



NTNU – Trondheim
Norwegian University of
Science and Technology

Mobile News

Design, User Experience and
Recommendation

Kent Robin Haugen

Master of Science in Computer Science

Submission date: June 2013

Supervisor: Jon Atle Gulla, IDI

Norwegian University of Science and Technology
Department of Computer and Information Science

Problem Description

This thesis focuses on three main topics, namely a mobile application, with the use of a recommender system to help filter the content, narrowed down to a news application's perspective and following are the research questions that this thesis aims to give answers to.

- What are possible perspectives on news in a personalized news recommender system and how are they related?

In which way is it possible to represent news articles and how can one view representing a news article be translated to another view. What type of information is represented in each of the different views and can one view be misleading or suffer from a lack of information compared to another view, if presented a part from the other perspectives?

- What are particular features of mobile user interfaces that affect the mobile news user experience?

In which way are mobile user interfaces different from, for instance, desktop computer user interfaces? What considerations has to be done when designing a mobile user interface and how do the interaction methods differ from one another?

- What are relevant perspectives on news in current mobile news apps?

Subjecting a handful of already developed mobile news applications, how is the news presented? Which types of perspectives are used in the different apps? Are different perspectives used in the same app, if so, how are linked together and what do they represent, in terms of information?

- How can these perspectives be supported on a mobile platform to increase user experience and provide maximum flexibility?

What is the main goal for the different perspectives and are they successful in terms of presenting the information that the user wants? What do they convey and is this perspective contributing to a better user experience? Do these perspective depend on other technologies or APIs found on the mobile device, like GPS or gyroscope?

Abstract

As news consumption is shifting towards the digital domain, with mobile devices in particular, mobile news applications are becoming increasingly popular.

This thesis focuses on a combination of mobile applications, with the use of recommender systems, in a news application's perspective, by identifying the state of mobile news applications available today and analyzing how they are accomplished, as well as developing a mobile news recommender application, as a use case associated with this project to see how this can be realized from a developer's perspective.

Most of the commercial applications examined in this project follows a similar design pattern concerning user interfaces and user interaction, but there are some that breaks with this trend. The use of personalization and recommendation techniques as an underlying technology for news applications are becoming more wide spread, especially the use of active personalization making use of implicit user feedback.

Sammendrag

Som et resultat av at nyhetskonsumering stadig er i forandring mot det digitale mediet, og da spesielt med tanke på mobile enheter, har mobile nyhetsapplikasjoner blitt mer og mer populært.

Denne avhandlingen fokuser på en kombinasjon av mobile applikasjoner, som tar i bruk anbefalingssystemer, i et nyhetsapplikasjons perspektiv, ved å identifisere tilstanden til mobile nyhetsapplikasjoner som finnes i dag og ved å analysere hvordan de er gjennomført. I tillegg til at det blir utviklet en mobil anbefalende nyhetsapplikasjon som et use-case tilknyttet denne avhandlingen for å se hvordan en slik applikasjon kan bli realisert, sett fra et utviklers perspektiv.

De fleste av de kommersielle applikasjonene som er undersøkt i forbindelse med dette prosjektet følger et lignende designmønster med tanke på brukergrensesnitt og brukerinteraksjon, men det er noen som bryter med denne trenden. Bruken av personalisering og anbefalingsteknikker som en underliggende teknologi for nyhetsapplikasjoner blir stadig mer populært, spesielt da aktiv personalisering som tar i bruk implisitt tilbakemeldinger fra brukerne.

Preface

This report holds the work carried out in my master thesis as part of my Master of Science degree at the Norwegian University of Science and Technology. The topic chosen for this thesis was based on my personal interest in mobile application design and development, as well as a genuine curiosity towards trending technologies and applications.

I would like to thank my supervisors Professor Jon Atle Gulla at the Department of Computer and Information Science, NTNU and Erik Berg at Telenor for the guidance and valuable feedback throughout the project.

I also wish to thank my fiancé for her patience and support, considering we both are continuously juggling studies, work and parenting our one year old daughter.

Trondheim, June 9, 2013

Kent Robin Haugen

Contents

| | | |
|-----------|---|-----------|
| I | Introduction | 1 |
| 1 | Introduction | 3 |
| 1.1 | Problem | 3 |
| 1.1.1 | Research questions | 4 |
| 1.2 | Approach | 5 |
| 1.3 | Results | 6 |
| 1.4 | Report Structure | 7 |
| II | Theoretical Background | 9 |
| 2 | Theoretical Overview | 11 |
| 2.1 | Mobile User Interface | 11 |
| 2.2 | Personalization | 11 |
| 2.3 | Recommender System | 12 |
| 2.3.1 | Filtering | 12 |
| 2.4 | News Application | 13 |
| 2.4.1 | News Recommendation | 13 |
| 3 | Related Work | 15 |
| 3.1 | Personalization on mobile devices | 15 |
| 3.2 | Personalized news | 16 |
| 3.2.1 | Filtering | 19 |
| 3.2.2 | Mobile News | 20 |
| 3.3 | News consumption | 23 |
| 3.4 | Designing news applications | 23 |
| 3.5 | Commercial mobile news applications | 25 |
| 3.5.1 | Zite | 26 |
| 3.5.2 | Flipboard | 28 |
| 3.5.3 | Pulse | 30 |
| 3.5.4 | Summly | 31 |

| | | |
|------------|---|-----------|
| 3.5.5 | News360 | 33 |
| 3.5.6 | Circa | 35 |
| 3.5.7 | Wavii | 37 |
| 3.5.8 | Prismatic | 39 |
| 3.5.9 | Taptu | 42 |
| 3.5.10 | Feedly | 43 |
| 4 | The Mobile News Recommender Use Case | 47 |
| 4.1 | Client application | 47 |
| 4.1.1 | User Interface Design | 48 |
| 4.1.2 | Technology | 52 |
| III | Mobile News User Interfaces | 57 |
| 5 | Perspectives | 59 |
| 5.1 | The Different Perspectives | 59 |
| 5.2 | Relations | 64 |
| 5.3 | Purpose of and Flow Between Perspectives | 66 |
| 5.4 | Perspectives and Their Connections in the Different Apps | 69 |
| 5.4.1 | Zite | 69 |
| 5.4.2 | Flipboard | 70 |
| 5.4.3 | Pulse | 70 |
| 5.4.4 | Summly | 71 |
| 5.4.5 | News360 | 72 |
| 5.4.6 | Circa | 72 |
| 5.4.7 | Wavii | 73 |
| 5.4.8 | Prismatic | 74 |
| 5.4.9 | Taptu | 74 |
| 5.4.10 | Feedly | 75 |
| 5.4.11 | Use Case | 75 |
| 5.5 | How the Perspectives Are Supported on the Mobile Platform | 76 |
| 6 | Mobile Features | 79 |
| 6.1 | Display | 79 |
| 6.2 | Interaction | 80 |
| 6.3 | Mobility | 81 |
| 6.3.1 | User Location | 81 |
| 6.3.2 | Connectivity | 81 |
| 6.3.3 | Scanning | 81 |
| 6.4 | Software | 81 |

| | |
|---|------------|
| <i>CONTENTS</i> | xi |
| 6.5 Monetizing | 82 |
| 6.5.1 Advertising | 82 |
| 6.5.2 Paid Subscription | 83 |
| 6.5.3 In-app Purchases | 83 |
| 6.5.4 Paid Applications | 84 |
| 7 Comparing the Applications | 85 |
| 7.1 Table Comparison | 85 |
| 7.1.1 Explanation | 85 |
| 7.1.2 The Table | 89 |
| 7.1.3 Key characteristics | 93 |
| IV Discussion and Conclusion | 95 |
| 8 Discussion | 97 |
| 8.1 Perspectives | 97 |
| 8.2 Mobile News Applications | 98 |
| 8.2.1 Recommendation Apps | 99 |
| 8.2.2 Summarization Apps | 99 |
| 8.2.3 Advanced RSS Apps | 100 |
| 8.3 Use Case Implementation | 101 |
| 9 Conclusion | 103 |
| 10 Further Work | 105 |
| Appendices | 113 |
| A Implementation of Use Case Application | 115 |
| A.1 Overview | 115 |
| A.1.1 Technical Specifications | 115 |
| A.2 External Libraries | 116 |
| A.3 Screenshots | 117 |

List of Figures

| | | |
|------|---|----|
| 1.1 | The main focus area of the thesis. | 4 |
| 2.1 | The shifting of how people access news from 1991 to 2012 shown in percent of people asked. | 14 |
| 3.1 | The learning curves of the different models used in the adaptive news access framework. | 17 |
| 3.2 | Experience rating of the aggregated news system. | 17 |
| 3.3 | Growth of adaptive news at 11 national US and UK websites, 2007-2010. | 18 |
| 3.4 | Precision rates showing the hybrid system versus the content-based system. | 19 |
| 3.5 | Live traffic click ratios for comparing PLSI and MinHash algorithms. | 21 |
| 3.6 | The ePaper shown on two different devices. | 21 |
| 3.7 | The flow chart of the MONERS mobile news recommendation system. | 22 |
| 3.8 | Screenshot from News Sync showing the search result for Wattergate. | 24 |
| 3.9 | Screenshot from 247 showing the individual UI design along with an example of individual usage. | 25 |
| 3.10 | Screenshot from 247 showing the peripheral UI design along with an example of peripheral usage. | 26 |
| 3.11 | Screenshots from Zite showing the Top Stories feed, a single news article, and categories list. | 27 |
| 3.12 | Screenshots from Flipboard showing the top categories, a single news article, and the settings view. | 29 |
| 3.13 | Screenshots from Pulse showing the start page, a single news article, and the categories/settings view. | 31 |
| 3.14 | Screenshots from Summly showing the category selection page, a single news article, and the category settings view. | 33 |

| | | |
|------|---|----|
| 3.15 | Screenshots from News360 showing the top stories feed, a single news article, and the category settings/selection view. | 34 |
| 3.16 | The work flow of the News360 news system. | 36 |
| 3.17 | Screenshots from Circa showing the top stories feed, a single news article, and the category selection view. | 37 |
| 3.18 | Screenshots from Wavii showing the top stories feed, a single news article, and the category selection/settings view. | 39 |
| 3.19 | Screenshots from Prismatic showing the top stories feed, a single news article, and the category selection/settings view. | 40 |
| 3.20 | An illustration of how the prismatic user's interest profile are stored created by Aria Haghghi, co-founder of Prismatic. | 41 |
| 3.21 | Screenshots from Taptu showing the top stories feed, a single news article, and the category settings view. | 43 |
| 3.22 | Screenshots from Feedly showing the top stories feed, a single news article, and the category settings view. | 45 |
| 4.1 | Screenshots from the client application showing the start page and the category selection screen. | 49 |
| 4.2 | Screenshots from the client application showing the RSS view and the RSS view after triggering the share/save menu. | 50 |
| 4.3 | Screenshots from the client application showing the full article view and the map view. | 51 |
| 4.4 | Conceptual drawing showing the architectural view of the whole Smartmedia Mobile News Recommender System. | 55 |
| 5.1 | A screenshot from Prismatic showing an example of a full article perspective. | 60 |
| 5.2 | A screenshot from Feedly showing an example of an RSS perspective. | 60 |
| 5.3 | A screenshot from NewsCloud showing an example of an entity perspective. | 61 |
| 5.4 | A screenshot from Wavii showing an example of an event perspective. | 62 |
| 5.5 | A screenshot from Flipboard showing an example of a web perspective. | 63 |
| 5.6 | A screenshot from Summly showing an example of a summary perspective. | 63 |
| 5.7 | A screenshot from Circa showing an example of a map perspective. | 64 |
| 5.8 | A simple navigation chart showing the connection between perspectives in the mobile news application Zite. | 69 |

| | | |
|------|---|-----|
| 5.9 | A simple navigation chart showing the connection between perspectives in the mobile news application Flipboard. | 70 |
| 5.10 | A simple navigation chart showing the connection between perspectives in the mobile news application Pulse. | 71 |
| 5.11 | A simple navigation chart showing the connection between perspectives in the mobile news application Summly. | 71 |
| 5.12 | A simple navigation chart showing the connection between perspectives in the mobile news application News360. | 72 |
| 5.13 | A simple navigation chart showing the connection between perspectives in the mobile news application Circa. | 73 |
| 5.14 | A simple navigation chart showing the connection between perspectives in the mobile news application Wavii. | 73 |
| 5.15 | A simple navigation chart showing the connection between perspectives in the mobile news application Prismatic. | 74 |
| 5.16 | A simple navigation chart showing the connection between perspectives in the mobile news application Taptu. | 74 |
| 5.17 | A simple navigation chart showing the connection between perspectives in the mobile news application Feedly. | 75 |
| 5.18 | A simple navigation chart showing the connection between perspectives in the use case mobile news application. | 76 |
| 8.1 | Statistics from the use case application showing how many times the top stories and settings screen are shown in a month. | 101 |
| A.1 | Screenshot of the use case application showing the front page screen. | 118 |
| A.2 | Screenshot of the use case application showing the category selection screen. | 118 |
| A.3 | Screenshot of the use case application showing the settings screen. | 119 |
| A.4 | Screenshot of the use case application showing the RSS perspective. | 119 |
| A.5 | Screenshot of the use case application showing the RSS perspective after triggering the share/save control. | 120 |
| A.6 | Screenshot of the use case application showing the Twitter share screen. | 120 |
| A.7 | Screenshot of the use case application showing the Facebook share screen. | 121 |
| A.8 | Screenshot of the use case application showing the email share screen. | 121 |

| | |
|--|-----|
| A.9 Screenshot of the use case application showing the full article perspective. | 122 |
| A.10 Screenshot of the use case application showing the map perspective. | 122 |
| A.11 Screenshot of the use case application showing the full text screen. | 123 |

Abbreviations and Acronyms

| | |
|---------------|---|
| AHP | Analytic Hierarchy Processing |
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| App(s) | Application(s) |
| BN | Bayesian Network |
| GPS | Global Positioning System |
| JSON | JavaScript Object Notation |
| NER | Named Entity Recognition |
| NFC | Near Field Communication |
| NLP | Natural Language Processing |
| NPR | National Public Radio |
| OS | Operating System |
| P2P | Peer-to-Peer |
| PC | Personal Computer |
| PDA | Personal Digital Assistant |
| PLSI | Probabilistic Latent Semantic Indexing |
| QR | Quick Response |
| REST | Representational State Transfer |
| RFID | Radio-Frequency Identification |
| RSS | Rich Site Summary |
| SDK | Software Development Kit |
| SMNRS | Smartmedia Mobile News Recommender System |
| SMS | Short Message Service |
| UI | User Interface |
| URL | Uniform Resource Locator |
| XML | Extensible Markup Language |

Part I
Introduction

Chapter 1

Introduction

From the rise of the smart mobile era, starting around the mid 2000s, content consumption has shifted more and more towards the mobile scene, as the mobile devices has gotten more powerful, with larger screens and Internet access has gotten more ubiquitous. Information that previously was only accessible in paper form, i.e. books and newspapers, also has had a shifting trend towards the digital platform. With all this content and information available to almost everybody and almost everywhere in the world, the importance of being able to filter it and getting only the content that is relevant or interesting to a single user is a highly relevant and an important area of research. Just as important to be able to filter this information, is how to present the filtered information. With the small screens that follows with the mobile devices, compared to for instance desktop computers, how the information is presented, is crucial to a mobile application's usability.

1.1 Problem

This thesis will focus on a combination of these three main topics, namely a mobile application, with the use of a recommender system to help filter the content, narrowed down to a news application's perspective, as shown in figure 1.1.

This will result in a mobile news recommender application, developed for iOS on the iPhone, where the main focus will be on the user interface part of the system, and how this part of the application can be created. Figure 4.4 shows how the iPhone application to be developed in association with this thesis, links to the other parts of the system as a whole. This thesis does not have as a goal to evaluate or develop recommendation and personalization techniques, but focuses on the client part of the whole news recommendation

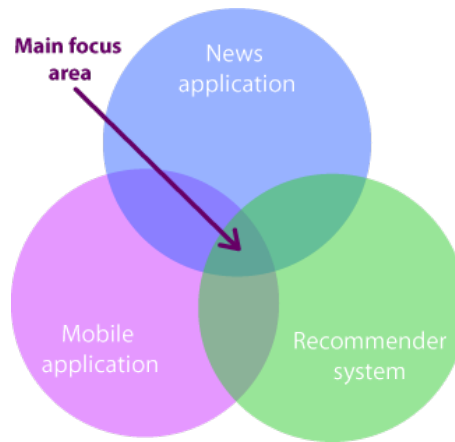


Figure 1.1: The main focus area of the thesis.

system.

1.1.1 Research questions

Following are the research questions that this thesis aims to give answers to.

What are possible perspectives on news in a personalized news recommender system and how are they related?

In which way is it possible to represent news articles and how can one view representing a news article be translated to another view. What type of information is represented in each of the different views and can one view be misleading or suffer from a lack of information compared to another view, if presented a part from the other perspectives?

Different perspectives to present news via will be presented, as well as how they are related and corresponds with each other.

What are particular features of mobile user interfaces that affect the mobile news user experience?

In which way are mobile user interfaces different from, for instance, desktop computer user interfaces? What considerations has to be done when designing a mobile user interface and how do the interaction methods differ from one another?

The most relevant differences between a mobile user interface and other interfaces where news can be accessed will be presented. Also how these

differences may have an impact on the news reading abilities on a mobile device will be discussed.

What are relevant perspectives on news in current mobile news apps?

Subjecting a handful of already developed mobile news applications, how is the news presented? Which types of perspectives are used in the different apps? Are different perspectives used in the same app, if so, how are linked together and what do they represent, in terms of information?

Approximately a dozen of commercial and non-commercial mobile news application will be targeted and the different perspectives in each application and how they are related in the application will be presented.

How can these perspectives be supported on a mobile platform to increase user experience and provide maximum flexibility?

What is the main goal for the different perspectives and are they successful in terms of presenting the information that the user wants? What do they convey and is this perspective contributing to a better user experience? Do these perspective depend on other technologies or APIs found on the mobile device, like GPS or gyroscope?

The different perspectives presented will be evaluated in terms of functionality and how they can contribute to improving the news reading experience, or on the other hand, create more confusion than supplying the user with additional useful information.

The Smartmedia Mobile News Recommender Use Case

As a use case for this thesis, the Smartmedia Mobile News Recommender system¹ will be applied. A mobile application will be developed on top of the existing back end, as shown in figure 4.4, limited to the iOS platform, for the sake of simplicity.

Further the use case will be compared to existing mobile news applications in terms of available perspectives and functionality.

1.2 Approach

Following are the main approaches to answer the research questions.

¹The Smartmedia Mobile News Recommender system's project web site can be found at <http://smartmedia.idi.ntnu.no>.

1. Select a particular use case to design and develop within the domain of news recommendation, and make use of an already developed recommender system that provides real-time news from all the major Norwegian newspapers.
2. Identify the state of commercial and non-commercial contributions and news applications with their technology and perspectives.
3. Compare and discuss how the contributions are related and realized.
4. The application will also be presented and discussed with representatives from the media sector and researchers on web applications, after the thesis has been delivered.

1.3 Results

The result of this project focuses upon identifying the state of mobile news applications that are available today through looking at commercial applications, as well as published papers on the topic, and especially how they are realized in terms of design, navigation logic, recommendation, and the mobile features to support all of this. A use case application is also a part of the result in terms of an practical approach on how to realize a mobile news recommendation application.

Implementing the client part of a news recommendation application to make it comparable to commercial news applications, is not a very difficult task, if the developer has a bit experience within the domain. The use case application was developed by a single student alongside writing this thesis. The main issue is to realize a working recommendation system, which the client application makes use of, seeing that such a back-end system craves a lot more effort and competence to be able to create, than a simple client application that makes use of the data it gets served.

In the use case application, one of the focus areas were to limit the use of buttons, and replacing them with gestures wherever possible, to not clutter the small screen with other elements than the information the application is trying to convey. This works to some extent, particularly for navigational purposes buttons are easily replaceable, although the UI may not be as intuitive as when using buttons that tells the user what is going to happen when a button is pushed. One can probably not replace all buttons with gestures, as this would require much learning before being able to use an application as intended.

1.4 Report Structure

Chapter 2: Theoretical Overview

A quick introduction to important terms and technologies associated with mobile devices, news applications, and recommendation.

Chapter 3: Related Work

A presentation of related work done on one or more of the topics that this thesis focuses upon, mainly within some aspect of a mobile news recommender application.

Chapter 4: The Mobile News Recommender Use Case

The presentation of the use case associated with this thesis, both front-end and back-end.

Chapter 5: Perspectives

The most common perspectives used in mobile news applications, their correlations and purpose will be presented, as well as how they are used in the different mobile news applications presented in section 3.5.

Chapter 6: Mobile Features

Which mobile features that are common in mobile news applications and how they are different from other systems, as well as how they support a mobile news application will be presented.

Chapter 7: Comparing the Applications

The applications examined in section 3.5, as well as the use case application presented in chapter 4, will be compared.

Chapter 8: Discussion

The different perspectives' usability will be discussed, and the examined news applications will be classified into three main categories. The implementation of the use case will also be discussed.

Chapter 9: Conclusion

The conclusion drawn from this study will be presented.

Chapter 10: Further Work

A set of proposals for further work will be listed and briefly explained.

Part II

Theoretical Background

Chapter 2

Theoretical Overview

This chapter will explain terms and technologies associated with personalization and recommendation, mobile user interfaces and digital news applications, to form a basis of what is often used when creating a mobile news recommendation application.

2.1 Mobile User Interface

After the rise of the smart phone, mobile content consumption has seen an incredible growth and is still growing[12], making the mobile user interface more and more common to people everywhere. The focus here will be on the smart phone, and the iPhone in particular, but applies to a lot of other smart phones manufactured as well.

Mobile devices brings with it a lot of opportunities, but has its limitations as well, compared to a desktop computer, for instance. Mobile screens are smaller than PC screens and are often touch-based, making the design approach for a mobile user interface quite different.

A PC screen can normally hold more information in one view than a mobile device, so to display the same amount of information on a mobile screen, several views needs to be used and some form of navigation logic between them needs to be established.

2.2 Personalization

According to Cylogy, personalization is "the process of deciding - given a large set of possible choices - what has the highest value to an individual" [17].

[49] defines it as "A form of user-to-system interactivity that uses a set of technological features to adapt the content, delivery, and arrangement of

a communication to individual users' explicitly registered and/or implicitly determined preferences."

So simple and a bit blunt stated: "give the users what they want, without them asking for it."

2.3 Recommender System

"The goal of a Recommender System is to generate meaningful recommendations to a collection of users for items or products that might interest them." [32]

Recommender systems are becoming widely used and is a necessary approach to deal with the ever growing information overload situation that are occurring nowadays. These systems can be found in many different domains, like online shopping, reading news, listening to music or streaming other types of visual media, like movies or series.

2.3.1 Filtering

When building a recommender system there are three main approaches to follow, namely content-based filtering, collaborative filtering or a hybrid of these two.

Content-based Filtering

Content-based filtering is a filtering technique where the items proposed are chosen based on similar items to what the system thinks the user is interested in. The items have some sort of classifying meta data, that makes them comparable to other items, and the item with the best match to the user's profile will be the item proposed. The user profile can be built by extracting and classifying events from the user's click log, for instance, or by explicit data the user supplies in terms of categories of interest etc.

