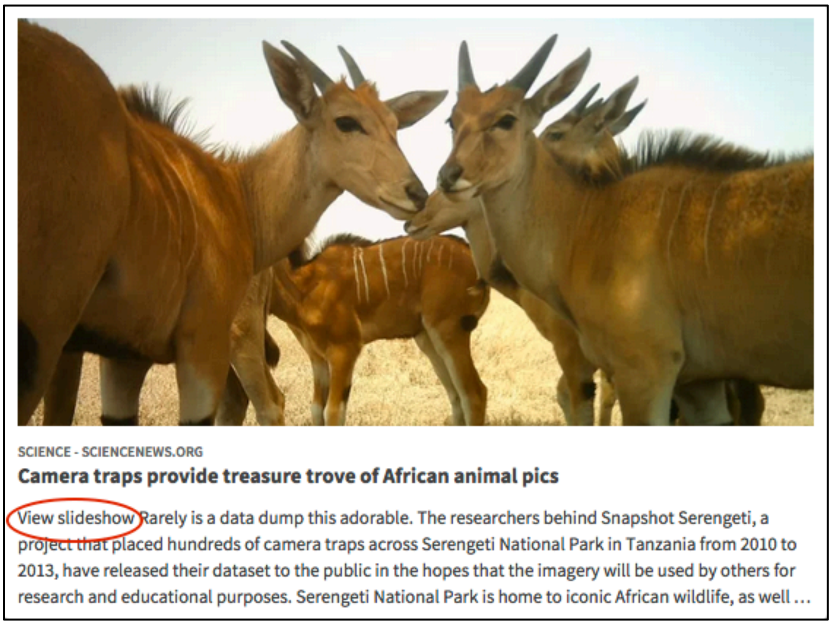
NEWS - ABCNEWS.COM

Greece Makes New Proposals to End Bailout Impasse

Greece has made new proposals aimed to end a long standoff with international bailout creditors and unlock vitally needed rescue funds, officia...

IMAGE 10. Discover article cards with and without an image

Another important factor is the quality of description text for an article. Discover gets a description text by obtaining it from the source webpage. With the use of hints provided to the crawler by content managers, the position of the article text is determined. Sometimes it can happen that the wrong portion of the text is captured, making the summary look improper.



SCIENCE - SCIENCENEWS.ORG

Camera traps provide treasure trove of African animal pics

View slideshow Rarely is a data dump this adorable. The researchers behind Snapshot Serengeti, a project that placed hundreds of camera traps across Serengeti National Park in Tanzania from 2010 to 2013, have released their dataset to the public in the hopes that the imagery will be used by others for research and educational purposes. Serengeti National Park is home to iconic African wildlife, as well …

IMAGE 11. Improper text portion displayed in article's summary text

6 WAYS TO IMPROVE RELEVANCY & FRESHNESS OF CONTENT IN DISCOVER

In the previous chapter some of the drawbacks of the Discover feature were identified. Those problems affect relevancy and freshness of the suggested content and can result in a situation, where user will decide to use another news aggregator, because the suggested content will not reflect his/her needs and interests.

Generally speaking, there are two main directions for improving recommendations within the service:

- General recommendations improvements – changes in Discover that will affect all users simultaneously, or large groups of users.
- Personalized content improvements – improvements that will lead to suggesting different content for each specific user.

Relevancy of content is a major factor for both of these directions. It is important that the user receives relevant and fresh content no matter which way of improvement is chosen.

Discover is a constantly improving service. Since 2013, when the Discover feature was first released, a lot of changes have been made to provide better recommendations for users. At the moment the team is actively working on introducing a personalized content approach. With personalization, each user will get news he/she is most interested in, meaning a great betterment in content relevancy. Nevertheless, personalization will work well only for recurring users. Occasional visitors of Discover should be considered as well, meaning that general relevancy is also an important subject for improvements.

