Analyzing Discourse in Collaborative Learning
Chat Conversations with Multiple Participants

Ștefan TRĂUȘAN-MATU**, Traian REBEDEA*, and Mihai DASCÂLU*

*“Politehnica” University of Bucharest, Department of Computer Science and Engineering,
**Research Institute for Artificial Intelligence of the Romanian Academy,
Email: stefan.trausan@cs.pub.ro, traian.rebedea@cs.pub.ro, mihai.dascalu@cs.pub.ro

Appeared in Tutuș, D., Corina Forăscu (eds.), Multilinguality and Interoperability in Language Processing with Emphasis on Romanian, Editura Academiei, Bucharest, pp. 313-330, 2010

Abstract: The paper introduces a discourse model for instant messenger conversations (chats) with multiple participants, based on Mikhail Bakhtin’s dialogic theory. Some chat environments for Computer-Supported Collaborative Learning (CSCL) support and encourage the existence of parallel discussion threads by providing explicit referencing facilities. The model considers such parallel discussion threads that inter-animate, sometimes in a polyphonic, counterpointal way. An implemented system is also presented, which analyzes such chat logs for detecting additional, implicit links among utterances and threads and, more important for CSCL, for detecting the involvement (inter-animation) of the participants in problem solving. The system begins with a NLP pipe and concludes with inter-animation identification in order to generate feedback and to propose grades for the learners. The system may be used for both English and Romanian languages.

Key words: Discourse analysis, conversation, chat, dialogism, polyphony, Computer-Supported Collaborative Learning.

1. Introduction

This paper presents a theoretical model of discourse in instant messenger (chat) conversations with multiple participants. The model is starting from the dialogism theory of Mikhail Bakhtin (1981, 1993) and was used for the implementation of the PolyCAFe system (Polyphonic Conversation Analysis and Feedback generation), developed under the project LTfLL (Berlanga, 2009).

Until now, the theories and the developed computer programs mainly considered conversations with two participants, transcribed telephone conversations (Allen & Core, 1997). Now, in the internet chat systems (e.g. Yahoo Messenger in a conference style or similar environments), conversations are rather written, not spoken (and afterwards transcribed). This context allows more than two users to participate to the chat conversation. However, a problem is
that multiple discussion threads may occur in parallel and, for example, if two participants write a question at a very short time interval, and a third answers, it may be very difficult to determine to whom it replied. A solution to this problem was spontaneously found: Users refer the addressing person explicitly with the “@” sign in front: “@john you’re right”. Another solution is provided in the VMT environment (Stahl, 2009, see also http://gerrystahl.net/vmt/), used in our system: For referencing, a user may click on the previous utterances. In addition, referencing is extended also to the whiteboard which is provided. The reference is explicitly displayed by the interface (see Figure 1). In addition to these explicit references, Natural Language Processing (NLP) techniques (for example, co-references, adjacency pairs and argumentation) may be used for detecting implicit links (Trăușan-Matu & Rebedea, 2010).

Written chat conversations with environments that allow explicit referencing encourage the existence of parallel threads of discussion. More important, inter-animation processes appear among these threads, following patterns similar to those of counter-point in polyphonic music (Trăușan-Matu, Stahl & Sarmiento, 2007; Trăușan-Matu & Rebedea, 2009).

Chat conversations, due to their popularity, are now used also for teaching purposes, in a new approach, Computer Supported Collaborative Learning (CSCL, see Stahl, 2006), becoming an alternative or supplement to classical learning. A typical considered case is that of small virtual groups using chat systems for learning together (Stahl, 2006; Trăușan-Matu, Stahl & Sarmiento, 2007). CSCL is a change of vision on learning replacing the idea of the transfer of knowledge from a human or a written source to the student. The new idea is that learning should empower the students to become participants in a discourse: “rather than speaking about ‘acquisition of knowledge’, many people prefer to view learning as becoming a participant in a certain discourse” (Sfard, 2000). A natural consequence is that in order to provide automatic assessment and feedback generation, the system should be able to analyze students’ discourse and therefore theories on discourse are needed.

Mikhail Bakhtin’s dialogism (Bakhtin, 1981; 1993) was proposed by Koschmann (1999) as a paradigm for CSCL, its key features being multivocality and polyphony. Wegerif (2006) also considered dialogism as a theoretical starting point which can be used for developing tools for teaching thinking skills. Moreover, he considers inter-animation as very important for the success of collaborative learning. Our approach follows exactly these ideas and is investigating how Bakhtin’s theory of polyphony and inter-animation can be used for analyzing the discourse in chat conversations with multiple participants. Moreover, several computer tools were implemented and investigated (Trăușan-Matu, Rebedea, Drăgan & Alexandru, 2007). Until now, almost no investigations and developments were performed on how these ideas could effectively be used for the analysis of CSCL dialogs and for the implementation of supporting computer tools.
The paper continues with a presentation of the polyphonic theory on discourse. The third section describes the PolyCAFe system while the fourth section presents the widgets that were developed to visualize the results generated by PolyCAFe.