Content-based filtering is dependent on knowledge about the items to be able to classify and cluster them, and sophisticated methods based on machine learning and NLP are often used to fulfill this purpose.

Collaborative Filtering

Collaborative filtering is a filtering technique that proposes items based on which items other similar users have been accessing. Often large amount of user data from other users' activities or preferences are gathered and analyzed and the users are put into segments based on their behavior. If a user from

one segment accesses an item, this item can be proposed as a recommended item for another user within this segment.

An advantage of this approach as opposed to content-based filtering, is that collaborative filtering does not need to have an understanding of the item itself to be able to recommend it to a user. On the other hand, collaborative filtering often suffers from cold start[25], scalability and data sparsity[16] problems.

Hybrid Filtering

Hybrid filtering is an approach where content-based filtering and collaborative filtering are combined in some way, and it has shown that this approach is more effective in some cases, as presented in section 3.2.

One way of conducting the hybrid approach is to implement collaborative filtering and content-based filtering separately, and then combining them. Another way is to use some capabilities from one approach and add it to the other approach. One can also merge them together, and apply them as a single model.

2.4 News Application

A news application is an application where a user can access news digitally via the Internet from one or more news publishing sources. This application can be provided by the content publisher itself, or it can be provided by a third party redistributing the content.

News applications are becoming widely used, and the consumption of news are rapidly shifting towards accessing news online and via mobile devices (see figure 2.1)[8].

2.4.1 News Recommendation

News recommendation is when a news application makes use of a recommender system to provide an online news reading user with a news service that are filtered or personalized to the user's preferences and likings.

A challenge with news recommendation, compared to for instance movie or music recommendation, is the freshness of the news. News have a short lifespan, and are best served fresh, making approaches like collaborative filtering a difficult task considering the cold start problem and data sparsity problem.

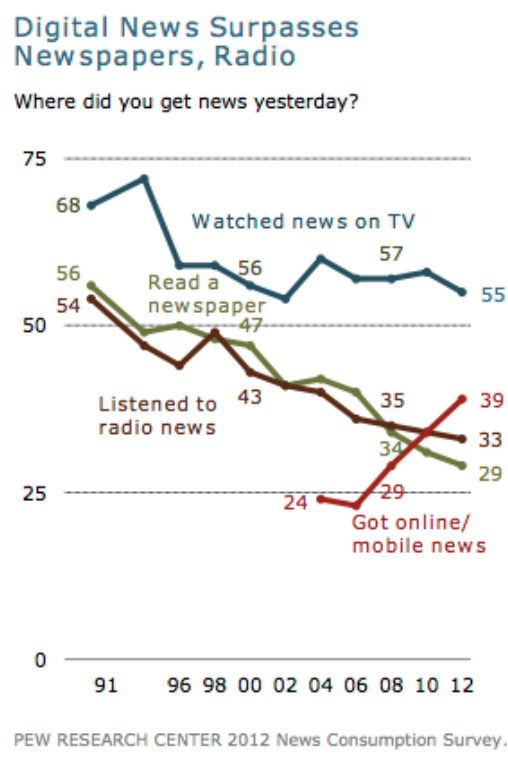


Figure 2.1: The shifting of how people access news from 1991 to 2012 shown in percent of people asked.

Chapter 3

Related Work

In the ever expanding universe of accessible information and the information overload that follows with it, the work on dealing with this overload never sleeps. This chapter will look at work done on handling information overload, especially considering news consumption, and news consumption on mobile devices. Topics like personalization, mobile news applications and news applications in general, designing mobile news applications and information aggregation will be presented in more detail.

3.1 Personalization on mobile devices

Information handling on regular desktop computers compared to mobile devices comes with different problems, possibilities and focus areas, considering for instance hardware capabilities and area of usage.

[38] lists several issues when creating recommender systems for mobile devices:

- The screen is a lot smaller than on desktop computers resulting in a smaller scrollable area which craves a higher workload from the user to display all the information, which makes the probability of the user seeing all the information smaller.
- Mobile devices often has limited input and interaction capabilities, which makes the effort of querying higher than on a regular keyboard.
- Mobile Internet browsing sessions are often shorter than on desktop computers, i.e. in the time span of a couple of minutes.
- Most web pages are not optimized for mobile devices which may result in a badly formatted displaying of the page.

- The cost of retrieving the data using mobile data through Edge or 3G are usually much higher than on personal Internet connections at home or office.

3.2 Personalized news

Personalized news and news applications taking advantage of personalization and recommendation technology is a trending topic now, considering three of the most popular mobile news application has been acquired by major software companies the last couple of months[26][50][2]. Different news personalization approaches has also been researched and proposed the last years.

[3] presents a system with focus on design, deployment and evaluation of a client/server-based framework for adaptive news access. Two news agents are described and deployed, one for the web and one on a PDA. The web version uses explicit feedback, and the PDA version uses implicit feedback, and the individual user models based on this feedback are induced by using machine learning algorithms. This feedback are meant to constantly keep the user profile up to date, and empirically build the user profile.

The feedback are translated into short term and long term models, whereas the short term model can propose articles related to stories that has already been read, or not proposing articles similar to ones that already are read. A user can be interested in two different articles concerning a topic, but will most likely not be interested in two similar articles covering the same story although the topic is considered interesting. The long term model can update the user profile's interests with regard to categories the user finds interesting. If the user has read a lot of articles in one specific category over a longer period of time, a safe bet is that this is an interesting category, and can also be added to the user profile as an interesting category.

The framework also proposes a hybrid model where the short term and long term models are combined in such a way that if the short term model cannot classify the feedback, the long term classification is used.

Figure 3.1 shows the learning curves of the different models, where F1 is a function to measure the performance of the model. The figure shows that the hybrid approach outperforms both the short term and long term models when used individually. The results of the empirical study of this framework also suggests that effective personalization are obtainable without explicit feedback from the user.

[39] proposes a framework for aggregating news articles, extracting the text, images, videos and TV news, turning it into a news event and display-

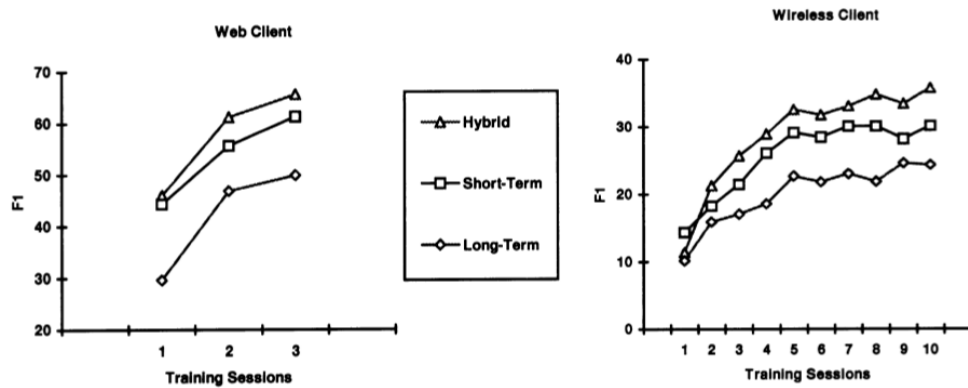


Figure 3.1: The learning curves of the different models used in the adaptive news access framework.

ing this in a lucid and clear user interface. The user is then able to select aggregated events based on semantic data, instead of just news articles.

The user interface has three main views, the overview of the different events, the single event view and the tag cloud. The overview shows a list of different events, grouped by date and ordered by their descending score. The single event view shows a particular event with corresponding related articles, images, videos and a timeline graph showing the evolution of the event over time in terms of number of resources published per day. The tag cloud is a collection of entities that the personalization engine has set as the user's current topics of interests. The bolder font, the more popular that particular topic is, and the users can remove and add topics as they wish.

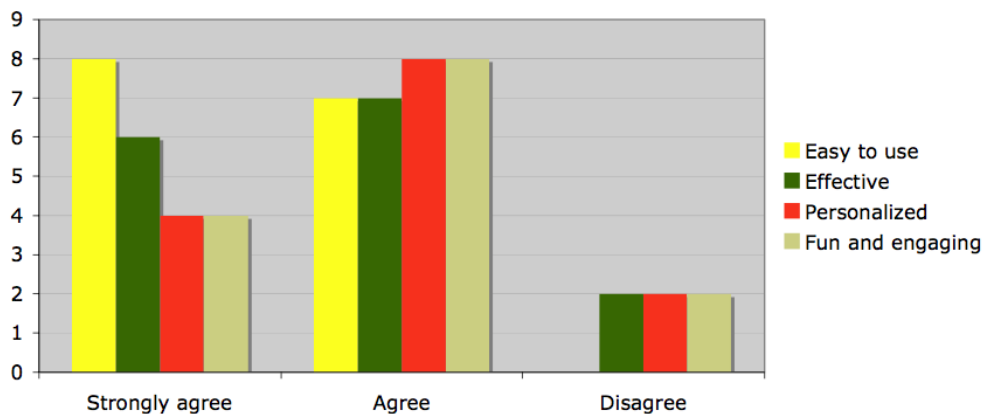


Figure 3.2: Experience rating of the aggregated news system.

The UI allows an easy browsing experience as well as an in-depth under-

standing of an event. Figure 3.2 shows the user feedback of the proposed UI, and it shows that the majority of the users are satisfied with the system.

From the middle of 2007 to late 2010, [49] performed a longitudinal study covering how personalization is done at 11 national news websites in the UK and US. Results from the three content surveys conducted in midst 2007, late 2009 and late 2010, showed that there were an increase of 69 per cent in the number of distinct adaptive news categories, from 70 to 118.

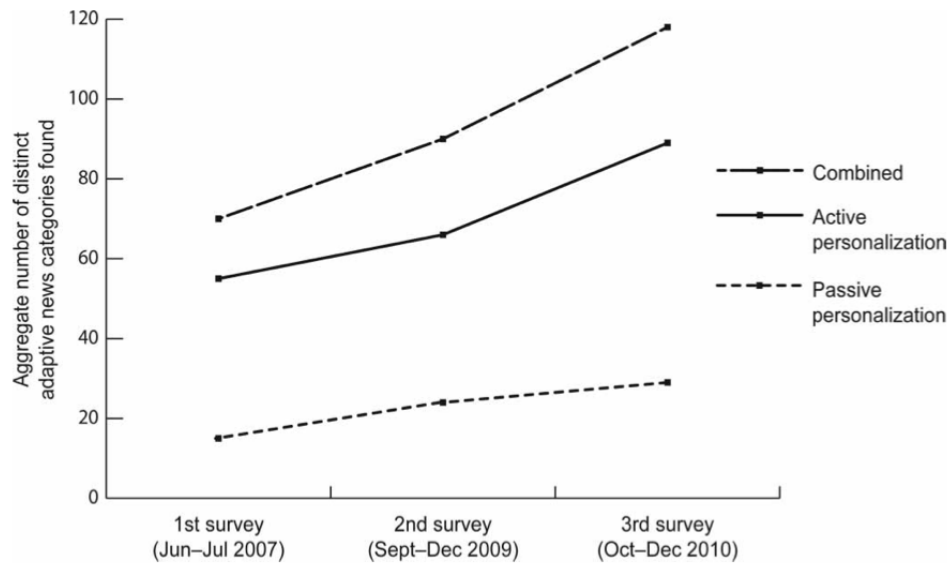


Figure 3.3: Growth of adaptive news at 11 national US and UK websites, 2007-2010.

Another finding was that passive personalization was decreasing while active personalization was increasing, and combined there still was an increase, as shown in figure 3.3. It also states that news content providers needs to increase their use of personalization to target advertising better to each specific user, as static advertising is decreasing and dynamic advertising is increasing.

[29] proposes a scalable two-stage personalized news recommender system, SCENE. The system has two levels of news representation, namely the topics relevant to the user preferences and the specific news articles. Further, the system consists of three main components; clustering of news articles, construction of the user profile and recommendation.

The clustering is done by using probabilistic language models. The user profile construction is done by extracting accessed news content, similar access patterns and preferred name entities from the user's browser log. The

recommendation is done by selecting the most similar news group by looking at the clustered news articles and the user profile. Further, from this group, the recommendation is modeled as a "budgeted maximum coverage problem" [24] and solved in a greedy way. To finish it up, properties like recency and popularity are taken into account.

SCENE is demonstrated to support high quality recommendation result and an efficient clustering on newly published news articles, through extensive evaluation.

3.2.1 Filtering

[52] presents a personalized recommendation of news for web using a hybrid method consisting of both content-based filtering and collaborative filtering. This model provides an autonomous tool that has the ability to minimize tedious and repetitive web surfing for the user. A classification method, consisting of 5 steps, is used to auto-classify the web pages. Models representing the user's degree of interest to a specific topic; the user interest model, and the user's preference to use a web source; the preference model, are created using the Bayes theorem. Also a quantification of the time factor must be done, in addition to the preference model and user interest model, to be able to recommend the news item.

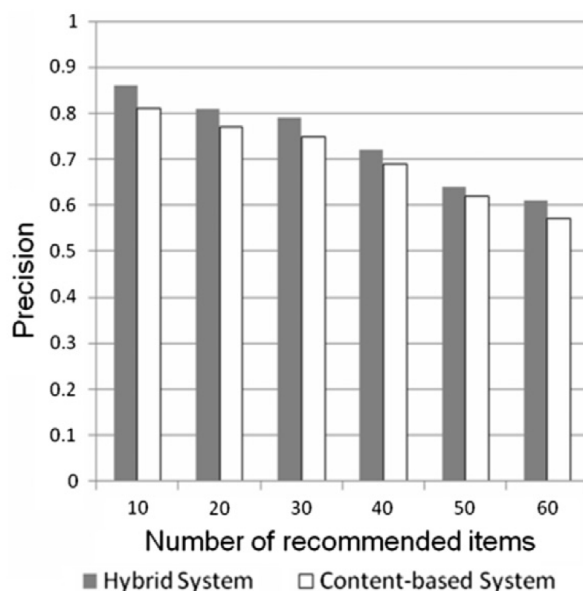


Figure 3.4: Precision rates showing the hybrid system versus the content-based system.

The user's interests are then matched with the contents of the news to present the recommended news item suited for the user. In figure 3.4 the precision rates of the hybrid model versus the same system with just the content-based filtering model approach is showing. The figure shows that the hybrid model has a higher precision than the content-based filtering model.

[30] also proposes a hybrid system where collaborative and content-based filtering are combined. A personalized news recommendation system for Google News is proposed. First a large-scale analysis of anonymized user click logs from Google News was conducted, to learn about the users' news interests. Further a Bayesian framework was developed to predict the users' current news interests based on activities of the current user, and the trending topics based on the activity of all the users, found from the log analysis. At last, the content-based recommendation mechanism combined with an existing collaborative filtering mechanism is used to generate personalized news recommendations.

The presented hybrid system is then tested with Google News and shows that the quality of the recommendations are improved and the number of visitors have increased, compared to when just the existing collaborative filtering mechanism was used.

[11] proposes a collaborative filtering approach for generating personalized recommendations for a large scaled system, namely Google News. These recommendations are generated following three approaches: Probabilistic Latent Semantic Indexing[15], MinHash[6] clustering, and co-visitation¹ count.

The test of the system used on a large fraction of the Google News traffic conducted over a period of several days, demonstrated the efficiency and scalability of the system, and figure 3.5 shows that the algorithms MinHash and PLSI combined almost always performs worse than when run separately.

3.2.2 Mobile News

[42] describes a personalized mobile newspaper system called ePaper, a research prototype system which aggregates content from several different news providers. The news items are classified as concepts from a news domain ontology. Each subscribed user can then receive an electronic newspaper.

Ontological content-based and collaborative filtering algorithms are applied to the user's profile and preferences to personalize the content from the newspapers. To maintain the user's profile, the user's activity log in the application are analyzed to provide an implicit and dynamic update. The

¹Co-visitation means that two different articles have been visited by the same user in a defined period of time[22].

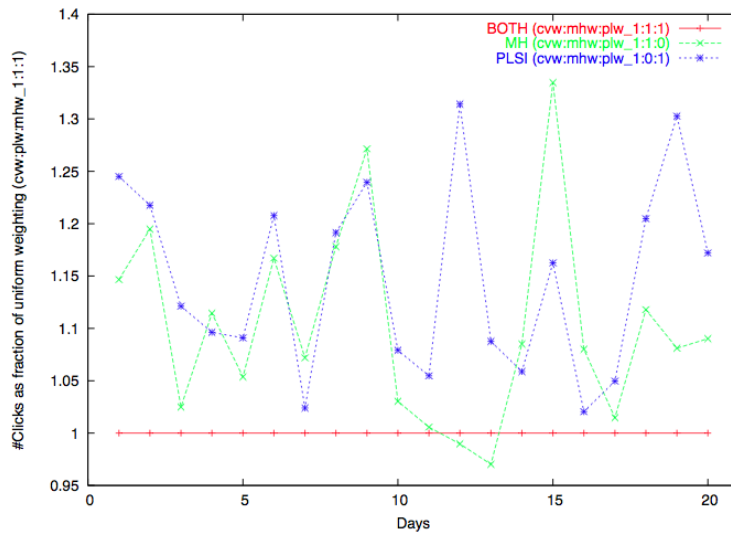


Figure 3.5: Live traffic click ratios for comparing PLSI and MinHash algorithms.

user can also access a news paper reading experience without any form of personalization through ePaper.



Figure 3.6: The ePaper shown on two different devices.

The ePaper is designed in a responsive way, meaning that it is adapted to the screen size and specifications of the device it runs on (see figure 3.6).

[28] proposes a mobile web news recommendation system called MONERS. This system combines the nature of mobile web news content and service with the characteristics of users.

A recommendation of a news article from MONERS is computed in several steps. The weight of an article is calculated by the degree of importance and the recency of that specific article. Users are assigned into category segments, where these segments are formed through k-mean clustering[48] using Euclidian distance measure[10]. The change in the user interests, measured as the weights of a category preference at the user's side, determines which category segments this user belongs to. The article preference of a user segment is also computed. These computations forms the recommendation that is delivered to the user. Figure 3.7 shows the flow of the mobile recommendation system.

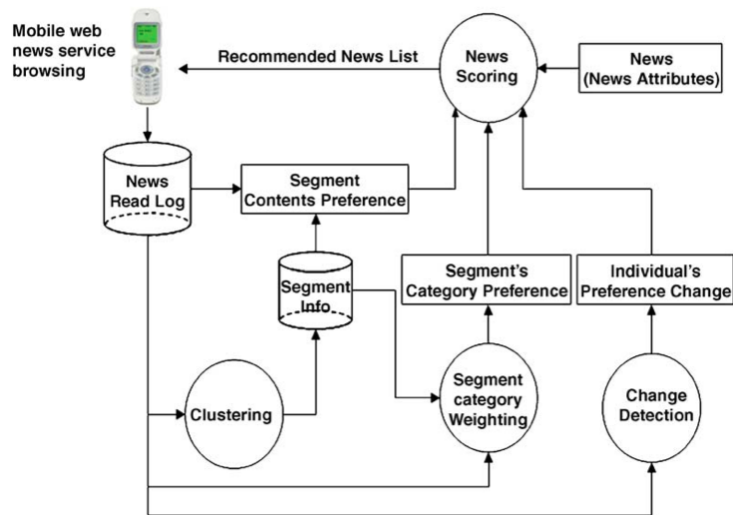


Figure 3.7: The flow chart of the MONERS mobile news recommendation system.

MONERS was tested on real users of a mobile service operator and the result showed that the news accessed by category had more page hits, while the recommended news articles had a higher read ratio than articles sorted by recency or category.

[56] presents an Analytic Hierarchy Process[40] model as a fundamental part of a proactive personalized mobile news recommendation system. The AHP model's function is to rate the relevance of news items and it supports both collaborative filtering and content-based filtering. To estimate the user's level of interest, a Bayesian Network[23] approach is used. BN was

chose because of its good performance, such as low memory usage and fast computations, as well as it helps solving the data sparsity problem. Further to push the proactive information, a hybrid P2P[43] system is applied.

The proposed model is implemented as a mobile application in Java, named PPNews, supplying a just-in-time personalized news service. First the Bayesian Network is constructed, followed by the assignment of the weights and then the rank of each item is computed by applying the Analytic Hierarchy Process model.

The application was tested on 10 research staff members and students, resulting in positive feedback, such as battery life lasting almost a whole day using the application, as well as the news pushed was generally matching the user's profile.

3.3 News consumption

[27] talks about how news consumption are shifting from accessing news from regular news sources on the Internet towards being presented and accessing news articles in a personalized manner via social news streams. A study done by Pew Internet on a sample of 2259 American adults, shows that approximately three fourths of the people asked who access news online, are accessing news via email or posts on social networking sites[37].

National Public Radio² also conducted a survey regarding social news consumption[7]. NPR asked their Facebook fans, which then summed up to about one million people, if people use social networks to access news and information. With over 40 000 answers to the survey, 74.6 per cent agreed that "Facebook is a major way for me to receive news and information from NPR".

Although NPR's survey cannot say if the people who answered they survey is a good representative of the general public, it shows the same trend as the survey conducted by Pew Internet.

3.4 Designing news applications

[51] presents a news service called News Sync. The system was created focusing on three different scenarios of use, either separately or combined; news covering a topic, a location or within a specified time period. News Sync

²National Public Radio is a public radio network serving over 900 public radio stations throughout the US.

proposes a combination of NLP, machine learning, visualization and search to create a more "captivating and sticky news consumption experience".

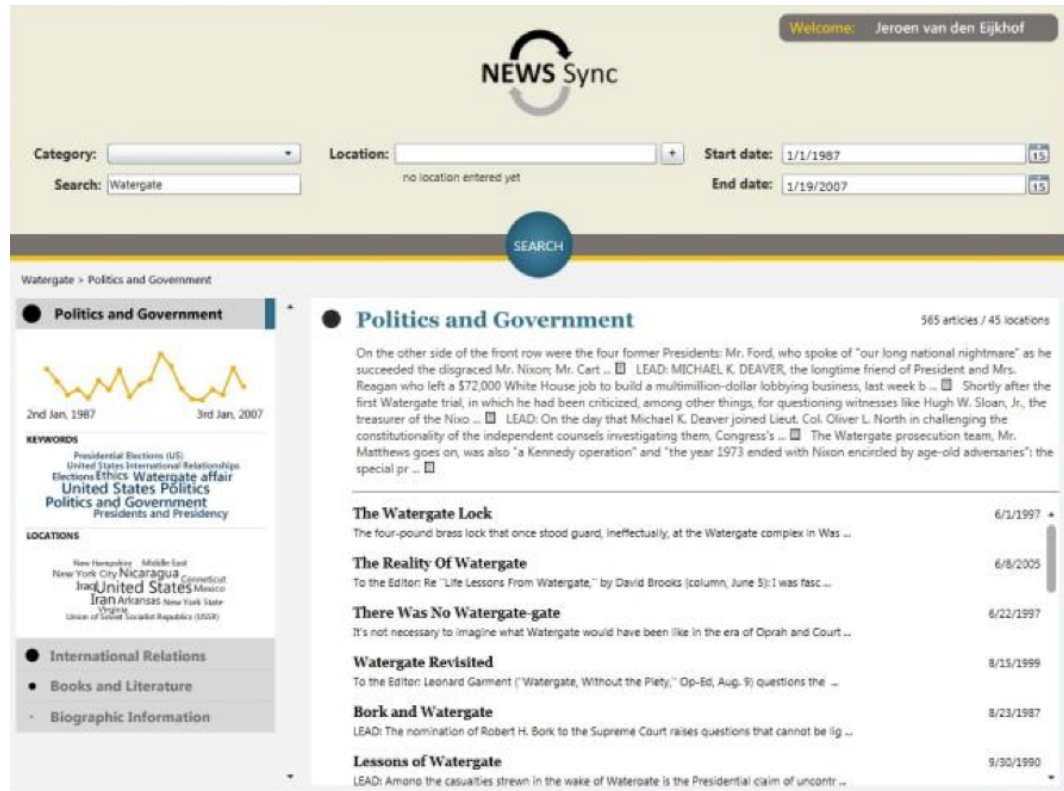


Figure 3.8: Screenshot from News Sync showing the search result for Watergate.