In this chapter a number of ways for improving the relevancy of content in Discover will be described. By integrating the suggested improvements it would be possible to achieve better relevancy of the suggested news, which, in turn, will lead to better user engagement and retention.

6.1 General relevancy improvements

General relevancy of content is an important aspect in engaging the audience to use Discover. Content relevancy is one of the main things users notice when opening a news RS. If the content is not relevant and fresh, there is a high chance that the reader will prefer to use another news recommender. At the moment the number of recurring users in Discover is lower than the amount of occasional visitors, so general relevancy should be considered as one of the main things to be improved.

6.1.1 Catching trending topics

A topic can be considered trending, if it is mentioned and discussed more often than any other topic. Several resources can be used to monitor trending topics:

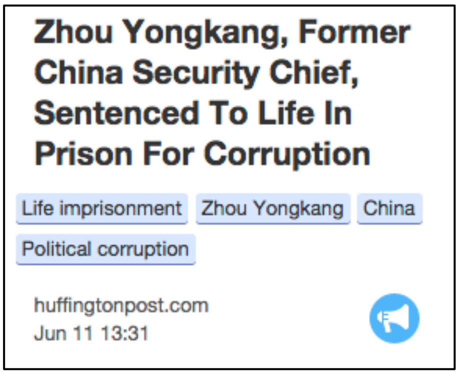
- Social networks (e.g. Facebook, VK, Google+, Twitter)
- Special trending sections on the biggest news websites (e.g. Buzzfeed, CNN, ABCNews, etc.)
- Search engines trends (e.g. Google trends, Yandex trends, etc.)

By combining the data from those sources, it is possible to receive a list of the most trending topics, which can be used in order to determine which news are to be shown first to the users.

However, knowing trending topics is not enough to provide better recommendations. In order for this to work properly, news recommendation systems should understand which news topics it has at its disposal. Therefore, news stories should be annotated. Annotation is about attaching names, attributes, comments, descriptions, etc. to a given text (Ontotext, 2015). The semantic annotation technology enables many new applications, such as highlighting, indexing and retrieval, categorization, structuring free texts and metadata generation. It is applicable for any kinds of texts – web pages, office documents, descriptions of a data table field, meta-information of a structured resource, additional comments for a document and so forth (Tang, 2011).

In case of Discover, an on-going experiment with using a special algorithm in order to extract annotations for news title texts – TAGME – is performed. TAGME is a powerful tool that is able to identify on-the-fly meaningful substrings (called "spots") in an unstructured text and link them to a pertinent Wikipedia page in a fast and effective way (Ferragina, Scaiella, 2012).

As an example, consider the image of an annotated article preview card below. There, the title of an article “Zhou Yongkang, Former China Security Chief, Sentenced To Life In Prison For Corruption” was annotated with tags “Life imprisonment”, “Zhou Yongkang”, “China” and “Political corruption”.



Zhou Yongkang, Former China Security Chief, Sentenced To Life In Prison For Corruption

Life imprisonment Zhou Yongkang China Political corruption

huffingtonpost.com
Jun 11 13:31

IMAGE 12. Article preview card with a title being annotated

Another option that can be used for retrieving the annotations from the article is to use `news_keywords` metatag. Introduced by Google in 2012, this metatag helps news crawlers to extract the topics the news article is about. Publishers can use it to specify a collection of terms that apply to a news article. The words used in a metatag don’t need to appear anywhere within the headline or body text, which allows writers to express their stories freely while helping crawl robots to properly understand and classify the content of the story. (Google News Blog, 2012)

For example, for an article “Prolific British Actor Christopher Lee Dies at Age 93” published on ABCnews, the website specifies the following tags:

```
<meta name="news_keywords" content="Arts and entertainment,
General news, Celebrity, Entertainment, Obituaries, Movies,
Christopher Lee, George Lucas, United Kingdom, Western Europe,
Europe" />
```

The metatag is also a good instrument to be used in order to disambiguate between related terms. For example, in an article on The Guardian website with the name “Armstrong admits fears over trial – and compares himself to Voldemort” tags are used as follows:

```
<meta name="news_keywords" content="Lance Armstrong, Cycling,
Drugs in sport, US sports, Sport"/>
```

The keywords help to understand what the story is actually about. To prove the point, below is a list of topics generated by TAGME algorithm for the same title:

- Neil Armstrong
- Fear
- Trial
- Lord Voldemort

As can be seen, automatic topic detection gave false results in this case, which can result is a story being irrelevant to current trending topics.

