

Identification and Classification of the Most Important Moments in Students' Collaborative Chats

Costin-Gabriel Chiru and Remus Decea

University Politehnica from Bucharest

Splaiul Independenței 313, Bucharest, Romania

costin.chiru@cs.pub.ro, remus.decea@gmail.com

Abstract

In this paper, we present an application for the automatic identification of the important moments that might occur during students' collaborative chats. The moments are detected based on the input received from the user, who may choose to perform an analysis on the topics that interest him/her. Moreover, the application offers various types of suggestive and intuitive graphics that aid the user in identification of such moments. There are two main aspects that are considered when identifying important moments: the concepts' frequency and distribution throughout the conversation and the chat tempo, which is analyzed for identifying intensively debated concepts. By the tempo of the chat we understand the rate at which the ideas are input by the chat participants, expressed by the utterances' timestamps.

1 Introduction

Nowadays, a chat system is an important element for any web-based business. On one hand, it allows the owners to find out about customers' opinion without increasing the expenses too much. On the other, it helps boosting the sales as the customers may find immediate help to surpass the problems they may stumble upon.

The importance of chat systems was also noticed by Zuckerberg, who said that messaging is “one of the few things people do more than social networking” (Hamburger, 2014).

Considering this increasing popularity of chat, many systems were developed to support it. However, simply helping the people talk about different facts is not always enough. Thus, there are situations when the information from such systems needs to be collected, analyzed and interpreted automatically. The computer supported collaborative learning (CSCL) systems are such examples.

CSCL is a pedagogical approach where learning takes place via interaction supported by several devices through Internet. Thus, technology becomes the main means of communication and resource sharing with the goal of obtaining new knowledge or distributing it. However, in this kind of systems, the tutor is not always available to answer the students' questions at the rate and moments when they are posed. Still, the information needs to be collected and analyzed and the results should be presented to the tutor for helping him/her in managing the class (either to answer or grade the students' comments). The most problematic situation is in the case of massive open online course (MOOC) systems, where there may be thousands of students enrolled for a class and, most of the time they may be geographically spread across the globe. This means that they may be online (and thus needing guidance) at different times during the day.

The collaboration theory (Stahl, 2002) states that technology should come with new media types that support “knowledge building”. Moreover, these technologies should also help in the process of changing the role of the teacher (seen as a bottleneck element) in the learning act. In this context has arisen the need for chat analysis tools.

In this paper, we will present a chat analysis application that was meant to improve the work of Chiru and Trausan-Matu (2012) by creating a web-based application which also considers the timestamp of the students' utterances in the chat.

The paper continues with some theoretical background that the application is based on. Afterwards, we will present an overview of the existing application, along with the improvements that we added. After that, we will go through some implementation details, presenting the graphics and functionalities offered by the app. The paper will end with our conclusions and direction for furthering even more the work presented here.

2 Theoretical Background

The starting point of the work presented in this paper is represented by the concept of voice introduced in the Polyphonic Theory (Trausan-Matu, 2009; Trausan-Matu and Rebedea, 2010). A voice can be perceived as a point of view shared by one or several participants in the chat. These voices interact throughout the conversation generating relationships that are similar to the sequential, transversal or counter-point ones from polyphonic music (Bakhtin and Emerson, 1993). According to Trausan-Matu's claim (2013), following Bakhtin's perspective, a voice may range from a word or an idea to a reply in a chat or even to an entire book. Words play an essential role: "the idea resembles the word, with which it forms a dialogical unity. Like the word, the idea wants to be heard, understood and 'answered' by other voices, from other positions" (Trausan-Matu, 2013).

Bakhtin extended this model from music to discourse, stating that the voice of others is an expression of what they affirm, write or think. Thus, the knowledge is gained through the interaction with the other existing voices: "rather than speaking about 'acquisition of knowledge,' many people prefer to view learning as becoming a participant in a certain discourse" (Sfard, 2000).

Starting from this concept, Chiru and Trausan-Matu (2012) came up with the idea that the most important moments in a discourse may be identified at the interaction of different voices from that discourse. Thus, considering the type of these interactions, they classified the important moments in 4 different categories: pivotal, convergent, singular and divergent. A pivotal moment appears when, in the same utterance, one voice "fades out" while another one appears for the first time. A convergent moment takes place when two or more voices appear for the last time in the same utterance. A singular moment may be found when several voices appear for the last time except for one, that continues through the discourse. A divergent moment is detected when two or more voices meet and afterwards they continue to appear in different areas of the conversation. Our work is based on the results obtained by Chiru and Trausan-Matu (2012) and is meant to improve them by also integrating the temporality in the analysis.