Figure 3.8 shows a screenshot of News Sync after a search for Watergate has been executed. The left-hand side consists of three main areas. A graph showing the distribution of articles containing the search phrase over the time specified, a tag cloud showing the keywords in the articles resulting from this search, and a tag cloud showing the locations concerning these articles. The higher concentration the bolder font, and all words in the tag clouds are clickable, for further browsing within this category or location.

The main view is showing the clustering of news articles from the selected keyword. The user can choose another keyword on the left-hand side, and another cluster of news articles will be shown in the main view containing articles within this cluster.

The user can share articles, summaries or stories to the most popular social networking sites, as well as saving the query for later access. All user activity are tracked and taken into account when building the personal user

profile, where this profile determines how the ranking and summarization of clusters are maintained to fit the user's profile.

[1] presents a proposal on how to design an aggregating news application called 247, on the Apple's iPad device with focus on combining the credibility of the editor, a social media layer and a fluent interaction experience. A main concern is how to move the conventional newspaper onto a digital medium, and how the digital medium can be used in everyday life to access news, either as an individual catching up on the latest news, or as a peripheral device presenting the latest headlines to whoever is looking at the device.

It includes field work on how people are using the iPad as a device, and how, when and where people are accessing news throughout the day, resulting in a design allowing both individual use, and peripheral use (see figure 3.9 and 3.10).



Figure 3.9: Screenshot from 247 showing the individual UI design along with an example of individual usage.

3.5 Commercial mobile news applications

There are a lot of different commercial mobile news applications, and since these are commercial systems, only a limited amount of information about their underlying technology has been published.

Following are a short description of ten of the most popular and well known mobile news applications that are available at this time of writing,



Figure 3.10: Screenshot from 247 showing the peripheral UI design along with an example of peripheral usage.

subjecting both the UI and how the recommendation is solved, if obtainable. These commercial applications are further discussed in chapter 7.

3.5.1 Zite

Zite is an intelligent magazine, which aims to guide the user towards discovering interesting things to read[18].

User Interface Design

The Zite application meets the user with an infinite vertical scrollable view consisting of the top stories. Each news story is framed within a rectangular tile, whereas each tile consists of at least a title and the publisher of the article. The tile can also include tags, lead text, the time since it was published and the article's image (see the first image in figure 3.11).

From the top stories the user has several choices. In the top left corner the user can access his "quicklist" of categories and entities (see the last picture in figure 3.11). In the top right corner the user can search for topics like music or entities like Rolling Stones, for instance. These search results can be added to the "quicklist" to access them later. If an article is tapped, that article is showed in a new view (middle picture in figure 3.11). Here the user can read the Zite's version of the article or open the publisher's version. The user can share the article through numerous social websites and services. The article can be rated with thumbs up or thumbs down buttons. The user also has the ability to change the text size of the article or block the publisher of the article to never get another article from this publisher, from this screen.

Zite has a clean and simple UI design with a minimum of surprises regarding navigation. All is more or less straight forward, with an exception of the swipe left on the main screen to trigger the next element on the "quicklist",

which might come as a surprise to the common user.

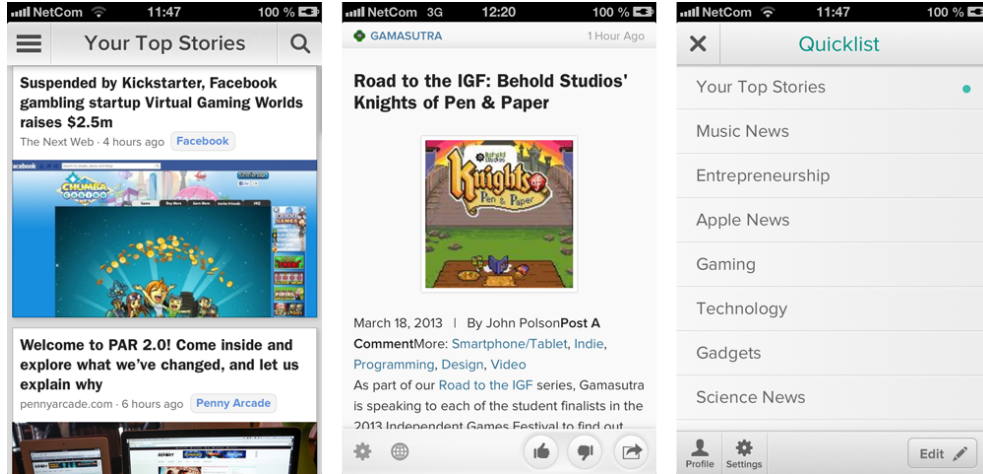


Figure 3.11: Screenshots from Zite showing the Top Stories feed, a single news article, and categories list.

Technology

Zite uses different approaches to personalize the news magazine for the user[19]. It uses automated algorithms to retrieve articles that are trending now by looking at how the different articles are discussed and shared on different websites, blogs, popular social network services, like Twitter and Facebook, and other news applications like Pocket and Google Reader. If an article is considered less trending by the algorithm, it is less likely to appear in the news stream, although the user might consider this article interesting.

The user can also choose to login with one or more of the aforementioned social network services and news applications, and get news recommendations based on what is shared by the user itself and friends of the user, or based on what is stored by the user in the news applications.

Zite offers to save the user profile so the user can access its personalized news feed on different devices, but it does not limit the use of the application if a user chooses not to sign in.

The user can heavily influence what kind of news that are retrieved by rating different articles with the "thumbs up" or "thumbs down" buttons. The more rating the user gives, the better the personalization gets. Zite also keeps track of which articles the user is reading and which articles that are shared by the user and the user's connections on the social networks. This tracking influences the content delivered to the user.

In addition the user can search for different entities of interests, and further choose to like this type of entity by clicking a heart shaped button. If it is liked, this type of news will start to show in the main news stream. The user can also choose to add this entity to the "quicklist" to be able to quickly access only this type of news in the news stream.

A single source can also be blocked completely, as explained in the previous section.

3.5.2 Flipboard

Flipboard is a digital news magazine combining news of all sorts and events and feeds from social networks. Flipboard has gained much attention because of their joyful and easy-to-use flip-design.

User Interface Design

When launched, Flipboard greets the user with a front page consisting of different categories (see left image in figure 3.12). These categories are set by the user by clicking the "Your Flipboard" button shown in the right picture in figure 3.12. This settings screen is triggered by clicking the red ribbon in the top right corner of the start screen. While in the start screen the user can navigate further down by swiping, or clicking one of the categories.

When a category is tapped, news articles from this category can be browsed by swiping up or down. Each category stream uses a full size view for each article, meaning that one article's preview uses the whole mobile screen. Each article's preview shows at least the title of the article, publisher and/or author, and the time since it was published. The preview can also include an image, lead text, and how many retweets³ this article has. The user can also bring up the settings screen from this screen by tapping the magnifying glass in the top right corner.

When the article's preview is tapped, the user is presented with the article screen (see the middle picture in figure 3.12). Here the user can swipe up and down to read the whole story. If the article is retrieved from a social network, the user has the opportunity to interact with the social network the article is gathered from, like retweet and favorite for Twitter or "like" for Facebook. The user also has the ability to share the article via other services or choose to read it later by clicking the "out of the box" arrow on the top.

Flipboard has an easy to use interface design, and has gotten a lot of attention because the app is considered to have a beautiful design[20] and

³A retweet is if someone on Twitter shares a Twitter message created by someone else.

a special flip-like animation that is used when scrolling the pages.[14][21]. Flipboard was also picked as "App of the Year" by Apple in 2010[13].

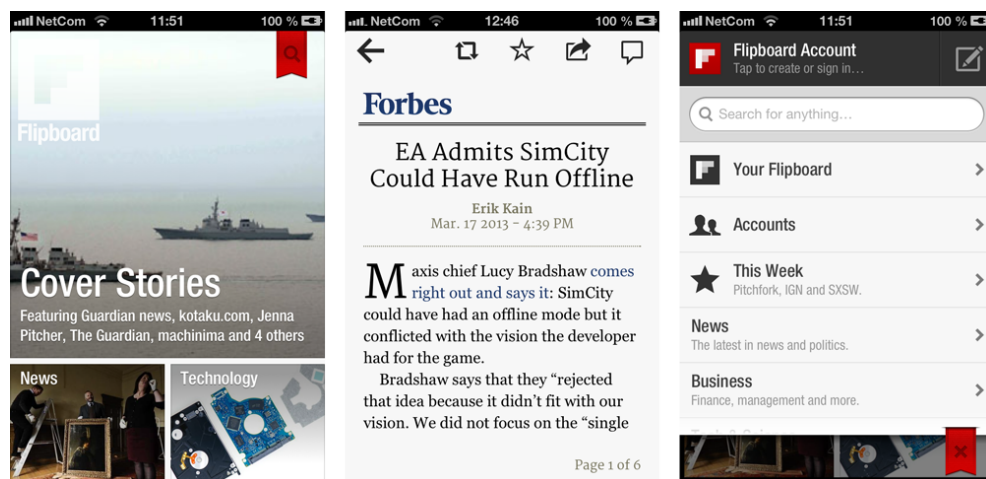


Figure 3.12: Screenshots from Flipboard showing the top categories, a single news article, and the settings view.

Technology

Flipboard allows users to include their own social feeds from Twitter, Facebook, Google+, along with RSS feeds. Flipboard also crawls the Internet for trending news articles and categorizes them accordingly.

Flipboard acquired the Ellerdale startup early on[4]. The Ellerdale project included a semantic data-analysis technology which started as a Twitter trend analysis before it was acquired by Flipboard. Using Twitter's firehose API⁴ the Ellerdale Project was able to process all tweets, thousands per second, and categorize them by topic, rather than keywords[5] being able, in theory, to differentiate between Internet Explorer, Ford Explorer and Dora the Explorer. Further, Flipboard uses this technology to crawl social networks for trending topics and news, and being able to categorize them by topics to present useful and interesting news to the user.

The application offers the user the ability to create a Flipboard account to be able to access the personal feed on several devices. The application is still highly usable without creating an account, but some of the functionality is limited when not logged in. For instance, the user cannot save articles for later reading, like or comment articles without signing in.

⁴The Twitter Firehose API allows third party developers to get access to all tweets that are composed in real-time.

3.5.3 Pulse

Pulse is a news reading application for iOS, Android and web browsers that supports HTML 5. It was released for the first time in 2010 and was awarded with the Apple Design Award of 2011[36].

User Interface Design

When first booted, the user can login, either by a Pulse account or via Facebook. When launched the first time, the user is presented with a limited set of news categories which the user can choose from to add to its interests.

When done selecting category interests, the user is met with the main view, which is shown as the start up screen from now on, as shown in the first image in figure 3.13. This screen shows one category with several publishers sorted in rows. By swiping horizontally in a publishers row, the user can browse the different articles from this publisher. By swiping vertically the user can browse the different publishers in this category. If the user swipes to the bottom of the screen, it can click the "Add Content" button, and add more sources of liking.

The sources can be removed by tapping the "Edit" button in the top right corner, and to change the displaying category, the user can hit the menu button in the top left corner. The user then has the ability to select another category (see the last image in figure 3.13), or add more content to the news feeds.

The main screen has a lot of information showing at the same time and can feel somewhat cluttered and overloaded.

When an article is tapped, the single article screen is presented (see the middle image in figure 3.13) to the user. This screen shows whatever information available from the source feed, like title, author, when the article was published, image and a lead text or article text if available. By hitting the action button in the top right corner, the user has the ability to save the article for later reading or sharing via Facebook, Twitter, LinkedIn, SMS or email.

By swiping horizontally the user can switch between the different articles from the showing publisher, similarly to the main screen. The user also has the ability to show the whole article by either tapping the title or pressing the "Read on Web" button at the bottom of the article. By doing so, the full article is presented in a web view, showing the publishers own website with the given article.

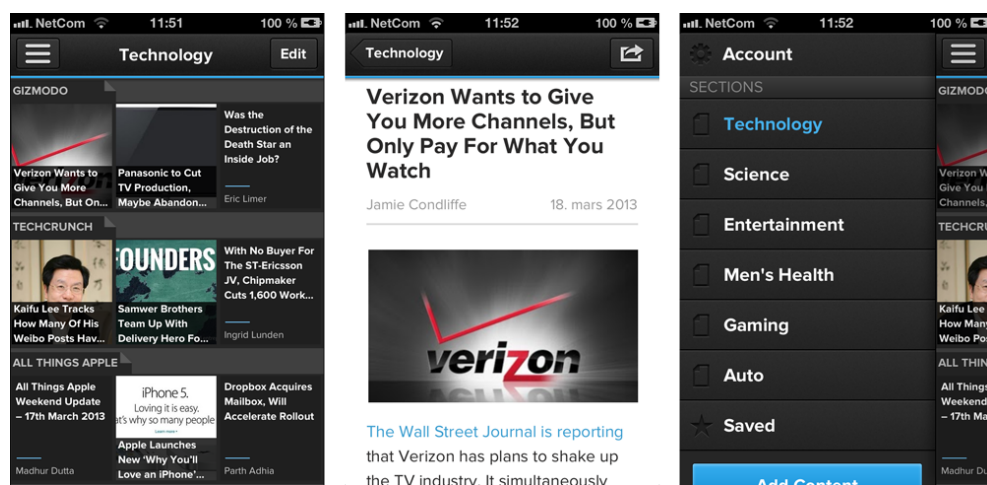


Figure 3.13: Screenshots from Pulse showing the start page, a single news article, and the categories/settings view.

Technology

On first startup, after login, Pulse lets the user choose from a dozen or so different categories to start off the news reading experience. These different categories gets news from different predefined sources added by the developers.

Pulse is mainly based on news publisher's RSS feeds, but allows the user to add their own social feeds from several social networking sites, e.g. Instagram, Facebook and YouTube.

The application also has the ability to search for feeds and add personal RSS feeds to a predefined category or a custom category. This way a user can add the newspaper feeds of their interests. Pulse can also check for the devices location and search for feeds that are nearby by using the GPS coordinates.

If logged in with Facebook, Pulse crawls the users news feed and finds articles previously shared or otherwise interacted with, and shows these article's sources in the recommended section.

3.5.4 Summly

Summly is a gesture oriented news aggregating application which was developed by Nick D'Aloisio in 2011. The main idea was not to personalize the news reading experience, but rather create intelligent summaries of the most trending news, by using advanced text analysis and natural language processing methods[57]. Summly was sold to Yahoo! for reported £18 million

GBP in March 2013[2], and was shortly after removed from the App Store, for Yahoo! to use the summary technology elsewhere[55]. On April 22nd, Yahoo! released their new news application using the NLP algorithms and machine learning that previously was used in Summly[31].

User Interface Design

The front page in the Summly application shows the user how many unread Summlys⁵ the user has. It also shows the title, publisher and category of a trending article as of now. The title of the trending article is also a button, and when clicked, the application opens the single article view showing the clicked article.

The single article view (see the second image in figure 3.14) shows the article's image, title, publisher, time since it was published, number of words in the full Summly and the short Summly. With horizontal swipes the user can navigate between different articles in the current showing category. To read the full Summly the user can double tap the anywhere on the screen. In the full summary the user can choose to close the full Summly by double tapping the screen again or open the article at the publisher's website in a web view by tapping a blue right arrow button.

In the single article view the user can also access the full article at the publisher's site by swiping down. To share or save this article, the user simply holds down one finger and a share view is presented where the user can save the article, or share it on Twitter, Facebook or email.

By swiping up in the single article view the category view (first image in figure 3.14) is shown. Here the user can choose to read from another category or add new categories, entities or topics by clicking the plus button at the bottom of the category screen. The screen where categories are managed are shown in the last image in figure 3.14.

Technology

Summly has the ability to add custom categories, topics and entities as well as giving the user the latest and most trending news within these categories, topics and entities. Summly does not focus on personalization or recommendation, but rather on creating intelligent summaries of news articles, as earlier mentioned.

Summly states when talking about how the text summarization is done: "instead of a robotic, linear algorithm, Summly uses a genetic algorithm to mimic how a human actually thinks, using organic metrics to extract the

⁵A Summly is a summarized version of a website or article.

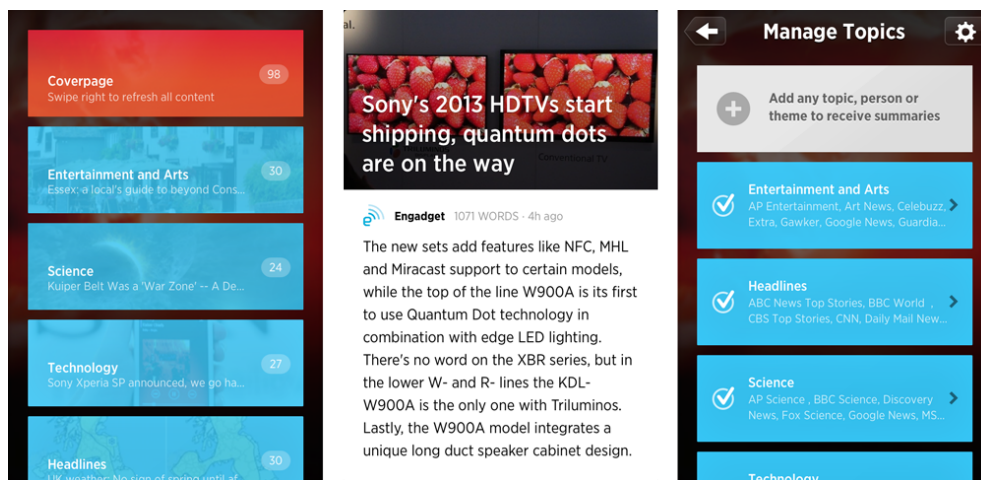


Figure 3.14: Screenshots from Summly showing the category selection page, a single news article, and the category settings view.

most critical components”[45]. Summly uses an advanced algorithm based on machine learning and NLP to create their 400 character long summaries of the news articles. At first the summarization technique was rather basic, but it got more complex when Summly teamed up with SRI International[46].

As a result not focusing on personalization and news recommendation, Summly does not have any form for log in.

3.5.5 News360

News360 is a news recommendation application and according to themselves a ”smart and elegant app that learns what you like and brings you stories from across the web”[34], available for iOS, Android and the Windows 8 series.

User Interface Design

On the first load, the user can choose to log in or continue without an account. Next, the user is prompted with a set of categories to select as interesting, a possibility to search for other topics to add to the feed, and an option to crawl the user’s social news feeds for articles interacted with to add these topics to the personalization feed.

When the personal news feed is finished building, the home screen is presented to the user (see the first image in figure 3.15). This is also the first screen that meets the user when the application is launched after the initial setup is done the first time. News360 shows one news article per screen, and

horizontal swiping is used to navigate to other stories. This screen shows the article image, if any, which category the story resides to, the title, publisher, how long ago since it was published and how many similar or related stories News360 has indexed.

By swiping down on this screen, the user can access the share screen, the "thumbs up" and "thumbs down" buttons to rate the article, and a save button, if the user wants to read the article later. By swiping up the user is presented with the lead text of the article and from here can choose to read the whole article by tapping the article or by hitting the "Continue" button.

When the single news article view is presented, the user has the same action buttons as by swiping down on the previous view, share, rate and save (see the second image in figure 3.15 in the top right section.). In this view the user can choose to read more of the story, view the story from the publishers website, or choose a similar story by another publisher, to get the story from another angle or view. The user can also get more stories from this publisher by clicking the publisher's name. If the publishers name is tapped, the user can navigate to other stories by this publisher, see the publishers profile and choose to subscribe to this publisher.

By tapping the button in the top left corner in the main view (see first image in figure 3.15), the menu view is shown (see the last picture in figure 3.15). From here the user can navigate to the other categories or topics that the user has added. The user can also get news by location, edit existing topics, add new topics, search for news, read saved stories, log in or out, and navigate to the settings section.

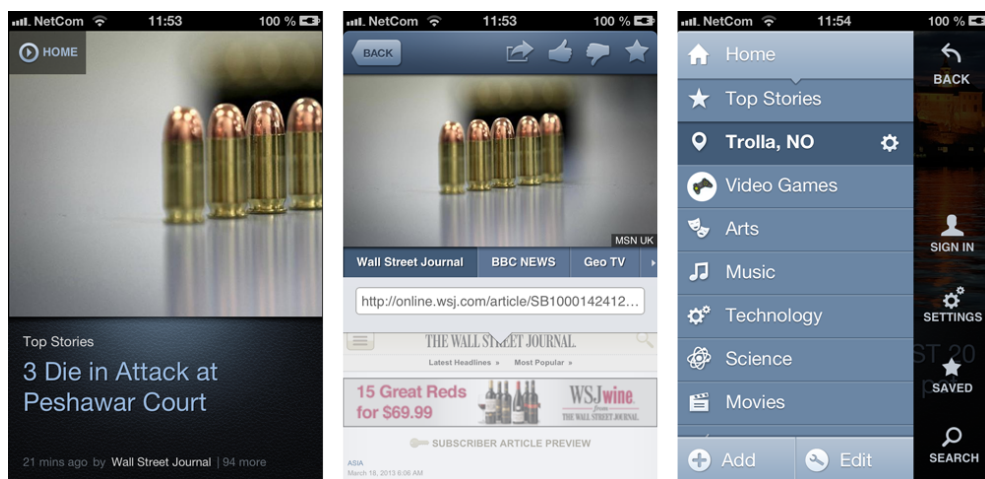


Figure 3.15: Screenshots from News360 showing the top stories feed, a single news article, and the category settings/selection view.

Technology

News360 uses a lot of advanced technology to be able to recommend news to the reader[35]. The semantic analysis platform is created from seven years of natural language analysis and development experience. News360 uses a self made sophisticated linguistic analysis engine to perform tasks like entity extraction, fact extraction, text classification, dossier generation and clustering.

Further this engine is used in combination with a complex news-gathering system to analyze more than 100 000 articles a day in real time. From this analysis approximately 700 000 different people, locations, brands and companies are identified. Then the articles are tagged with locality and topics, and stored in clusters.

To understand which stories that are important, News360 uses a ranking algorithm which checks the impact of the sources that are aggregated, and all the articles that are published in that source. The system also checks the audience and credibility of the source and author, text characteristics and the velocity of the news event as it is happening.

News360 also gives the user the ability to tailor these recommended news by letting the application check the user's social feeds to find articles and sources interacted with. The user can rate each article with "thumbs up" or "thumbs down" buttons to have an impact on which news are shown.

