

Social Interaction for Interactive Storytelling

Edirlei Soares de Lima¹, Bruno Feijó¹, Cesar Tadeu Pozzer², Angelo E.M. Ciarlini³,
Simone Diniz Junqueira Barbosa¹, Antonio L. Furtado¹,
and Fabio A. Guilherme da Silva¹

¹PUC-Rio – Department of Informatics, Rio de Janeiro, RJ – Brazil
{elima,bfeijo,simone,furtado,araujo}@inf.puc-rio.br

²UFSM – Department of Electronics and Computing, Santa Maria, RS – Brazil
pozzer@inf.ufsm.br

³UNIRIO – Department of Applied Informatics, Rio de Janeiro, RJ – Brazil
angelo.ciarlini@uniriotec.br

Abstract. In recent years interactive narratives emerged as a new form of digital entertainment, allowing users to interact and change stories according to their own desires. In this paper, we explore the use of social networks as a way of interaction in interactive narratives. We present the interaction interface of an interactive storytelling system that allows users to interact and change stories through social networks. To validate our approach we conducted a user study with 24 participants. The preliminary results show that our method improved the user satisfaction and experience. The proposed methods can be adapted to other applications that require social interaction.

Keywords: Interactive Storytelling, Social Networks, Social Interaction.

1 Introduction

Interactive narratives have been presented as a new form of digital entertainment, in which users can interact and change stories according to their own desires. Although the idea of interactive narratives can be traced back to the 1970s [1] and important experiments on agent-based storytelling can be found in the 1990s [2], the 2000s can be considered the decade of the most intensive and influential research works on interactive storytelling systems. In more recent years, we have been exposed to new demands for richer interactive experiences in storytelling, such as transmedia storytelling [3] and social interaction between groups [4]. We believe that those new demands require the development of new interaction mechanisms and, in particular, the use of social networks as an interaction interface. Williams *et al.* [4] argue that after playing a game, gamers love to discuss the events that just took place and comment on the memorable events, contributing to the culture of the game. The same can be said of interactive storytelling on digital TV.

An interactive narrative designed for TV and shared by thousands of viewers requires new interaction mechanisms and interfaces that support the social interaction between viewers. Moreover, attractive interaction mechanisms are necessary to incentive viewers to interact with the narratives. The popularization of social network services (such as Facebook, Twitter and Google+) puts social computing in the everyday

life. Social networks are like windows to virtual worlds, where people can interact with friends, family, or even strangers.

In this paper, we explore the use of social networks as a way of interaction in interactive stories. We implemented a prototype of the proposed interaction interface in Logtell [5], an interactive storytelling system based on temporal logic and non-deterministic planning. The interactive story implemented in our system corresponds to a short story in the genre of swords and dragons (Figure 1).

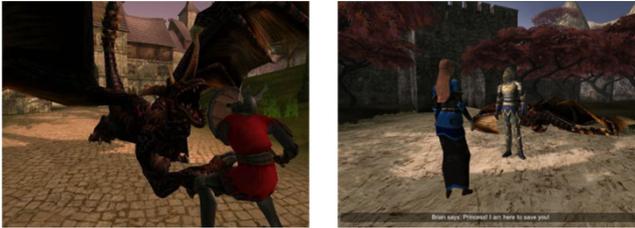


Fig. 1. Dramatization of the swords-and-dragons story performed by Logtell

The paper is organized as follows. Section 2 presents the previous works. Section 3 proposes the architecture of the social interaction interface. Section 4 presents the proposed methods of social interactions. Section 5 describes a preliminary evaluation of the proposed system. Section 6 contains concluding remarks.

2 Previous Works

There are several works on forms of interactions for interactive storytelling in the literature. They cover the subject from traditional GUI interfaces [6][7] to more complex interaction mechanisms, such as speech recognition [8][9][10], body gestures combined with speech [11][12][13], hand-drawn sketches [14][15] and physiological inputs [16]. However, few of these works consider multiuser interactions. Moreover, none of them proposes the use of social networks as a form of interaction interface. The need for social interactions between groups watching interactive TV is analyzed by Williams *et al.* [4]. However, the authors do not propose any specific form of interaction.

Likavec *et al.* [17] propose a framework that allows a social/collective narrative from several story fragments created by several authors interacting on social networks. However, the authors are not dealing with interaction interface methods. Moreover they are not considering interactive digital storytelling narratology. Pittarello [18] presents an architecture for mobile interactive storytelling that includes Facebook. However, his work does not propose forms of social interaction.

3 Social Interaction

The method of interaction presented in this paper is based on the idea of using social networks (such as Facebook, Twitter and Google+) as an interaction interface. This method, here called “**social interaction**”, expands the boundaries of human-computer

interaction towards new forms of multiuser applications. It can be used in any storytelling system that generates plots organized by chapters. This idea is demonstrated in this paper through an interface prototype that we created to our interactive storytelling system called Logtell [4].