To sum up, if at least one of the annotated tags is considered as a trending topic, the story should get a higher rank and, therefore, be placed more to the top in the Discover interface. This will make the story more noticeable and, in turn, will increase the relevancy of the content presented in the service. However, in order to avoid the problem of disambiguation, metatag `news_keywords` is always of better use, if present on the source's website.

By detecting trending topics, several factors of relevancy, such as freshness, importance and popularity, can be achieved.

6.1.2 Gathering user feedback and providing tools for news publishers

Even though there is an on-going debate on whether news aggregators actually decrease or increase the revenue of news publishers, providing news stories to news RSs can actually be very beneficial. Most news aggregators are based on the concept of fair use, meaning that they are required to:

1. Show the original source/publisher of the article
2. Limit the amount of content displayed to a short ingress
3. Drive traffic back to publishers via links to the original article

By maintaining these rules, news aggregators ensure that news publishers' monetization methods are not damaged. Discover service obeys the rules of fair use, which means that news websites can benefit from their content being displayed. This gives Discover a lot of potential partners.

News aggregators can benefit a lot from the cooperation with news publishers as well. Having a large database of publishers can ensure a better variation in content, which, in turn, will lead to better relevancy. At the moment the list of publishers for Discover is determined by content managers only, who search for best news websites for each of the countries. It is clear that by using this approach some of the good news providers are left out, as it is not possible to analyse millions of news websites available on the Internet manually. With this said, by providing publisher tools within the Discover user interface, it would be possible to increase the database of news sources available in Discover at the moment.

"Publisher tools" is used as a general term for a number of web tools that will allow news publishers to communicate directly with the content management team in Discover. By using these tools news providers can request:

- To include their website into Discover service

- To update the information about their website (i.e. add new categories, change displayed URL, etc.)
- To fix some display problems present in Discover interface for their website (i.e. include pictures, correct the summary text, fix the category news is displayed in, etc.)
- To remove their website from Discover

By introducing publisher tools, Discover will be able to improve relevancy a lot, because better quality and variation of content will be achieved by getting direct help from news providers.

It is important not to forget about the end users of Discover as well. At the moment Discover does not have any feedback tool, and because of that the only way users can express their opinion on the content is by commenting on the Opera news blogs. Of course, Opera also has a public bug report system, but it is mostly used by profound users only. By introducing the feedback tool in Discover, content managers will be able to receive instant response from the end users, which will facilitate to improve quality of the content, increasing the relevancy of the service in general. Users can be given an option to report inappropriate articles, inform about some visual and contextual bugs for article previews, and even an option to suggest new sources for Discover.

6.1.3 Prioritizing top news websites

As already mentioned in chapter 4.2, freshness of content in Discover is suffering a lot because of the procedures that have to be executed before an article can be shown to the reader. This is a common problem for most of the news aggregators and not much can be done to improve it. The delay in news presentation is not that vital for feature articles, but affects breaking news a lot. If the user does not get a breaking story right after it is published, there is a high chance that he/she will read it somewhere else before it becomes available in Discover.