With their semantic analysis, News360 also offers the user the ability to read similar stories, but from different publishers, to have the ability to view the news event from several points of view and angles.

A summary of the work flow, gathered from [35], is shown in figure 3.16.

3.5.6 Circa

Circa is a mobile news application for the iPhone that summarizes news for the user by using real editors on the most trending news and breaking these news stories down to short paragraphs and bullet points[9].

User Interface Design

When first loaded, the user can login, create an account, login with Facebook or choose not to log in at all.

Circa has three main views, the news feed, the single news view, and the category selection view, which is shown in the three images in figure 3.17 respectively.



Figure 3.16: The work flow of the News360 news system.

The news feed is the start screen when the application is launched. The user is presented with images from the most recent and trending news at the top in a horizontal scrollable view, followed by an endless vertical scrollable view with news stories from the category chosen. From here the user can browse through more news stories by swiping up in the vertical scrollable view.

To enter the whole story (second picture in figure 3.17) the news story has to be tapped. The single news view has three buttons in the top right corner. The first, an "i" button, shows which sources Circa's editors has used to create the summaries that are presented to the user. These sources are also clickable, if the user wants to open the whole article in a web view. The second, a share button, lets the user share the whole article or some part of the summary via Facebook, Twitter, mail or SMS. The last, a plus button, lets a user subscribe to the news story, meaning that the user will be notified if there is new information added to the news story, which happens when an editor changes a news article in some way.

To head back to the news feed the user can either tap the back arrow in the top left corner, or swipe right. To bring up the category view (last image in figure 3.17, a user simply taps the button in the top left corner in the news feed view or swipes to the right. Here the user can choose another category by tapping the wanted one and the chosen category will be loaded and shown in the news feed.

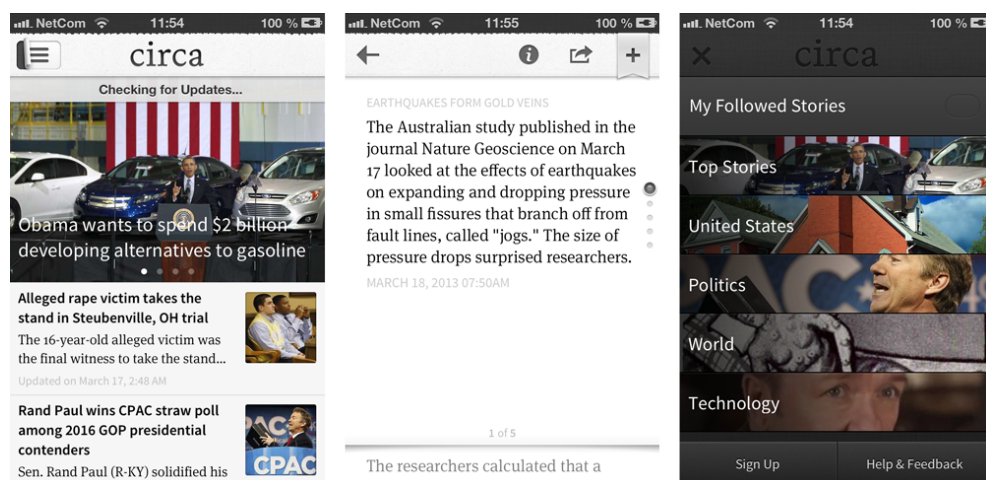


Figure 3.17: Screenshots from Circa showing the top stories feed, a single news article, and the category selection view.

Technology

Similar to Summly, Circa does not focus on the personalization and recommendation part of the news reading experience, but more on the part of bringing the most essential parts of a news article to the user by creating summaries.

While Summly uses NLP and machine learning to do so, Circa has an own editorial staff, working solely with creating summaries for the news reader, in terms of bullet points, facts, quotes, photos, maps and related stories[9].

3.5.7 Wavii

Wavii is something in between a news application and a social network application, and has been called "A Facebook for Topics"[54]. The main idea is that a user can follow topics and interact with these topics, rather than friends on Facebook, for instance. The user gets a news feed of interesting topics, where the sources of these news feed instances can be news sites, Twitter, blogs, etc., and the user can share his/hers reaction to the story, comment and share with friends. In April 2013, Wavii was acquired by Google, and later removed from the app market for their NLP technology to be used in Google's own products[50].

User Interface Design

To be able to start using Wavii, the user has to login, either using Twitter, Facebook or by creating a Wavii account with an email address. After login the user is prompted with a screen to start following topics.

The start screen is a news feed showing followed topics and is an aggregated endless feed of all topics that is followed (see first image in figure 3.18). Each story in the feed is an event and includes one or more topics that is followed. The story can be interacted with in several different ways, like share reaction in terms of different predefined facial expressions shown as simple drawings, commenting, sharing on social networks and poll questions.

The top bar is scrollable in a horizontal direction, and here the user can change the main category of the news feed, in terms of categories extracted from the different topics followed. By hitting the menu button in the top left corner, the navigation view, shown in last image in figure 3.18, is revealed. Here the user can navigate to the different main views of the application, like the main feed, user profile, discover new topics, search for friends, and settings.

When a story in the main feed is tapped, the single story view, shown in the second image in figure 3.18, is shown. In the single story view the news story, twitter feed, or whatever the source of this event is, is shown. The user also has the ability to read a part of the source that caused the event, or open the whole source in a web view. The single story view also shows which stories or events that are concerning the showing event, in terms of clickable boxes, and lets the user follow that story or event by simply clicking it. At the bottom of the single story view, a dozen of related topics and events are shown, these also as boxes, and by clicking these the user can easily start following it.

Technology

To help you find interesting topics, Wavii can crawl the user's social feeds, and propose a set of topics to follow, based on what the user and the friends of the user has interacted with. The user can also search for topics to follow.

To extract topics and events, Wavii uses a proprietary algorithm for NLP[44]. Oren Etzioni, a professor in computer science and a specialist in the field of AI, at the University of Washington, who also is an advisor to Wavii, says that Wavii uses a state-of-the-art extraction algorithm to gather topics and events[54].

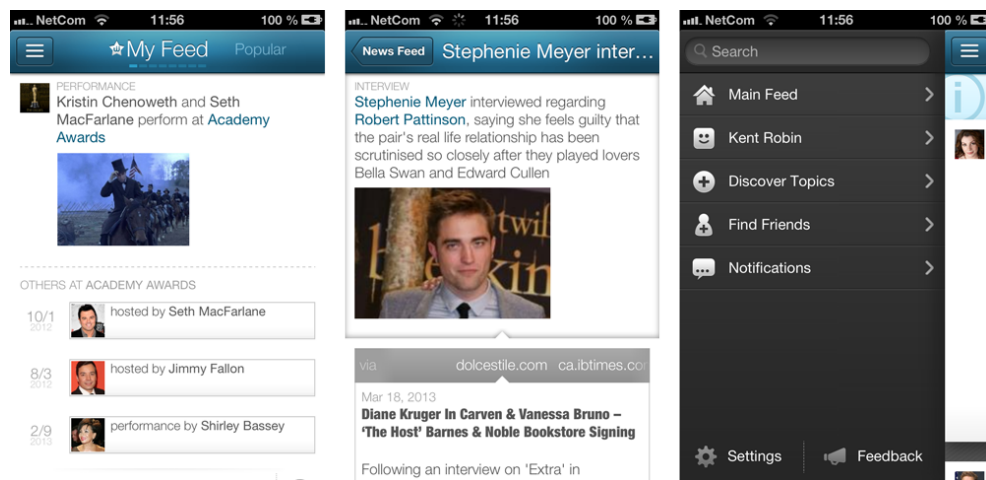


Figure 3.18: Screenshots from Wavii showing the top stories feed, a single news article, and the category selection/settings view.

3.5.8 Prismatic

Prismatic is a news recommendation application that gives the user a personalized news feed based on social network aggregation and applies machine learning algorithms to filter content based on the user's interests[53].

User Interface Design

To get started with Prismatic, the user has to login, either with Facebook, Google+ or Twitter. The user can also register a "Stealth"-account, as Prismatic calls it, if the user do not want to connect the application with their social account.

When logged in for the first time the user starts a tour of the application to add interests and get to know how to use the application. The interests suggestions are based on localization and the user's social network feeds and interactions. The user can also search for topics to be included in the news feed.

The news feed is shown in the first image in figure 3.19, and this is also the main screen of the application. Here the user is greeted with an endless feed of news stories based on the user's interests. Each story can have a title, an image, the source of the story, a lead text and zero or more topics or entities that is connected to the story.

When a story is tapped the story is presented in the single news view, as shown in the second image in figure 3.19. Here the user can read the whole story in a vertical scroll view, get a list of related stories by tapping the text

that states how many related stories there are and open the sources for the story in a web view by tapping the keyword associated with this source.

A press and hold gesture wherever in the application will bring up three action buttons to interact with the selected news story. The three buttons are save the story for later reading, add the story to the favorites list, and share the story. If the share button is clicked, it will bring up a new screen, where the user can share the story by email, Facebook or Twitter.

By swiping left anywhere in the application, the category selection/settings view is brought up, shown in the last image in figure 3.19. Here the user can search for new topics or sources, deselect already chosen topics or sources, access the user's activity in terms of articles that are shared, saved, read and starred, and get suggestions for new stories, topics and sources.

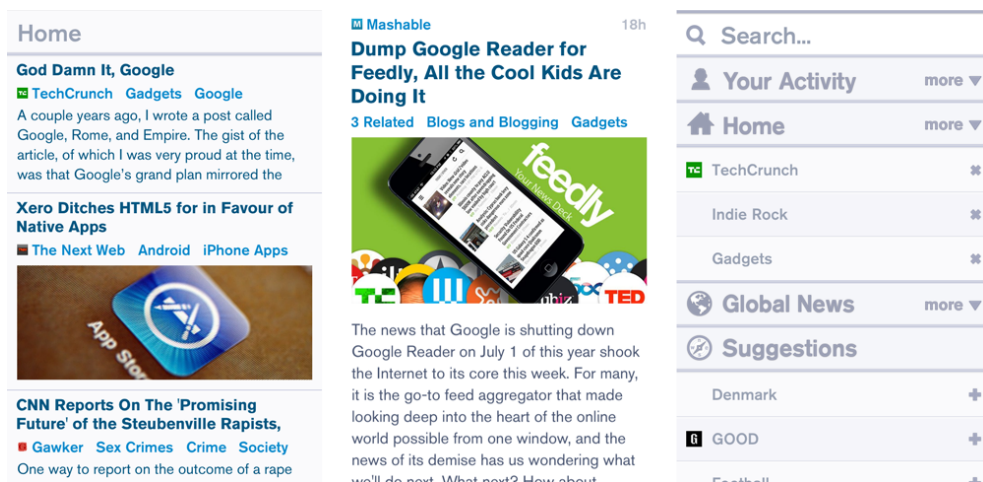


Figure 3.19: Screenshots from Prismatic showing the top stories feed, a single news article, and the category selection/settings view.

Technology

Instead of just crawling the user's social feeds for news that are shared or otherwise interacted with by the user or the user's friends, Prismatic categorizes and extracts keywords and entities from those news and finds news that are related to the news that are shared. Prismatic also tracks all the moves the users does to further build the user's interest profile. Each user profile has a set of stored interests with a weight of how important that interest is, illustrated in figure 3.20. The user profiles are also updated in real time, and the news reading habits of the user will immediately affect the user's profile and which news that are recommended[41].

To gather source for the news articles, Prismatic fetches millions of new news articles, blog posts, as well as other type of shared content, every day. All these sources are analyzed and processed with the use of machine learning and machine language training[41].

3.5.9 Taptu

Taptu is a social news aggregating application with the slogan "DJ your News". This means that the user can create their own stream, combine or split streams or add your social network feeds as own streams inside the application. The application comes with a lot of predefined feeds from a lot of different sources, sorted by categories, but the user can also search for other streams as well. Taptu is available for iOS, Android and web.

User Interface Design

At the initial launch, Taptu offers to connect the application with Facebook, Twitter, LinkedIn or Twitter. Further the user can choose topics of interests to be included in the news feed. This also includes the user's social news feed, if the user chose to connect with one or more of the social networks.

The main view which holds all the different feeds are shown in the first image in figure 3.21. Each feed is horizontal scrollable to browse the different news stories in this feed. All the feeds are put in a vertical scrollable view to be able to access all the different feeds. At the bottom there are four buttons, namely "Streams", "StreamStudio", "Add Streams" and "Settings".

The "Streams"-button shows the main view, just described. The "StreamStudio"-button shows a view where the user can manage the streams, like merge, split, rename, and remove sources from the stream. When the "Add Streams"-button is tapped, the view shown in the last image in figure 3.21 is presented. Here the user can add streams from one of the predefined categories by tapping the desired stream. The user can also search for other streams, or choose to login with Google Reader to include streams the user already has created in Google Reader. The last button shows the settings view where the user can change the layout of the application to the user's liking, in terms of fonts, text size, and background color. The user can also manage the social user profiles and feeds in this section.

In the second image in figure 3.21, the single article view is shown in a vertical scroll view. This view can consist of an image, title, author, and a text. At the bottom of the scroll view, Taptu also lists a couple of related topics which the user can chose to follow by tapping them. At the bottom of the page there are three buttons, the "Globe", "Star", and "Arrow" buttons.

The "Globe" button opens the showing article in web view and shows the original article at the news provider's web page. The "Star" button adds the article to a bookmarks stream for easy access later. The "Arrow" button or "Action" button, as it is often called, brings up a share screen where the user can share the story via numerous different channels, like email, Twitter, Facebook, and LinkedIn.

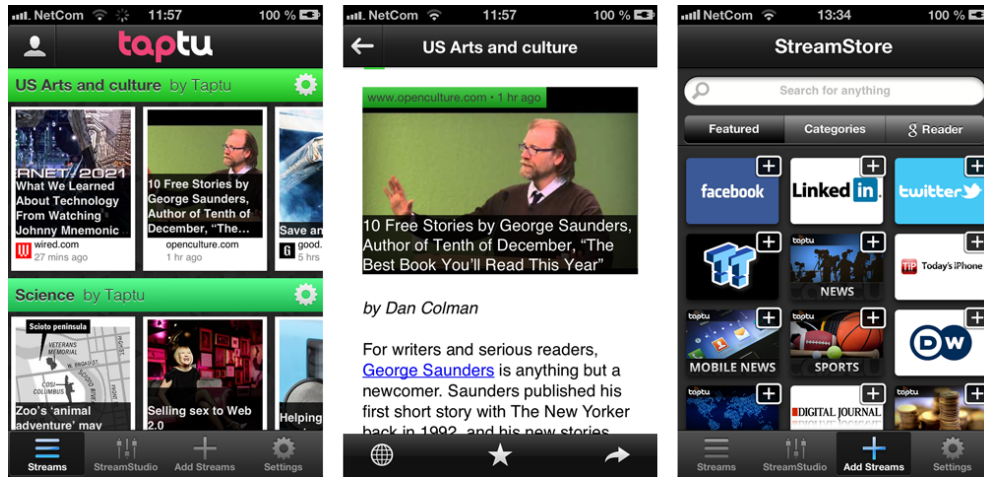


Figure 3.21: Screenshots from Taptu showing the top stories feed, a single news article, and the category settings view.

Technology

Taptu crawls social networks and retrieves the most shared news and posts, and lets users add own streams, in addition to the predefined streams, but Taptu does not have any advanced form of personalized recommendation. Their approach is more towards letting the users themselves choose what kind of sources to be displayed.

It does however lets the user enable an option to the streams that is called "Taptu Magic", which follows the user's reading habits and shift articles more and more towards a subject that is more read than another[47]. For instance, if a user has a sports stream, and the user reads in general more basketball news than baseball news, "Taptu Magic" will start showing more basketball news and less baseball news in this stream.

3.5.10 Feedly

Feedly is a news aggregator application for iOS, Android and web. It is quite similar to Taptu in terms of functionality and purpose. Basically it is an

advanced RSS-reading application.

User Interface Design

On the initial load, Feedly gives the user a quick introduction to the application and points the user in the direction of how to add feeds to the application. The application is highly usable without any login, but if the user wants to add own feeds and store these to later, a sign up with Google is required.

To find new feeds, the user can search by keywords in the search field or by browsing the predefined topics, as shown in the last image in figure 3.22. If the user has signed in, a source can be added to a feed, otherwise the previous feed that was shown will be replaced by the new feed that is selected.

The main screen, shown in the first image in figure 3.22, consists of a news feed contained in a vertical scroll view and a toolbar at the top. The toolbar has four buttons, the menu, title, add, and search button.

The menu button in the top left corner, shows a category selection and settings menu if the user is signed in. If not, the user is prompted to sign in via Google. When signed in the user can navigate to the different feeds, show all feeds as an aggregated feed, manage feeds, add new content, and change Feedly application settings.

The title, in the top middle, also works as a button. When tapped it brings the user to the start of the feed again.

The add button, lets the user add the current displaying feed to an already defined category or to a new one.

The search button is located in the top right corner, and this opens the search view shown in the third image in figure 3.22.

When a single article is tapped in the main view, the single article view, as shown in the second image in figure 3.22, is brought up. The single article view has a toolbar at the top with four buttons, and a vertical scroll view which shows whatever information is available from that feed, like title, number of likes, author, time since it was published, image, video, lead text, full article text. At the bottom of the scroll view there is a button to open the article at the publishers site in a web view.

The four buttons at the top consists of a back button, which takes the user to the previous page, a custom button which is set by the user, a save button and a share button. The custom button can be, for instance, share to Twitter or open in Safari. The save button stores the article for accessing later or via another device. The share button opens a small view with numerous different buttons to share via different services, like Facebook, Google+, Buffer, mail,

and Instapaper.

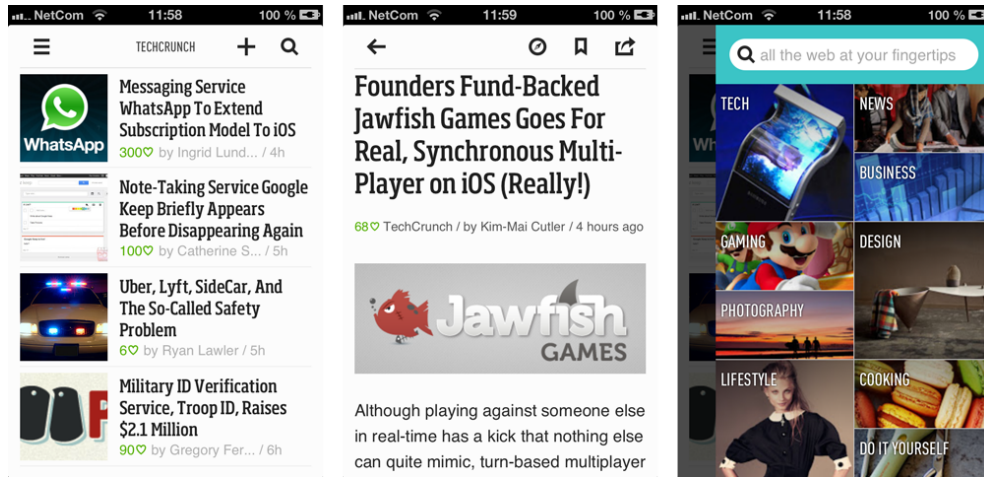


Figure 3.22: Screenshots from Feedly showing the top stories feed, a single news article, and the category settings view.

Technology

As Feedly is more of an advanced RSS-reader, than a news recommendation application, and the focus of the application seems to be on creating a highly functional one, Feedly lets the user be in total control of the content, and does not offer any personalized recommendations based on social networks, or behavior. Feedly includes a search functionality that makes it very easy to add new feeds of choice and manage these feeds as the user see fit.

Chapter 4

The Mobile News Recommender Use Case

The use case following this thesis is the Smartmedia Mobile News Recommender system. The SMNRS is a program at the Department of Computer and Information Science at the Norwegian University of Science and Technology with a close collaboration with the Scandinavian media industry. It was established in 2012, and the program focuses on examining new technologies to help the news industry with the information overload situation and looking for ways to deliver news more efficiently and attractive to their readers.

The technologies essential to the SMNRS project are:

- Big Data architectures
- Information retrieval and recommendation
- Semantics
- Text analytics and sentiment analysis
- Mobile platforms

4.1 Client application

As of now, there is only one client application making use of the SMNRS's back-end, which is an iOS application developed for iPhone running on iOS 6.0 and above. The client is developed as a part of this thesis and as a part of the SMNRS program. The iPhone client application's role in the whole project is shown in figure 4.4 and a video presenting the application can be found at <https://www.youtube.com/watch?v=3HgvnlqZ67A>.

4.1.1 User Interface Design

When the application is first launched, the user is met with a pop-up box where the user can agree or disagree to send the device's geolocation. Further the user is guided through an introduction phase showing how to use the application and which possibilities the app has. When this phase is completed, the user is met with the start screen where every application launch initiated from now on starts.

The start screen, as shown in the foremost image in figure 4.1, shows the title of the latest recommended news article and how long it is since it was published. The title also works as a button and will reveal the corresponding article in the RSS view. In the top left corner it shows how many unread top stories there are, and in the top right corner the user has the ability to search for any topic or category it desires. Swiping to the right and releasing will trigger an update for the top stories, a swipe down will reveal the last read top story article, a swipe up will reveal the settings screen and a swipe to the left will reveal the category selection screen, as shown in the rear image in figure 4.1.

The categories in the category selection screen are categories that are defined by the back-end by analyzing the article's content, as well as the categories set in the RSS feeds provided by the content publishers. The user can also access any stored articles from this menu. By clicking a category in the category selection screen the RSS view will be presented showing the articles from the category clicked. The RSS view is shown in figure 4.2.

At the top of the RSS view the title of the article is shown on top of the article's image, if any. The view also shows the publisher, when it was published, which categories the article resides to, the page number the article has of all the articles retrieved in this category and the lead text. To navigate back and forth between the different articles, a horizontal swipe is used. A double tap anywhere on the screen will trigger an update of the news in this category and bring the user to the foremost article. The share/save menu, as shown in the rear image in figure 4.2, is triggered by holding down one finger anywhere on the screen for a short amount of time. From this menu the user can share the article via Facebook, Twitter or mail, as well as storing the article for later reading, by tapping the star icon. By swiping up in this screen the user will be sent back to either the category selection screen or the start screen, depending on which of the two screens triggered the presentation of the RSS view. By swiping down on the RSS view, the full article view is presented to the user. The full article view is shown in the foremost image in figure 4.3

The full article view are quite similar to the RSS view, but in addition



Figure 4.1: Screenshots from the client application showing the start page and the category selection screen.

