Discerning Actuality in Backstage
Comprehensible Contextual Aging

Julia Hadersberger, Alexander Pohl and François Bry

Institute for Informatics
University of Munich
http://pms.ifi.lmu.de
{hadersberger,pohl}@pms.ifi.lmu.de, bry@lmu.de

Abstract. The digital backchannel Backstage aims at supporting active and socially enriched participation in large class lectures by improving the social awareness of both lecturer and students. For this purpose, Backstage provides microblog-based communication for fast information exchange among students as well as from audience to lecturer. Rating enables students to assess relevance of backchannel messages for the lecture. Upon rating a ranking of messages can be determined and immediately presented to the lecturer. However, relevance is of temporal nature. Thus, the relevance of a message should degrade over time, a process called aging. Several aging approaches can be found in the literature. Many of them, however, rely on the physical time which only plays a minor role in assessing relevance in lecture settings. Rather, the actuality of relevance should depend on the progress of a lecture and on backchannel activity. Besides, many approaches are quite difficult in terms of comprehensibility, interpretation and handling. In this article we propose an approach to aging that is easy to understand and to handle and therefore more appropriate in the setting considered.

Keywords: Enhanced Classroom, Backchannel, Relevance, Aging

1 Introduction

Lectures with large audiences is a much-noticed appearance of modern education. In large class lectures students seldom actively participate, despite the fact that active participation is vital for learning success. Several circumstances that favor passivity are provoked by large class lectures [1]: students are often inhibited to speak in front of many peers. They are wary about interrupting the lecturer to ask because their question might only be of minor relevance to the others and thus would merely disturb the lecture; they are also afraid of appearing incompetent when asking many questions [2]. Often students have also difficulties in formulating a question or a comment, especially when dealing with a quite unknown topic. When lectures proceed at a high pace students only have little time to think about the topic and only few opportunities to ask or comment. Besides, in the lecture hall only one person can speak at a time. Whenever several group members engage in a joint discussion, moderation is necessary.

To remedy the shortcomings of large class lectures, much effort has been put in investigating the use of CMC[1] and social media for learning (e.g. [3–5]). We argue that the synchronous use of CMC in the form of a digital backchannel carefully designed for the use in lectures may help to improve the social experience in the classroom. For example, a student may assess the relevance of her question and request for social support. Exchanging on a backchannel allows her to gain confidence to raise a hand. But also the lecturer can utilize the communication on the backchannel to lower the barrier and to stay connected with the audience. The system Backstage [6, 7] is a digital backchannel specifically tailored for the use in large class lectures as part of a research project that aims at advances in both e-learning and social media. Backstage provides carefully designed microblog-based communication by which students can rapidly exchange opinions, questions and comments (cf. Section 2).

Communication on a backchannel can quickly become confusing and incomprehensible without further structuring and filtering, even when the number of participants is small. Furthermore, the relevance and quality may vary entailing the need to filter out irrelevant messages. Therefore, students may rate, i.e. approve or reject, messages. Rating plays an important role for the lecturer: because of the outstanding role and the short time spans during which she can pay attention to the backchannel while lecturing it is hardly possible for her to get a meaningful overview of the backchannel communication without the help of the audience. Rating makes possible to provide her with a top-k ranking of the relevant messages. Also, rating is important for the students because it serves as an instrument to collectively direct the lecturer's attention to what they find particularly relevant for their good reception of the lecture.

However, relevance of lecture-related messages sent during the lecture is of temporal nature. Thus ratings and rankings, for that matter, should depend on time. As the lecture proceeds, topics might change and some questions or comments might become obsolete with respect to the progress of the lecture (while staying relevant and available for discussions and exchange after the lecture). Therefore, some kind of aging is needed. That is, the importance of messages should gradually degrade over time. With aging, attention during the lecture is directed to recent and active messages. Though, determining age on the basis of the physical time does not seem to be reasonable for our purposes. Lectures usually vary in progress. For example, introductory slides might be presented at much a higher pace than a difficult mathematical proof. Aging should rather depend on a lecture-specific measure of time like the activity on the backchannel. The approaches found in the literature seem to be too involved for our needs and difficult to handle in the context of a backchannel for large class lectures. In this article we present an approach to aging that is based on the backchannel activity, and that is highly focused on ease in comprehensibility and handling. It should be noted that although our approach is specifically conceived for Backstage it

[1] Computer-Mediated Communication

might also be interesting for other microblogging platforms like Twitter[2], which is discussed in Section 5.