2. A POLYPHONIC THEORY ON DISCOURSE

2.1. Emitter-receiver discourse analysis

Discourse may be defined as “a coherent structured group of sentences” (Jurafsky & Martin, 2007) and in NLP it is usually considered different in monologues and dialogues. However, in both cases the same idea of an emitter-receiver channel is used, the difference being the uni- respectively bi-directional communications.

In monologues, an unidirectional model of communication is considered, from a speaker to a listener (Jurafsky & Martin, 2009). One of the main ways of analyzing discourse in text is the detection of local relations and measuring coherence. Some structures are searched, as the Rhetorical Schema Theory (RST) (Mann & Thomson, 1987), which considers a hierarchical decomposition of a text. Centering Theory (Grosz et al., 1995), co-reference resolution systems (Jurafsky & Martin, 2009) are also considered.

Dialogue analysis has as prototype phone-like (or face-to-face) conversations. A typical approach starts in analysis from local and two-participant data and tries to identify speech acts, dialog acts and afterwards, adjacency pairs (Jurafsky & Martin, 2009). Even if there are attempts to analyze conversations with multiple participants, considering more global, collaboration-based perspective, like transacts (Joshi & Rose, 2007), the approach is also based on a two interlocutors’ model.

Until now, the goals of discourse analysis in existing approaches were to detect topics and links (Adams & Martell, 2008), dialog acts (Kontostathis, 2009), lexical chains (Dong, 2006) or other complex relations (Rose et al., 2007). The used techniques are TF-IDF (Adams & Martell, 2008; Schmidt & Stone), Latent Semantic Analysis (Schmidt & Stone; Dong, 2006; Manning & Schutze, 1999), Social Network Analysis (Dong, 2006), Machine Learning (for example, Naïve
Bayes (Kontostathis, 2009), Support Vector Machines and Collin’s perceptron (Joshi & Rose, 2007), and the TagHelper environment (Rose et al., 2007)). The lexical ontology WordNet (wordnet.princeton.edu) (Adams & Martell, 2008; Dong, 2006) is also used.

In our approach we start from identifying words and patterns in utterances which are indicators of links among them and, afterwards, we make an analysis based on the graph of these links (as a social network) and on threads and their interactions. In our analysis we start from the dialogism and polyphony theory of Mikhail Bakhtin, the details being introduced in the next section.

2.2. Multithreaded discourse

In phone and face-to-face dialogs only one person usually speaks at a given moment in time, generating a single thread of discussion. This is, of course, determined by the physical, acoustical constraints (if two or more persons are speaking in the same moment, it is impossible to understand something). In chat environments, like the one used in the Virtual Math Teams (VMT) project (Stahl, 2009), any number of participants may write utterances at the same time. As discussed in a previous section, the VMT environment offers also explicit referencing facilities that allow the users to indicate to what previous utterance(s) they refer to. This facility is extremely important in chat conversations with more than two participants because it allows the existence of several discussion threads in parallel. Moreover, the co-occurrence of several threads gives birth to inter-animation, a phenomenon similar to polyphony, where several voices jointly play a coherent piece as a whole (Trăușan-Matu, Stahl & Sarmiento, 2007; Trăușan-Matu & Rebedea, 2009).

Mikhail Bakhtin (1981, 1993) emphasized that polyphony occurs in any text. He considers that dialog is characterizing any text, that “all our utterances (including creative works), is filled with others’ words” (Bakhtin, 1986). The voice becomes a central concept, has a more complex meaning. A voice is not limited to the acoustic dimension, it may be considered as a particular position, which may be taken by one or more persons when emitting an utterance, which may have both explicit (like those provided by the VMT chat environment (Stahl, 2009)) and implicit links (for example, lexical chains, co-references or argumentation links) and influence other voices. Each utterance is filled with ‘overtones’ of other utterances (Bakhtin, 1993). Moreover, by the simple fact that they co-occur, voices are permanently inter-animating, entering in competition, generating multi-vocality in any conversation and even in any text (in Bakhtin’s dialogic theory everything is a dialog) or, as Bakhtin calls it, a “heteroglossia, which grows as long as language is alive” (1981).