Thus, another direction that is relevant for our work was undertaken by Reimann (2009) and Chiu and Khoo (2005) who considered that temporality is a key component of learning concepts

in the CSCL setting. Reimann introduced temporality analysis techniques based on sliders. Starting from their research, we investigated whether the time factor could play a role in identifying the important moments from conversations. Thus, using a slider, the developed application filters discourse as much as the user wants and points out the chat fragments that are "hotter" (where there are more utterances in a short time period).

3 Application Description

The developed application is a web app that offers visualization techniques for identifying the important moments from the students' collaborative learning chats. In this section, we will present a quick overview of the already existing application (as described in (Chiru and Trausan-Matu, 2012)) to understand its basic functionality. Afterwards, in section 4, we will detail the new features developed to extend this functionality so that to incorporate the utterances' temporality in the analysis.

3.1 Application Overview

The initial version of the application was built in Java and thus was only locally available. Its purpose was to semi-automatically identify the important moments that appeared in a collaborative learning chat, starting from the voices (important concepts) that were selected by the user. These moments were then shown to the end user in three different tabs that were related to the "voice visualization" process ("Word level", "Sentence level" and "Important moments") plus another one ("View File") where the chat was displayed along with the words selected by the user (or words belonging to the same lexical chain) which were highlighted for a better comprehension of their distribution. The "Word level" tab provides an analysis of the distribution of voices throughout the conversation, at the word level. In the "Sentence level" tab, the analysis is done at utterance level. Finally, the "Important moments" tab points out the important moments from a chat, classified as specified above. In these analyses, a dot represents either a word (as in "Word Level" tab) or an utterance (for the other two visualizations). The position of the dot is given by the relative position of that word/utterance in the whole text and the representation is similar to the position of a word in a written page (matrix representation, having a maximum number of dots per raw and continuing afterwards with the left side of the next raw).



Figure 1: Word level graphic.

4 The Improved Application

The main problems that we identified at the existing application were related to its availability (only locally available), the user's intervention ((s)he had to choose the voices of interest) and the lack of use of temporal timestamps of the utterances. Thus, these are the main issues that we addressed and are presented in this paper. The new application's purpose is to automatically identify and suggest important moments that appear in a collaborative learning chat. These moments are then presented to the end user by 2 means: "voice visualization" graphics and "hot" moments identification. The additional information is presented on two new tabs: "List of moments", that presents all the important moments from a chat that were automatically identified; and "Hot moments", which adds-in the analysis relying on the time factor.

4.1 Application Overview

In this subsection, we will present the general graphics and functionalities of the web app, that were ported from the initial application. The first graphic shows the distribution of voices at a word level. Regardless of the conversation length, this graphic offers a suggestive illustration of the distribution of concepts by mapping them in a fixed-size window (see Fig. 1). As a new feature, by clicking a word from the graphic, the user is re-directed in the "View File" tab to see the content of the chat, with the clicked word highlighted.

In Fig. 1 is depicted an example with the selection of 3 voices: one can easily see that the "blog" voice was present in the beginning of the conver-

sation, while the "chat" topic was addressed in the later part of it. Also, in the "History" fragment of the screen, the arrays of positions of the respective concepts are shown. It should be noted that the "History" area is shared by other visualizations ("Sentence Level" and "Important Moments") and thus, by adding/removing/cleaning concepts to/from any of these graphics will affect all of them.

To add a word to the graphic, the user would select that word from the vocabulary (not shown in Fig. 1) and press "Draw". To remove the last concept, the user should press "Erase", and to remove all the concepts, (s)he should press "Clean". The "Color" button changes the latest added concept's color, with a randomly generated one.

By clicking one of the dots, the screen will be re-directed to the "View Chat" tab, pointing to the position of that word's occurrence in the chat. This behavior is illustrated in Fig. 2, where the user pressed the third dot of the "blog" voice from Fig. 1. By adding this re-direction, the user may read in the context of the chat the utterance where the "blog" concept appeared (namely, at position 133, counting all the words from the chat).

The "Sentence Level" graphic is very similar with the previous one, the difference being that now the occurrences are not indexed by the position of the word itself, but by the position of the utterances they belong to.

The "Important Moments" view is a key part of the application, illustrating the most important moments from a chat. In the developed web app, the following shapes are used for illustrating them: pivotal moment - Δ ; convergence moment - \square ; singular moment - \diamond ; divergent moment - \circ . An example of such moments is provided in Fig. 3.



Figure 2: Highlight option for the “Word level” tab.