2 A Short Overview of Backstage

Backstage is a digital backchannel for the use in large class lectures. The central part of Backstage is CMC on the basis of microblogs akin to Twitter. Microblogs are short messages comprising only a few words. They seem to be apt for the synchronous use during lectures, since they only contain one information item and may be read and written quickly. Unlike common microblogging, Backstage requires messages to be assigned to predefined categories, e.g. Question or Answer. One rationale behind categories is to convey to the students the kind of communication sought on the backchannel. As mentioned above, messages may be rated by the students to express acceptance or rejection of a message in terms of quality and relevance for the lecture.

A major goal of Backstage is to provide communication and promote student-to-student as well as student-to-lecturer interaction conducive for learning. For this reason, Backstage guides the user's interactions [8]. To provide for context on Backstage the presentation slides are integrated into the users' dashboards (cf. Figure 1).

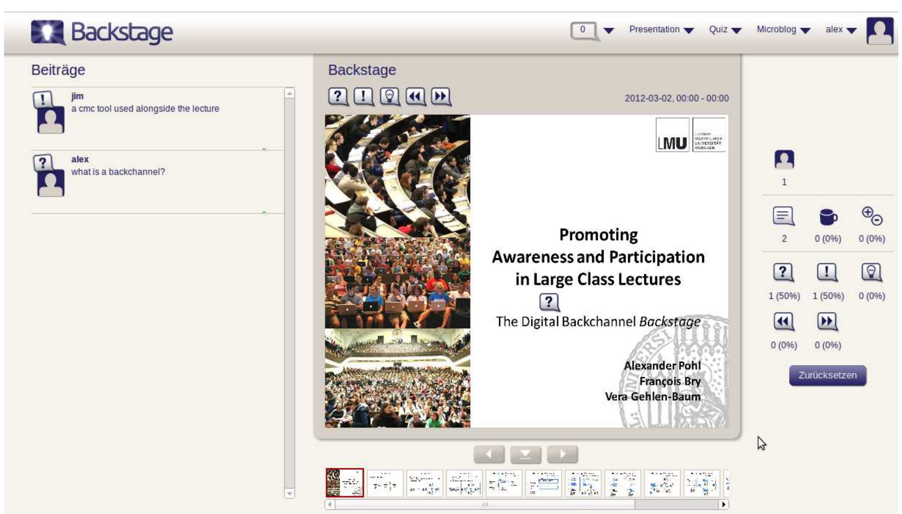

Fig. 1. The lecturer's dashboard on Backstage: the message stream is shown at the left-hand side. The slides are displayed at the center with the categories of messages on top. At the right-hand side the aggregated topic overview is displayed.

[2] http://www.twitter.com

To align the backchannel communication with the slides the creation of a message is a well thought process simple and intuitive to perform that, in a manner, is inspired from scripts [9]. It is realized by an iconic drag-and-drop onto the slides to direct users to messages profitable for learning: to write a message the student has to be aware of what she wants to say (both in terms of category and content), and to which part of the slide the message refers to (cf. [8]). That is, on Backstage messages annotate slides, which is also referred to as explicit referencing [10]. As already mentioned, messages on Backstage can also be read, and possibly answered, both by students and by the lecturer at any time after the lecture.

Backstage also provides means to improve the lecturer's awareness. Since due to the script-based user interface every message is necessarily assigned to some predefined category (e.g. Question, Answer, Remark, Too Fast) an aggregated overview showing the distribution of the communication to the categories can be given. For example, such an overview makes possible for the lecturer to quickly become aware during the lecture of many students getting lost, which presumably results in a notable increase in Question- and Too Fast-messages. Besides a topic-related overview, a top-k ranking of messages can be generated showing the k messages that the audience finds particularly relevant. Such a ranking is based on the ratings of students. Thus, rating allows students to direct the lecturer's attention to what that they find relevant. Both kinds of overview, the distribution of messages to the categories and a content-related overview by a top-k ranking supports the lecturer in staying attached to the backchannel.

To support active participation, Backstage allows the conduct of quizzes that are reminiscent to audience response systems (e.g. [11, 12]). Recently, audience response systems have gained much attention. They not only allow to playfully assess students' retention but also help to structure the lecture and activate students at a regular basis. When a quiz is conducted on Backstage, students can only answer the quiz; other functionalities are disabled. After the quiz is finished the results are integrated as ordinary slides that can be annotated and viewed. That is, quizzes can be used for introducing some gamification into the lecture thus providing a kind of break and sustaining the students' attention.