Interactive storytelling applications usually require multiuser settings, especially the ones designed for interactive television (iTV), where narratives are shared by thousands of viewers. The Logtell system fits in this category and requires an interaction mechanism that is at the same time attractive to users and allows such a large-scale multiuser interaction.

3.1 Basic Architecture

Figure 2 illustrates the basic architecture of the social interaction interface and how it communicates with the interactive storytelling system. The Storytelling Server generates and controls the execution of the stories, administrates multiple clients who share the same narrative, and informs the Social Interaction Server about valid suggestions for the next chapters. The Social Interaction Server is the interface between interactive storytelling system and social network. It is responsible for accessing the social networks looking for user interactions, translating the users desires into valid story suggestions, and informing the Storytelling Server about the user's choices. The Client Drama Viewer is responsible for simply displaying the generated plots to the viewers. Users can access the social networks using their own social network applications. The interaction system also has a special page in the social networks. Users who want to participate and interact with the story must “follow” (Twitter and Google+) or “like” (Facebook) this page, so they are able to receive the update messages from the interaction system.

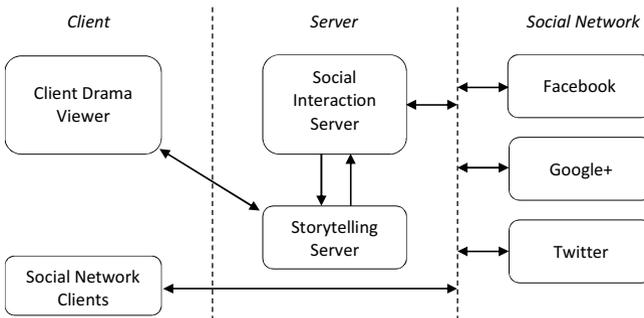


Fig. 2. Basic architecture of the proposed interactive storytelling system with social interaction

3.2 Flow of Activities

Our method is designed for storytelling systems that are organized by chapters. In these systems, users' desires can be fulfilled in the next chapter or during current dramatization. The storytelling system is constantly sending messages to induce facts to the user or provoke him/her, which we denominate “induction messages”. We can

use Activity Diagrams [19] to specify the dynamic behavior of the system, where messages are exchanged and dramatizations are performed. Figure 3 presents the activity diagram for the proposed system, in which the moments of social interactions are indicated by a cloud illustration and a shaded box. When the system starts, the induction message is an introduction to the story. Users receive this message (Facebook and Google+) or use hashtags (Twitter) to indicate suggestions. The introduction message describes the story characters, places, gives some tips about what could happen in the story and incentive the users to comment what they would like to see happening in the story. In our prototype, we present a fairy tale as shown in Figure 4.

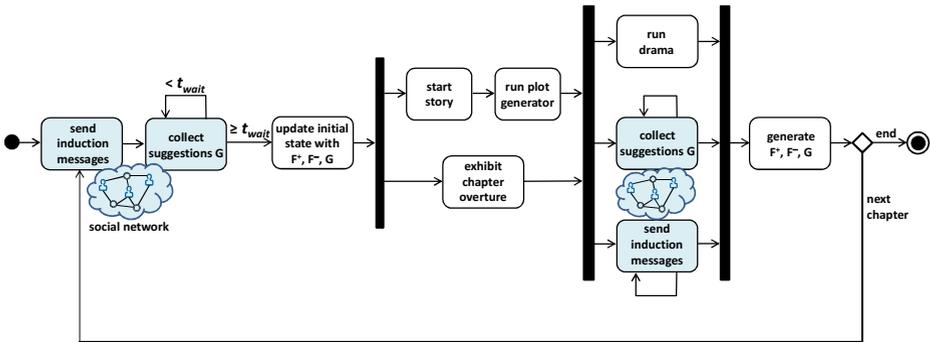


Fig. 3. Activity diagram of the proposed system

“Once upon a time there was a charming princess, called Marian, lady of the White Palace, and two brave young men, sir Brian and sir Hoel, knights of the Gray Castle. Not far away in the sinister Red Castle, lurched Draco, the evil dragon, ready to seize the princess, despite her guardians, and keep her with super-human strength. But there was also the silent wizard of the Green Forest, Turjan the mage. Whoever approached him with due courtesy could hope for a gift of great fighting power. Uncountable stories can be told in this world of fantasy. Will the princess be abducted by the dragon? Or killed by the monster? Will one of the knights save her or revenge her death, with or without the mage’s help?”

A new interactive story is about to begin. Comment here what you would like to see happen with the characters of this story.”