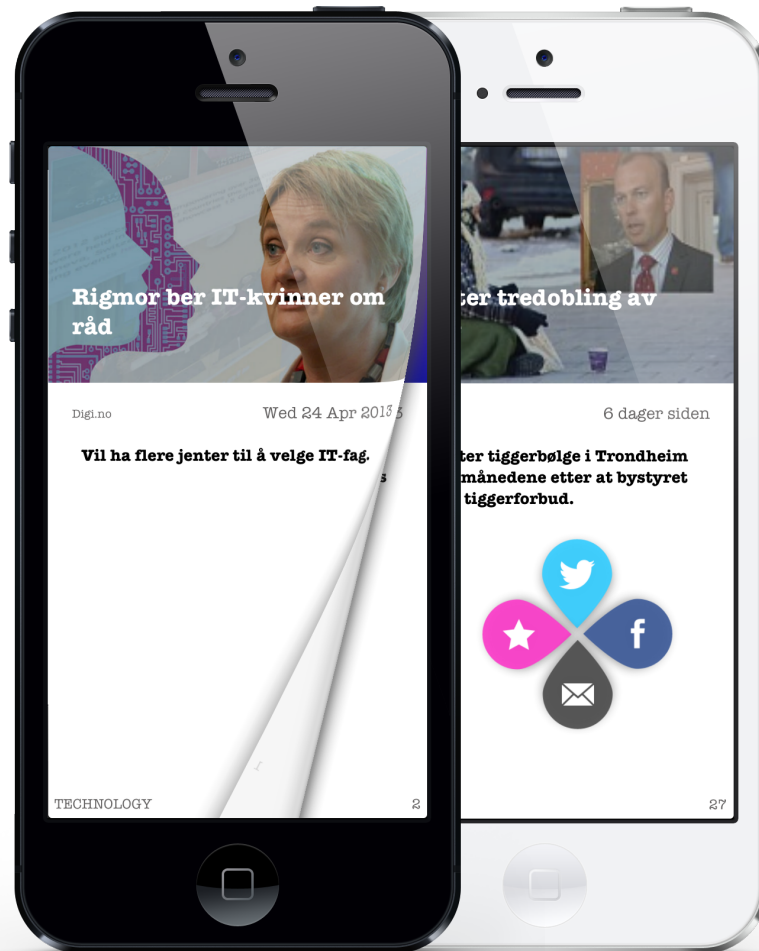


Figure 4.2: Screenshots from the client application showing the RSS view and the RSS view after triggering the share/save menu.

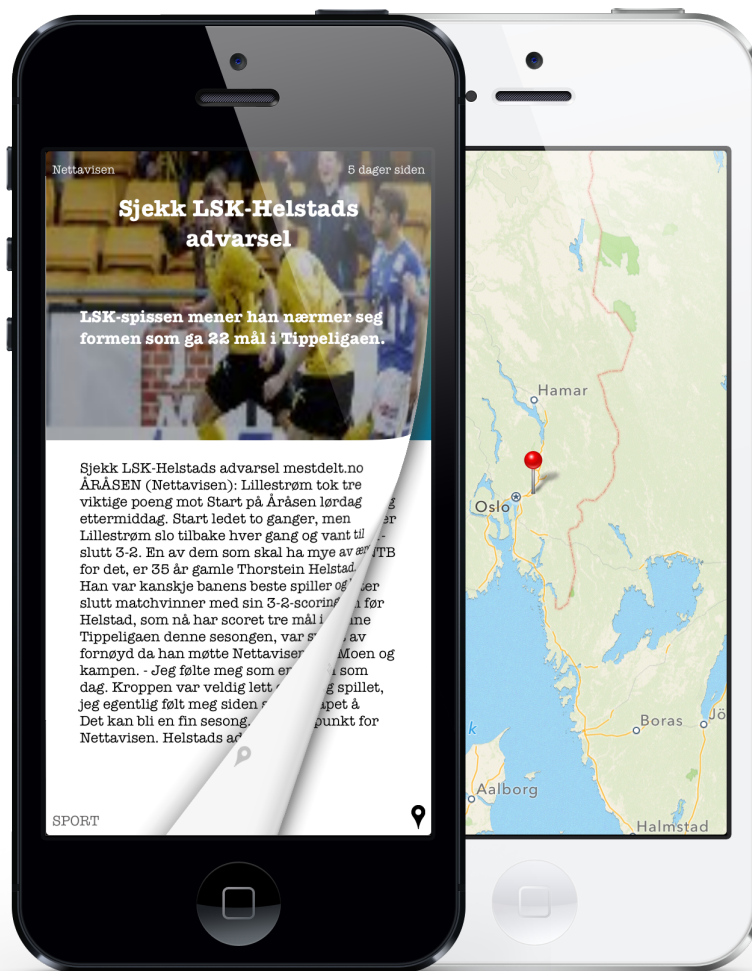


Figure 4.3: Screenshots from the client application showing the full article view and the map view.

to all the information found in the RSS view, the full article view has access to the full article text, related articles and the possibility to show where the article resided on a map. The start of the full article text is shown in the lower part of the screen, and if the user wants to continue reading the the article text, a press and hold with one finger anywhere on the screen will bring up a new view showing only the article text in a full screen format. Here the user can read the whole story without minding any other information besides the article text. To close the article text view the user holds down one finger for a short amount of time again. If the article has any locations connected with it, a map pin is shown in the bottom right corner of the screen. By tapping this map pin or by double tapping the screen, a full screen map view (rear image in figure 4.3) is presented showing on a map, all the locations that are connected with this article. To close the map view, the user simply double taps it again. The back-end system also delivers related articles to any article retrieved, if there exist any. These can be viewed by performing a horizontal swipe, similar to the RSS view. The related articles are shown in the same way as the full article view.

As mentioned earlier, the settings view are presented when performing a swipe up gesture in the start screen. The settings view shows how the user profile is stored in the back-end system, with regard to which criteria that are weighted most when retrieving news. These criteria are freshness and location, for instance. These values are set by the back-end system based on the user activity, and the user has the ability to override these settings and storing them, as well as wiping the user profile clean and start the learning from scratch. The user can also override the GPS location if the user wants news from a different place than where the device is currently residing.

4.1.2 Technology

The SMNRS, as shown in figure 4.4, is quite complex and consists of several advanced technologies and APIs.

Back end

The back end starts by subscribing to different RSS feeds, consisting of different publishers and categories, provided by various news content providers. The preprocessing part consists of six steps:

1. Extract RSS entries
 - The RSS entries are extracted using an RSS parsing library for Java, called ROME.

2. Download HTML contents

- The news pages are scraped using a library called Apache Tika.

3. Extract body text

- Extraction of the body text are done using the same library as for downloading the HTML contents, Apache Tika.

4. Named Entity Recognition

- For the NER, a library called Apache OpenNLP is used, and the model used is especially trained for the Norwegian language.

5. Geo Code tagging

- Geo Code tagging is realized through the Google Location API.

6. Index document

- The documents are indexed in an Apache Solr solution, which supports queries that combine geographical and content-related elements.

Further these news article documents are stored in a search engine/document store service, using Apache Solr¹.

To be able to get recommended news based on user activity and preferences, user profiles and user event logs are stored in a MongoDB². An Apache Hadoop³ batch job is running to constantly process the user events and updating the user profiles in the MongoDB.

A middleware layer is put on top of the Solr and MongoDB systems using a REST API⁴ to make querying news, storing event logs and updating user profiles an easier and more universal task.

¹Apache Solr stems from the Apache Lucene project and provides an open source enterprise search platform. More information at <http://lucene.apache.org/solr/>.

²MongoDB is a type of database for storing structured data. More information at <http://www.mongodb.org/>.

³Apache Hadoop is a framework for distributed processing of large data sets. More information at <http://hadoop.apache.org/>.

⁴A REST API is way of providing an interface to underlying technologies to make it easier to retrieve and store information without worrying about how the underlying technologies communicate.

Client

To retrieve the recommended news, the client queries the REST API for news, providing the back end with a unique user id and geolocation to get the news that best fits the user and the user's location, represented as JSON objects. If the user clicks a category, the category string is added to the query and if the user opens an article in the full article view, the client application queries the REST API for related news providing a news article id, in addition to the user id and geolocation.

The client also tracks the user's activities, like if the user opened the full article view of an article, this counts as a positive event for this category or topic and will affect the user's profile. Which again will have an impact on the news delivered to this user. These events are pushed and stored in the MongoDB via the REST API every time the client makes a news query, and further processed by the Hadoop batch job, which will update the user profile according to the events pushed.

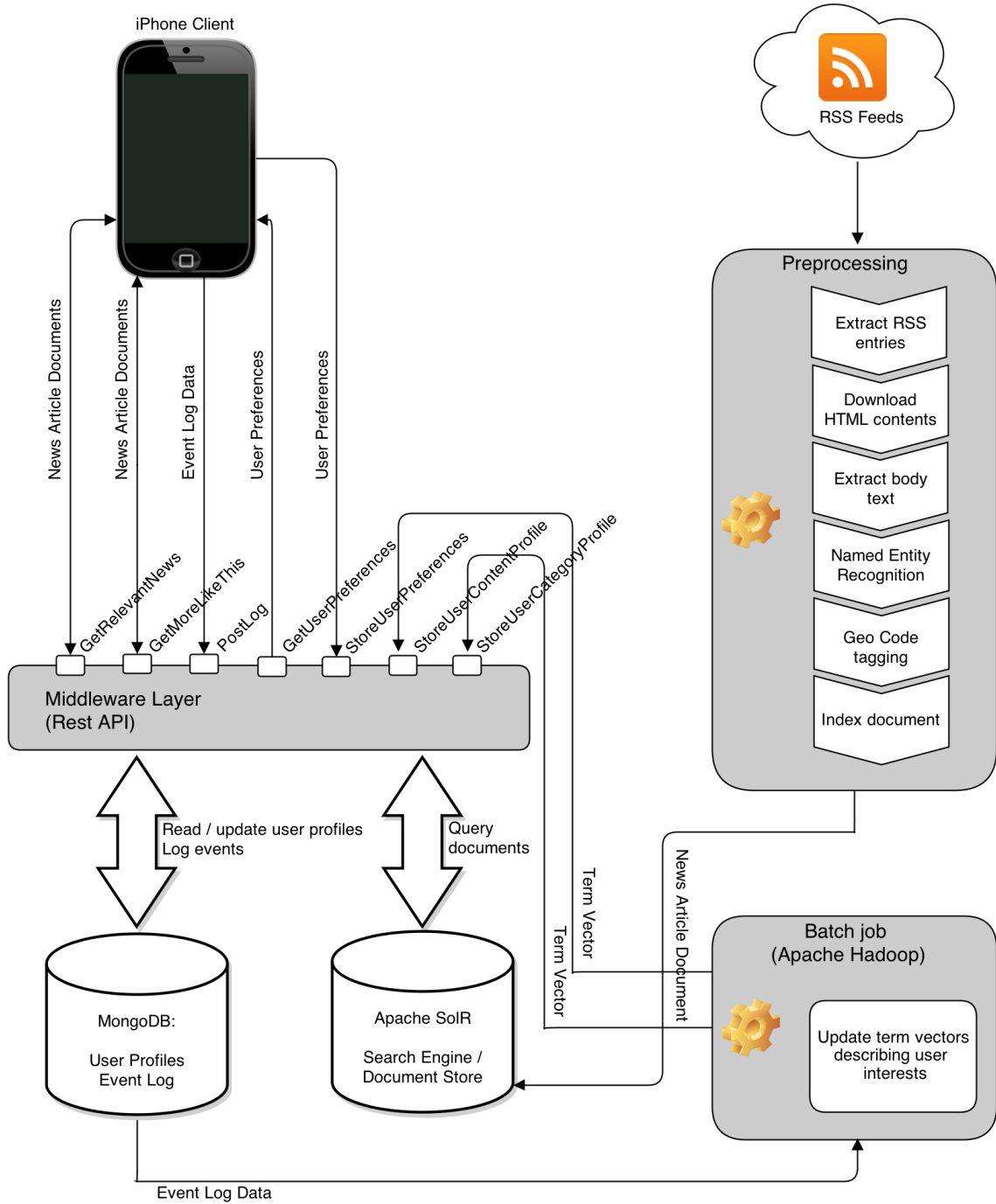


Figure 4.4: Conceptual drawing showing the architectural view of the whole Smartmedia Mobile News Recommender System.

Part III

Mobile News User Interfaces

Chapter 5

Perspectives

The different perspectives that can display a news article will be presented and following in this chapter is a more thorough examination of how the different perspectives are related, how one perspective translates to another, the transition between them, and which perspectives fits which purposes.

5.1 The Different Perspectives

There are a lot of different ways that news can be presented, considering how much information from a news article that are shown, how it is shown graphically, what the information shown means in different contexts etc. Following are description of the perspectives that are most used in different news applications and the applications referred to in this section are examined and presented in section 3.5.

Full Article

A full article perspective, as the name implies, is perspective that normally holds all the information available for that particular news story. The most important information in this perspective is the full article text, as this is the information that most often is lacking in the other perspectives. Figure 5.1 shows an example of a full article perspective.

RSS

An RSS perspective is a perspective showing just some of the information from an article to give the user a quick overview of what the news article concerns. The name stems from the RSS feed technology which has a certain set of information like title, lead-text, when the article was published, and

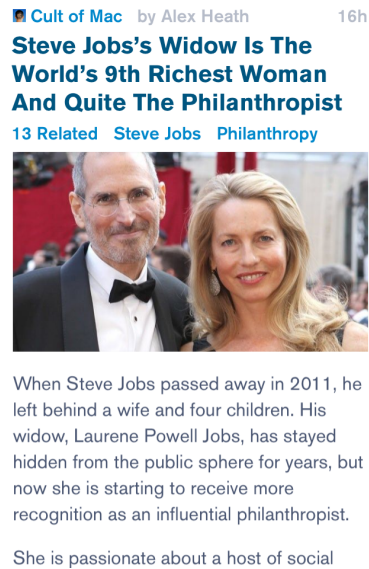


Figure 5.1: A screenshot from Prismatic showing an example of a full article perspective.

sometimes an image. The RSS view may have other information as well, but the ones mentioned are the most common. Figure 5.2 shows an example of how an RSS perspective may be presented.

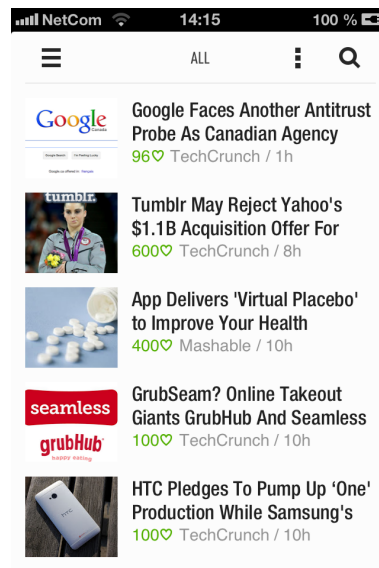


Figure 5.2: A screenshot from Feedly showing an example of an RSS perspective.

Entity

An entity perspective is a perspective showing a set of keywords extracted from a news article by the use of a keyword extracting algorithm or other NLP technologies. Figure 5.3 shows an example of how an entity perspective can be displayed to the user. In this application if a keyword is selected, all the other keywords that are connected to this keyword is highlighted. The articles these keywords are extracted from are shown in a view on the right-hand side as links that will take the user to the corresponding article if clicked.

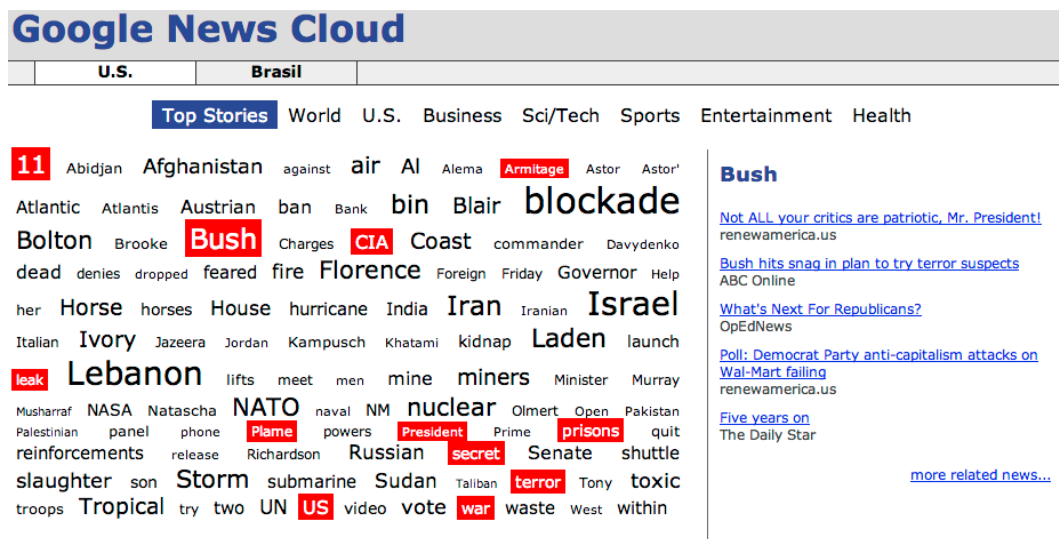


Figure 5.3: A screenshot from NewsCloud showing an example of an entity perspective.

Event

An event perspective is a perspective that displays certain events that occurred by analyzing news articles and extracting those events by using NLP or other text-analyzing techniques. Figure 5.4 shows three events extracted from different news articles. The first shows that two celebrities broke up, the second shows that a theatrical trailer was released by a movie publisher and the third shows that a software company released a new application for the iOS platform.

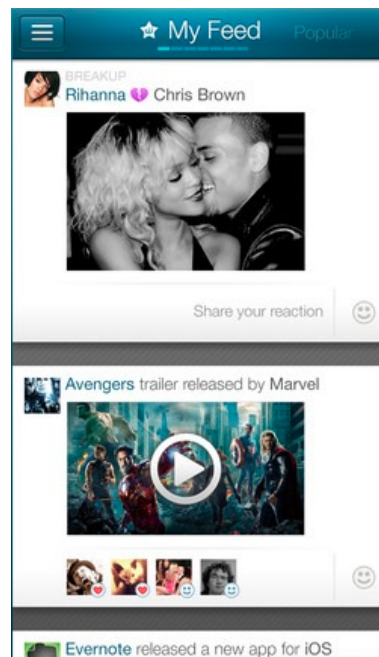


Figure 5.4: A screenshot from Wavii showing an example of an event perspective.

Web

The web perspective is quite similar to the full article perspective and usually includes the same amount of information. The main difference is that the web perspective is the news article shown at the publisher's website through an application by the use of a web browser inside a third party application. This is a widely used perspective as a third party application cannot normally show a full news article from another publisher without an agreement with the publisher itself. Showing a full article without a contract with the publisher is most likely a copyright infringement. Figure 5.5 shows an example of how a web perspective may be presented.

Summary

The summary perspective is a perspective that shows a summary text of the full article text, where this text is created by the use of NLP technologies, like the Summly application, or hand crafted by real editors, like the Circa application. Figure 5.6 shows a screenshot of the Summly application presenting a summary perspective.

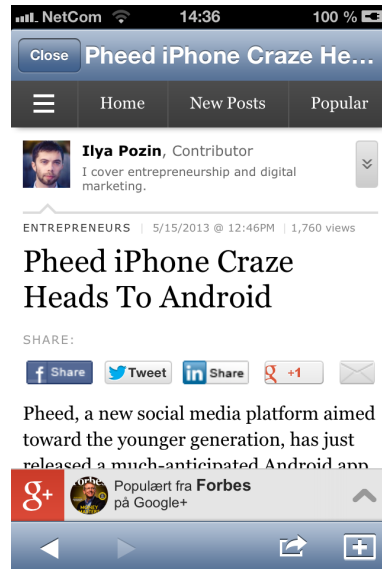


Figure 5.5: A screenshot from Flipboard showing an example of a web perspective.

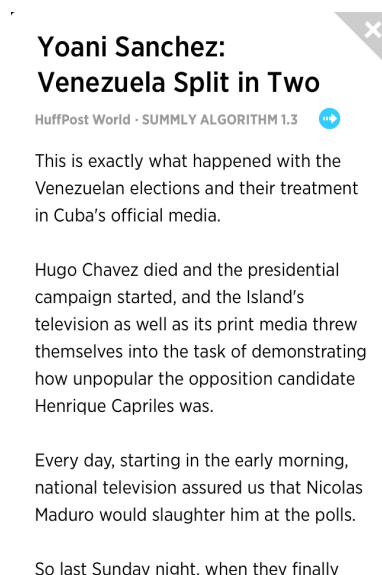


Figure 5.6: A screenshot from Summly showing an example of a summary perspective.

Map

The map perspective is a perspective that shows where in the world a news story concerns by the use of a map view. Figure 5.7 shows a news story residing in Lisbon, Portugal in the Circa application.

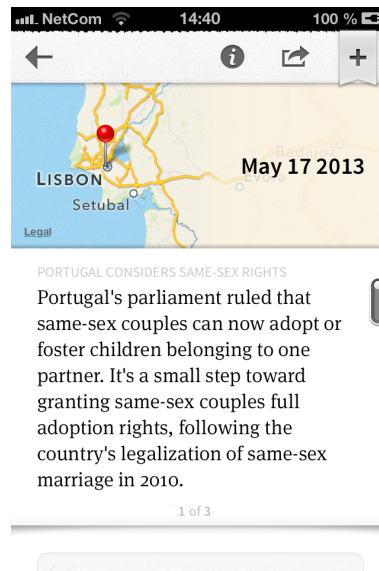


Figure 5.7: A screenshot from Circa showing an example of a map perspective.

5.2 Relations

A perspective can present all the information available from a news article or just a subset of the information, depending on which purpose the perspective has and what type of information the perspective is trying to convey. Following is an explanation of how the different perspectives are related.

Full Article Perspective and Web Perspective

The full article perspective and the web perspective are the two perspectives that holds the most information and are most related. The main difference between the two is that the full article perspectives presents its information in a native way, following the UI design that goes with rest of the application, while the web perspective is presented as a web page in a web browser wrapper inside the native application. The full article perspective is designed by the creator of the native application, while the web perspective usually

is designed by the content publisher itself as a responsive web page¹. If a native app developer wants to show all information available from a news article, it usually has to be done via a web perspective to avoid copyright infringement. The developer can present it as a full article perspective if the app developer has some kind of agreement with content publisher who has the copyright, or the app developer is the content publisher itself.

RSS Perspective

All the other perspectives presents information that can somehow be derived from the full article perspective or the web perspective. The RSS perspective normally holds the information that are available from an RSS feed, hence the name, which is information that usually are available and free to use from the content publishers itself, as long as it links to the web perspective of the same article.

Entity Perspective

The entity perspective holds information in terms of keywords. Normally these keywords are not available from any content supplier, but has been processed from the news article's title, lead text, and full article text, by the use of some sort of keyword extraction algorithm or other NLP technologies. For the best result the event perspective is derived from the full article perspective or web perspective to be able to process all of the information available, but it can also be derived from the RSS perspective, as the title and lead text alone often holds important keywords that are sufficient to understand what the articles comprises.

Event Perspective

Similar to the entity perspective, the event perspective holds information that are derived and processed from the full article perspective or the web perspective. To extract the information shown in the event perspective, advanced NLP technology needs to be applied and a deeper understanding of the article needs to be established. Deriving information from the RSS perspective is most likely not sufficient to overcome this. An event may be extracted from the RSS perspective, but titles and lead texts alone may be misleading or not convey fully what the full story text tries to communicate.

¹A responsive web page is web page that adapts to the screen size and resolution of the device opening the web page.

Summary Perspective

NLP is a fairly important domain when processing news articles and extracting useful information, and the summary perspective is another example of just this. As the name implies, the summary perspective most often shows a summary of the full article text derived from the full article perspective or web perspective. The summaries can be created by the use of NLP, like the Summly application, or by real editors like the Circa application. The summary perspective are not limited to a text summary, it can also hold information from the entity perspective and the event perspective as these also are a form of summaries or compressed information from the full article perspective or web perspective.