The ideas of Bakhtin are driving to a musical metaphor for discourse and for learning: “the voices of others become woven into what we say, write, and think” (Koschmann, 1999). Therefore, for analyzing discourse in chats we should investigate how voices are woven, how themes and voices inter-animate in a polyphonic way (Trăușan-Matu, Stahl & Sarmiento, 2007). This is important not only for understanding how meaning is created but also for trying to design tools for support and evaluation.

As an example, let us consider the chat, from which an excerpt is presented in Figure 2. The threads generated by repeating words (‘interact’, ‘polyphony’ and ‘voice’) become voices, which inter-animate (see the thin curly arrows in Figure 2).
Analysis of Discourse in Collaborative Learning Chat Conversations with Multiple Participants

Fig. 2 Inter-animating threads

In our implemented solution, we focus on the idea of identifying voices in the analysis of discourse in chats, which are, in our view, important utterances with an important impact on the conversation as a whole. For this aim, we analyze the links (adjacency pairs, repetitions, lexical chains and argumentation links) between utterances, we construct a graph, we identify threads and we compute for each utterance a value reflecting its importance, given not only by its content but also based on the ‘echoes’ of other previous utterances present in it and on the echoes generated by it and detected in other further utterances.

3. AUTOMATIC ANALYSIS OF DISCOURSE IN CHATS WITH MULTIPLE PARTICIPANTS

3.1. The architecture and functionality of the PolyCAFe system

The PolyCAFe system is based on the research started by the first author in 2005, during his joint work with the VMT project (Stahl, 2009). The idea of considering Bakhtin’s vision for chats in Computer-Supported Collaborative Learning (Trăușan-Matu, Stahl & Sarmiento, 2007) was the basis for the design and implementation of a first analysis system, using text mining techniques and social network analysis (Trăușan-Matu, Rebedea, Drăgan & Alexandru, 2007). Meanwhile, another analysis system, based also on Bakhtin ideas and using also Social Network Analysis and, additionally, Latent Semantic Analysis was developed (Dascălu et al., 2008; 2010). PolyCAFe integrated the ideas and experience of these previous systems.
PolyCAFe is one of the modules of the system developed in the LTfLL project. It is available as a series of web widgets, which may be accessed from different platforms that allow the integration of widgets, like Moodle. PolyCAFe analyses chats in which each participant is associated to a given topic (for example, students that should debate in an assignment, each of them having to support one topic). PolyCAFe generates a series of feedback and assessment information regarding both the content (if the students have discussed what was expected) and if they participated to the building of discourse. An interactive graphical representation of the discussion and of the explicit and implicit links is also provided. Users may select to see discourse threads by clicking on utterances or by indicating words in that threads (see section 4).

The architecture of PolyCAFe is presented in Figure 3. The NLP pipe is needed for pre-processing the text log for the content analyzer and implicit link identification. The inter-animation analyzer is processing the threads in the chat, which are built upon the explicit and implicit links in the conversation.

3.2. The NLP pipe and the XML schema

The NLP pipe receives as input a chat log or the text of a discussion forum, in the XML format presented below in Figure 4 and has the following component modules:

- Spelling correction;
- Tokenizer;
- Named Entity Recognizer. It uses a gazetteer which has to be loaded with specific names for the considered teaching domain;
- Stemmer (Lemmatizer);
- POS tagger and parser;
- NP-chunker.

In the first version, the modules of the NLP pipe were those provided by the Stanford NLP software (http://nlp.stanford.edu/software), with the exception of the spellchecker (implemented with Jazzy, see http://jazzy.sourceforge.net/ and http://www.ibm.com/developerworks/java/library/j-jazzy/).
Two alternative NLP pipes are under experimentation, integrating modules from GATE (http://gate.ac.uk) and LingPipe (http://alias-i.com/lingpipe/).

The first version of the NLP pipe included only modules for the English language. However, the design and implementation were performed with the facility of considering other languages as well. The second version of the system includes modules for the Romanian language.

```
<?xml version="1.0" encoding="UTF-8"?>
<Dialog time="2005-01-11 09:26:11" description="this is an assignment for the NLP course" file="chat_input_1.xml" id="Social networks13_6_200610_57_10" language="en|fr|ro"
  name="chat-12-A" subject="about pragmatics" team="12">
  <Participants>
    <Person nickname="Alex" realname="Bibi Ionescu"/>
    <Person nickname="Adrian"/>
    <Person nickname="vvalceia" realname=""/>
  </Participants>
  <Topics>
    <Itemset description="NLP - pragmatics">
      <Item description="cf. Grice's theory">implicature</Item>
    </Itemset>
    <Itemset>
      <Item>
        <Utterance genid="1" ref="0" time="2005-01-11 09:26:03">hello all</Utterance>
      </Item>
      <Turn>
        <Utterance genid="2" ref="0" time="2005-01-11 09:27:18">hi</Utterance>
      </Turn>
      <Turn>
        <Utterance genid="3" ref="1" time="2005-01-11 09:29:29">Hello Alex</Utterance>
      </Turn>
    </Itemset>
  </Topics>
</Dialog>
```