Fig. 4. Example of an introduction message

The system keeps collecting suggestions for t_{wait} minutes (as indicated by the looping arrow), after which the initial state of the chapter is update. Three sets of facts are considered: F^+ (facts generated by the system that are added to the state), F^- (facts generated by the system that are removed from the state), and G (suggestions created by the user that may be considered by the plot generator if they are not inconsistent with the ongoing story). The system considers the G facts that are more frequently mentioned by the users.

The thick black bars indicate parallel activities. We should notice that a chapter overture is exhibited (*e.g.* audio, text, and/or video) while the system runs the plot generator module (the most demanding processing time). Also we can see that interactions occur during the dramatization process (*i.e.* in parallel with the box “run drama” of Figure 3). The induction messages in this dramatization stage are suggestions for the next chapter (*e.g.* “*Would you like to see Turjan giving strength to Brian in the next chapter?*”) or an invitation for a poll (*e.g.* “*The villain should defeat the guards? Yes or No?*”). The users do not have to wait or pay attention to induction messages – they can keep sending any kind of messages (*e.g.* “*The princess should die!*”) anytime.

The last activity in the process is to generate the sets of facts F^+ , F^- , and G for the next chapter. This set G will probably be expanded or edited in the beginning of the new cycle, during the initial t_{wait} minutes.

During the interaction process, the behavior of the users is analyzed by the interaction system. If it detects that only a small number of users is contributing with story suggestions through comments, the system creates a poll in the social network where they are able to select and vote by clicking on the poll option. The votes are combined with the votes from the other interaction methods to decide the events for the next chapters.

4 Methods of Social Interactions

We propose three basic ways of interacting with the stories: (1) **interaction by comments** – where users explicitly express their desires through comments in natural language; (2) **interaction by preferences** – where users express satisfaction or state preferences; and (3) **interaction by poll** – where a poll is created and users vote in what they want. Adaptations are necessary for each type of social network. For example, Twitter does not allow direct comments on the posts as Facebook and Google+, but allow users to use hashtags to indicate suggestions to the stories.

The above-mentioned methods of interaction define the architecture of the Social Interaction Server (Figure 5). In this architecture, the Suggestion Manager controls the interaction mechanisms and centralizes the users’ suggestions and poll results. The next sections describe those three methods of interaction in more details.

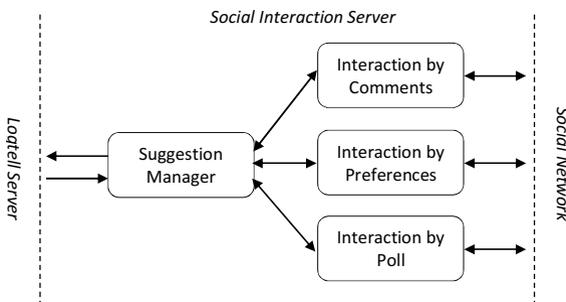


Fig. 5. The architecture of the Social Interaction Server

4.1 Interaction by Comments

The interaction by comments allows users to explicitly express their desires through comments on the social network. The interaction system accesses the users’ comments and extracts valid suggestions using natural language processing techniques. The most often mentioned suggestions are incorporated into the story.

The process of extracting valid suggestions from the users’ comments involves natural language processing [20]. A traditional natural language processing task consists of two main phases: (1) **syntax parsing**, where the syntax tree and the grammatical relations between the parts of the sentence are extracted; and (2) **semantic analysis**, which is the extraction of the meaning of words or phrases.

In the proposed interaction system, we adopted the Stanford Parser to perform the syntax parsing of the sentences [21]. The Stanford Parser [22] is a probabilistic parser that represents all sentence relationships as typed dependency relations instead of using phrase structure representations. However, it also produces phrase structure trees.

The Stanford Parser produces 55 different typed dependencies [23]. These dependencies reflect the grammatical relationships between the words. Such grammatical relations provide an abstraction layer to the pure syntax tree and provide information about the syntactic role of all elements. Figure 6 (a) shows a phrase structure tree generated by the Stanford Parser for the sentence “*Draco should kill Marian!*”. The corresponding typed dependencies are listed in Figure 6 (b). Typed dependencies facilitate the analysis of semantic relationships between words based on both their grammatical relationships and overall sentence syntactical structure.

