Enabling User to User Interactions in Web Lectures with History-Aware User Awareness

Markus Ketterl*, Robert Mertens**, Christoph Wiesen** and Oliver Vornberger***

* Center for Information Management and Virtual Teaching
  University of Osnabrück, Heger-Tor-Wall 12
  49074 Osnabrück, Germany
  Email: markus.ketterl@uni-osnabrueck.de

** Fraunhofer IAIS
  Schloß Birlinghoven, 53754 Sankt Augustin, Germany
  Email: robert.mertens@iais.fraunhofer.de
  christoph.wiesen@iais.fraunhofer.de

*** Dept. of Computer Science, University of Osnabrück, Albrechtstrasse 28
  49069 Osnabrück, Germany
  Email: oliver@uni-osnabrueck.de

ABSTRACT
User awareness has become a popular feature in many social web applications. In classic text-based web-systems, user awareness features show how many users are online in a web application or how many users are accessing the same web page. When time-based media like web lectures are concerned this approach comes to its limits since time-based media are inherently different from classic text-based media. The main difference in these two types of media is that text-based media can easily be skimmed at a glance while video- or audio-objects have to be fully replayed when users want to grasp their content. This paper presents an approach that employs time-based usage statistics for all users in a web lecture system in order to provide users with a means to communicate with other users who are online and have watched those parts of a media object that the user is interested in. The work is implemented as a prototype application in the context of the Opencast Matterhorn project – an open source based project for producing, managing and distributing academic video content.

Keywords: Lecture Recording, Opencast, Social Navigation, User Awareness, Web 2.0

1. INTRODUCTION
In many web portals, user awareness is designed to tell users which of their friends are online. While this information is useful in a leisure time scenario, it is only of limited use in an e-learning scenario or, for that matter, in any scenario in which users work with the content. In a learning related scenario, user awareness can be used to indicate whether other users are working with the same content or possess knowledge about that content. The user can then direct questions about that content directly to other
users that are both online and familiar with the content. In classical text and picture-based media, detecting users that work with the same piece of content and are currently online is relatively easy. Access statistics from a web server can be used to determine which object (like a specific chunk of text) a user is currently accessing. If these objects are reasonably sized, which is the case in most web content, a user accessing it can easily tell the contents of that object. However, web lectures and other time-based media differ from classical text and picture based media in that a user can hardly tell the content of those passages of the media-object he has not already replayed (Mertens et al., 2004). Hence, the fact that other users are online and retrieving the same media-object does not deliver any cue as to whether these other users have actually visited the parts of the media object that are relevant to the user. The approach presented in this paper solves this problem by matching viewing statistics of the current user with viewing statistics of all users that have accessed the respective media-object. The user interface provides a feature to specify arbitrary parts within a web lecture video so that users can ask questions about specific passages in a media object.

The remainder of this paper is structured as follows: Section 2 discusses related work in the field of user awareness in text based media as well as current developments bringing social web approaches to web lectures and other time based media. Section 3 explains in which context the work presented in this paper is being used. Section 4 describes the users’ perspective on the system. The technical background as well as perspectives for future work of the application is explained in sections 5, 6 and 7. A conclusive discussion of the approach is given in section 8.