However, some adjustments to the crawling algorithm can be made in order to improve the time required for a published article to appear in Discover. There are a number of news sources that are very good at publishing breaking news. Most of the time, these are huge news corporations, like BBC, CNN, Aljazeera, Reuters, etc. Usually, they also have a “breaking news” section, where most important and fresh news stories are displayed. By prioritizing those sections to be crawled more often, the freshness of content in Discover could be increased. Breaking stories, in turn, can be either gathered in a separate special category, or displayed on the first page of Top Stories or categories screens, with a special label (e.g. “breaking story”, “important”, “currently happening”, etc.). This way the visibility of breaking stories will be increased, which will ensure that the reader notices that news prior to reading some other articles.

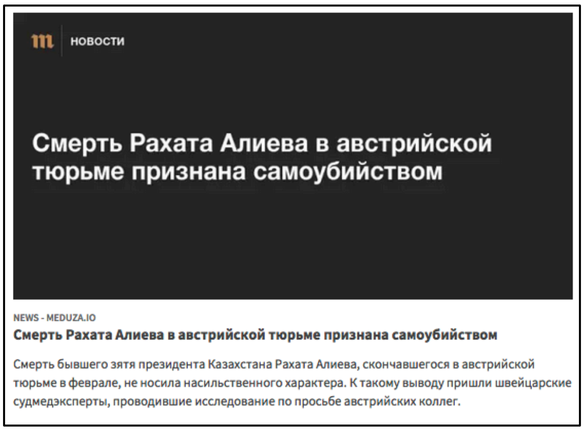
6.1.4 Generating media content for articles without images

Quality of news articles can be affected a lot by media enrichment. Articles with pictures always look more attractive than articles containing just text (see image 10). That is why Discover should try to prioritize media enriched news pieces.

Nonetheless, articles without pictures or with pictures that are not good enough (i.e. small resolution) can contain more important information and be more relevant to the user, than news stories with pictures. Taking that into account, an important question arises: should the news recommender sacrifice the look of its interface in order to not miss important stories that have no media content provided, or not. In order to eliminate both of these problems, news RS can add a simple image for the articles that do not have it. There are several ways that can be used in order to generate a picture, and the most popular ones are presented below:

- Title text can be used to generate a simple image containing this text
- Images database can be used to provide an image corresponding with the content of an article
- If an article has an image that is not good enough to be included because of low resolution, blur effect can be applied to increase image resolution
- Image containing website’s logotype can be used

Image below (13) contains an example of a picture with article text being used. This picture comes from the website medusa.io, from where the article was taken, but the similar approach can be applied by a news RS to generate images.



m НОВОСТИ

Смерть Рахата Алиева в австрийской тюрьме признана самоубийством

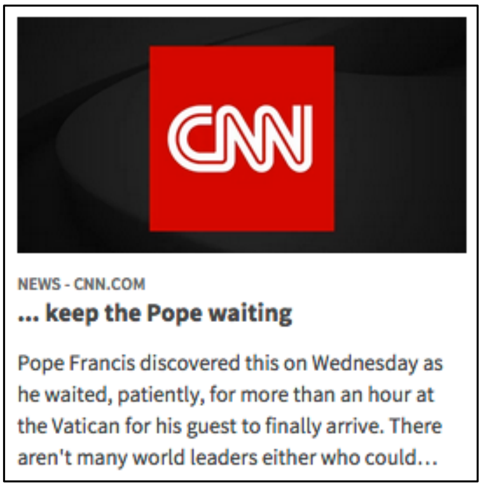
NEWS - MEDUZA.IO

Смерть Рахата Алиева в австрийской тюрьме признана самоубийством

Смерть бывшего зятя президента Казахстана Рахата Алиева, скончавшегося в австрийской тюрьме в феврале, не носила насильственного характера. К такому выводу пришли швейцарские судмедэксперты, проводившие исследование по просьбе австрийских коллег.

IMAGE 13. Article title used as an accompanying image to an article

Another example (image 14) contains the logotype of a news website (CNN in this case) that is used as a picture for an article.