3 Related Work

Prior to presenting our approach to ranking with aging it is reasonable to provide the reader with a short overview of the field. As mentioned above we want to determine a ranking of messages upon the students' ratings. What is understood as rating and ranking is in many cases not so clear, however. Making matters worse, rating and ranking often occur interleaved, since rankings are frequently determined on the basis of ratings. Though, we distinguish between rating and ranking as follows: rating refers to the process of assigning some concrete value to a single message, e.g. "plus" and "minus" or "approve" and "reject". Ranking, in turn, relates two or more messages to each other, thereby specifying a relative (strict) order, for example pairwise comparison of the form "Message A is more relevant than message B".

3.1 Rating

Rating has been applied in various situations. In the Internet it is especially known for its use on commercial websites (rating products or sellers) and in Web 2.0 applications, to get feedback and find high-quality [13, 14]. Basically rating schemes can be distinguished in two main groups, namely explicit and implicit rating.

The first group – explicit rating – comprises all algorithms that necessitate an intentional vote of a user, which means she is conscious of her evaluation [15, 16]. This kind of obvious rating forces the user to actively think about her judgment, but this can also be seen as an effort so that the user might get discouraged if there is no kind of reward for it [17]. The simplest solution for explicit rating is solely giving users the possibility to "like" an item by voting for it [16], maybe even on a five-star rating scale. Normalization is often used to keep the score within a certain range. The downside of normalization is that the reliability of the average score is not apparent to the users. For example, an item with an average rating of two of five stars voted by only one person does not seem as bad as an item with the same average rating voted by, say, twenty people. However, the opinion of the larger group seems to be more reliable. Another explicit form of rating scheme gives the possibility to not only vote positively for an item, but also negatively or even express neutrality [18–21]. Negative ratings are sometimes desired to give users the possibility to "punish" inappropriate items or behavior. Disadvantageously, calculations with negative values can become complicated, chances are that positive and negative values cancel each other out. As a result, no received votes and a balanced average of votes might be observed as the same overall score.

The second group – implicit rating – extracts rating information from non-rating interactions or data that, however, is interpreted as votes. The user is often unconscious about her influence, since rating happens in the course of using the application [15, 16]. Implicit rating helps to overcome data sparsity, since the user does not need to be motivated to particularly provide for ratings. Different kind of interactions depending on the context can be chosen as a source of rating. Clicks on links or items can be seen as interest and positive feedback, but there could also be "misclicks" which are then misinterpreted [18, 13]. Other interactions might be more reliable, like adding an item to someone's favorites, printing or buying an item or even measuring the time that was spent on an item [17]. Furthermore, answering a question on a discussion board can also be considered as interest in an item and thus, as a positive vote.

Although implicit rating seems to be more complex to handle, since much data has to be analyzed and stored, the retrieval of more reliable data collection in a more timely fashion is possible. On the other hand, explicit rating is the only way to force the user to really consciously form an opinion about an item.

3.2 Ranking

Ranking can be found in various situations, for example online for listing the best game players or ordering search results. For Backstage a ranking is needed that melts the opinions of the users into one single ranking. Basically, there are two different ways to get a collective ranking: aggregating individual ratings or aggregating individual rankings. Furthermore, the collective ranking can be split in two groups, namely non-parametric and parametric solutions.

Non-parametric solutions do not rely on any externally set parameters or weights. The first way, getting a collective ranking by aggregating individual ratings, comprises some simple mathematical solutions that were already mentioned in Section 3.1, like summing up values or calculating the arithmetic mean. As already mentioned, these basic solutions entail different disadvantages, for example positive and negative values cancel each other out. Furthermore, the arithmetic mean makes it easier for new items to get a better overall rating than older ones, since an item can only receive the highest positive overall rating if every rating was that high [14]. More complex ideas entail more complex problems, like finding experts in question-answer-portals. Although it seems to be a good idea to count the number of people a user has already helped, the problem remains that it is not known if she only answered to lay people or other experts [22]. To get individual rankings that can be aggregated to a collective ranking users can be asked to directly order the items according to their opinion. As it is very challenging for users to order many items, comparison based methods are frequently used. Therefore, two ore more items are shown to the user, who has to decide which one she prefers. Repetitive comparison of the winning item against the other alternatives until no items are left result in an individual ranking. This can also be done implicitly, for example while browsing a website with several links on it choosing one link can be interpreted as preference for the clicked link over the other ones [15]. The so-called Hasse method [23] offers the opportunity to create a ranking of items by directly comparing their two or more properties. Disadvantageously items could be incomparable if they are not better or worse in all properties. Hence, the Hasse method might result only in a partial order. Afterwards it can be ranked according to the average positions. The so-called Copeland Score [23] combines the idea of the Hasse method with the direct comparison of items. Like the Hasse method, items with several properties are compared against each other. The Copeland Score for each item denotes the number of wins minus the number of defeats (incomparability is equivalent to zero) while it is compared to all alternatives. Afterwards the items are ranked according to their descending Copeland Score. It has to be noticed that the Copeland Score results in a total, but not necessarily in a strict order, which means there can be two items with the same score.