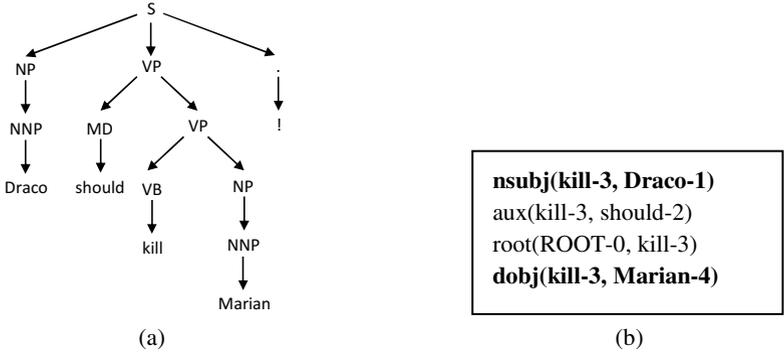


Fig. 6. Phrase structure tree (a) and the typed dependencies (b) of “*Draco should kill Marian!*”

The typed dependencies are all binary relations, where a grammatical relation holds between a “governor” and a “dependent”. In the above example, the relation *nsubj* (nominal subject) relates the noun “*Draco*” with the corresponding verb “*kill*”, whereas the relation *dobj* (direct object) relates this verb with the object “*Marian*”. In this way, the sentence elements are extracted and the sentence structure can be translated into simple first-order logic sentences. In the above example, the following sentence is extracted:

kill(Draco, Marian)

which means that “*Draco*” must perform the action “*kill*” and the victim is “*Marian*”.

In the present work we generate simple logic sentences composed by a disjunction of predicates, e.g. from “*Brian and Hoel fights against Draco*” is generated the sentence *fight(Brian, Draco)* and *fight(Hoel, Draco)*.

With this dependency chain, the system is able to extract “subject – direct object” relationships from sentences. However, for this pattern to be valid, four conditions must be met: (1) a nominal subject (*nsubj*) dependency must exist; (2) the dependent of the *nsubj* dependency must be a family member (in the phrase structure tree); (3) the governor of this dependency must be a verb, which means that a family member is the head noun of the subject of a clause which is predicated by the verb; and (4) a direct object (*dobj*) dependency must exist and the governor of this dependency must match the index of the governor of the *nsubj* dependency – then we assume that the dependency of the *dobj* relation is paired with the family member found initially.

In the example above, the extracted logical sentence already contains the semantic meaning necessary to our interaction system infer a valid suggestion to the story. However, there are some cases where the subjects are not directly referenced. For example, in the sentence “*Brian save Marian and marry her.*”, the pronoun “*her*” refers to “*Marian*”. However, when we compute the typed dependencies for this sentence (Figure 7), we see in the relation “*dobj(marry-5, her-6)*” that the pronoun “*her*” was not resolved and, in some cases, it’s not possible to solve it using only the phrase structure tree. The process of resolving what pronoun or a noun phrase refers to is called **anaphora resolution**. To solve this problem, we used another tool from the Stanford Natural Language Processing Group, the Stanford Deterministic Coreference Resolution System [24], which is able to indicate precisely the correct reference of any unknown pronoun.

```

nsubj(save-2, Brian-1)
nsubj(marry-5, Brian-1)
root(ROOT-0, save-2)
dobj(save-2, Marian-3)
conj_and(save-2, marry-5)
dobj(marry-5, her-6)

```

Fig. 7. Example of anaphora problem in the sentence “*Brian save Marian and marry her.*”

The parser also verifies the occurrence of negations. For example, in the sentence “*Draco should not kill Marian!*”, the adverb “*not*” completely changes the meaning of the sentence. To identify negations, the parser analyses the occurrence of negation modifiers (“*neg*”) in the typed dependency list. Figure 8 illustrates the typed dependency for the example above and occurrence of the negation modifier.

```

nsubj(kill-4, Draco-1)
aux(kill-4, should-2)
neg(kill-4, not-3)
root(ROOT-0, kill-4)
dobj(kill-4, Marian-5)

```

Fig. 8. Example of negation in the sentence “*Draco should not kill Marian!*”

After translating the “subject – direct object” relations into first-order logic sentences, the parser also needs to validate the sentences. For example, the predicate “*fight*(CH_1, CH_2)” requires a nominal subject CH_1 that is a valid character and a direct object CH_2 that also is a valid character in the story context. Moreover, the verb “*fight*” also must be a valid action. To perform this validation, the parser has access to a list of valid actions, characters and places. In this way, the parser is able to identify the elements that the words represent. However, almost all words have synonyms and to deal with this, the parser also incorporates a dictionary of synonyms associated with each action, character and place. So, it is able to parse sentences such as “*The hero should annihilate the villain!*”, where the verb “*annihilate*” is an synonym of the action “*kill*”, and the objects “*hero*” and “*villain*” are, respectively, the roles of the characters “*Brian*” and “*Draco*”.