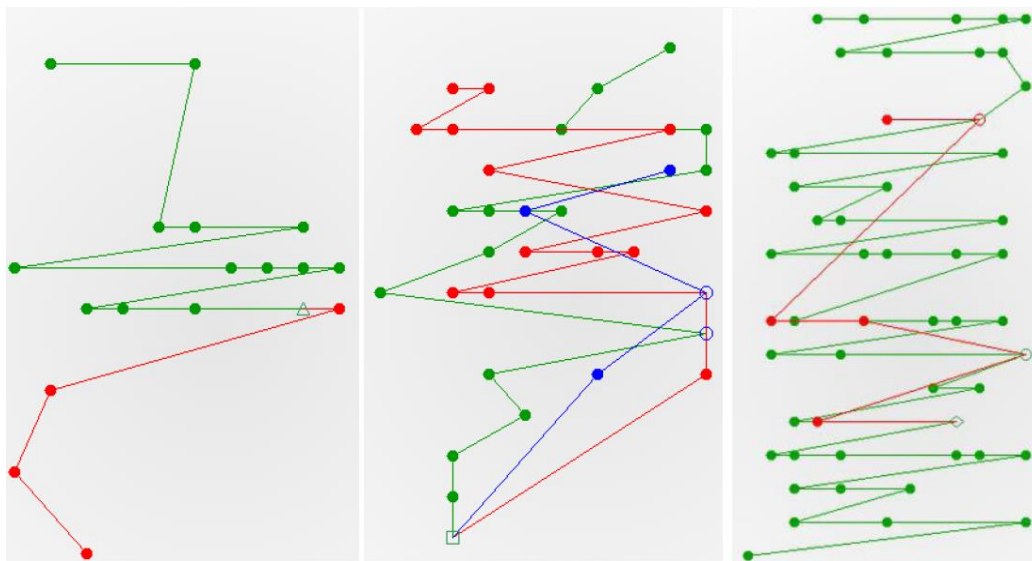


Figure 3: Examples of important moments.

4.2 List of Moments

In the initial version of the application, the user had to choose different voices to identify the important moments of the chat. This way, (s)he may oversee some of these moments, if wrong voices are chosen. Thus, we decided to add another view (“List of Moments”) to provide an automatic analysis of the voices from the chat, that doesn't depend on the user's choice. Therefore, in this tab (see Fig. 4), the user has the option to select how many voices (s)he wants to see interacting and then the app detects all the important moments that can be identified using the chosen number of voices. That is, if the user chooses n voices, all the possible combinations of n concepts from the chat are considered, and then, all the moments that are identified are classified according to the criteria developed by Chiru and Trausan-Matu (2012). This way, the user's intervention is minimized, the only thing that (s)he has to input being the cardinality of the set of concepts to be analyzed.

4.3 Hot Moments

The last important element that we added to the existing application was the analysis of the utterances timestamps, which resulted in the “Hot Moments” view. The graphic from this tab is different from the others presented so far, as it doesn't consider the distribution of voices throughout the chat, but it relies on a different factor: time. Thus, it captures the moments when the participants are intensely disputing a subject. However, it should be noted that this functionality is available only for the chats that have the utterances' timestamps.

For this visualization, the user is allowed to adjust the maximum accepted delay between utterances, which is initially computed as the chat duration divided by the number of utterances that it contains. Thus, by adjusting the delay to a lower percentage of this value, some parts of the chat are cut off, the analysis consisting only in the parts of the chat that are “hotter”. In Fig. 5 can be seen the