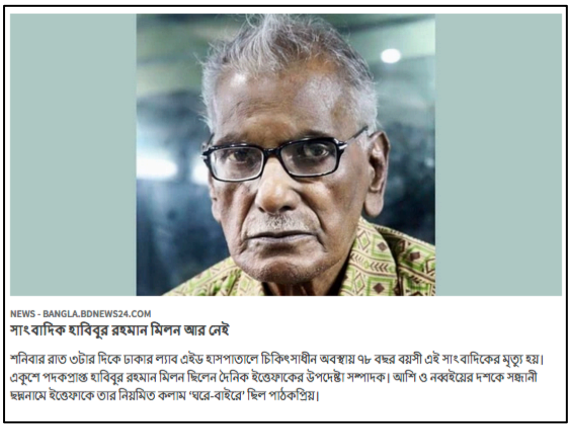
NEWS - CNN.COM

... keep the Pope waiting

Pope Francis discovered this on Wednesday as he waited, patiently, for more than an hour at the Vatican for his guest to finally arrive. There aren't many world leaders either who could...

IMAGE 14. Website's logo as an article picture

In the next example (image 15), resolution of the image was not good enough to be included in Discover. However, by applying blur effect to the corners of an image, the resolution was artificially increased.



NEWS - BANGLA.BDNEWS24.COM

সাংবাদিক হাবিবুর রহমান মিলন আর নেই

শনিবার রাত ৩টার দিকে ঢাকার ল্যাব এইড হাসপাতালে চিকিৎসাধীন অবস্থায় ৭৮ বছর বয়সী এই সাংবাদিকের মৃত্যু হয়। একুশে পদকপ্রাপ্ত হাবিবুর রহমান মিলন ছিলেন দৈনিক ইত্তেফাকের উপদেষ্টা সম্পাদক। আশি ও নব্বইয়ের দশকে সন্ধানী ছদ্মনামে ইত্তেফাকে তার নিয়মিত কলাম 'ঘরে-বাইরে' ছিল পাঠকপ্রিয়।

IMAGE 15. Blur effect applied on the sides of an image with a low resolution

By using the described ways of picture generation, a better look and quality of articles in Discover can be achieved.

6.1.5 Duplicate detection and removal/clustering of similar articles

Another important aspect of quality that will help to keep relevancy high is duplicate detection. News recommenders containing several articles on the same topic with similar content will not be good enough, as the same topic will be repeated over and over again. Even though articles will be coming from various sources, the user might find the described situation irritating. To eliminate this problem, news aggregators can apply algorithms for duplicate detection. Duplicated articles can either be removed (especially if articles are identical), or clustered (combined) by topic (if articles are on the same topic but provide different information).

There are a wide variety of methods that can be used in order to detect similar or duplicate articles. News aggregators can decide, which method is the most relevant to be used depending on the situation with duplicated articles and different technical aspects.

By using duplicate detection algorithms, Discover will be able to increase articles' relevancy, as better quality of content will be achieved.

6.2 Personalized suggestions

As already stated, general relevancy of content will work well for occasional users of Discover. However, it might be insufficient for regular users, as they would want the news recommender to adapt to their needs and interests based on their reading habits. In order to keep regular users of Discover engaged by the service, personal recommendations can be used to build a different set of news stories corresponding to each of the readers' personal interests.

Filtering methods for recommendation systems were discussed in chapter 3. Some of the filtering techniques are already applied by Discover (see chapter 3.5), but there is still a lot of potential for recommendations betterment. As Discover mostly uses filtering techniques for general recommendations, a large field of application for those algorithms in personalized recommendations remains open for improvements.

A key for providing personal recommendations is to monitor and log user activity. By knowing a user's habits and behaviour patterns, it is possible to create a personalized stream of news that will correspond to those patterns. Some of the easiest factors to monitor are to be discussed further.

6.2.1 Analysing user behaviour

Chapter 3.5.3 describes an algorithm used to improve Top Stories relevancy. There, click events are monitored to push the most clicked articles in each of the countries further to the top of the screen. A similar approach with additional metrics can be used to create a personalized stream of news for individual users.