Map Perspective

To be able to show on a map where news articles resides, the information from the full article perspective or web perspective needs to be thoroughly processed and analyzed, and again NLP is an important part of this. Places can be extracted from the title, lead text or the full article text, but this information is not sufficient to say where an article's content concerns. There are several challenges to overcome to accomplish an accurate map perspective.

The name of a place without knowing the coordinates of the place, can point to several places throughout the globe. For instance, Heimdal is a place near Trondheim, Norway, but is also a place in North Dakota, USA. To be able to pick the right one, a deeper understanding of the article is necessary.

Also an article can mention several places in the full article text, without all of these places necessarily concerns the article itself. For instance, the article can state "The rock and roll band Aerosmith from Boston, Massachusetts is playing the O2 arena in London next week, with support from the electronica band Röyksopp from Tromsø, Norway". In this example London would be the place the article concerns, and Boston and Tromsø are places that would be misleading to pin to a map with same significance as London.

5.3 Purpose of and Flow Between Perspectives

The different perspectives serves different purposes, and have various areas of usage. Following are a description of the perspectives' main functions and where they convey their information to the fullest, ordered after when they are most likely to appear to the user. The more information the perspective

can hold the farther down in the navigation hierarchy they are likely to be.

RSS Perspective

The RSS perspective often serves as an entry point to a news article and its purpose is to give a quick overview of the content of the article and/or to trigger a curiosity at the user's end to make them wanting to read more. The information in the RSS perspective are often authored by a journalist from the content publisher to achieve exactly this purpose, to drive more traffic to the content publisher's web page, which again will give the content publisher more advertising revenue. With this in mind the RSS perspective is on a thin line between supplying enough information to give the user an understanding of what the topic and main content of the article is, but at the same time create enough curiosity to make the user wanting to read more.

Summary Perspective

The summary perspective is located somewhere in between the RSS perspective and the full article perspective or web perspective in the navigation hierarchy. For the users that want a little bit more information than what the RSS perspective can provide, but less than the full article perspective has to offer, it can serve as a replacement for both. It can work well as an entry point to an article and at the same time provide enough information to the user to make it satisfied with the information it has gained from the summary text.

It can also serve as an additional level in the navigation hierarchy, between the RSS perspective and the full article perspective or web perspective, to give the user smaller steps between the perspectives and the amount of information presented.

Event Perspective

The event perspective shows important events extracted from the news article and is in a way a short summary consisting of the most important happenings in an article. For users browsing through many articles, this is a simple and efficient way to get the most important parts of an article. For these users an event perspective can work as an entry point to an article and replace the normally used RSS perspective. For some users the event perspective can also work as a replacement of the summary perspective, given that the events covers the essential content in the article. Similar to the summary perspective, the event perspective also can serve as an additional level in the navigation hierarchy to supply the user with some extra information in

addition to the RSS perspective before deciding to navigate to the full article perspective.

Entity Perspective

The entity perspective can serve two main purposes. If presented as a supplemental perspective to the full article perspective or web perspective in the same level in the navigation hierarchy, it can serve as a way to find related articles and topics based on a news article. If, for instance, the entities in an entity view are "war, Bush, Al-Qaeda, Middle East, Iraq", these keywords can work as buttons to display a cluster of news about one of those keywords, or they can be added to the user profile as an interesting topic to be included when recommending news articles.

On the other hand, the entity view can serve as a way of getting a quick overview of what the main topics of an article are and then decide to go further into the navigation hierarchy to read more about it. Considering the example given over, a user can easily decide if this is a topic worth reading more about, or skip through. With this approach the entity perspective could replace the RSS perspective, and even the summary or event perspective for some users.

Full Article Perspective

The full article perspective has all the news article's information available, and are usually the last level in the navigation hierarchy. It is the perspective for the most interested users, who wants to have access to all the information an article has to offer, and to be able to read the whole story. This is a perspective primarily used by applications that have legal access to all the content of a story, like the content publishers themselves or third-party developers with special agreements with the content publishers.

Web Perspective

The web perspective serves the same purpose as the full article perspective, to satisfy the most interested users. This perspective are also usually located at the end of the navigation hierarchy, normally linked to from the RSS perspective, or one of the other perspectives that can replace the RSS perspective or work as a middle layer between the RSS perspective and web perspective, like the summary, event, or entity perspective.

Map Perspective

The map perspective has two main purposes. It can work as an additional perspective to the full article perspective or web perspective to show the user on a map where an article resides, to provide the interested user even more information about the content.

It can also work as an entry point, but this can become a foul experience if not done right and then defeat its purpose. If used as an entry point, it would probably not be of much use to show a lot of articles spread out all over the world on a such small device, and have the user click every map pin to see if this could be an interesting story or not. On the other hand if the device can collect the users location and retrieve news that are nearby with coordinates that are accurate to where the news story concerns, the user can probably have a better understanding of the location, since the user is located nearby and probably knows the area better and can easier relate to the location. Then if a map pin is clicked the user can be brought further down in the navigation hierarchy.

5.4 Perspectives and Their Connections in the Different Apps

Following is a simple navigation chart showing how the different perspectives in the different applications are connected.

5.4.1 Zite

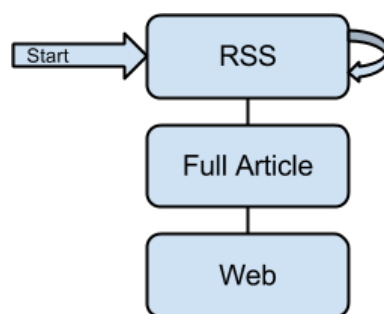


Figure 5.8: A simple navigation chart showing the connection between perspectives in the mobile news application Zite.

Figure 5.8 shows that Zite first presents the user with the RSS perspective showing the top stories based on the user profile. Further the user can

navigate from this perspective to the full article perspective by clicking one of the articles or navigate to a new RSS perspective listing articles from a topic. To move to the topic RSS perspective, the user can either search for a topic, click one the keywords that is connected to the story showing in the top stories RSS perspective, or selecting one of the topics already stored by the user. The RSS perspective can always open a new RSS perspective by clicking one of the topics associated with an article. Whenever a story is clicked in the RSS perspective, the full article perspective is shown, and from the full article perspective the user can navigate to the web perspective

5.4.2 Flipboard

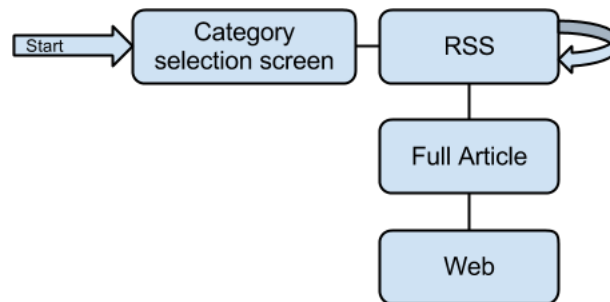


Figure 5.9: A simple navigation chart showing the connection between perspectives in the mobile news application Flipboard.

Figure 5.9 shows that Flipboard greets the user with a category selection screen. From here the user can select a category that is predefined by the user or search for a topic, which in either way brings the user to the RSS perspective. From the RSS perspective the user can again search for a topic which will navigate to new RSS perspective or click an article and be presented with the full article perspective. From the full article perspective, the user can navigate to the web perspective.

5.4.3 Pulse

Figure 5.10 shows that Pulse greets the user with the RSS perspective. This RSS perspective lists a number of RSS feeds within a certain category. The user can navigate from here to another RSS perspective listing RSS feeds from another category or search for other content and open this content in the same RSS perspective. If a story is clicked, the user is presented with another RSS perspective, but for a single article. From this single article

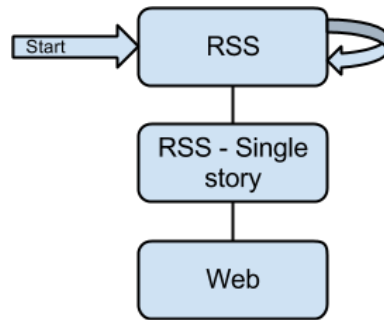


Figure 5.10: A simple navigation chart showing the connection between perspectives in the mobile news application Pulse.

RSS perspective, the user can navigate to the web perspective to show the whole article at the content publishers web site.

5.4.4 Summly

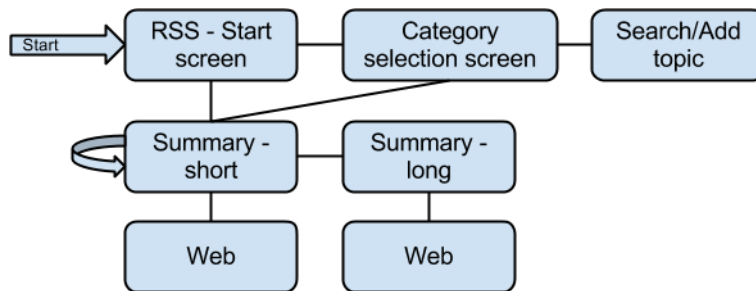


Figure 5.11: A simple navigation chart showing the connection between perspectives in the mobile news application Summly.

Summly starts off by presenting the user with an RSS perspective holding one trending news article, as shown in figure 5.11. From here the user can open this news article in a short summary perspective or navigate to the category selection screen. In the category selection screen the user can search for and add new topics to the category list or click a category or topic to present the short summary perspective. Once in the short summary perspective, the user can navigate between different articles in this category by performing a horizontal swipe, open a long summary perspective by double tapping the article or open the web perspective showing the whole article at the content publishers own web site. The user can also navigate to the web perspective from the long summary perspective, but not navigate between articles from here.

5.4.5 News360

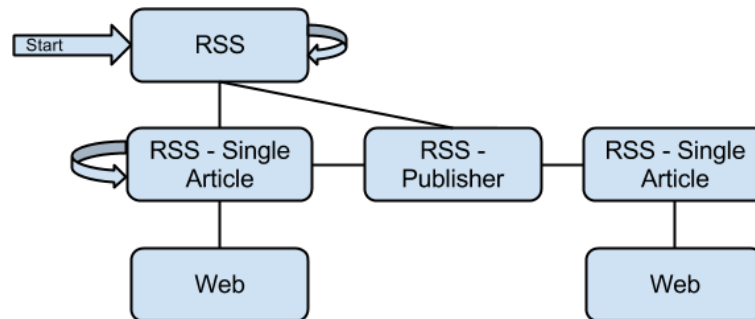


Figure 5.12: A simple navigation chart showing the connection between perspectives in the mobile news application News360.

News360 starts by presenting the user with an RSS perspective containing the top stories based on the user profile, as shown in figure 5.12. In this perspective, the user can navigate between the stories in this category or topic by performing a horizontal swipe. From here the user can also reload the RSS perspective by selecting a different category or by searching for another topic or category. The user can navigate from this perspective to an RSS perspective showing a single article or to an RSS perspective showing a list of articles published by a single publisher by clicking on the publisher name as well. The publisher RSS perspective can open single articles from this publisher in an RSS perspective, and further open a web perspective showing the same article.

The single article RSS perspective on the left-hand side in figure 5.12, shows the article clicked in the first RSS perspective. In this perspective the user can access the web perspective showing this article at the content publishers website. The user can also choose to open other similar articles covering the same story in this perspective. The RSS perspective holding a list of a publisher's stories, can also be accessed from this perspective.

5.4.6 Circa

As shown in figure 5.13, the user is presented with an RSS perspective showing the top stories. The user can reload this perspective by selecting another category to show articles from. The user can also access an RSS perspective holding stories that are followed by the user from the initial RSS perspective. Further both the latter RSS perspectives links to a summary perspective. From this perspective the user can navigate to the web perspective to

5.4. PERSPECTIVES AND THEIR CONNECTIONS IN THE DIFFERENT APPS⁷³

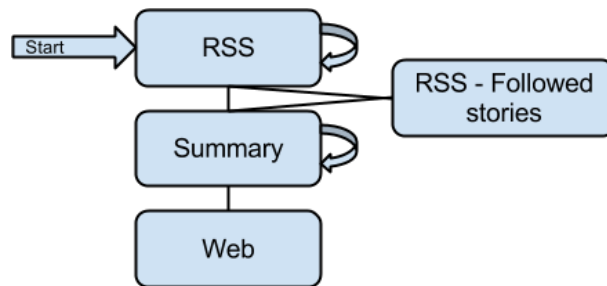


Figure 5.13: A simple navigation chart showing the connection between perspectives in the mobile news application Circa.

show the articles that the summaries are based upon, or the user can navigate to summaries of other related articles. The summary perspective can also contain a map perspective, but the map perspective does not have any interactions connected to it, it only shows where the articles concerns.

5.4.7 Wavii

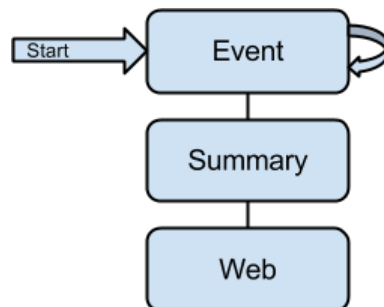


Figure 5.14: A simple navigation chart showing the connection between perspectives in the mobile news application Wavii.

Wavii presents the user with a event perspective holding a list of different events within a category or topic, as shown in figure 5.14². The user can reload the event perspective with other topics or categories the user has chosen to follow. If an event is clicked, Wavii shows a summary of the stories that caused the event in a summary perspective. Further the user can choose to open the stories that caused the event in a web perspective.

²The navigation chart for Wavii might not be completely accurate as the chart was based on previous notes because of the application's removal from the App Store

5.4.8 Prismatic

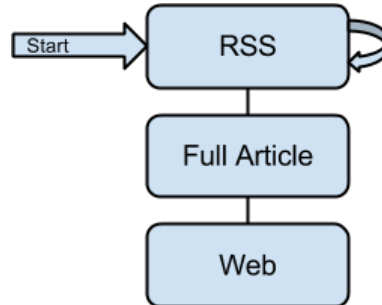


Figure 5.15: A simple navigation chart showing the connection between perspectives in the mobile news application Prismatic.

Prismatic greets the user with an RSS perspective holding news that are recommended for the user based on its profile, as shown in figure 5.15. From here the user can reload the initial RSS perspective by clicking one of the topics that are connected to an article, clicking topics that are suggested by Prismatic based on the user profile or searching for other topics. When an article is clicked it is shown in a full article perspective. From here the user can load a new RSS perspective showing related stories, stories based on topics connected to a story, or stories from a single content publisher. The story itself also contains a lot of highlighted keywords that, when pressed, opens a web perspective showing content regarding this keyword. This content can be another news story, or a page explaining a certain term etc.

5.4.9 Taptu

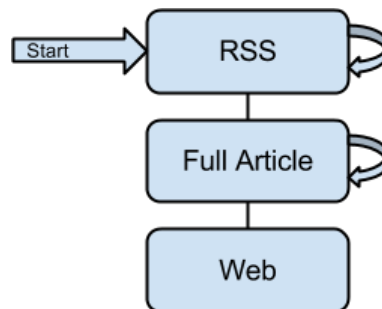


Figure 5.16: A simple navigation chart showing the connection between perspectives in the mobile news application Taptu.

5.4. PERSPECTIVES AND THEIR CONNECTIONS IN THE DIFFERENT APPS75

As shown in figure 5.16, Taptu greets the user with an RSS perspective that holds several lists with different categories and feeds. These lists can be modified in several ways, like merged and split, and new feeds can be added by searching or choosing some predefined feeds. When an article is clicked, the full article perspective is shown containing that article. The user can navigate between articles in this list by performing a horizontal swipe, and also open the showing article in a web perspective. The article may also contain references to other content, which is opened in a web perspective if clicked.

5.4.10 Feedly

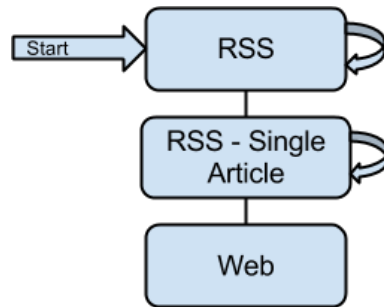


Figure 5.17: A simple navigation chart showing the connection between perspectives in the mobile news application Feedly.

Feedly starts off by presenting the user with a RSS perspective, as shown in figure 5.17, containing a news list from all of the topics or feeds the user has stored. This RSS perspective can be reloaded, either by searching for a topic or by selecting a category or topic that the user has already stored. If an article is clicked, another RSS perspective is shown, but holding the article that was clicked. The user can navigate between the articles in the list by swiping horizontally in the single article RSS perspective and also open the article in a web perspective containing the full article at the content publisher's web site.

5.4.11 Use Case

The use case application starts off by presenting the user with an RSS perspective, as shown in figure 5.18, consisting of the latest top story from the recommendation based on the user profile. From here the user can navigate to the category selection screen, and from the start screen or the category

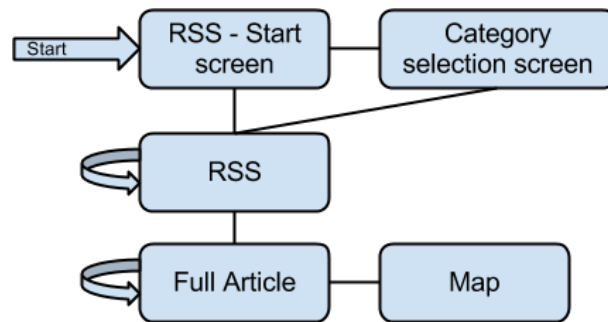


Figure 5.18: A simple navigation chart showing the connection between perspectives in the use case mobile news application.

selection screen, the user can navigate to the RSS perspective holding articles from the top stories, the category clicked in the category selection screen or by searching for a topic in the start screen. This RSS perspective shows only one article at a time, but by performing a horizontal swipe the user can navigate between the different articles in this category. From this RSS perspective the user can navigate to the full article perspective, which also holds only one article at a time, but to access related stories the user can swipe horizontally. The map perspective, which shows where the article concerns are also accessible from the full article perspective.

5.5 How the Perspectives Are Supported on the Mobile Platform

Following is a description of how the different perspectives are supported on the mobile platform with regard to which technologies and opportunities they make use of, and how they are realized. This is, as mentioned earlier, from a iOS point of view, but applies to most other smart phones as well, as these technologies or opportunities are widely used on several mobile platforms.

To represent data, the data must be gathered and parsed in some way. News publishers most often gather and create their own data through the work of journalists. Third-party developers creating news applications often start by gathering different RSS feeds and process these further. Either way, mobile news applications need to get their data from somewhere, parse this data and represent it in some way. Raw data transferred over the Internet often is represented by XML or JSON, and parsing this data is the first step for being able to represent any of the perspectives. Some of the mobile operating systems has native support for parsing XML or JSON, and if not, there

5.5. HOW THE PERSPECTIVES ARE SUPPORTED ON THE MOBILE PLATFORM77

are numerous external libraries that support this that are easy accessible and free to use.

Further when the data is parsed, a way of representing the data has to be found. As all of the major mobile operating systems has native support for displaying text, images, sound and video, there just a matter of how to display it in a usable manner.

All perspectives listed in section 5.1, except the web perspective and map perspective, normally uses parsed raw data, and can be represented as the developer wishes. Also these perspectives normally does not represent the data in any other form than text, image, video or sound, which are all natively supported.

The map perspective uses a map to pin the article's location on. It may also show some sort of text, like the title, connected to the map pin. Opening a map view inside an application is also natively supported on all the major mobile operating systems, making this, if not a trivial task, a fairly easy task.

The web perspective however, is not a perspective the developer can control much over. Opening a web page inside a mobile application is supported by the operating systems, but the layout of the content and how it is represented is controlled by the content publishers managing the web page. The native application can control the zoom level of the content and it usually has to support some sort of navigation controls inside the web perspective. This because, the web perspective acts much like a regular web page, and all links on the web page are initially clickable, which makes it possible for the user to navigate around on the web page inside of the application. If, for instance, the web perspective is showing an article, and this article links to another related article inside the web perspective, the user can click this link and view the related article. Hence, the web perspective should have some navigation support like moving backwards and forward, similar to a regular web browser. In many cases it is possible to override these settings by turning off the ability to click links, but this may cause a confusing and frustrating element at the user's end if the user cannot click to view a related article and the web page does not act like a normal web page to which the user has grown custom to.

Chapter 6

Mobile Features

Having a news application in mind, following are a description the most important aspects when creating a mobile application and how it is different from traditional desktop applications, considering the various features and abilities that are involved.

6.1 Display

The mobile displays are smaller than regular PC screens and the mobile user interfaces needs another approach than the PC UIs when designing them. News read on a desktop computer is mainly accessed through a web browser by visiting the content publishers' own websites or by other news aggregating services available on the Internet. Most news related native desktop applications available today are just RSS readers, which links to a web page, and the news article is still accessed via a web browser. This means that the designing of the news providing application is OS independent. Also most PC screens are large enough to display a web page with its content, not having to worry about the different screen sizes and adapting the content to every possible size, as when designing for mobile devices.

As of now, the iPhone only has two different screen sizes, where only the length of the screen differ, making the design process of adapting the application to both devices less comprehensive compared to other mobile devices. For instance, applications developed for mobile phones running on Android or Windows Phone, has a lot of different resolutions and screens sizes to take into account when designing the UI for the application.

Given that all information that should be displayed in a news web page fits inside a PC screen, it will most likely not fit inside a mobile screen. The design approach on a mobile UI will probably result in dividing the informa-

tion up in smaller pieces and views, and some navigation logic between them has to be added.

6.2 Interaction

The interaction pattern for PCs and smart mobile devices are quite different. PCs are usually controlled with a keyboard and a mouse, while smart phones are normally controlled through a touch based screen.

On systems where a mouse is used, which accounts for most computers, the possibility of bringing up information when the mouse hovers over an element is widely used, but on mobile user interfaces with touch screens, the hover possibility is not an option, ergo another way of showing the same information has to be created.

On the other hand a touch device has a lot of possibilities that are not available or not as intuitive on a standard desktop computer. With a touch device, a user interacts directly with the element on a screen, compared to using a mouse. Gestures like dragging, pinching, swiping etc. can make a mobile user interface intuitive and easy to use, and more similar to gestures people do with actual objects. For instance, a swipe to the side to flip a page, similar to flipping a page in a real newspaper, is a more authentic experience to the real life than moving the mouse cursor to a page, clicking the mouse button and holding it down, and then drag it to flip the page.

Also a lot of mobile touch devices can register a lot of different event triggers, like touches, gestures, and movements, to mention a few. It is possible to differentiate between a touch with two fingers versus a touch with four fingers, a swipe with one finger versus a swipe with three fingers, and these gestures can also be combined. For instance, a user can touch and hold a box on the screen with two fingers for one second to trigger an edit mode, and further move this box around by starting to swipe two fingers while still holding on to the box, to position it wherever the user prefers, and then place the box by removing the fingers from the screen. Another possibility is to use a gyroscope, to have movement trigger some event in the mobile user interface. If a mobile phone is placed on a table facing up, and the phone starts ringing, turning the phone face down can trigger the silent function, for instance.

6.3 Mobility

An essential difference between PCs and mobile devices is the mobile device's mobility. A mobile smart phone is carried around with the user all day making it an easier accessible device, which again most likely makes the threshold for starting a device session lower. Also the mobile device has become more personal than the personal computer, and it is probably safe to say that a mobile phone can be connected with one single user, while a PC more frequently are used by several users.