Each of the above mentioned non-parametric solutions can be combined with, and influenced by, parameters and hence become a parametric solution. Setting the parameters is crucial and can influence the overall computation significantly [23]. Therefore, many experiments are required to find the right configuration. Two interesting projects, using individual ratings to get a collective ranking, shall be mentioned here. The Backchan.nl project [20] is similar to Backstage and includes a formula that combines the so-called voteFactor with the ageFactor. The voteFactor is based on the proportion of positive votes for a message and the number of votes the message received compared to all other messages. This solution is already designed for a very specific context, as it does not only reward positive items but also highly discussed ones. Another algorithm is the Real-Life-Rating [14], an extension of the so-called Bayesian Rating, which is shortly explained in Section 4. This algorithm involves the expertise of users for certain domains and the friendship between users additionally to the rating itself. The Real-Life-Rating algorithm is very elaborate, but also very specific. It seems to be adequate to rather make use of the Bayesian Rating to keep it simple. The last example shows the combination of individual ratings and rankings aggregated at the same time to get a collective ranking. The ranking algorithm for microblog search [24] is based on three different properties. First, the FollowerRank which denotes the number of followers of one user normalized by the total number of her followers and the users she follows. Second, the LengthRank which is the comparison by percentage of this message to the longest message within the search results. Finally, the URLRank is set to a positive constant if the message contains a URL, otherwise it is set to zero. Although this solution can be criticized, as containing a link or being very long does not necessarily constitute a good message, it is a very good example for the smooth transition between rating and ranking. Although all three properties seem to be a ranking due to their name, in fact the two URLRank and FollowerRank are independently set or calculated values without any comparison to other items.

3.3 Aging

In most projects aging is a negative process of losing influence as time goes. Therefore, aging is naturally expressed as some kind of weight decreasing over time and expressing a remaining relevance. The older an item is, the lower its influence on the overall score. As we will see in Section 4 the notion of the term "age" is important.

One solution is based on the half-life parameter as known from the modeling of nuclear decay processes. Therefore, a time-dependent monotonic decreasing function $f(t)$ is included in the algorithm [25], for example the exponential or logistic function. The authors define the time function as $f(t) = e^{-\lambda t}$, where λ is the decay rate $\frac{1}{T_0}$. This algorithm depends on the setting of T_0, which specifies how long it takes to reduce the weight by half. The lower T_0, the faster the decay of the weight and the lower the influence. Another algorithm concerning the freshness of items on social tagging sites [26] divides the timeline in discrete and equi-distant time intervals. The time function a^{m-s} is included into the formula, where a denotes a decay factor between zero and one. While m counts the number of all time slices up to now and s is a indexed variable from one to m, $m = s$ is the current time slice. The fresher a tagging the smaller is the exponent, and the bigger the whole factor. Fresher items have a bigger influence.

In contrast to the above mentioned algorithms, the ageFactor of the system

Backchan.nl [20] is not so clear. As the ageFactor is combined with the voteFactor by multiplication it seems obvious at first sight that the aging here is once again some kind of weight. Examples show that voteFactor and ageFactor are inconsistent with one another. Therefore, we solely focus on the ageFactor formula here. The age of a message is defined by the average age of the last five votes the message received. The authors use a constant $\tau = 10^4$ by which the average age is divided to reduce the influence of the age factor. This solution seems to be very intuitive, but it has several drawbacks. First of all, the parameter τ has to be set individually according to each context. It could happen that the ageFactor becomes larger than one if enough time goes by. The inconsistency of this algorithm lies in the fact that with increasing age of an item the ageFactor also increases. Using the ageFactor as defined in [20] with increasing age the influence of such a post is also increased instead of reduced.