2. RELATED WORK

Making web activity more apparent to users is a major part of today’s web experience. In literature one can find different types of user awareness. Liechti (Liechti and Sumi, 2002) differentiates four main categories, which are not mutually exclusive: group, workspace, contextual, and peripheral awareness. Dourish and Bellotti (Dourish, P. and V. Bellotti, 1992) proposed one of the first definitions for awareness: "Awareness is an understanding of the activities of others, which provides a context for your own activity". Group awareness conveys information about the state and activity within a team. Workspace awareness facilitates the coordination among users and allows communication and collaboration. Contextual awareness adapts the idea of presenting information to the user situation. Peripheral awareness works by filling users’ peripheral attention with information without distraction (Cadiz et al., 2001). An example would be the usage of peripheral audio for information updates (Pacey and MacGregor, 2001) while working with a computer. Web 2.0 and social network applications inherent many of this ideas and combine them in an attractive way. Early ideas for a social web (and social navigation) have been presented by (Palfreyman and Rodden, 1996) or (Dieberger, 1997). Nowadays social network applications allow users inside a network to get information about what is happening on the site or in the application (e.g. who checked my profile, left a comment). Further examples for awareness of web activity and description can be found in (Gellersen and Schmidt, 2002) or (Schmidt and Gellersen, 2001). Group awareness is an important issue in e-learning applications. Lecturers as well as students need to know the activities and background knowledge of the members of their group in order to support learning effectively. Ad-hoc communication and collaboration among users can support online learning or foster coordination work. Establishing and maintaining awareness has been reported to be difficult without appropriate supporting tools (Gutwin and Greenberg, 2002). Fundamental privacy techniques and tradeoffs for awareness systems have been presented in (Hudson and Smith, 1996). In (Hu et al., 2002) an awareness component model with a notification mechanism has been presented that enables group awareness within web-based learning components. Researchers have reported a significant effectiveness of applying media technologies, such as chatting tools, shared workspace tools, video and audio tools, visualization representation tools, email and notification tools to support awareness through groupware systems (Tsai et al., 2005).
Examples for the visualization of social data that conveys information about the online world and its participants can be found in Lifeline (Plaisant, 1996), Conversion Map (Sack, 2000), Netscan (Smith et al, 2001) or the Personal Map project (Farnham et al., 2003). Social browsers that support awareness and social interaction can be found in the work presented in (Lee et al., 2004) The authors explain the use of a people browser to visualize the relations among the members of a community and also the human interaction over time. In web lectures, user awareness has – to the author’s knowledge - only been implemented in the approach presented in (Fox et al., 2009) In that approach, three user awareness features are implemented: a “who is online”-list that shows which other users are online in the system, a social scrubber that shows thumbnail pictures of other users who are currently watching the same web lecture on the web lecture’s timeline and a cumulative footprint feature that shows how often a passage of the video has been accessed by other users. The “who is online”-list is not different from classical user awareness features found in many web portals; it simply shows the users who are currently logged into the system. The social scrubber is an interesting social feature that is tailored to the use in time-based media. It does, however, face the problem that it only shows who is currently watching the same web lecture. It is thus not history aware and loses valuable information. The social footprints help the user to identify what parts of a web lecture have been watched most by other students but they have no relation to other users who are currently online in the system as the footprints are a purely cumulative feature (Merten et al., 2006).

3. CONTEXT OF THE WORK

Web lectures have become a reliable learning companion for students at many places. Applications and research in the field of lecture recording have grown exponentially across the world. For the most part, these systems and technologies were originally introduced as research projects and evolved to meet local institutional or academic needs. To counter this trend, an alternative concept, Opencast\(^1\), was introduced by UC Berkeley to explore an Open Source alternative for the production, distribution, management of and engagement with audiovisual content. Opencast recognized the numerous academic efforts emerging in isolation, and has created a landscape on which institutions can combine efforts and increase innovation in the context of one large project (Ketterl et al., 2010). The work that is being presented in this paper is part of the development activity in Opencast.

Opencast Matterhorn Engage Applications

Most of us would not consider watching a lecture recording as "interactive learning", especially if it is a replay of ex-cathedra teaching. Lecture recordings alone, even before adding advanced learning features, can be more engaging by their very nature. With a system that can capture more than the information delivered in a lecture, a video has the potential to engage the student on a more personal level than other kinds of e-learning content. It allows the learner to make stronger connections to the lecturer as an actual person, as well as the other learners in the classroom, and benefit from the non-verbal and visual cues of the experience. Opencast Matterhorn engage applications go beyond a pure lecture video experience. The applications do support fine-grained segment based navigation (based on video OCR), in-video search and multi-stream replay. Application and multimedia accessibility has been a high priority throughout the development that allows full assistive technology support on many platforms. Figure 1 depicts a multi-stream engage application that shows the lecturer and a content stream in a synchronized way. In addition further content based navigation elements have been implemented.

---

\(^1\) http://www.opencastproject.org/
The history aware user awareness extensions are being implemented in the context of these applications and are currently in a prototype status. The ideas behind the user awareness features are being described in the next sections.

4. USERS PERSPECTIVE
The context-aware user-awareness feature described in this paper is designed for a special use case. When a user watches a web lecture and has a question about a specific passage in that lecture, he or she can find other users who are online and have already watched that very passage and are thus likely to be capable of answering that question. The overall workflow from a user’s perspective is shown in figure 2.
Figure 2: Workflow from a user's perspective

In this early implementation the user marks a passage in the web lecture he or she is currently viewing. This is done by selecting the passage’s starting and ending points on the timeline (1). After the user has selected a passage, the system checks if any users who have watched that passage already are online (2). If no such users can be found, the system informs the user. If such users can be found, a message field opens in the interface. The user then types a message – preferably a question – about the passage (3). When the user sends this message, the system forwards it to all users who have watched the passage and are online (4). These users can then start chatting with the user who posted the message (5). The message is posted to all online users who have watched the passage in question. However, the user who has posted the question is not aware of who these users might be unless they answer his or her question. This way, the other users’ privacy is protected.