At the moment the Discover team actively works on the development of user profiles, where activity of each user is saved for later use. Example below shows a data structure for a test user (some parts of the code are omitted for better readability):

```
{
  "a":    {
            "e12a22e3265dae8bd38a0a95fa4ce3d015dff2d2":
            "20150531T165900Z",
            "779036b04d2a613d6dc329c5b9de2743be022ce1":
            "20150521T155000Z",
            "a89aef0690b71fe285a07224355e88ce60569d37":
            "20150522T180600Z"
          },
  "c":    {
            "de_de": 9, "no_no": 5, "za_en": 1, "it_it": 9,
            "zz_en": 121
          },

  "g":    {
            "ga": 9,"mo": 22, "sp": 1, "tr": 21, "ne": 33,
            "fo": 10, "sc": 21, "te": 27, "he":1
          },

  "ann":  {
            "Automobile Racing Club of America": 1,
            "Lewis Hamilton": 1,
            "Risk management": 1,
            "NASCAR": 2,
            "International Space Station": 2,
            "Astronaut": 2
          },
  "s":    {
            "9077": 1, "12923": 2,"12922": 1, "9072": 1,
            "5015": 1, "5016": 1, "2653": 1, "12897": 3
          },
  "u":    ["android", "opera mobile 30", "android"]
}
```

Here, each ID of an article that a user reads is saved under “a” key; with a respective time an article was accessed. User might read news in different countries, so “c” key contains countries and the count of articles read in each country. Key “g” contains the count of read articles for each of the category present in Discover. Articles' annotations are saved under “ann” key. Each source ID is saved in “s” key; with the number of times source was accessed. Finally, “u” field contains user agents that were used when accessing Discover.

This data set gives variety of possibilities for creating personalized suggestions:

- By knowing the countries a user has read stories from it is possible to combine news from different countries in one list. This way the reader will not have to switch country every time he/she wants to read news pieces in another language. The count of articles accessed in each country can help to distribute news for the user according to the popularity of these countries.
- With the data on categories access available, Discover can suggest a new set of articles, prioritizing the categories user is most interested in.
- Annotations from the articles user was interested in can help in suggesting new content on similar topics.
- By knowing the sources that user chooses most of the time, more content from those news publishers can be suggested.

While creating a personalized stream of news, it is important not to forget about other content available in Discover. Even if the user is not so actively reading news for some of the categories, it is vital to display those articles anyway. This approach will help to maintain a high level of relevancy, and at the same time will provide an element of discovery.

Improvements, suggested in this chapter, will rely on content-based filtering. The characteristics of news articles (e.g. category, language, annotations, original source, etc.) will be used by RS in order to find similar items to be presented to the end user. By providing personalized content-based suggestions, relevancy of content in Discover for each individual user will be improved. News stories will become more important and popular (as recommender will know the taste of the user), and more variation in content will be achieved.

6.2.2 Comparing user profiles

User profiling provides a wide range of possibilities for suggesting personalized content. Collected data can be used not only to provide content-based suggestions, but also to apply some collaborative filtering techniques. As similar users tend to have some common interests, articles that one user has already read can be suggested to another user whose reading habits are found to be similar by a recommender system.

By applying collaborative filtering, Discover will be able to suggest content that might be interesting to identical users by comparing the similarity between user profiles. An example below shows simplified profiles for two similar users.