4 Discerning Actuality in the Ranking of Messages

For the presentation of aging we first assume that the rating procedure is a black box that yields numeric values for the backchannel messages. According to these ratings messages are sorted in order to obtain a ranking. Various rating schemes of different complexities and requirements may be employed. For the big picture, however, we present in a few words the rating currently used in Backstage. Users may rate a message positively (approval) or negatively (rejection) only once. The overall rating $r(m)$ of a message m is calculated by the following weighted average (e.g. cf. [14]):

$$r(m) = \frac{1}{\langle \mathrm{NR} \rangle + \mathrm{nr}(m)} \left(\langle \mathrm{NR} \rangle \cdot \langle \mathrm{R} \rangle + \mathrm{nr}(m) \cdot \frac{\mathrm{pos}(m)}{\mathrm{nr}(m)} \right)$$

In the formula above $\langle \mathrm{NR} \rangle$ denotes the average number of ratings of all messages, $\mathrm{nr}(m) = \max(1, \mathrm{pos}(m) + \mathrm{neg}(m))$ denotes the total number of ratings for the message m, $\mathrm{pos}(m)$ is the number of positive ratings the message m received, $\mathrm{neg}(m)$ the negative ratings for m, and $\langle \mathrm{R} \rangle$ denotes the average rating of all messages. As can be seen, positive and negative ratings do not cancel each other out, but the negative ratings weaken the influence of the positive ratings. If the total number of ratings for a message $\mathrm{nr}(m)$ is much smaller than the average number of ratings $\langle \mathrm{NR} \rangle$ the message's rating is dominated by the average rating $\langle \mathrm{R} \rangle$, meaning that not much credit is given to the users who rated the message m. Conversely, if the number of ratings for m is greater than the average number of ratings for a message the rating for m is dominated by the users who rated it. Thus, the rating scheme is biased in as much as it favors the appraisal of the collective over that of the few. However, whether this rating scheme is appropriate for Backstage needs to be investigated in an experiment in the near future.

4.1 Measuring Time and Age in Backstage

To better reflect the progress of a lecture we propose to use the backchannel activity during the lecture to promote aging. The logical time on the backchannel advances after each n-th interaction on Backstage. Both the number n and the specification of what is considered as activity is defined by the lecturer. Activities may comprise sending of messages of certain categories and rating. Since on Backstage rating is performed by (automatically) sending messages of a special rating category, specifying activity amounts to nothing else than selecting categories. Both the number of interactions after which the time advances and the specification of activity on Backstage are very intuitive parameters that can easily be handled by the lecturer, even during a lecture.

A first idea for measuring the age of messages would be to calculate the difference between the current time and the time of creation. However, this solution is inappropriate, since messages that are regularly rated, i.e. active messages, would age at the same pace as messages which are disregarded by the audience. On Backstage, active messages shall age at a lower pace than inactive messages. Thus, it is reasonable to also consider the age of a message's ratings. Hence, aging depends on the attention a message receives: it is promoted when the focus by the audience of a message recedes. A naive approach to determining age might be to calculate the difference of the current time and the time of the most recent rating a message received. This is problematic, though. For example, imagine that many students have rated a post a long time ago, i.e. the message is actually obsolete, but one student revives the message by rating, the message would suddenly, and inexplicably, rejuvenate.

The arithmetic mean over all ratings would solve this issue. However, it is very sensitive to outliers. Many ratings at the same time would be needed to assure that the age of this message can be considered robust. To overcome these difficulties we favor the use of the median as the average age of a message. The median of a frequency distribution is the sampled value of an (artificial) instance that bisects the distribution. For a sequence $(x_1, x_2, \ldots, x_k)$ of k sampled values the median $\overline{x}$ is computed as follows:

$$\overline{x} = \begin{cases} x_{\frac{k+1}{2}} & \text{if k is odd} \\ \frac{1}{2}\left(x_{\frac{k}{2}} + x_{\frac{k}{2}+1}\right) & \text{otherwise} \end{cases}$$

The median is an interesting representative of the central tendency, since it is quite robust against outliers but likewise sensitive enough to reflect relevant changes in the data (cf. [27]). Thus, to determine the age of a message, we determine the median from the ratings' age and from the creation time of the corresponding message. Also considering the time of creation is necessary in the case that a message has not received any ratings at the beginning. Otherwise, the message would not be considered by aging.

4.2 Aging in Backstage

After each n interactions on Backstage aging is promoted and the ranking is updated. We therefore propose the procedure given as pseudo-code in Listing 1.