5. Capturing Usage Data

When a user watches a video, so called footprints are generated automatically. Figure 2 depicts this data as viewing statistics in the timeline. Those footprints are further evaluated as depicted in figure 3. They store what passages of the video have been watched. As seen before a passage is denoted by the time index where the user hit play and the time index where replay was stopped (1). This information can be used by the individual user to visually indicate what passages he or she has watched. In addition, it can be useful to find new information or to find a passage the user remembers only vaguely. Those footprints are stored in a database and are being normalized relative to the total lecture length of the corresponding lecture (2). The footprints per user are then associated with the corresponding lecture video the user has watched (3). If a user selects a certain passage of the lecture video later, all footprints associated with the lecture video are checked, if they cover the selected passage. The fact whether a user is online is also stored in the datawarehouse (4).
Figure 3: When a lecture video is watched the start and endtime are converted into footprints. They are then stored in the datawarehouse and associated with the corresponding lecture.

The system also stores metadata associated with the user. These metadata can vary depending on the video-player’s context. The video-player can be embedded in different contexts (open website, portal, learn-management-system with authentication). In some contexts only a limited amount of information about the user is available. In case of a social network, a complete user profile exists. Each of these contexts poses its own demands on the structure and granularity of the metadata associated with a footprint. The remainder of this section describes four prototypical classes of contexts.

The first class of contexts is a nearly context-free environment like a website without any text and just the video player embedded. In this case, only general information like the client’s IP-address, time and date and possibly webpages linking to the site can be obtained. In this class of metadata for footprints (class I), geo-coding could be used to analyze geographical disparities in learning behavior. This approach is, however, not elaborated on in this paper.

A second class of contexts (class II) are webpages with machine-readable content. This is the case whenever the video-player is embedded in a webpage that contains text. In this case, the metadata of class I can be enriched with information on the webpage’s content. For this purpose, keywords can be extracted from the text and title of the surrounding webpage.

Class III contexts are given for web lectures that are accessible via a learn-management-system (LMS) with authentication only. In an LMS, users are linked to courses and a simple user profile that gives information about the user’s status field of study. Also, users are identifiable since they had to log in using a personal password or similar authentication method. Standalone web lecture systems that require a user to log in also fall in this category.

The last class of contexts (class IV) can be found in web lectures that are embedded in social network applications (Fox et. al, 2009). Facebook or applications that support OpenSocial are a good example for
social network applications. Social network applications usually come with user profiles that can store information about a user’s professional life, private life and study activities. The amount of information stored is, however, highly dependent on how much information the user leaves and can also be a privacy issue. Additionally, a social graph exists, that captures relations between users (friend, study mate, co-worker, etc.). Users can also be members of a group. Members of a group share a status, an activity or a field interest. Hence, usage data from gathered web lectures in this context come with a wealth of information.

The approach described in this paper is realized by sending messages to users that have already watched the passage in question. Ideally these users need to be identifiable from one session to the next. This is only given for class III and IV contexts.

6. Finding Users
In the datawarehouse, all users that have already watched the passage in question are retrieved. This is done by analyzing all footprints stored for the web lecture in question.

A key challenge is finding those users who are suited best to answer the user’s question. The precondition for a user to be associated with a passage is of course that the user has indeed watched the passage, or at least parts of the passage. After having selected all users who have watched all or parts of the marked passage the users can be sorted according to a list of criteria. These criteria could be:

- **How often has the user watched the passage?**
  This criterion is a double sided one. On the one hand, a user should be quite familiar with a passage he or she has watched more than once (for instance to prepare for an exam). On the other hand, repeated viewing could indicate that the user had problems understanding the passage’s content. In the current implementation, the criterion is regarded positively, i.e. more views increase the ranking.

- **How long ago did the user watch the passage?**
  If the user watched the passage a long time ago he or she might not be able to remember the content. Thus, the longer ago a user watched the passage the lower should be the ranking.

- **Has the user watched other videos from the same lecture?**
  Having watched other videos from the lecture give a user a broader knowledge about the topic. These users are thus ranked higher.

- **Is the user online?**
  This is a knock-out criterion. If the user is not online, he or she cannot answer the question online. Hence users who are not online are omitted.

- **Which parts of the passage did the user watch?**
  In the current implementation, messages are only sent to users who have watched the whole passage. Percentages of coverage could also be used as a criterion. If they are used, a certain limit should be set as a minimum percentage a user must have watched. If the percentage is smaller than the limit the user is not considered. However, the limit should not be static but should rather depend on the total length of the passage. Very short passages should require a higher percentage limit than longer passages. Another issue is the weighting of the percentages watched once they are above the knock out limit. A linear weighting is not ideal, since the user tends to select a passage starting a little bit before and ending a little bit after the actual section he or she is interested in. The beginning and end are thus not as important as the middle section. The can be considered by using a Gaussian function over the total passage length and correlating it with the part a user has seen.