**Fig. 4** The XML schema

An XML schema was designed for encoding chat conversations and discussion forums (see Figure 4). Each utterance has a unique identifier (‘genid’) and the existing explicit references (‘ref’) to previous utterances, which were specified by the participants using the facility provided by the VMT environment. In addition to annotating the elements of a chat, the schema also includes at the end annotations generated by the system or added using a manual annotation tool (Dascălu et al., 2008).

In order to ensure flexibility, the input data may be in different formats besides the above XML schema, for example, saved chats from Yahoo Messenger in text format, other text format chats and VMT format. A pre-processing module transforms these formats into our XML schema.

### 3.3. Content analysis

The content analysis module identifies the main concepts of the chat or forum using the NLP pipe, cue-phrases and graph algorithms. It also identifies speech acts and argumentation types of utterances (as in Toulmin’s theory: Warrant, Concession, Rebuttal and Qualifiers (Toulmin, 1958)).

The first step in finding the chat subjects is to strip the text of irrelevant words (stop-words), text emoticons (like “:)” or “:P”) special abbreviations used while chatting (e.g., “brb,” “np” and “thx”) and other words considered irrelevant at this stage. The next step is the tokenization of the chat text. Recurrent tokens and their synonyms are considered as candidate concepts in the analysis. Synonyms are retrieved from the WordNet lexical ontology (http://wordnet.princeton.edu). If a concept is not found on WordNet, mistypes are searched. If successful, the synonyms of the suggested word will be retrieved.
The last stage for identifying the chat topics consists of a unification of the candidate concepts discovered in the chat. This is done by using the synonym list for every concept: if a concept in the chat appears in the list of synonyms of another concept, then the two concepts’ synonym lists are joined. At this point, the frequency of the resulting concept is the added frequencies of the two unified concepts. This process continues until there are no more concepts to be unified. At this point, the list of resulting concepts is taken as the list of topics for the chat conversation, ordered by their frequency.

In addition to the above method, used for determining the chat topics, there is an alternative technique we used to infer them by using a surface analysis technique of the conversation. Observing that new topics are generally introduced into a conversation using some standard expressions such as “let’s talk about email” or “what about wikis,” we have constructed a simple and efficient method for deducing the topics in a conversation by searching patterns containing specific cue phrases.

The topics of the chat may also be detected starting from the connected components in the interaction graph constructed from the explicit and implicit links described in the next section.

Speech acts were introduced by Austin and then elaborated by Searle and others (Jurafsky & Martin, 2009). They are classifications of utterances according to the action they fulfil. The list of speech acts considered by the system is derived from DAMSL (Allen & Core, 1997):

<table>
<thead>
<tr>
<th>Statement</th>
<th>Conventional</th>
<th>Maybe</th>
<th>Greeting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info Request</td>
<td>Agreement</td>
<td>Answer</td>
<td>Noise</td>
</tr>
<tr>
<td>Declarative Question</td>
<td>Accept</td>
<td>Answer</td>
<td>Continuation</td>
</tr>
<tr>
<td>Wh-Question</td>
<td>Reject</td>
<td>Thanks</td>
<td>Exclamation</td>
</tr>
<tr>
<td>Action</td>
<td>Partial Accept</td>
<td>Sorry</td>
<td></td>
</tr>
<tr>
<td>Directive</td>
<td>Partial Reject</td>
<td>Opinion</td>
<td></td>
</tr>
</tbody>
</table>

### 3.4. Implicit links identification

In addition to explicit links, stated in chats by the referencing facility of the VMT environment and in forums by the ‘reply to’ link, implicit links are also identified. The advanced NLP and basic discourse analysis sub-layer uses the results of the previous two sub-layers to identify various types of implicit links:

- Repetitions (of ordinary words or named entities);
- Lexical chains, which identify relations among the words in the same post or utterance or in different ones, by using semantic similarities (the semantic sub-layer);
- Adjacency pairs (Jurafsky & Martin, 2009) – pairs of specific speech acts – answers to a single question in a limited window of time (in which the echo of the “voice” of the question remains), greeting-greeting, etc.;
- Co-references.