TABLE 4. User profiles for two similar users

User 1	User 2

"a": {
"article_id1":
"access_date1",
"article_id2":
"access_date3",
"article_id3": "
access_date5"
}
	"a": {
	"article_id1":
	"access_date2",
	"article_id2": "
	access_date4"
	}

"ann": {
"annotation1": 1,
"annotation2": 2,
"annotation3": 2,
"annotation4": 2
}
	"ann": {
	"annotation1": 1,
	"annotation2": 2,
	"annotation3": 2
	}

"s": {
"id1": 1,
"id2": 2,
"id3": 1,
"id4": 1,
"id5": 3
}
	"s": {
	"id1": 1,
	"id2": 2,
	"id3": 1,
	"id4": 1
	}

As can be seen, User 1 and User 2 have similar reading habits: article IDs 1 and 2 are the same, annotations 1, 2 and 3 are the same, and sources IDs 1, 2, 3 and 4 are the same as well. By analysing this data, it can be concluded that both users tend to have common interests. User 1, however, has more data available in the profile (highlighted with grey colour). Thus, this data can be used to suggest new content for User 2. In this case, User 2 can be suggested the following content: article with ID 3, articles having annotation 4, and source with ID 5.

Of course, this is an idealised example presented to show the idea of profiles comparison. In a real situation, more data from more profiles should be analysed before making any suggestions to similar users. However, the concept behind finding similarities in user profiles is clear.

It is important to note, that even though profiles of users might contain a lot of similarities, there is no guarantee that they will actually share some common interests. Because of that, collaborative filtering should be used with high caution, and the suggestions obtained by this method should get a lower rank than content-based suggestions.

Nevertheless, by applying collaborative filtering, it would be possible to keep reader's interest in the service high enough, which will result in better user engagement and greater content relevancy. By keeping the right ratio between content-based and collaborative suggestions, Discover might achieve improved results in relevancy of the recommended content.

7 CONCLUSION

The aim of this thesis work was to conduct research on relevancy and freshness of the suggested content in Discover and provide ideas for improvements. It was concluded that while the Discover feature has a lot of potential, there are some drawbacks that influence content relevancy. By analysing those weaknesses, several ideas for relevancy improvement were suggested. While it was not possible to test the suggested improvements, they are expected to increase content relevancy according to the theoretical background provided.

The objectives set in the thesis were completed as research progressed:

- General information on Opera Software and Discover service was retrieved
- Research on filtering techniques applicable in recommendation systems was conducted
- Filtering techniques currently used in Discover were analysed
- The concept of content relevancy was revealed
- Relevancy criteria that can help to improve recommendations were stated
- Current issues in relevancy of content in Discover were highlighted
- Possible ways to improve content relevancy were suggested

By completing this thesis, the author has gained a lot of knowledge in the area of recommender systems and content relevancy. The results of research executed are considered to be beneficial in relation to later working life.

During the process of thesis writing, the author has experienced several difficulties that were mostly connected with retrieving information on the required topics. While a lot of sufficient information is present on the subject of recommender systems and filtering techniques on the Internet and in scientific papers, the issue of content relevancy for news aggregators is not studied enough yet. As a result, a lot of work had to be done to find applicable information on that matter and apply it in practise.
Another challenge of the thesis was to structure all the material in the right form to make this paper easily readable. Therefore, some of the information had to be omitted from the theoretical part and some parts were added later to cover some topics more broadly.

Even though all the challenges were overcome as the research progressed, there is still room for improvements. More data could be gathered on the topic of content relevancy to present different opinions, and more interviews could be conducted with the Discover team to obtain additional information. Live testing of the suggested improvements could be held to prove the theory in practise.

Nevertheless, accomplished results and recommendations are considered to be reliable, as they were based on various theoretical research of the current problems in Discover.

REFERENCES

Lawson B. 2013. *300 million users and move to WebKit* [web publication]. Opera Developer News [accessed 13 May 2015]. Archive copy available from: http://web.archive.org/web/20130214043259/http://my.opera.com/ODIN/blog/300-million-users-and-move-to-webkit

Authors. *The Chromium Project* [electronic file]. Chromium [accessed 13 May 2015]. Available from: https://src.chromium.org/viewvc/chrome/trunk/src/AUTHORS

Barth A. 2013. *Blink: A rendering engine for the Chromium project* [web publication]. Chromium blog [accessed 10 May 2015]. Available from: http://blog.chromium.org/2013/04/blink-rendering-engine-for-chromium.html