- **Has the user written comments about this passage or a nearby one?**
  A user who has written comments is very likely to have good knowledge about the passage.
- Has the user already answered questions about this passage or a nearby one?

A user who has already answered questions about the passage is also very likely to have good knowledge about the passage.

In the current implementation, only the criteria “is the user online?” and “has the user watched the whole passage?” are taken into account. A message is send to only those users who have watched the whole passage in question and are online.

7. PERSPECTIVES FOR FUTURE WORK

This section discusses a number of possible alternative solutions and extensions that are not implemented in the current prototype.

Asynchronous Interactions

In case a private message system exists that allows for sending persistent messages also users who are offline could get a message. Once they are online they could decide if they want to answer this message. The user who selected the passage would then be notified by email.

If the lecture system has many users, a forum providing topics for popular lectures might be helpful to store this valid interaction information. The forum topics based on online discussions could be automatically displayed in a preview mode when a popular lecture is being watched.

This asynchronous interaction is a planned feature for the near future. It has been already successfully implemented in previous work presented in (Fox et al., 2009) and needs to be updated to fit in the Opencast Matterhorn architecture.

Preselecting Popular Passages

If privacy protection is not an issue, a posted question does not have to be dealt with anonymously in the first place. This permits other mechanisms to be realized. A lecture video could be automatically segmented into “popular” passages. A passage is popular, when a certain number of users have watched the passage. For all popular passages the corresponding users who are online and have watched the passage could be displayed including the users’ names and profile picture. However this results in a lot of traffic between the server and the clients and requires many database queries to find out which of the users associated with popular passages are online.

In order to find popular passages, every watched passage of a lecture video must be correlated with every other watched passages of that video. Given a passage A with a starting time \( S_{tA} \) and an ending time \( E_{nA} \) and a passage B in the same lecture video with a starting time \( S_{tB} \) and an ending time \( E_{nB} \). Ideally the watched passages should be exactly the same (\( S_{tA} = S_{tB} \) und \( E_{nA} = E_{nB} \)), but also overlaps should be considered. In general bigger overlaps should be preferred. This can be achieved by representing the passages by continuous functions and correlating those functions. By using a Gaussian like function to represent a passage the fact is considered that a user usually selects a passage starting time a little bit before and the ending time a little bit after the actual section he or she is interested in.

The result of the correlation can then be used to create a ranking. At the beginning a ranking is 0. Whenever two passages are correlated, the rankings are increased by the correlation result. Some exemplary overlaps and the resulting rankings are displayed in figure 4.

Popular passages can thus be defined by setting a ranking level that has to be exceeded. This level can either be the same for all lecture videos or it can be calculated relative to the total number of views of the lecture video.
Preselecting “Online” Passages
A second possible mechanism in case of lower privacy requirements is the preselection of passages that at least one currently online user has watched. It is thus not necessary to find the most popular passages but for each passage the user having watched the passage must be found and it must be checked if this user is online.

The Importance Of The Passage End Time
Usability is most important to make this feature available for the broader Opencast Matterhorn community. Different prototype for the required presentation of user interactions have been discussed and tested. Accessibility has been a concern as well as an easy specification of the start and end time of a recording fragment from a user’s perspective. The descriptions above imply that a user question consists of a start time, question and as well an end time. This data is required for the algorithms to identify suitable users based on the interaction data in the system. However the specification of a question’s end time is optional for the user. Like mentioned before in most cases one can expect a rather unclear question exit point. In this cases the system is going to use the next possible (slide) segments start time as question boundary. If the recording does not include further segmentation a default time frame based on the recording length can be used. Further experiments are being conducted.

8. CONCLUSION
User’s privacy seems to be more important than ever before. Social networks and social applications tempt to collect all kinds of data – oftentimes it is not transparent to the users what kind of data is collected and how it will be used in the future. User feedback from interviews and comments indicate that in particular student users do prefer a strict separation between a fun or leisure usage of these networks and the tools and applications they use for learning at universities.

We believe that social technology like the awareness features presented in this paper help to build better tools for learners to engage with the material. Group and workspace awareness (see related work) as well as privacy are important in trying to build sustainable tools for the learners. Standards like Open
Social help to develop applications as independent as possible and allow control about the data that is being recorded, analyzed and published. Opencast Matterhorn is an ideal platform for building academic audio and video applications for learners. The work presented in this paper is one of the first steps to bring the social web also to this platform.

REFERENCES