Implicit links, with the exception of lexical chains and co-references (detected with the BART system, see http://bart-coref.org/) are detected using a cue phrase identification system (the PatternSearch module of PolyCAFe system developed in the LTfLL project, see Trăuşan et al., 2009) and Latent Semantic Analysis. The “PatternSearch” module was implemented for searching occurrences that match expressions specified by the user, in a chat log. The module, in addition to a simple regular expression search, allows considering not only words, but also synonyms, words’ stems and their corresponding part of speech (POS). Another novel facility is the consideration of
utterances as a search unit, for example, specifying that a word should be searched in the previous \( n \) utterances and that two expressions should be in two distinct utterances. The search is made at utterance level. The program checks the utterances one by one and if there is a match between a part of the utterance and the searched expression, both the utterance and the specific text that matched are indicated.

For example, the expression \(<S "convergence">\) \([\ast]\) cube searches pairs of utterances that have a synonym of “convergence” in the first utterance and “cube” in the second. One result from a particular chat is the pair of utterances 1103 and 1107:

\[
1103 \# 1107. \quad \text{overlap} \# \text{cube [that would still have to account for the overlap that way]} \# \quad \text{[an idea: Each cube is assigned to 3 edges. Then add the edges on the diagonalish face.]}\]

### 3.5. Discourse analysis for determining the involvement in a chat

Communication between participants in a chat is conveyed through language in written form. Lexical, syntactic, and semantic information are basic levels used to describe the features of written utterances (Page & Paulus, 1968), and are taken into account for the analysis of a participant’s discourse, as a measure of his involvement in a chat. First, surface metrics are computed for all the utterances of a participant in order to determine factors like fluency, spelling, diction or utterance structure (Page & Paulus, 1968). All these factors are combined and a mark is obtained for each participant without taking into consideration a lexical or a semantic analysis of what they are actually discussing. At the same level readability ease measures are computed.

The next step is grammatical and morphological analysis based on spellchecking, stemming, tokenization and part of speech tagging. Eventually, a semantic evaluation is performed using LSA (Landauer et al., 1998). For assessing the on-topic grade of each utterance a set of predefined keywords for all corpus chats is taken into consideration.

Moreover, at the surface and at the semantic levels, metrics specific to social networks are applied for proper assessment of participants’ involvement and similarities with the overall chat and predefined topics of the discussion.

In order to perform a detailed surface analysis two categories of factors are taken into consideration at a lexical level: Page’s essay grading proxes and readability. Page’s idea was that computers could be used to automatically evaluate and grade student essays as effective as any human teacher using only simple measures – statistically and easily detectable attributes (Page & Paulus, 1968). The main purpose was to prove that computers could grade as well, but with less effort and time, therefore enabling teachers to assign more writing. So the goal was to improve the student’s capabilities by practice, having at hand the statistical capabilities of computers for writing analysis.

### 3.6. Social networks analysis

In addition to quantity and quality measures computed starting from utterances, social factors are also taken into account in our approach. Consequently, a graph is generated from the chat transcript in concordance with the utterances exchanged by the participants. Nodes are participants in a collaborative environment and the edges are generated based on explicit (obtained from the explicit referencing facility of the chat environment used (Holmer et al., 2006), which enables participants to manually add links during the conversation for marking subsequent utterances derived from a specific one) and implicit links.

From the point of view of social networks, various metrics are computed in order to determine the force of the voices of participants in chat: degree (indegree, outdegree), centrality (closeness
centrality, graph centrality, eigen–values) and user ranking similar to the well known Google Page Rank Algorithm (Dascălu et al., 2008). These metrics are applied first on the effective number of interchanged utterances between participants providing a quantitative approach. Second, the metrics are applied to the sum of utterance marks based on a semantic evaluation of each utterance. Starting from the results obtained for each utterance, a new graph is built on which all social metrics are applied. This provides the basis for a qualitative evaluation of the chat.

All the identified metrics used in the social network analysis are relative in the sense they provide markings relevant only compared with other participants in the same chat, not with those from other chats. This is the main reason why all factors are scaled between all the participants, giving each participant a weighted percentage from the overall performance of all participants.