Algorithm 1 AgingRank: Ranking with Aging

```
Require: the number k of messages that constitute the ranking
Require: the interaction counter n
  if clockTick(n) then
    candidatePosts := getCandidatePosts()
    {promote aging}
    for all post in candidatePosts do
      updateAge(post, calculateMedianAge(post))
    end for
    rankingByAge := sortDescendingByAge(candidatePosts)
    rankingByRating := sortDescendingByRating(candidatePosts)
    {we assume lists to be 1-indexed}
    for indexAge := 1 to maxIndex(rankingByAge) do
      post := getElement(indexAge, rankingByAge)
      {get the index of the post in the ranking by rating}
      indexRating := getIndex(post, rankingByRating)
      updateScore(post, indexAge * indexRating)
    end for
    {sort the candidates by just updated score values}
    relevantPosts := sortDescendingByScore(candidatePosts)
    resolved := resolveConflicts(relevantPosts)
    result := firstElements(k, resolved)
    updateRanking(result)
  end if
```

As can be seen in the given procedure the rating score is obtained by multiplying the positions of a post in the two rankings built upon age and ratings. Since it is possible that two posts may be assigned the same rating they may share the same position in the respective ranking. The final top-k ranking is then computed by sorting the list of relevant messages according to the messages' scores. However, the given procedure may result in conflicts. For example, two messages, say, m_1 with indexRating = 2 and indexAge = 3, and m_2 with the positions conversed, that is indexRating = 3 and indexAge = 2 would receive the same score 6. Both messages m_1 and m_2 would be assigned the same position in the final ranking. Thus, conflict resolution is necessary.

We propose a simple but eligible approach to conflict resolution: we let the lecturer decide which of the conflicting messages should get higher priority. Therefore, the lecturer specifies in her profile, whether she favors a conservative ranking, i.e. older messages stay in the ranking, or a progressive ranking in which older messages are replaced by newer ones whenever possible. In case of further remaining conflicts we may eventually establish a strict order by resorting to the physical age, since the conflicting messages can then be considered equal in terms of relevance and logical age.

To determine the follow-up ranking it is not necessary to consider the entire message stream. It rather suffices to determine a set of candidates, the number of which depends on the number of interactions n by which aging is promoted.

Certainly, the messages listed in the current ranking are also candidates for the follow-up ranking. However, other messages may be candidates as well. For this purpose, consider the example timeline in Figure 2.

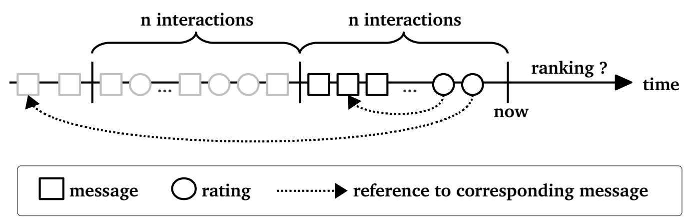

Fig. 2. Example Timeline of Interactions. The rectangles illustrate the points in time at which messages are sent, the circles illustrate the points in time at which messages are rated. The dotted arrows connect the ratings with the rated messages. The dots at the timeline indicate further interactions.

Between two ticks of the logical clock, n interactions are carried out by the users. These interactions may comprise the creation of $x \leq n$ new messages and $y = n - x$ ratings for existing messages. The ratings may refer to up to y messages created during the recent or some earlier time span. All these messages have recently been in the focus of the audience. Thus, besides the currently ranked messages, both the newly created and the newly rated messages are also candidates for the follow-up ranking. Reckoned up, the set of candidate messages comprises not more than $k + n$ messages.

5 Discerning Actuality in Twitter-based Tools

Although we conceived timely ranking by aging for the digital backchannel Backstage, it most likely might also be of interest for other microblogging platforms based on Twitter. To employ our approach it is sufficient to provide for means to determine relevance ratings, to measure activity, and to set the strategy for updating the rankings. This section illustrates possible applications in both e-learning and non-e-learning fields.

Twitter is the most prominent publicly available generic microblogging service and gained much attention not only by e-learning researchers. Twitter allows to relate microblog messages, so-called tweets, by hashtags. Thus, hashtags make possible to retrieve a coherent line of communication on a topic. Users can follow other users, i.e. become their followers. The tweets of the followed users are displayed at one's own message stream. One may forward messages of followed users to their own followers by a special form of citation, so-called re-tweets: the original message is copied and prefixed with the keyword "RT" followed by the origin user. Thus, a retweet is usually of the form "RT @*originUser* [*original text*]".