Ideally, the parser expects sentences that contain at least one verb, one nominal subject and a direct object. However, it not always happens, in some cases the subject, the direct object, or both are omitted. For example, the sentence “*Kill the princess!*” do not express directly who should perform the action “*kill*”, but indicates the direct object “*princess*” (Figure 9). In this case, the parser is still able to generate a partial logic sentence to represent it:

kill(* , *Marian*)

which means that someone “*” must perform the action “*kill*” and the victim is “*Marian*” (identified by its role in the story (“*princess*”). The operator “*” can be replaced by any valid character to complete the logical sentence.

root(ROOT-0, Kill-1) det(princess-3, the-2) dobj(Kill-1, princess-3)

Fig. 9. Example of omitted subject in the sentence “*Kill the princess!*”

The entire process of extracting valid first-order logic sentences from text phrases is illustrated in Figure 10. In the Syntax Parsing step, the Stanford Parser receives a text phrase S_x as input and generates a Dependency Tree and the Typed Dependencies for the sentence. Using this information, in the Semantic Analysis phase, the parser performs the Anaphora Resolution process to resolve the pronouns of the sentence and find valid synonyms using the Synonym Dictionary. Finally, the parser checks the integrity of the sentences using some Logic Rules and returns a list of valid first-order logic sentences (P_x^n).

After extracting all users’ suggestions from the comments in the social network, the interaction system relates comments that express the same suggestions and count how many votes the suggestions received. In the case of Facebook and Google+, besides writing a comment, users are also able to “like” or “+1” a comment of another user, which indicates that they liked what the comment says. In this way, the interaction system considers the number of users that directly wrote that something should happen and the number of users that liked the respective comments. After counting

the number of votes of each suggestion, the module that manages the Interaction by Comments sends this information to the Suggestion Manager (as illustrated in Figure 5).

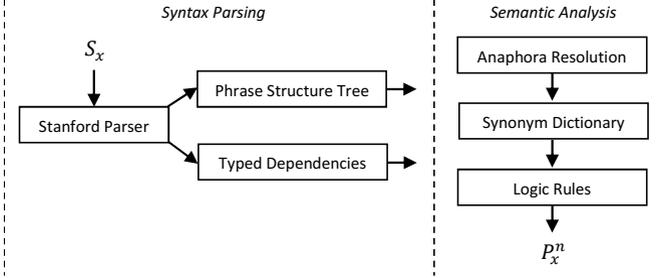


Fig. 10. The process of extracting valid first-order logic sentences. S_x is the input text phrase and P_x^n is the output list of predicates

4.2 Interaction by Preferences

The interaction by preferences allows users to express their satisfaction with the story suggestions through social networks. Instead of directly writing a comment expressing a desire, users are able to “like” (Facebook) or “+1” (Google+) a suggestion generated by the interaction system. Users can also write comments on the generated suggestions expressing their satisfaction with the proposed events. The interaction system checks the users’ responses and the most well received suggestions are incorporated into the story.

The process of extracting user’s satisfaction also involves natural language processing, more specifically the area of Sentiment Analysis [25]. However, in this work, instead of using complex sentiment analysis techniques, we adopted a more simplistic approach to solve this problem. As the generated suggestions are more like questions (e.g. “*Would you like to see Draco attacking the White Palace in the next chapter?*”), the users usually respond it positively or negatively, i.e. agreeing or disagreeing. In this way, the parser only needs to identify positive and negative answers in the users’ comments.

The approach adopted by our parser to identify positive and negative answers uses a list of words, where each word W_i is associated with a numerical score $W_i^s \in [-1.0, +1.0]$. High negative scores represent very negative words and high positive scores represent very positive words. Considering C_x a user commentary, the sentiment $St(C_x)$ is given by:

$$St(C_x) = \frac{1}{n} \sum_{i=1}^n W_i^s \quad \text{if } (W_i \in C_x)$$

where $St(C_x) \in [-1.0, +1.0]$ indicates if C_x is a positive commentary ($C_x > \beta$) or a negative commentary ($C_x < \alpha$), in which α and β defines a precision threshold where uncertain commentaries are ignored (classified as neutral).

To illustrate this process, let's consider $\alpha = -0.3$ $\beta = +0.3$, and the following user commentaries for the suggestion $S_1 =$ “*Would you like to see the princess Marian dying in the next chapter?*”:

1. “*Yes!! :)*”
2. “*I would love to see it happening!!! ;)*”
3. “*No!! I love the princess... :(*”
4. “*This story is boring... :(*”

For case (1), the word “*Yes*” and the emoticon “*:)*” have both the score +1.0; giving the sentiment $St(C_1) = +1.0$ and classifying it as a positive commentary. In case (2), the word “*love*”, “*see*” and the emoticon “*;*” have the scores +0.8, +0.5 and +0.9 respectively; giving the sentiment $St(C_2) = +0.73$ and classifying the sentence as a positive commentary. In case (3), the word “*No*”, “*love*” and the emoticon “*:(*” have the scores -1.0, +0.8 and -1.0 respectively; giving the sentiment $St(C_3) = -0.4$ and classifying the sentence as a negative commentary. Finally, in case (4) the word “*boring*” and the emoticon “*:(*” have the scores -0.7 and -1.0 respectively; giving the sentiment $St(C_4) = -0.85$ and classifying the sentence as a negative commentary.