3.7. LSA and the corresponding learning process

Latent Semantic Analysis (LSA) is a technique based on the vector-space based model (Wiemer-Hastings & Zipitria, 2001; Miller, 2004). It is used for analyzing relationships between a set of documents and terms contained within, by projecting them in sets of concepts related to those documents (Landauer et al., 1998). LSA starts from a term-document array which describes the occurrence of each term in all corpus documents. LSA transforms the occurrence matrix into a relation between terms and concepts, and a relation between those concepts and the corresponding documents. Thus, the terms and the documents are now indirectly related through concepts (Landauer et al., 1998; Manning & Schütze, 1999). This transformation is obtained by a singular-value decomposition of the matrix and a reduction of its dimensionality.

Our system uses words from a chat corpus. The first step in the learning process, after spell–checking, is stop words elimination from each utterance. The next step is POS tagging and, in case of verbs, these are stemmed in order to decrease the number of corresponding forms identified in chats by keeping track of only the verb’s stem (the meaning of all forms is actually the same, but in LSA only one form is learnt). All other words are left in their identified forms, adding corresponding tagging because same words, but with different POS tags have other contextual senses, and therefore semantic neighbours (Wiemer-Hastings & Zipitria, 2001).

Once the term-document matrix is populated, Tf–Idf (term frequency - inverse document frequency (Manning & Schütze, 1999)) is computed. The final steps are the singular value decomposition (SVD) and the projection of the array in order to reduce its dimensions. According to (Lemaire, 2008), the optimal empiric value for k is 300, a value used in current experiments at which multiple sources concord.

Another important aspect in the LSA learning process is segmentation which is the process of dividing chats taking into consideration units with similar meaning and high internal cohesion. In the current implementation, the chat is divided between participants because of the considered unity and cohesion between utterances from the same participant. These documents are afterwards divided into segments using fixed non-overlapping windows. In this case contiguous segments are less effective because of intertwined themes present in chats and these aspects will be dealt with in future improvements of the marking system.

LSA is used for evaluating the proximity between two words by using the cosine measure:

$$\text{Sim}(\text{word}_1, \text{word}_2) = \frac{\sum_{i=1}^{k} \text{word}_{1,i} \cdot \text{word}_{2,i}}{\sqrt{\sum_{i=1}^{k} \text{word}_{1,i}^2} \times \sqrt{\sum_{i=1}^{k} \text{word}_{2,i}^2}}.$$  \hspace{1cm} (1)

Similarities between utterances and similarities of utterances related with the entire document are used in order to assess the importance of each utterance compared with the entire chat or with a predefined set of keywords referenced as a new document:
Vector(utterance) = \sum_{i=1}^{n}(1 + \log(no\_occurrence(word_i)) \cdot \text{vector}(word_i).

Sim(utterance_1, utterance_2) = Sim(Vector(utterance_1), \text{Vector}(utterance_2)).

3.8. The utterance and participants’ evaluation process

The first aspect that needs to be taken care of is building the graph of utterances which highlights the correlations between utterances on the basis of explicit references. In order to evaluate each sentence, after finishing the morphological and lexical analysis three steps are processed:

1. Evaluate each utterance individually taking into consideration several features: the effective length of initial utterance; the number of occurrences of all keywords which remain after eliminating stop words, spell-checking and stemming; the level at which the current utterance is situated in the overall thread (similar to a breadth-first search in the utterance space/threads based only on explicit links); the branching factor corresponding with the actual number of derived utterances from current one; the correlation / similarity with the overall chat; the correlation / similarity with a set of predefined set of topics of discussion.

This mark combines the quantitative approach (the length of the sentence starting from the assumption that a piece of information should be more valuable if transmitted in multiple messages, linked together, and expressed in more words, not only to impress, but also meaningful in the context) with a qualitative one (the use of LSA and keywords).

In the process of evaluating each utterance, the semantic value is computed with the help of the likelihood between the terms used in the current utterance (those after preliminary processing) and the whole document, respectively those from a list of predefined topics of discussion.

The formulas used for evaluating each utterance are:

\[
\text{mark}_{\text{empirical}} = \frac{\text{length(initial \_ utterance)}}{10} + 9 \times \sum_{\text{word}} \text{mark}(\text{word}) \times \text{emphasis}.
\]

\[
\text{mark}(\text{word}) = \text{length(utterance)} \times (1 + \log(\text{no\_occurrences})) .
\]

\[
\text{emphasis} = (1 + \log(\text{level})) \times (1 + \log(\text{branching \_ factor}) \times
\times \text{Sim(utterance, whole \_ document}) \times
\times \text{Sim(utterance, predefined \_ keywords)}.
\]

2. Emphasize Utterance Marks. Each thread obtained by chaining utterances based upon explicit links has a global maximum around which all utterance marks are increased correspondingly with a Gaussian distribution:

\[
p(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \text{ where:}
\]

\[
\sigma = \frac{\max(\text{id \_ utter \_ thread}) - \min(\text{id \_ utter \_ thread})}{2};
\]

\[
\mu = \text{id \_ utterance \_ with \_ highest \_ mark}.
\]

Therefore each utterance mark is multiplied by a factor of 1 + p(current_utterance).