Twitter provides a rich API[3] upon which custom microblogging applications can be built. One e-learning backchannel similar to Backstage is Twitterwall[4] [28]. The platform allows the retrieval and display of multiple message streams by specifying hashtags. Furthermore it extends Twitter in that it provides rating of tweets. To extend Twitterwall with aging, the rating scheme that is already integrated can be used. Activity can be measured by the number of messages containing certain hashtags. As for each hashtag Twitterwall displays a separate message stream, it might also be interesting to provide rankings for each of those streams that underly distinct aging.

Also, discerning actuality in tweet rankings directly on Twitter can be accomplished in much the same way as is proposed for Backstage. As mentioned above, Twitter does not provide rating of tweets. However, rating of a tweet can be mimicked by counting the number of users retweeting the tweet. That is, a tweet that is frequently retweeted is heavily focused on by users and may thus be considered relevant. On Twitter, activity can be measured by the number of messages containing certain hashtags and by the number of retweets of those messages. Obtaining a timely ranking of tweets may provide interesting insights into trends in social news broadcast on Twitter.

Another quite interesting field of application might be stock microblogging, e.g. TweetTrader[5] [29]. Among other things, users of TweetTrader estimate in tweets the performance of stock quotations. Using special processable syntax, those tweets are evaluated and aggregated to determine the collective estimation of near-future stock developments. Discerning actuality in a ranking of those estimations might be of great interest for stock microblogging. Ratings in this case might be based on the content of the tweets, i.e. the users' assessments of the stock development. The activity may be specified by the number of tweets sent, for example. A progressive update strategy is likely to be preferred for a ranking in order to always be aware of the most recent estimations. Also, timely ranking of stock quotations might yield interesting outcomes in the analysis of trends.

6 Conclusion and Future Work

This article proposes an intuitive and easy-to-handle approach to discerning actuality in Backstage, a backchannel carefully designed for the use in large class lectures. We show how aging can be used to provide the lecturer with a ranking that considers actuality. The approach in this article favors the activity on the backchannel as the time measure according to which aging of messages is promoted, since the physical time only plays a minor role in determining a lecture's progress. Potential fields of applications are sketched. Since during the development of the presented approach Backstage has undergone several changes, the integration is not yet finished. Furthermore, its usefulness needs

[3] Application Programming Interface
[4] http://twitterwall.tugraz.at
[5] http://tweettrader.net

to be investigated in an experimental setting. Promoting aging also seems to be valuable for other purposes. For example, further functionalities that aim at supporting the awareness of students and lecturer and making interactions more personal and affectionate are currently under development. Some of these functionalities also depend on a sort of time and might also require aging. These topics are going to be discussed in a forthcoming paper.

References

1. van de Grift, T., Wolfman, S.A., Yasuhara, K., Anderson, R.J.: Promoting Interaction in Large Classes with a Computer-Mediated Feedback System. Proceedings of the International Conference CSCL (2003) 119 – 123
2. Schworm, S., Fischer, F.: Academic Help Seeking (In German). Handbuch Lernstrategien [Handbook of Learning Strategies] (2006) 282–293
3. Ebner, M., Schiefner, M.: Microblogging – More than Fun? In Immaculada Arnedillo, S., Isaias, P., eds.: Proceeding of IADIS Mobile Learning Conference 2008. (2008) 155–159
4. Ebner, M.: Introducing Live Microblogging: how Single Presentations can be Enhanced by the Mass. Journal of Research in Innovative Teaching **2(1)** (2009) 108–111
5. Holotescu, C., Grosseck, G.: Using Microblogging to Deliver Online Courses. Case-Study: Cirip.ro. Procedia - Social and Behavioral Sciences **1**(1) (2009) 495 – 501 World Conference on Educational Sciences, Nicosia, North Cyprus, 4-7 February 2009 - New Trends and Issues in Educational Sciences.
6. Bry, F., Gehlen-Baum, V., Pohl, A.: Promoting Awareness and Participation in Large Class Lectures: The Digital Backchannel Backstage. In: Proceedings of the IADIS International Conference e-society, Spain, Avila (March 2011) 27–34
7. Pohl, A., Gehlen-Baum, V., Bry, F.: Introducing Backstage – A Digital Backchannel for Large Class Lectures. Interactive Techology and Smart Education **8**(4) (2011) 186–200
8. Baumgart, D., Pohl, A., Gehlen-Baum, V., Bry, F.: Providing Guidance on Backstage, a Novel Digital Backchannel for Large Class Teaching. In: Education in a Technological World: Communicating Current and Emerging Research and Technological Efforts. Formatex, Spain (2012) 364–371
9. King, A.: Scripting Collaborative Learning Processes: A Cognitive Perspective. In Fischer, Frank and Kollar, Ingo and Mandl, Heinz and Haake, Jörg M., ed.: Scripting Computer-Supported Collaborative Learning. Springer US, Boston (2007) 13—37
10. Suthers, D., Xu, J.: Kükäkükä: An Online Environment for Artifact-Centered Discourse. In: Proceedings of the 11th World Wide Web Conference. (2002)
11. Golub, E.: On Audience Activities During Presentations. Journal of Computing Sciences in Colleges **20**(3) (2005) 38–46
12. Kay, R.H., LeSage, A.: Examining the Benefits and Challenges of using Audience Response Systems: A Review of the Literature. Computers & Education **53**(3) (2009) 819 – 827
13. Bian, J., Liu, Y., Agichtein, E., Zha, H.: Finding the Right Facts in the Crowd: Factoid Question Answering over Social Media. In: Proceedings of the 17th International Conference on World Wide Web. WWW '08, New York, NY, USA, ACM (2008) 467–476