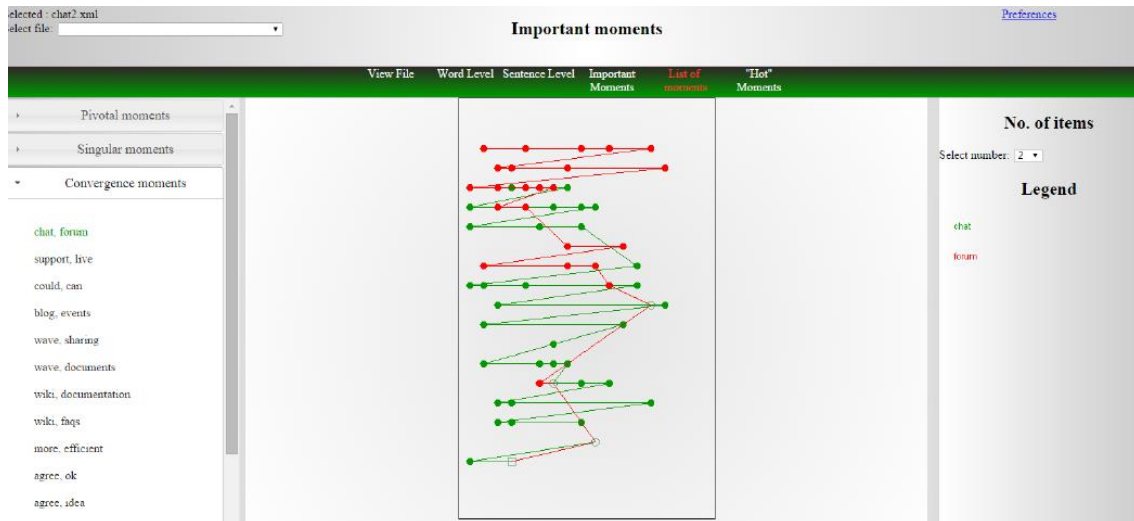


Figure 4: The List of Moments tab.

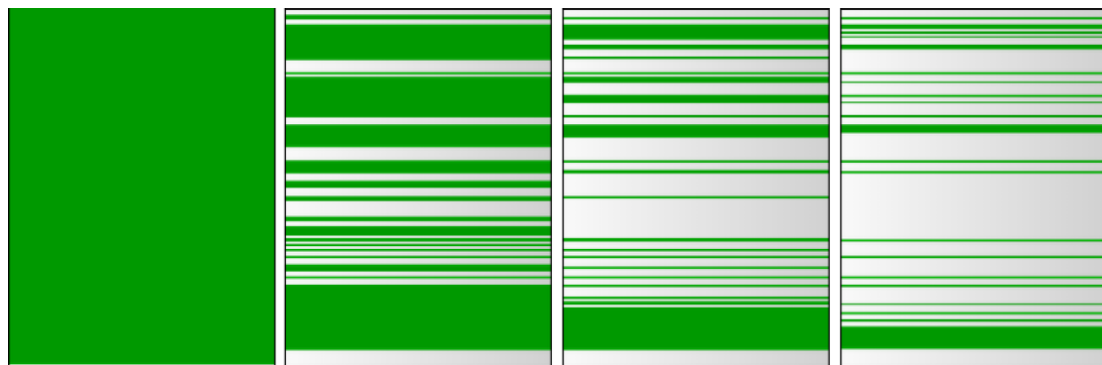


Figure 5: Hot moments sketch evolution (maximum delay is set in order from left to right to 100%, 75%, 40% and 15%).

hot moments sketch's evolution when the maximum delay is set, in order, to: 100%, 75%, 40% and 15% of the original value. The green areas represent the chat fragments where the delay is less than the chosen percentage of the maximum delay, resulting in considering only highly debated fragments (from the temporality perspective).

The final aspect that we are going to mention here is the combination of “Hot Moments” and “Sentence Level” functionalities. Using this combination, the user can view if there is a strong connection between the appearances of certain concepts and the parts from the chat where an intense dispute is present. This way, the user can observe whether a concept was more intensely disputed compared to others.

For an easier readability, we also provide an overlap percentage between the sentences where the considered concepts were present and the ones that remained after filtering the chat according to the selected maximum delay. In Fig. 6, we provide an example of this functionality. Seeing this anal-

ysis, one can draw the conclusions that the “blog” (73%) and “wiki” (64%) concepts were more disputed than the “chat” (8%) one.

5 Conclusions and Future Developments

In this paper, we presented the improved version of an application for the identification and classification of important moments from the students' collaborative chats. Compared to the initial version, two factors were considered in the analysis: the concepts distribution throughout the chat and the intensity of dialogue in various fragments.

Using the visualization graphics, the user may see the main topics that were debated and may speculate whether the participants reached agreement or not. Moreover, the “Hot Moments” functionality offers the option to view the intensively disputed parts of a chat. Furthermore, by combining the functionalities, it can be seen whether the “Hot Moments” overlap with the voices from the chat, leading to the discovery of inflammatory