There are some cases where the users also complement their opinion with other story suggestions (e.g. “*No!!! The hero must save the princess!*”). For this reason, the interaction system also performs the process of extracting story suggestions from user comments (described on section 4.1) in the comments created on posts designed for the interaction by preferences. In the above example, the system would be able to classify the commentary as a negative commentary (counting a negative vote for the system suggestion) and extract a new vote for the suggestion “*save(Brian, Marian)*”.

When the system completes the process of classifying the comments as positive or negative, the number of votes is computed. Positive comments count as positive votes to the suggestion described in the post and the negative comments count as negative votes. The number of users that “like” (Facebook) or “+1” (Google+) the suggestion also count as positive votes. Then, the module that manages the interaction by preferences sends this information to the Suggestion Manager (as illustrated in Figure 5).

4.3 Interaction by Poll

The interaction by poll allows users to choose what they want through polls in the social network. Instead of directly writing a comment or waiting for the desired suggestion appear (posted by the interaction by preferences), they are able to see all available options and vote in the suggestion of their choice. The interaction system checks the poll results, and the most voted suggestions are incorporated into the story.

The process of extracting users’ choices from a poll does not require any complex algorithm. However, the importance of this method should not be underestimated, because it provides an easy way of interaction where users that don’t like to write or don’t know exactly what they want are able to interact just by clicking on a poll option.

After computing the number of votes of each poll option, the module that manages the Interaction by Poll sends this information to the Suggestion Manager (as illustrated on Figure 5).

5 Evaluation

To evaluate the social interaction interface, we performed two tests: (1) a user evaluation test to check the interface usability from a Human-Computer Interaction (HCI) perspective, and (2) a technical test to check the performance and accuracy of the proposed methods of interaction. The following sections describe these tests.

5.1 User Evaluation

We have conducted a preliminary user evaluation with 24 high school students, 14 male and 10 female, aged 16 to 18 (mean of 17). Sixteen of them play video games at least weekly. None of them had previous experiences with interactive storytelling systems. Twenty of the participants use social networks at least once a day; the other four use few times a week. We divided the participants in two groups and conducted separated evaluation sessions for each group. The participants of each session were on the same room, but we asked them to don't interact between them physically.

We asked participants to interact with two versions of our interactive storytelling system, one based on a traditional GUI interface (described in [26]) and the other using the social interaction interface here presented. In order to reduce learning effects, one group used the traditional GUI interface first, and the other used the social interaction interface first. Facebook was the most popular social network between the participants, so all participants used it to interact with the social interaction interface. Six participants decided to use their cell phones to access the social network, while the others used desktop computers. The GUI interface was used by all participants through desktop computers.

After using each version, the participants filled out a questionnaire derived from the IRIS Evaluation Toolkit [27][28]. We evaluate the system usability, the correspondence of system capabilities with user expectations (user satisfaction), the interaction effectiveness and the user experience (curiosity, flow and enjoyment). Each statement was given on a five-point Likert scale ranging from "strongly disagree" (1) through "neutral" (3) to "strongly agree" (5). After having interacted with both versions of the system, the participants were interviewed about their experience.

Figure 11 summarizes the results of the questionnaires. The GUI interface produces slightly better usability and effectiveness when compared with the social interaction interface, probably because it requires less effort for the interaction. On the other hand, the social interaction interface clearly increases the user satisfaction and improved the user experience. As far as the interviews are concerned, all participants stated that they preferred to interact using the social network, because it was more interesting, attractive, exciting, and innovative. Some participants pointed that they had some difficult to understand what kind of suggestions they could write. We believe that this difficult can be overcome by adding some discrete tips in the story (*e.g.* by making the characters to think out loud what they could do). We noticed during the analysis of the results that users that used cell phones to access the social network had better experiences, especially regarding the usability of the system.

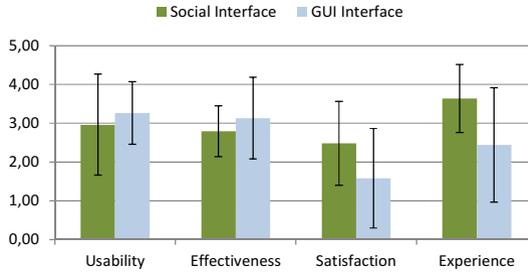


Fig. 11. Average and standard deviation of questionnaire topics in both versions of our system

5.2 Technical Evaluation

The technical evaluation concerns the accuracy and the real-time performance of the interaction system. The evaluation was based on two experiments: (1) the recognition rate test, to check the accuracy of the predicted suggestions; and (2) the performance test, to check the time needed to process the input comments and recognize the suggestions as first-order logic sentences. For both tests we used the comments derived from the user evaluation experiment described on section 5.1. In this way, we were able to evaluate the methods with real comments and check how well the system performed in the experiment.