6.3.1 User Location

Most mobile devices has one or more ways of collecting the device's location, either by a GPS, using the WiFi the device is connected to or using the base station the mobile network is connected to. When obtained, the device's location can be used to provide the user with news that resided nearby, either fresh news or historical news about events that happened a while back.

6.3.2 Connectivity

Mobile devices has the ability to stay connected to the Internet almost everywhere and all the time, making it quite effortless to check the latest news, and using the mobile device as a pastime activity while waiting for someone or riding the bus. In this manner, people are shifting towards a longer and more frequent mobile news consumption trend[33].

6.3.3 Scanning

Many mobile device has the ability to scan QR codes, RFID chips or other types of NFC units. If, for instance, a user walks by a poster advertising for a concert that is about to happen, and the poster has a QR code containing the information of the name of the artist, where it is playing etc. the mobile user can scan this code with its mobile device, and by some processing on the back-end side, bring up news about this happening, if there is any.

6.4 Software

Most mobile devices can only run one application at a time, making the possibility to integrate features from other applications seamlessly an important aspect when designing and developing a mobile application. For instance,

when using a mobile news application many users may feel the need to share a story they find interesting on a social network. This should be doable without costing the user too much effort, and without having the user to lose its news reading focus by having to perform a lot of operations to get it done. By integrating this possibility in an effortless way, sharing a news story to social network can be done in matter of a few clicks.

All the most popular mobile operating systems, like iOS, Android and Windows Phone, allows for an easy integration between applications, to make sharing content to social networks an effortless task. In the mobile news application developed as part of this thesis an article can be shared to Facebook, Twitter or via mail by simply holding down one finger on the RSS perspective that are showing the news article to bring up the sharing control, and then choosing the desired sharing channel. Then an OS integrated sharing view is brought up and the user can share the story right from the news application without having to switch to another application.

This approach extends much further than just sharing content on social sites. If the external service the application wants to make use of, is not incorporated in the OS, the external service most likely has an SDK that can be included in the application to make use of the service directly from the application. This be showing a place on a map, sharing content, saving stories for reading later or watching movie clips.

6.5 Monetizing

As the trend of consuming news content is shifting from purchasing and reading real newspapers to accessing free news content on the Internet, news publishers has to use other approaches to earn money. There are several monetizing approaches to earn money through a news mobile application, for instance advertising, paid subscription, in-app purchases or paid applications.

To buy applications or performing other types of purchases on the different mobile application stores, the user usually has to store its credit card information on beforehand, making later purchases a quick and easy process. This can make the threshold for paying for content lower, and heighten the income.

6.5.1 Advertising

Advertising is a widely used approach on news web sites to earn money. The content providers can sell advertising space on their web sites directly to companies that want to advertise, or they can use a middle layer advertising

service that provides the content publishers with ads, and in this way not having to get advertisers themselves.

A problem with using ads on web sites on desktop computers is that there are several plug-ins for the various web browsers that can remove ad banners for the user. On the mobile platform this is not a big problem yet, as the browsers used on mobile phones are not as configurable as browsers on desktop computers.

There are a lot, probably several hundred, of different mobile advertising services, which offers different types of services and integrates nicely with a mobile application, the main issue is just to find the right one.

6.5.2 Paid Subscription

Paid subscription is a way for the content publishers to give the paying customer more content than what is available to all the other non-paying users, and it also creates a form of a lock-in, as the subscriptions normally renews themselves and is automatically deducted from the users account. The news applications that offers a subscription are often just a digitalized version of the paper version of the same newspaper, and are available through the different mobile applications stores. These applications are also mainly provided by the news publishers themselves and not by third parties. The subscription based applications are also often a special type of application, available mainly for newspapers and magazines to digitally distribute their content.

There are also other ways of offering paid subscriptions, through web applications for instance, where a subscription can be bought through other payment services than the ones offered by the mobile application stores. This is an approach preferred by many content providers as the application stores often takes quite a large percentage of the revenue made through the app stores.

6.5.3 In-app Purchases

In-app purchases is a way of selling content or services inside a mobile application. As the credit card information normally already are stored, this is an easy way for the user to purchase content and services. A common strategy is to offer a free mobile application to make the threshold of downloading it lower, and further include in-app purchases to gain revenue inside the application. It is also a pretty trivial process for the developers and to include and execute an in-app purchase inside the application. An in-app purchase can, for instance, remove advertisement inside the application for a certain

amount of money, for a certain amount of time, or reveal premium content not available to non-paying customers.

6.5.4 Paid Applications

Applications can also be sold on the application stores for a certain amount of money. This may be the least profitable approach, as users most likely want to try the application before paying. This can however be solved by offering a free application in addition with less functionality or content, so the user has the ability to try before buying.

Chapter 7

Comparing the Applications

Following are a comparison of all the commercial mobile news applications listed in section 3.5, as well as the application created associated with the use case.

7.1 Table Comparison

A lot of the commercial news recommender applications are quite alike on several levels. In table 7.1 the key features of the apps are extracted and sorted in to a more clean view. Using this table makes it easier to see where the apps differ and where they are similar.

Because of Google's acquirement of Wavii resulting in a removal from the app stores, some of the information for this application was not obtainable when comparing the different news apps.

7.1.1 Explanation

Following is an explanation of what the different comparison criteria means.

News Sources

News sources are which types of sources the news applications make use of. There are three main types, namely:

- News publishers
 - Traditional news content providers like The New York Times or BBC
- Blogs

- Independent blogs like an artist’s own blog or larger scale blogs like the tech blog Engadget.
- Social networks
 - Include friends’ feeds in the app or content friends shared on their social networks, like Facebook, Twitter, Google+.

Perspectives

Perspectives are which different perspectives the current app provides. The different perspectives are presented in section 5.1.

Login

If the app provides an opportunity for logging in, and which consequences this has for the use of the app. There are four different types:

- Yes
 - The login is mandatory to be able to use the application.
- No
 - There is no login opportunities.
- Voluntary
 - It is optional to login, and there are no limitations when using the app, other than not being able to get the devices preferences synced with other devices.
- Voluntary, limited use
 - The login is voluntary, but if choosing not to log in some of the features are not available, in addition to not being able to sync the preferences with other devices.

Input mode

Input mode are the different ways of interacting with the application, there are three main types:

- Gestures

- The input is registered in terms of different gestures like swiping, hold and drag, or long press.
- Buttons
 - Buttons are used to interact with the application.
- Text input
 - Text input is used when interacting with the application. This is primarily used to write search phrases inside the application, and not as part of any navigation logic.

User profiling

The different ways the application uses to gather information when building user profiles and recommending news to the user. The different kinds are:

- Thumbs up/down
 - The user gives explicit feedback about an article in terms of thumb up button or thumb down button, which rates as positive or negative feedback respectively.
- Log analysis
 - The application collects implicit feedback by logging how the user is using the application and the log is then sent to the back-end for further analysis.
- Stored topics
 - When a topic is stored by the user, this event is taken into account when building the user profile and recommending news.
- Likes
 - The user gives explicit feedback to an article by in some way liking this article via a button or gesture.
- Social activity
 - The application crawls the user's social networks searching for content that the user has shared or friends of the user has shared with the user. The content that is shared is further analyzed to affect the user profile or recommendation in some way.

- Blocked sources
 - The user gives explicit feedback by blocking a content source or a specific author.
- Location
 - The device's location is gathered and the app can recommend news based on the device's location.
- Saved articles
 - Articles that are in some way saved are analyzed and taken into account when building the user profile and recommending news.
- Overriding user profile
 - The user can override its own profile if he or she means that the user profile that has been created by explicit and/or implicit feedback is somehow inaccurate or in other ways wrong.

Filtering

The different ways an application provides to filter the content. The different approaches is as follows:

- Search
 - News articles can be filtered by searching for a topic or keyword
- Category
 - News articles can be filtered by categories
- User profile
 - News articles can be recommended according to the user profile.
- Related stories
 - News articles can be filtered by related stories to the current displaying story.
- Similar stories
 - News articles can be filtered by similar stories to the current displaying story.

- Tags
 - News articles can be filtered by tags that are somehow connected to a news article.
- Social behavior
 - News articles can be filtered by social behavior, for instance by number of likes on Facebook, or number of retweets on Twitter.
- Location
 - News articles can be filtered based on the device's location if the news article has location information.
- Publishers
 - News articles can be filtered by a single content publisher.

Trends

Trends is if an application has a feature for showing trending news over time. For instance a graph showing the amount of articles that has been published on a single topic over a given amount of time.

Sharing

Which types of channels the application offers to share content via. These can be social networks like Twitter or Facebook, or other types like Email or SMS.

Storing

Which types of storing an article the application supports. These can be other commercial solutions like Pocket or Instapaper, or just an own developed solution provided by the application itself.

7.1.2 The Table

Table 7.1: Comparing the different news recommender apps.

| <i>Feature / App</i> | Zite | Flip-board | Pulse | Summly | News360 | Circa | Wavii | Prismatic | Taptu | Feedly | Use case |
|-----------------------------|-------------------------------|---|---|----------------------------------|-------------------------------|------------------------|---|-------------------------------|---|-------------------------------|-------------------------------|
| News sources | News publishers, blogs | News publishers, blogs, social networks | News publishers, blogs, social networks | News publishers, blogs | News publishers, blogs | News publishers, blogs | News publishers, blogs, social networks | News publishers, blogs | News publishers, blogs, social networks | News publishers, blogs | News publishers |
| <i>UI</i> | | | | | | | | | | | |
| Perspectives | RSS, web, full article | RSS, web, full article | RSS, web | short summary, long summary, web | RSS, full article, web | RSS, summary, web, map | Events, summary, web | RSS, full article, web | RSS, full article, web | RSS, web | RSS, full article, map |
| Login | Voluntary | Voluntary, limited use | Voluntary | No | Voluntary | Voluntary | Yes | Yes | Voluntary | Voluntary, limited use | Voluntary |
| Input mode | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input | Gestures, buttons, text input |
| <i>Technology</i> | | | | | | | | | | | |

Continued on next page

Table 7.1 – Continued from previous page

| Feature / App | Zite | Flip-board | Pulse | Summly | News360 | Circa | Wavii | Pris-matic | Taptu | Feedly | Use case |
|-----------------------|---|--|--------------------------------|------------------|--|--------------|-----------------------------------|---|------------------|------------------|---|
| User profiling | Thumbs up/-down, log analysis, stored topics, likes, social activity, blocked sources | Blocked authors, stored topics, social activity | Social activity, user location | N/A | Thumbs up/-down, social activity, log analysis, saved articles | N/A | Social activity | Social activity, log analysis (in real time), stored topics, saved articles | N/A | N/A | Log analysis, user location, overriding user profile |
| Filtering | Search, category, user profile, related stories, tags | Search, category, social behavior(cover stories) | Search, category, location | Search, category | Search, category, user profile, location, similar stories | Category | Search, category, social behavior | Search, category, user profile, related stories, tags, publishers | Search, category | Search, category | Search, category, related stories, user profile, location |
| Trends | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A |

Continued on next page

Table 7.1 – Continued from previous page

| Feature / App | Zite | Flip-board | Pulse | Summly | News360 | Circa | Wavii | Prismatic | Taptu | Feedly | Use case |
|----------------------|--|---|---|--------------------------|---|-------------------------------|--------------|--------------------------|------------------------------------|---|--------------------------|
| Sharing | Email, Twitter, Facebook, Google+, LinkedIn, SMS | Email, Twitter, Facebook, Google+, LinkedIn | Email, Twitter, Facebook, LinkedIn, SMS | Email, Twitter, Facebook | Email, Twitter, Facebook, Google+ | Email, Twitter, Facebook, SMS | N/A | Email, Twitter, Facebook | Email, Twitter, Facebook, LinkedIn | Email, Twitter, Facebook, Google+, Buffer | Email, Twitter, Facebook |
| Storing | Evernote, Instapaper, Pocket | Instapaper, Pocket, Readability | Own storing | Own storing | Own storing, Evernote, Instapaper, Pocket | Own storing (follow story) | N/A | Own storing | Own storing, Instapaper, Pocket | Own storing, Instapaper, Pocket | Own storing |

7.1.3 Key characteristics

The different commercial applications examined in this section has a lot of similarities, but differs in some aspects as well.

All applications, except Summly and Wavii, has some RSS type of entry point to a news article, followed up by a full article view, a web view or both. They all have ways of sharing and storing content, and they all have a similar user interaction pattern, making use of gestures, buttons and text inputs. Even though all applications uses text input, it is mostly because the application offers a way of searching for content, and Circa being the only application that does not offer a search functionality, has no text input interaction, also non of the examined applications has any support for showing news trends over a given timespan.

Where they differ the most are their degree of recommendation, and with this the amount of filtering tend to follow. Zite, News360 and Prismatic has the most advanced user profiling and they also have the most filtering opportunities.

Also the degree of login differs quite a lot, but this does not seem to have any relation to what kind of application it is and what features or technologies it offers.

Part IV

Discussion and Conclusion

Chapter 8

Discussion

This chapter contains a discussion on how the perspectives are best used and how they can fulfill or replace each other. Further the examined news applications will be classified into different categories and the implementation of the use case will be discussed.

8.1 Perspectives

There are a handful of different perspectives presented in section 5.1, and they all seem to have different purposes and functions in a mobile news application. However there are several of the perspectives presented that in one way or another can overlap each other, or even replace another perspective completely. Some may be combined, and others may be split down into several perspectives. Some screens that are used throughout the different apps, like for instance a category selection screen or search screen, are not classified as a news perspective since it does not contain any information directly from a news article.

At a basis, a mobile news application should at least consist of two perspectives. An entry perspective and a perspective providing more information to an article linked from the entry perspective. The entry perspective should have as a main function to show articles or topics in a clear and lucid manner to be able to present the user with an easy way of understanding what the article or topic is mainly about. If the user finds it interesting and wants to access more information about the article or a given topic, the entry perspective should point to a perspective containing this. Further other perspectives can be added in addition to the two main perspectives to provide a higher navigation hierarchy, or replacing some of them by breaking down into smaller or other types of perspectives serving the same purpose.

A basic news model containing only the two main perspectives mentioned above, could be a simple RSS reader which consists of an RSS perspective as the entry perspective and a web perspective as the perspective holding more information about a single news article. This is an approach that is widely used, but if the developers wanted to extend this model in some way to convey the information in another way, this could be done using some of the other perspectives. For instance, the RSS perspective could be replaced by the event perspective to quickly tell what the article was about, further the event perspective could be linked to a summary perspective to give a little bit more depth to the article before the summary perspective sends the user further down the navigation hierarchy to a full article perspective containing all there is to know about this article.

Some of the perspectives presented could also be combined. The event perspective and the entity perspective could be melted down to one perspective by, for instance, having a view showing the events that the news article concerns, and at the same time highlighting the different entities that are identified in the events that are listed. Further the click of an entity could send the user to a new RSS perspective showing a list of articles that are connected to this entity.

As well as combining perspectives, some perspectives can be broken down into smaller pieces as well. Take, for instance, the full article perspective. This is a perspective that holds a lot of information, and say that the user only wants to access the article's full text, this could be presented as an own perspective, only showing the article's full body text, and excluding all the other information about the article that are not interesting at the given moment, and at some point could be perceived as noise.

It is also important to use the different perspectives where they are best intended and serves the best possible purpose. Presenting a full article perspective as an entry point with an article the user may not be interested in at all, may be perceived as too much information and even noise by the user. Showing a map perspective with no contextual information may be very confusing to the user, not knowing why that pin is placed at that place or what the article is about in the first place.

8.2 Mobile News Applications

Table 7.1, which presents the different news applications that are examined, shows that there are a lot of similarities between the applications. However, each application has some features or is lacking some features, that make them classifiable.

The different applications can be broken down to three main categories, depending on their features and what their main area of focus are. The three categories are recommendation apps, summarization apps and advanced RSS apps, where these names are created solely to try to give an understanding of what the main focus areas or features of the application are.

8.2.1 Recommendation Apps

These are the applications that focuses mainly on giving users the news they think they want, by using some sort of advanced recommendation and personalization technologies, based on the user's explicit and/or implicit feedback. Following are the apps put in this category.

- Zite
- News360
- Prismatic
- Use case

All these application are put in this category because they all track the user's activity to gain implicit feedback from the user, and some of them also uses explicit feedback for even better user profiling. They all use some kind of advanced recommendation technology to recommend news based on the user profile, and their main focus is on the recommendation technology and providing the user with news that best fits the user profile.

8.2.2 Summarization Apps

These are the applications that focuses mostly on creating quality content for their users, by summarizing and extracting the most important parts of the news articles, and not focusing as much, or at all, on recommending news. Following are the apps put in this category:

- Summly
- Circa

Both Summly's and Circa's main focus are on extracting the most important information from a news article and presenting it to the user. Neither of them have any user profiling or recommendation of news to the user. Both deliver the most trending news and trusts the user to find the news that

they want themselves. Summly has the ability to search for and add news categories or topics, while Circa only has a finite set of categories and news articles to choose from. However, with Circa the user can choose to follow different stories, and get notified when a story changes or new information to a story is added.

Even though their put in the same category and their main focus is the same, their approach to create summaries are totally different. Summly uses NLP and machine learning to analyze text and extracting the main content, while Circa has real human editors that creates their summaries.

8.2.3 Advanced RSS Apps

These are the applications that works similarly to normal RSS readers, and does not strive to create content summaries or recommend news on an advanced level, but may include more functionality than a basic RSS reader like proposals to news categories for the users, searching for content, or crawling social sites for content that is shared. Following are the apps put in this category.

- Flipboard
- Pulse
- Wavii
- Taptu
- Feedly

The applications put in this category are applications that neither recommend news based on user profiles or creates summaries for the user. These application can be thought of as advanced RSS readers, which helps the user with getting content, but having the user adding content explicitly. All applications has a certain set of preloaded categories or content the user can choose to add to the news feed, but news are not recommended based on what the user is doing. Some of these applications also has a more social aspect allowing the user to add their social feeds into the application. The news that are delivered to the user are more based on what is hot and trending with regard to most shared on Twitter or Facebook, for instance, than tailoring it specifically for the single user.

Wavii is somewhat different than the others though, since this application creates events from news content, and present these in a Facebook type of manner, but it still presents news events that are trending and not tailored for the single user.

8.3 Use Case Implementation

The use case application presented in section 4.1 were meant to include all of the perspectives described in section 5.1 to see how it could be solved in navigational and presentational manner, but this was not feasible at this point due to several factors. How to implement it has been thought of and will be presented in chapter 10.

To sum up how the application's perspectives are implemented in a brief manner, the RSS perspective is presented after the user has chosen recommended news or another category. From the RSS perspective, the user can either go back or open the full article perspective which links to related articles and the map perspective.

The application uses mostly gestures for navigational purposes to save space on the already limited screen size that comes with mobile devices, and buttons are kept to a minimum. This may lead to less intuitive user interface, as mentioned by some of the users when testing the application¹.

Some of the users stated that the navigation between perspectives sometimes was counter intuitive, meaning that they were presented with the previous perspective when they wanted to go forward and vice versa. For instance was the category selection screen shown when they wanted to view the full article perspective. This was in most cases quickly adapted, after the application had been used a bit. This can also be reflected by the statistics gathered from the usage of the application (see figure 8.1), where it shows that the number of times the top stories was shown throughout a month was almost equal to the times the settings screen was shown, which probably was not the user's intention. The navigation scheme in the application may be a bit different as most mobile users probably are used to get new perspectives stacked from the right, as is the scheme in most mobile applications, and not stacked down from the top.

| | | | |
|---|-------------------------------------|----|--------|
| 2 | FrontPageView->ShowTopStories.Swipe | 88 | 15.9 % |
| 3 | FrontPageView->ShowSettings | 87 | 15.8 % |

Figure 8.1: Statistics from the use case application showing how many times the top stories and settings screen are shown in a month.

The navigation scheme can be seen as rotated 90 degrees to the left to what is mostly used in the other news applications. Instead of swiping up

¹The application was tested on five persons, all male, in the age 21-30, where the test persons used the application for ten to fifteen minutes and afterwards spoke freely of what they thought of the application and how it was to use. Also the application included checkpoints to build statistics of how the application was used.

to scroll down to see more, the user has to swipe left to scroll right to see more news, and instead of bringing new perspectives in from the right, the application stacks them from the top. As this maybe be confusing to some, it is consistently done throughout the whole application and was quickly adapted by most of the users that tested it. To help users to start using the application it also has small pop-ups, explaining how to navigate around the application, the first time it is launched.

It could have been wiser to just follow a well established navigation scheme, as this probably are more intuitive to the user and the testing showed that most of the users testing the application did not pay much attention to the pop-ups, but just started to use the application without reading what the pop-ups said.

The reason that this approach was chosen, was that it was a little different than the other most used design approaches, as well as it suited the swipe-based navigation scheme that was one of the main focuses to follow when developing the application. There are probably a lot of different approaches than could have been followed to design and implement the use case application, like the vertical news feed used in Zite for instance, or the two dimensional scroll views, used in Pulse and Taptu. The main reason the vertical scroll solution, found in Zite, was not chosen, was that it was too main stream, hence to familiar to the common user. A point was to not have a too familiar UI, to see how the navigation scheme developed were to be received at the user's end, without any pre-knowledge of how it would work. The main reason the two dimensional UI approach, found in Pulse was not chosen, was that it appeared too cluttered with too much information displaying on a small mobile screen.

The navigation scheme put aside, the application worked well with the connection between the perspectives and the users stated that the perspectives showed what they was expecting to see in the different perspectives. Again, there was no big surprises to how the information was presented, as all the perspectives used layouts that are pretty common in other applications.

Chapter 9

Conclusion

Mobile news applications has become very popular, and the popularity is only increasing. During this project three of the commercial mobile news applications examined, namely Summly, Wavii and Pulse, has been sold to large software companies, Yahoo!, Google and LinkedIn respectively. With the popularity of the mobile news application growing, the scale of mobile news consumption is growing increasingly as well, alongside with the technologies that these applications make use of. Recommendation technologies, for instance, has seen a rapid growth as a technology used in news applications, especially the use of active personalization making use of implicit user feedback.

By examining the different mobile news applications, one can see that there has been a common way of designing mobile news applications. Most applications greets the user with an RSS type of perspective as an entry point with a vertical scrollable view to give the user a short introduction to the different news articles, to further show a full article perspective or web perspective when an article is tapped. Pulse and Taptu uses a different approach where they also has a RSS type of perspective as an entry point with a vertical scrollable view, but they also put a horizontal scrollable view inside each element in the vertical list, which may come of as cluttered and too much information on a small screen.

The use case application developed associated with this thesis, chose a different approach similar to Summly and News360, where each article has a full screen for both the RSS perspective and the full article perspective, or the summary perspective in the case of Summly. This may also be the reason that some users that tested the use case application found it a bit confusing at first, since it did not follow the most used approach.

The idea with the use case application was to implement all the introduced perspectives in one consistent mobile UI, but this was not feasible at this

point due to certain factors. However there is no reason that this cannot be accomplished and a proposal to how this can be done is explained in chapter 10.

Another design pattern that was followed by all the applications examined, was the hierarchical approach to present a news article, where the perspective showing the least information was the first perspective that was presented to the user. The further down in the navigation hierarchy the user got the more information the perspective holding the news article contained.