3. Determine the final grade for each utterance in the current thread. Based upon the empiric mark, the final mark of the utterance is obtained for each utterance in its corresponding thread:

\[
\text{mark}_{\text{final}} = \text{mark}_{\text{final}}(\text{prev \_ utter}) + \text{coefficient} \times \text{mark}_{\text{empirical}},
\]

where the coefficient is determined from the type of the current utterance and the one to which it is linked to.
For the coefficient determination, the identification of speech acts plays an important role: verbs, punctuation signs and certain keywords are inspected. Starting from a set of predefined types of speech acts, the coefficients are obtained from a predefined matrix. These predefined values were determined after analyzing and estimating the impact of the current utterance considering only the previous one in the thread (similar to a Markov chain). The grade of a discussion thread may be raised or lowered by each utterance. Therefore, depending on the type of an utterance and the identified speech acts, the final mark might have a positive or negative value.

The in-degree, out-degree, closeness and graph centrality, eigen-values and rank factors, computed from the Social Network Analysis are applied on the matrix with the number of interchanged utterances between participants and the matrix which takes into consideration the empiric mark of an utterance instead of the default value of 1. Therefore, in the second approach quality, not quantity is important (an element \([i, j]\) equals the sum of mark_{empiric} for each utterance from participant \(i\) to participant \(j\)), providing a deeper analysis of chats using a social network’s approach based on a semantic utterance evaluation.

Each of the analysis factors (applied on both matrixes) is converted to a percentage (current grade/sum of all grades for each factor, except the case of eigen centrality where the conversion is made automatically by multiplying with 100 the corresponding eigen–value in absolute value). The final grade takes into consideration all these factors (including those from the surface analysis) and their corresponding weights:

\[
\text{final grade}_i = \sum_k \text{weight}_k \times \text{percentage}_{k,i},
\]

where \(k\) is a factor used in the final evaluation of the participant \(i\) and the weight of each factor is read from a configuration file.

After all measures are computed and using the grades from human evaluators, the Pearson correlation for each factor is determined, providing the means to assess the importance and the relevance compared with the manual grades taken as reference.

General information about the chat – for example overall grade correlation, absolute and relative correctness – are also determined and displayed by the system.

### 3.9. Optimization of each metric’s grading factor

The scope of the designed algorithm is to determine the optimal weights for each given factor in order to have the highest correlation with the manual annotator grades. A series of constraints had to be applied. First, minimal/maximum values for each weight are considered. For example, a minimum of 2% in order to take into consideration at least a small part of each factor, and maximum 40% in order to give all factors a chance and not simply obtain a solution with all factors 0% besides the one with the best overall correlation – 100%. Second, the sum of all factors must be 100%. Third, the goal is to obtain maximum mean correlation for all chats in the corpus.

In this case, the system has two components. A perceptron is used for obtaining fast solutions as inputs for the genetic algorithm. The main advantages for using this kind of network are the capacity to learn and adapt from examples, the fast convergence, the numerical stability; search in the weight space for optimal solution; duality and correlation between inputs and weights.

Secondly, a genetic algorithm is used for fine-tuning the solutions given by the neural network, also keeping in mind the predefined constraints. This algorithm operates over a population of chromosomes which represent potential solutions. Each generation represents and approximation of the solution - the determination of optimal weights in order to assure the best overall correlation, not the best distance between automatic grades and annotator ones. Correlation is expressed as an arithmetic mean of all correlations per chat because of the differences between evaluator styles.
The scope of this algorithm is to maximize the overall correlation, and specific characteristics of the implemented algorithm are:

- Initialization: 2/3 of initial population obtained via Neural Networks (perceptron), the rest is randomly generated in order to avoid local;
- Fixed number of 100 chromosomes per population;
- Fitness - overall correlation of all chats from the corpus evaluated as a mean of all individual correlations;
- Selection – roulette based or elitist selection - the higher the fitness, the greater the possibility a participant is selected for crossover;
- Correction – a necessary operator in order to assure that the initial constraint are satisfied: if above or below minim/maximum values, reinitialize weight starting from threshold and adding a random quantity to it; if overall sum of percentages different from 100% adjust randomly weights with steps of 1/precision;
- Crossover - is based on Real Intermediate Recombination which has the highest dispersion of newly generated weights - select a random alpha for each factor between [-0.25; 1.25]; the relative distance between 2 chromosomes selected for crossover must be at least 20% in order to apply the operator over them;
- Use CHC optimization, with a little modification - generate N children and retain 20% of the best newly generated chromosomes; 20% of best parents are kept in the new generation and the rest is made of the best remaining individuals;
- Multiple populations that exchange best individuals - add after 10 generations the best individual to a common list and replace the worst individual with a randomly selected one from the list;
- After reaching convergence of a population (consecutively 20% of the maximum number of generations have the same best individual), reinitialize population = keep best 10% of existing individuals, obtain 30% via neural networks, and generate the remaining randomly;

The solution for determining the optimal weights combines the two approaches in order to obtain benefits from both – numerical stable solutions from neural networks and the flexibility of genetic algorithms in adjusting these partial solutions.

4. VISUALIZATION

As mentioned in the second section, discourse analysis is an important part of the processing in PolyCAFe because in the CSCL paradigm successful learning is viewed as achieving the ability to utter meaningful utterances, to build a sound discourse, accepted implicitly as such by the other participants. Therefore, discourse is at the basis of measuring the degree of collaboration in a chat.

In PolyCAFe, among the seven provided widgets (Rebedea et al., 2010), four are providing data about the discourse. In the conversation feedback widget are presented the most frequent concepts in the chat and the topics that were considered relevant in the corpus used LSA training, in order to compare them. This information is related to the content of the discussion. Further information related to the collaboration process is also presented like the percentage of implicit and explicit links between utterances, the percentage of utterances that contain speech acts, argumentation, personal opinions, elaborations and requests for information and questions (Figure 5). The contributions of each participant at the discourse, such as the total score based on the content of the utterances issued by that participant, the ranking in conversation graph, SNA factors (e.g. out-degree and centrality) are displayed in the participant feedback widget (Figure 6).
The discourse related features of each utterance (speech acts, argumentation acts) and the associated computed value reflecting its importance are provided in the utterance feedback widget (Figure 7).
Analysis of Discourse in Collaborative Learning Chat Conversations with Multiple Participants

The most complex and appreciated widget in validation is the conversation visualization (Figure 8). Each participant in the chat has associated a horizontal line in the representation and each utterance is placed in the line corresponding to the issuer of that utterance, taking into account its positioning in the original chat file – using the timeline as an horizontal axis. Each utterance is represented as a rectangle aligned according to the issuer on the vertical axis and having a horizontal axis length that is proportional with the dimension of the utterance. The distance between two different utterances is proportional with the time passed between the utterances.

![Fig. 8 Conversation visualization](image-url)
The links between utterances are represented using colored lines that connect these utterances. The explicit and implicit links have different colors. The graphical representation of the chat has a scaling factor that permits zooming.

At the bottom of the graphical representation of the conversation, after the line corresponding to the last participant in the chat, there is a special area that contains a diagram of the evolution of the collaboration in the chat.

The visualization of discourse threads may be obtained in several ways. Clicking with the mouse an utterance colors automatically the thread which contains it. Also, the utterances in the thread are displayed in the second tab (see Figure 8). Discourse threads of utterances containing a specified repeated word may also be visualized in the third tab (see Figure 9).

![Fig. 9 Visualization of threads of repeated words](Image)

5. CONCLUSIONS

The validation performed on the PolyCAFe system showed that it can provide feedback and interactive analysis tools that were found useful for both students and teachers (Rebedea et al., 2010). As PolyCAFe is mainly based on discourse analysis for the visualization and evaluation of a participant’s overall contribution in conversation, we may conclude that we succeeded to analyse discourse in CSCL chats.

A new model was proposed for discourse starting from Bakhtin’s dialogism. A multiple-voiced analysis perspective is introduced which is also the basis for the implemented PolyCAFe system.

In present, evaluations and tuning of the PolyCAFe system are performed and a second version will be developed until the end of 2010. In this new version the Romanian language will be fully supported. In the first version, only several functionalities were language independent. For example, only the visualization of explicit links and of threads of repeated words were possible in Romanian or other languages (Figure 10).
ACKNOWLEDGEMENTS

The research presented in this paper was partially performed under the FP7 EU STREP project LTfLL. We would like to thank all the students of “Politehnica” University of Bucharest, Computer Science Department, who participated in our experiments and provided the inputs for generating our golden standard.

REFERENCES

4. BAKHTIN, M.M. Speech Genres & Other Late Essays, University of Texas Press, Austin, 1986.