During the user evaluation test we collected a set of 107 text comments, including 81 comments that were manually classified as valid suggestions. For the recognition rate test, we used our method to extract valid story suggestions from the comments and then compared the results with the results obtained through the manual classification. As result we get a recognition rate of 90.6%, with only 10 valid comments being incorrectly classified as invalid suggestions. The main reason for the incorrect classifications was the occurrence of spelling mistakes in the comments.

To evaluate the performance of our method, we again utilized the collection of 107 comments collected during user evaluation, and calculated the average time necessary to perform the recognition of the suggestions as first-order logic sentences. The computer used to run the experiments was an Intel Xeon E5620, 2.40 GHZ CPU, 24 GB of RAM using a single core to process the algorithms. As result we get the average time of 2.7 milliseconds to process an input comment and recognize the suggestion as first-order logic sentences (standard deviation of 1.3 milliseconds).

Similarly, we evaluated our method to recognize user satisfaction. During the user evaluation test we collected a set of 43 text comments expressing user satisfaction. We used our simplistic method of sentiment analysis to classify the comments as positive and negative comments then compared the results with the results obtained through a manual classification. As result we get a recognition rate of 97.6%, with only 1 positive comment incorrectly classified as negative. The time consumed by the algorithm is almost insignificant (less than 0.001 milliseconds).

In our experiments, the social interaction interface presented good results. However, natural language processing is not a trivial task; it is possible that our parser don't recognize correctly every possible valid sentence, but we believe that it is able to recognize the sentences in the most part of the cases without the audience be aware of

mistakes. The time necessary to process the user comments is small, but it grows according to the number of comments to be processed. With a large number of users interacting at same time, the parallelization of this process may be necessary to guarantee the real-time execution of the narrative.

6 Conclusions

In this paper, we explore the use of social networks as a way of interaction in interactive stories. We present the interaction interface of an interactive storytelling system that allows users to interact and change stories through social networks. This method, here called “social interaction”, expands the boundaries of human-computer interaction towards new forms of multi-user applications. As far as we are aware, this is the first time this form of interaction is explored in an interactive narrative.

The prototype was built over the Logtell system; however its architecture is generic enough to be adopted by any interactive storytelling system organized by chapters. Moreover, the interaction methods can be adapted to other systems that require a social user interaction. This form of interaction fits very well in the context of interactive narratives designed for digital TV. It doesn't require any direct interaction through the TV. Spectators interact through the social network clients, using smartphones, tablets, or personal computers without having to install any additional software. The activity that results from the user interactions in the social network may attract more viewers to the broadcasting channel (increasing the audience). In addition, viewers can make new friends through the interaction in the social network.

Acknowledgements. This work was partially supported by CAPES (Coordination for the Improvement of Higher Education Personnel, linked to the Ministry of Education) under grant RH-TVD 01/2007 No. 133/2008 and by Research Productivity Grants of CNPq (National Council for Scientific and Technological Development, linked to the Ministry of Science and Technology) under grant PQ 304090/2009-3.

References

1. Meehan, J.R.: TALE-SPIN, An Interactive Program that Writes Stories. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence, pp. 91–98 (1977)
2. Loyall, A.B., Bates, J.: Hap: A Reactive, Adaptive Architecture for Agents. Technical Report CMU-CS-91-147, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA (1991)
3. Cheshire, T., Burton, C.: Transmedia: Entertainment reimagined, Wired UK, <http://www.wired.co.uk/magazine/archive/2010/08/features/what-is-transmedia> (accessed April 20, 2012)
4. Williams, D., Ursu, M.F., Meenowa, J., Cesar, P., Kegel, I., Bergström, K.: Video mediated social interaction between groups: System requirements and technology challenges. *Telematics and Informatics* 28, 251–270 (2011)
5. The Logtell Project Website, <http://www.icad.puc-rio.br/~logtell> (accessed April 19, 2012)