14. Marmolowski, M.: Real-Life Rating Algorithm. Technical Report, DERI (2008)
15. Das Sarma, A., Das Sarma, A., Gollapudi, S., Panigrahy, R.: Ranking Mechanisms in Twitter-Like Forums. In: Proceedings of the 3rd ACM International Conference on Web Search and Data Mining. WSDM '10, New York, NY, USA, ACM (2010) 21–30
16. Lerman, K.: Advances in Web-Mining and Web Usage Analysis. In: Advances in Web Mining and Web Usage Analysis. Springer, Berlin, Heidelberg (2009) 77–96
17. Nichols, D.M.: Implicit Rating and Filtering. In Proceedings of the 5th DELOS Workshop on Filtering and Collaborative Filtering (1998) 31–36
18. Agichtein, E., Castillo, C., Donato, D., Gionis, A., Mishne, G.: Finding High-Quality Content in Social Media. In: Proceedings of the International Conference on Web Search and Web Data Mining. WSDM '08, New York, NY, USA, ACM (2008) 183–194
19. Bian, J., Liu, Y., Agichtein, E., Zha, H.: A Few Bad Votes too Many? Towards Robust Ranking in Social Media. In: Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web. AIRWeb '08, New York, NY, USA, ACM (2008) 53–60
20. Harry, D., Gutierrez, D., Green, J., Donath, J.: Backchan.nl: Integrating Backchannels with Physical Space. In: CHI '08 Extended Abstracts on Human Factors in Computing Systems, New York, NY, USA, ACM (2008) 2751–2756
21. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The Eigentrust Algorithm for Reputation Management in P2P Networks. In: Proceedings of the 12th International Conference on World Wide Web. WWW '03, New York, NY, USA, ACM (2003) 640–651
22. Zhang, J., Ackerman, M.S., Adamic, L.: Expertise Networks in Online Communities: Structure and Algorithms. In: Proceedings of the 16th International Conference on World Wide Web. WWW '07, New York, NY, USA, ACM (2007) 221–230
23. Al-Sharrah, G.: Ranking Using the Copeland Score: A Comparison with the Hasse Diagram. Journal of Chemical Information and Modeling **50**(5) (2010) 785–791
24. Nagmoti, R., Teredesai, A., De Cock, M.: Ranking Approaches for Microblog Search. In: Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01. WI-IAT '10, Washington, DC, USA, IEEE Computer Society (2010) 153–157
25. Ding, Y., Li, X.: Time Weight Collaborative Filtering. In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management. CIKM '05, New York, NY, USA, ACM (2005) 485–492
26. Huo, W., Tsotras, V.J.: Temporal Top-k Search in Social Tagging Sites using Multiple Social Networks. In: Proceedings of the 15th International Conference on Database Systems for Advanced Applications - Volume Part I. DASFAA'10, Berlin, Heidelberg, Springer-Verlag (2010) 498–504
27. Garcin, F., Faltings, B., Jurca, R.: Aggregating Reputation Feedback. In: Proceedings of the First International Conference on Reputation: Theory and Technology. Volume 1 of 1., Gargonza, Italy (2009) 62–67
28. Ebner, M.: Is Twitter a Tool for Mass-Education? In: Proccedings of the 4th International Conference on Student Mobility and ICT, Vienna (2011)
29. Sprenger, T.: TweetTrader.net: Leveraging Crowd Wisdom in a Stock Microblogging Forum. In: Proceedings of the 5th International Conference of Weblogs and Social Media, Barcelona, Spain (2011)