There is clearly a common way to design mobile news applications, but if this alone has any impact on the popularity of the application, is uncertain, given that the three commercial applications that were sold all had different design approaches.

Chapter 10

Further Work

Following are a proposal for further work to be carried out as a future part of this study.

Perspectives

A further study to see if there are other perspectives that are used in other applications, and if so, evaluating these to see how they are used and if they are suited for other purposes as well. A possible perspective that could be implemented, but not has been in any of the mobile news applications examined in this thesis, are a trend perspective, that shows a topic's or entity's popularity over time.

Implementation

The use case application presented in section 4.1 were meant to include all of the perspectives describes in section 5.1 to see how it could be solved in navigational and presentational manner.

As of now the application has the RSS perspective, full article perspective and map perspective, but how to implement the others as well has been thought of and should be implemented in future work.

The event, entity and web perspective can be put at the same level as the full article perspective, to provide additional information, or another form of viewing the information gathered from the full article view. This could be done by swiping down in the full article perspective to reveal one button for each of the other perspectives and allowing the user to choose which type of perspective it wants to view the news article from.

The summary perspective could be placed between the RSS perspective and the full article perspective in the navigation hierarchy, to provide the

user with an extra level, where more information about the article is shown, but not all of it, to get the notion of what the article is concerning before opening the full article.

Testing

Larger scaled and more comprehensive testing of the use case application and how the different perspectives are connected and presented in regard to user experience and usability, ought to be carried out. Also testing to see if perspectives used in a different manner, i.e. another navigation logic than the one that is most adapted by applications already developed, can be suited for users.

Research

A future research on why the different news applications are designed as they are and why there is an overweight of native third-party news applications, and not so much native news applications created by the content publishers themselves, could have been an interesting new depth to this study.

Bibliography

- [1] Benckert van de Boel Anna. Designing the future of the newspaper. 2011.
- [2] Charles Arthur. Teenager in need of £18m? there's a yahoo app for that. <http://www.guardian.co.uk/technology/2013/mar/25/summy-sold-yahoo-teenage-creator>, March 2013. [Accessed: April 16, 2013].
- [3] Daniel Billsus and Michael J Pazzani. User modeling for adaptive news access. *User modeling and user-adapted interaction*, 10(2-3):147–180, 2000.
- [4] The Ellerdale Project Blog. Ellerdale has been acquired by flipboard. <http://blog.ellerdale.com/>, July 2010. [Accessed: Mars 20, 2013].
- [5] Paul Boutin. Making money off twitter: The ellerdale project offers enterprise-scale search and analysis at 600 tweets per second. <http://venturebeat.com/2010/04/13/twitter-ellerdale/>, April 2010. [Accessed: Mars 20, 2013].
- [6] Andrei Z Broder. On the resemblance and containment of documents. In *Compression and Complexity of Sequences 1997. Proceedings*, pages 21–29. IEEE, 1997.
- [7] Andy Carvin and Noel Cody. Results from our survey of npr facebook fans. <http://www.npr.org/blogs/gofigure/2010/08/02/128928306/results-from-our-survey-of-npr-facebook-fans>, August 2010. [Accessed: May 10, 2013].
- [8] The Pew Research Center. Trends in news consumption: 1991-2012. <http://www.people-press.org/files/legacy-pdf/2012%20News%20Consumption%20Report.pdf>, September 2012. [Accessed: June 8, 2013].

- [9] Circa. Catch up quick. <http://cir.ca/>, April 2013. [Accessed: April 18, 2013].
- [10] Per-Erik Danielsson. Euclidean distance mapping. *Computer Graphics and image processing*, 14(3):227–248, 1980.
- [11] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*, pages 271–280. ACM, 2007.
- [12] eMarketer. Consumers spend more time with mobile as online growth slows. <http://www.emarketer.com/Article/Consumers-Spend-More-Time-with-Mobile-Online-Growth-Slows/1009431#IBhfumFlgTgxclz4.99>, October 2012. [Accessed: May 14, 2013].
- [13] Inside Flipboard. Apple picks flipboard as app of the year! <http://inside.flipboard.com/2010/12/09/apple-picks-flipboard-as-app-of-the-year/>, December 2010. [Accessed: Mars 20, 2013].
- [14] Inside Flipboard. Demo of flipboard for iphone. <http://www.youtube.com/watch?v=Y0QAGmXOERI&feature=share&list=UU0faC-F4a2MTRm37U02S17Q>, December 2011. [Accessed: Mars 20, 2013].
- [15] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.
- [16] Zan Huang, Hsinchun Chen, and Daniel Zeng. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems (TOIS)*, 22(1):116–142, 2004.
- [17] Cylogi Inc. Personalization overview. <http://venturebeat.com/2010/04/13/twitter-ellerdale/>, March 2013.
- [18] Zite Inc. Description. <https://itunes.apple.com/pl/app/zite/id419752338?mt=8>, March 2013. [Accessed: Mars 20, 2013].
- [19] Zite Inc. Faq. <http://blog.zite.com/228-2/>, March 2013.

- [20] Lauren Indvik. Flipboard: Behind mobile’s most beautiful newsreading magazine. <http://mashable.com/2012/06/05/flipboard-design/>, June 2012. [Accessed: April 16, 2013].
- [21] Lauren Indvik. Flipboard: Behind mobile’s most beautiful newsreading magazine. <http://mashable.com/2012/06/05/flipboard-design/>, June 2012. [Accessed: Mars 20, 2013].
- [22] Dietmar Jannach. Recommender systems - an introduction corrigenda list.
- [23] Finn V Jensen. *An introduction to Bayesian networks*, volume 74. UCL press London, 1996.
- [24] Samir Khullera, Anna Mossb, and J Naor. The budgeted maximum coverage problem. *Information Processing Letters*, 70:39–45, 1999.
- [25] Xuan Nhat Lam, Thuc Vu, Trong Duc Le, and Anh Duc Duong. Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, pages 208–211. ACM, 2008.
- [26] Frederic Lardinois. LinkedIn acquires pulse for \$90m in stock and cash. <http://techcrunch.com/2013/04/11/linkedin-acquires-pulse-for-90m-in-stock-and-cash/>, April 2013. [Accessed: May 6, 2013].
- [27] Vadim Lavrusik. How news consumption is shifting to the personalized social news stream. <http://mashable.com/2010/08/10/personalized-news-stream/>, August 2010. [Accessed: May 10, 2013].
- [28] Hong Joo Lee and Sung Joo Park. Moners: A news recommender for the mobile web. *Expert Systems with Applications*, 32(1):143–150, 2007.
- [29] Lei Li, Dingding Wang, Tao Li, Daniel Knox, and Balaji Padmanabhan. Scene: a scalable two-stage personalized news recommendation system. In *ACM Conference on Information Retrieval (SIGIR)*, 2011.
- [30] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*, pages 31–40. ACM, 2010.

- [31] Marissa Mayer. To summarize, the new yahoo! app delivers better mobile content discovery. <http://ycorpblog.com/2013/04/22/42779/>, April 2013. [Accessed: April 23, 2013].
- [32] Prem Melville and Vikas Sindhwani. Recommender systems. *Encyclopedia of machine learning*, 1:829–838, 2010.
- [33] Amy Mitchell, Tom Rosenstiel, and Leah Christian. Mobile devices and news consumption: Some good signs for journalism. <http://stateofthedia.org/2012/mobile-devices-and-news-consumption-some-good-signs-for-journalism/>, December 2012. [Accessed: May 24, 2013].
- [34] News360. About. <http://news360.com/>, April 2013.
- [35] News360. Technology. <http://news360.com/technology/>, April 2013.
- [36] Jason D. O’Grady. Wwdc 2011: Apple design award winning apps worthy of your download. <http://www.zdnet.com/blog/apple/wwdc-2011-apple-design-award-winning-apps-worthy-of-your-download/10339>, June 2011. [Accessed: April 15, 2013].
- [37] Kristen Purcell, Lee Rainie, Amy Mitchell, Tom Rosenstiel, and Kenny Olmstead. Understanding the participatory news consumer. *Pew Internet and American Life Project*, 1, 2010.
- [38] Francesco Ricci. Mobile recommender systems. *Information Technology & Tourism*, 12(3):205–231, 2010.
- [39] Cyril Rohr and Dian Tjondronegoro. Aggregated cross-media news visualization and personalization. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 371–378. ACM, 2008.
- [40] Thomas L Saaty. *What is the analytic hierarchy process?* Springer, 1988.
- [41] High Scalability. Prismatic architecture - using machine learning on social networks to figure out what you should read on the web. <http://highscalability.com/blog/2012/7/30/prismatic-architecture-using-machine-learning-on-social-netw.html>, July 2012. [Accessed: April 23, 2013].

- [42] Bracha Shapira, Peretz Shoval, Noam Tractinsky, and Joachim Meyer. epaper: A personalized mobile newspaper. *Journal of the American Society for Information Science and Technology*, 60(11):2333–2346, 2009.
- [43] Xuemin Shen, Heather Yu, John Buford, and Mursalin Akon. *Handbook of peer-to-peer networking*, volume 1. Springer Heidelberg, 2010.
- [44] Jonathan Strickland. How wavii works. <http://www.fwthinking.com/blog/wavii-works/>, March 2013. [Accessed: April 21, 2013].
- [45] Summly. Summly launch. <http://vimeo.com/52014691>, November 2012. [Accessed: April 17, 2013].
- [46] Summly. Summly technology. <http://summly.com/technology.html>, April 2013. [Accessed: April 17, 2013].
- [47] Taptu. Description. <https://itunes.apple.com/no/app/taptu-dj-your-news/id392240746?mt=8>, April 2013. [Accessed: April 25, 2013].
- [48] Kardi Teknomo. K-means clustering tutorial. *Medicine*, 100(4):3, 2006.
- [49] Neil Thurman and Steve Schifferes. The future of personalization at news websites: lessons from a longitudinal study. *Journalism Studies*, 13(5-6):775–790, 2012.
- [50] Alexia Tsotsis. Google buys wavii for north of \$30 million. <http://techcrunch.com/2013/04/23/google-buys-wavii-for-north-of-30-million/>, April 2013. [Accessed: May 6, 2013].
- [51] VG Vinod, Raman Ann Jim, and Chandrasekar Paradiso St George. News sync: Three reasons to visualize news better. *HCIR 2010*, page 45, 2010.
- [52] Hao Wen, Liping Fang, and Ling Guan. A hybrid approach for personalized recommendation of news on the web. *Expert Systems with Applications*, 39(5):5806–5814, 2012.
- [53] Wikipedia. Prismatic (app). http://en.wikipedia.org/wiki/Prismatic_%28app%29, March 2013. [Accessed: April 22, 2013].
- [54] Nick Wingfield. Wavii: A facebook for topics. <http://bits.blogs.nytimes.com/2012/04/13/wavii-a-facebook-for-topics/>, April 2012. [Accessed: April 18, 2013].

- [55] Yahoo. Yahoo! to acquire summy. <http://ycorpblog.com/2013/03/25/yahoo-to-acquire-summy/>, March 2013. [Accessed: April 16, 2013].
- [56] Kam Fung Yeung and Yanyan Yang. A proactive personalized mobile news recommendation system. In *Developments in E-systems Engineering (DESE), 2010*, pages 207–212. IEEE, 2010.
- [57] Ken Yeung. News curator summy launches to help simplify the way we consume news on mobile devices. <http://thenextweb.com/apps/2012/11/01/news-curator-summy-launches-to-help-simplify-the-way-we-consume-news-on-mobile-devices/>, November 2012. [Accessed: April 17, 2013].

Appendices

Appendix A

Implementation of Use Case Application

Following are a more detailed description of the iPhone application, developed as part of this project.

A.1 Overview

The implementation of the use case application, was conducted throughout the spring semester 2013, and was a continuously working progress. All implementation was done by the student itself, but the news delivery system that worked as the back-end was provided by the Department of Computer and Information Science, NTNU and other collaborators. A video demonstration of the application can be found at <http://youtu.be/3Hgvn1qZ67A>. No code samples will be provided in this appendix, but the whole project is submitted as an external file with the project delivery, and if the code itself is of interest, it is all included in the submitted file.

A.1.1 Technical Specifications

| | |
|-----------------------------|---|
| IDE | Xcode Version 4.6.2 (4H1003) |
| Devices | iPhone (3GS, 4, 4S, 5), iPod touch (3rd, 4th, 5th generation) |
| OS | Requires iOS 6.0 and above |
| Programming language | Objective C |
| Code lines | 4500 |
| Design pattern | Model-View-Controller |

A.2 External Libraries

Following are a quick description of the external libraries that the application makes use of.

SVPullToRefresh

<https://github.com/samvermette/SVPullToRefresh>

SVPullToRefresh is an extension control to UIScrollView that makes it possible to swipe and release to update the data in the UIScrollView.

It is used in the front page, figure A.1, to be able to refresh the top stories in from the front page.

SVProgressHUD

<https://github.com/samvermette/SVProgressHUD>

SVProgressHUD is an information control to help display messages to the user, be it progress, a success or failure message or loading. It is a singleton instance, which makes easy to use from wherever by calling its class methods.

The control is used a lot throughout the application, for instance when loading news, updating news, loading or saving the user profile, or when an article is saved.

SKBounceAnimation

<https://github.com/khanlou/SKBounceAnimation>

SKBounceAnimation is a library that makes it easier to implement a bounce effect when animating views.

This effect is used whenever a new view is loaded in the application.

TestFlight SDK 1.2.4

<https://testflightapp.com/sdk/>

The TestFlight SDK provides usage statistics, remote crash reports and logging, as well as making it possible to install the application for test purposes on remote devices.

This library is used to log checkpoints when a user is using the application, check remote crash reports and testing the application on remote devices.

CMPopTipView

<https://github.com/chrismls/CMPopTipView>

The CMPopTipView offers an easy way to implement pop-up views that point to certain views associated with some text.

This control is used on the first launch of the application to explain how it is used, and what features that are available.

KLExpandingSelect

<https://github.com/KieranLafferty/KLExpandingSelect>

KLExpandingSelect offers a neat control to share and star content. It does not include the sharing itself, but the UI to trigger the different events.

This control is used to offer the user an opportunity to share news articles in an easy way to Twitter, Facebook and mail, as well as saving articles for accessing later (see figure A.5).

A.3 Screenshots

Following are screenshots from all the screens in the application.

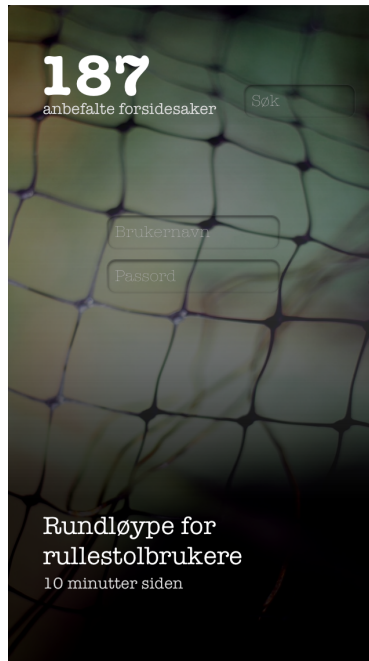


Figure A.1: Screenshot of the use case application showing the front page screen.

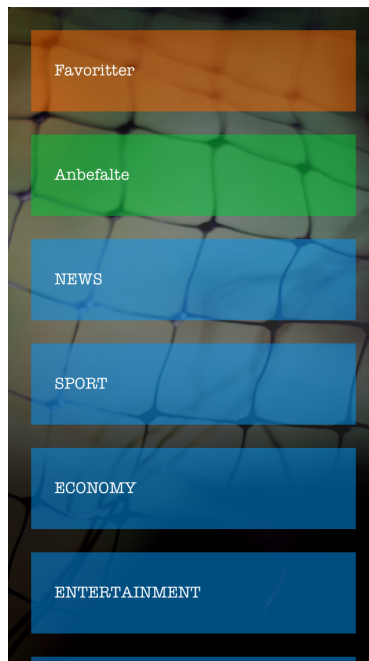


Figure A.2: Screenshot of the use case application showing the category selection screen.

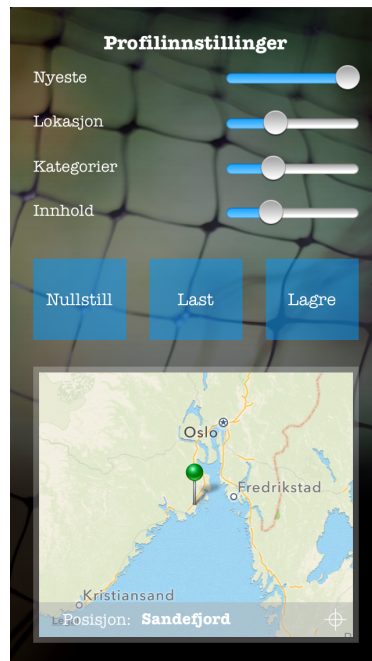


Figure A.3: Screenshot of the use case application showing the settings screen.

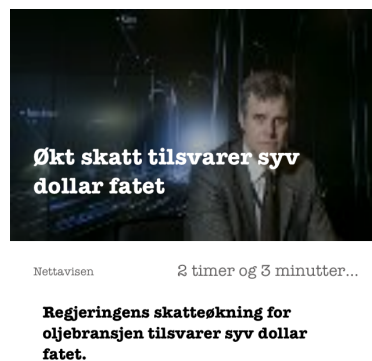


Figure A.4: Screenshot of the use case application showing the RSS perspective.

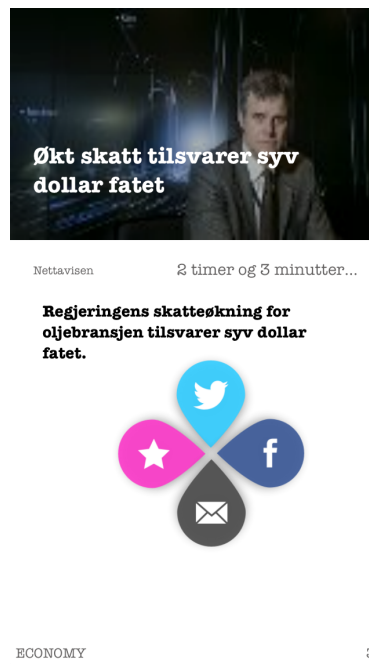


Figure A.5: Screenshot of the use case application showing the RSS perspective after triggering the share/save control.

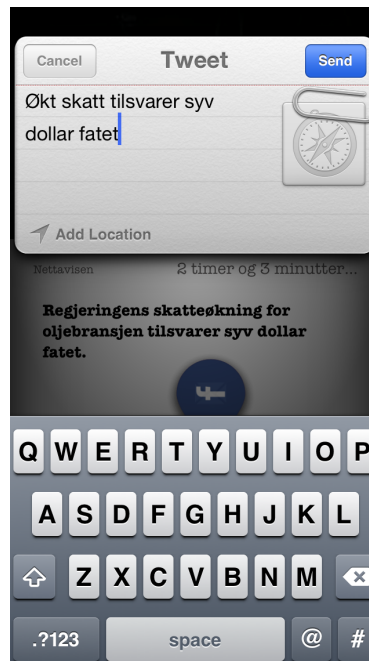


Figure A.6: Screenshot of the use case application showing the Twitter share screen.

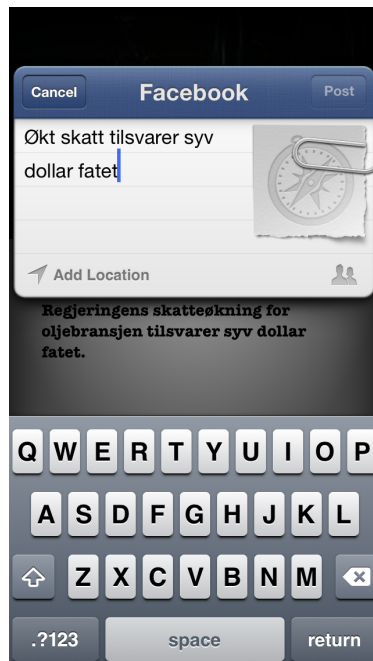


Figure A.7: Screenshot of the use case application showing the Facebook share screen.

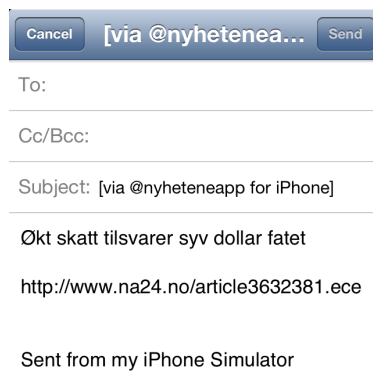


Figure A.8: Screenshot of the use case application showing the email share screen.



Figure A.9: Screenshot of the use case application showing the full article perspective.

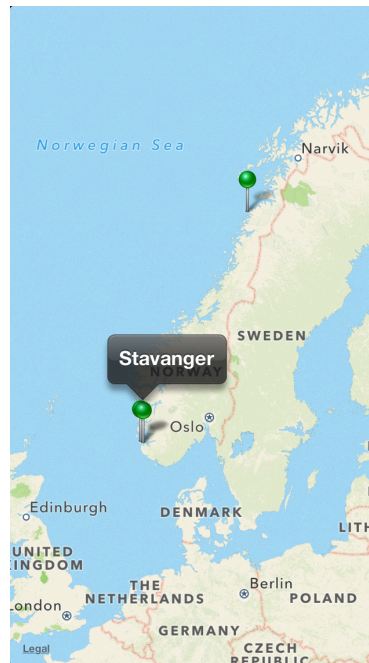


Figure A.10: Screenshot of the use case application showing the map perspective.

OL-vinnar Tone Edvardsen frå Randaberg har tatt eit val. Ho ser på det ho faktisk får til, i staden for det ho ikkje lenger klarar. - Fokuset mitt er å leva og ha det bra, seier 38-åringen. Naturglade Tone Edvardsen er stadig ute på nye turar i det fri, med fotoapparatet lett tilgjengeleg. Teodor (4) og Mikkel (2) frå Ørnes blir gjerne med tante Tone Edvardsen på tur, saman med pappa Åge Magnussen. Frå toppen av Randabergfjellet er utsikta god mellom anna til Kvitsøy. - Det er mange ting du kan få til om du veit det er trygt nok, legg ho til, medan ho styrer stolen framover på den gruslagde stien. Den 0,8 kilometer lange rundløypa på toppen av Randabergfjellet er ein enkel tur å ta for Tone. Sette seg fast ved elva - Det er ingen plass eg har sett meg så fast som ved Fossahølen i Tengselva. Der ville eg ned og sjå ein plass eg tidlegare hadde vore som gåande. Det måtte to mann til for å hjelpa meg laus, seier Tone og ler. Opp gjennom åra har Tone Edvardsen testa ut ei rekke av Aftenbladet sine Helge-turar i nærområdet. Nord-Jæren byr på ganske mange turar du kan klara sjølv om du sit i rullestol, meiner ho. Dagens tur går til Randabergfjellet. Det er eit av seks område i Randaberg som er tilrettelagt for at alle skal komma seg ut. Tone har vore med på å laga ein brosjyre som fortel i detalj korleis turane er, kvar det er bratte bakkar og som fortel om turen passar for synshemma og manuell

Figure A.11: Screenshot of the use case application showing the full text screen.