Opera Software ASA 2013. *Discover the new Opera* [web publication]. Opera Software [accessed 12 May 2015]. Available from: http://www.operasoftware.com/press/releases/desktop/2013-05-28

Streppone C. 2014. Interesting titbits about Discover in 2014 [email]. Recipient Anton Shakhov. Sent 12 December 2014 [accessed 12 December 2014]

Ricci F., Rokach L. and Shapira B. 2011. *Introduction to Recommender Systems Handbook* [web publication]. Springer Science + Business Media [accessed 15 May 2015]. Available from: http://www.inf.unibz.it/~ricci/papers/intro-rec-sys-handbook.pdf

Burke R. 2002. Hybrid recommender systems: Survey and experiments. *User modelling and user-adapted interaction*, 331–370.

Linden G., Smith B., York J. 2003. *Amazon.com Recommendations. Item-to-Item Collaborative Filtering*. IEEE Internet Computing.

Mild A., Natter M. *A critical view on recommendation systems* [web publication]. Austria: Vienna University of Economics and Business Administration [accessed 26 May 2015]. Available from: http://www.wu-wien.ac.at/am

Rajaraman A., Ullman J. D. 2011. Mining of Massive Datasets. *Data Mining* [electronic file]. Available from: http://i.stanford.edu/~ullman/mmds/ch1.pdf [accessed 26 May 2015]

Adomavicius G., Tuzhilin A. 2005. *Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions* [electronic file]. Available from: http://homepages.dcc.ufmg.br/~nivio/cursos/ri13/sources/recommender-systems-survey-2005.pdf [accessed 26 May 2015]

Burke R. 2002. Proc. of the User Modelling and User-Adapted Interaction, Vol. 12, No. 4. *Hybrid Recommender Systems: Survey and Experiments*, 331-370.

Murat G., Şule, G. Ö. 2010. Expert Systems with Applications, Vol. 37, No. 4. *Combination of Web Page Recommender Systems,* 2911-2922.

Streppone, Cosimo 2015. Operations Lead. Opera Software ASA. Oslo June 2015. Telephone conversation.

Chowdhury S., Landoni M. 2006. Online Information Review, Vol. 30. *News aggregator services: user expectations and experience*, 100-115.

The Pew Research Center's Project for Excellence in Journalism. *An annual Report on American Journalism 2013*. The State of the News Media [web publication]. Available from: http://www.stateofthemedia.org/2013/

Outsell 2009. *News Users*. Discussion paper.

Hjørland B., Sejer Christensen F. 2002. Journal of the American Society for Information Science and Technology, 53(11). *Work tasks and socio-cognitive relevance: a specific example*, 960-965.

Mitovska, Petya. Test major sites if they get feeds updated at the same time as page content [Opera bug tracking system]. 19 August 2014 [accessed 9 June 2015].

Kjellin, Simen. Are Top Stories really top stories? [Opera bug tracking system]. 4 August 2014 [accessed 9 June 2015].

Ontotext. Semantic Technology Products. Ontotext Semantic Platform. Semantic Annotation [web page]. [accessed 11 June 2015]. Available from: http://ontotext.com/

Tang, Y. 2011. *Onto-Ann: An Automatic and Semantically Rich Annotation Component for Do-It-Yourself Assemblage* [electronic file]. Vrije Universiteit Brussel. Available from: http://www.academia.edu/ [accessed 11 June 2015]

Ferragina P., Scaiella U. 2012. TAGME. Demo page [web page]. A³ Lab
Dipartimento di Informatica, University of Pisa. Available from: http://tagme.di.unipi.it/ [accessed 11 June 2015]

Google 2012. *A newly hatched way to tag your news articles* [web publication]. Google News Blog [accessed 11 June 2015]. Available from: http://googlenewsblog.blogspot.fi/