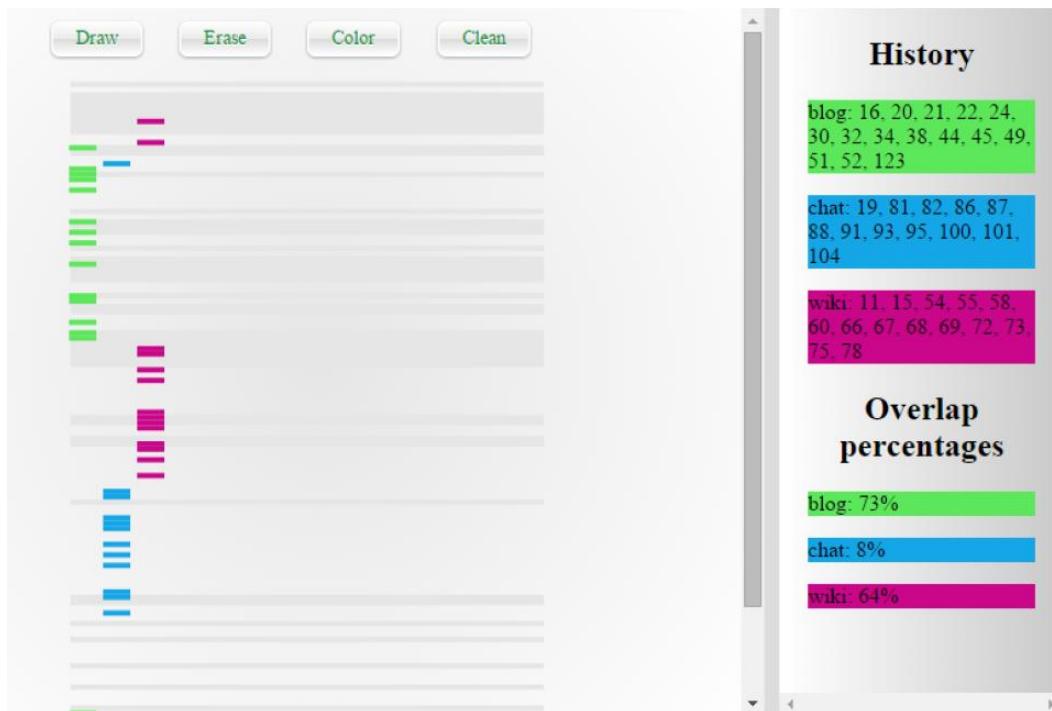


Figure 6: “Hot sentences”: overlap of “Hot Moments” and “Sentence Level” graphics.

topics, along with the participants that let themselves carried in such intense (contradictory) talks.

The application allows the user to choose the concepts to be analyzed, but also shows the moments that were automatically identified for various sets of concepts. This way the influence of the user in the analysis is minimized, thus overcoming one of the issues of the initial application.

In terms of future work, an interesting improvement could be the possibility to offer collaborative analysis tools, so that multiple persons could participate in the analysis of the same chat.

Acknowledgments

This work has been partially funded by University Politehnica of Bucharest, through the “Excellence Research Grants” Program, UPB – GEX. Identifier: UPB–EXCELENȚĂ–2016, 09/26.09.2016.

References

M Mikhail Mikhailovich Bakhtin and Caryl Emerson. 1993. *Problems of Dostoevsky’s poetics*. U of Minnesota Press.

Costin-Gabriel Chiru and Stefan Trausan-Matu. 2012. Identification and classification of the most important moments from students’ collaborative discourses. In *Proc. of ITS*, pages 330–339.

Ming Ming Chiu and Lawrence Khoo. 2005. A new method for analyzing sequential processes dynamic multilevel analysis. *Small Group Research* 36(5): 600–631.

Ellis Hamburger. 2014. Mark Zuckerberg finally explains why he forced you to download the standalone messenger app. <https://www.theverge.com/2014/11/6/7170791/mark-zuckerberg-finally-explains-why-he-forced-you-to-download-the>.

Peter Reimann. 2009. Time is precious: Variable-and event-centred approaches to process analysis in cscl research. *Int. J. of CSCL* 4(3):239–257.

Anna Sfard. 2000. On reform movement and the limits of mathematical discourse. *Mathematical thinking and learning* 2(3):157–189.

Gerry Stahl. 2002. Contributions to a theoretical framework for cscl. In *Proc. of CSCL: Foundations for a CSCL Community*, pages 62–71.

Stefan Trausan-Matu. 2009. The polyphonic model of hybrid and collaborative learning. *Handbook of research on hybrid learning models: Advanced tools, technologies, and applications*, pages 466–486.

Stefan Trausan-Matu. 2013. A polyphonic model, analysis method and computer support tools for the analysis of socially-built discourse. *Romanian J. of Information Science and Tech.* 16(2-3):144–154.

Stefan Trausan-Matu and Traian Rebedea. 2010. A polyphonic model and system for inter-animation analysis in chat conversations with multiple participants. In *Int. Conf. on Intelligent Text Processing and Computational Linguistics*, pages 354–363.