6. Ciarlini, A.E.M., Pozzer, C.T., Furtado, A.L., Feijo, B.: A logic-based tool for interactive generation and dramatization of stories. In: Proceedings of the International Conference on Advances in Computer Entertainment Technology, Valencia, pp. 133–140 (2005)
7. Grasbon, D., Braun, N.: A morphological approach to interactive storytelling. In: Proceedings of Cast 2001, Living in Mixed Realities, Sankt Augustin, Germany, pp. 337–340 (2001)
8. Mateas, M.: An Oz-Centric Review of Interactive Drama and Believable Agents. In: Veloso, M.M., Wooldridge, M.J. (eds.) Artificial Intelligence Today. LNCS (LNAI), vol. 1600, pp. 297–328. Springer, Heidelberg (1999)
9. Cavazza, M., Charles, F., Mead, S.: Character-based interactive storytelling. *IEEE Intelligent Systems*, Special issue on AI in Interactive Entertainment 17(4), 17–24 (2002)
10. Cavazza, M., Pizzi, D., Charles, F., Vogt, T., André, E.: Emotional Input for Character-based Interactive Storytelling. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems, Budapest, Hungary, pp. 313–320 (2009)
11. Cavazza, M., Charles, F., Mead, S.J., Martin, O., Marichal, X., Nandi, A.: Multimodal acting in mixed reality interactive storytelling. *IEEE Multimedia* 11(3), 30–39 (2004)
12. Cavazza, M., Lugin, J.-L., Pizzi, D., Charles, F.: Madame bovary on the holodeck: immersive interactive storytelling. In: Proceedings of the 15th International Conference on Multimedia (MULTIMEDIA 2007), pp. 651–660 (2007)
13. Lima, E.S., Feijó, B., Barbosa, S., Silva, F.G., Furtado, A.L., Pozzer, C.T., Ciarlini, A.E.M.: Multimodal, Multi-User and Adaptive Interaction for Interactive Storytelling Applications. In: Proceedings of the 10th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), pp. 1–10 (2011)
14. Lima, E.S., Feijó, B., Barbosa, S.D.J., Furtado, A.L., Ciarlini, A.E.M., Pozzer, C.T.: Draw Your Own Story: Paper and Pencil Interactive Storytelling. In: Proceedings of the 10th International Conference on Entertainment Computing (ICEC), Vancouver (2011)
15. Kuka, D., Elias, O., Martins, R., Lindinger, C., Pramböck, A., Jalsovec, A., Maresch, P., Hörtner, H., Brandl, P.: DEEP SPACE: High Resolution VR Platform for Multi-user Interactive Narratives. In: Proceedings of the 2nd Joint International Conference on Interactive Digital Storytelling, pp. 185–196 (2009)
16. Gilroy, S., Porteous, J., Charles, F., Cavazza, M.: Exploring Passive User Interaction for Adaptive Narratives. In: Proceedings of the International Conference on Intelligent User Interfaces (IUI). ACM (2012)
17. Likavec, S., Lombardi, I., Nantiat, A., Picardi, C., Dupré, D.T.: Threading Facts into a Collective Narrative World. In: Aylett, R., Lim, M.Y., Louchart, S., Petta, P., Riedl, M. (eds.) ICIDS 2010. LNCS, vol. 6432, pp. 86–97. Springer, Heidelberg (2010)
18. Pittarello, F.: Designing a Context-Aware Architecture for Emotionally Engaging Mobile Storytelling. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) INTERACT 2011, Part I. LNCS, vol. 6946, pp. 144–151. Springer, Heidelberg (2011)
19. OMG. UML Superstructure Specification, v2.4 (2011), <http://www.omg.org/spec/UML/2.4/Superstructure/Beta2/PDF> (accessed April 19, 2012)
20. Jurafsky, D., Martin, J.H.: Speech and language processing: An Introduction to Natural Language Processing. In: Computational Linguistics, and Speech Recognition. Prentice-Hall (2000)
21. Klein, D., Manning, D.C.: Accurate Unlexicalized Parsing. In: Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423–430 (2003)
22. Stanford, Stanford Parser, <http://nlp.stanford.edu/software/lex-parser.shtml> (accessed April 20, 2012)

23. Marneffe, M., Manning, C.D.: The Stanford typed dependencies representation. In: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation, Manchester, pp. 1–8 (August 2008)
24. Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., Manning, C.: A Multi-Pass Sieve for Coreference Resolution. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Boston, USA (2010)
25. Liu, B.: Sentiment Analysis and Subjectivity. In: Indurkha, N., Damerau, F.J. (eds.) Handbook of Natural Language Processing, 2nd edn. (2010)
26. Camanho, M.M., Ciarlini, A.E.M., Furtado, A.L., Pozzer, C.T., Feijo, B.: A model for interactive TV Storytelling. In: Proceedings of the 7th Brazilian Symposium on Games and Digital Entertainment, Rio de Janeiro, Brazil, pp. 197–206 (2009)
27. Klimmt, C., Roth, C., Vermeulen, I., Vorderer, P.: The Empirical Assessment of The User Experience In Interactive Storytelling: Construct Validation of Candidate Evaluation Measures. Technical Report, Integrating Research in Interactive Storytelling - IRIS (2010)
28. Roth, C., Vorderer, P., Klimmt, C.: The Motivational Appeal of Interactive Storytelling: Towards a Dimensional Model of the User Experience. In: Iurgel, I.A., Zagalo, N., Petta, P. (eds.) ICIDS 2009. LNCS, vol. 5915, pp. 38–43. Springer, Heidelberg (2009)