Squaring Stories:
Using Comics to Make Story Summaries

Ana Simões
INESC-ID / IST
Av. Professor Cavaco Silva, Tagus Park
2780-990 Porto Salvo, Portugal
ana.simoes@tagus.ist.utl.pt

ABSTRACT
Emergent narrative systems create stories which are always different from each other. Creating summaries of these stories is a challenge especially if we want to capture the richness of the characters. Our goal is to automatically generate summaries from emergent narrative using comics as the visual medium for the summary. We identify the most important situations in the story log looking at the emotional state of the characters, transform the resulting summary into a comics description and create the comic. We believe that a good summarization of a story that maintains the emotions of the characters together with an expressive visual representation is essential for the user to remember the story. We consider that the introduction of perspectives in a story is a good method to give more credibility to its characters.

Categories and Subject Descriptors
D.2.11 [Software Engineering]: Software Architectures – languages.

General Terms
Languages.

Keywords
Story Summarization, Emergent Narrative, Perspectives, Comics.

1. INTRODUCTION
As technology advances, the forms of entertainment have become more varied and more interesting. There are technologies through which we interact with characters and based on this interaction a non-scripted story is generated.

Some interaction technologies offer the possibility to review parts of its happenings, but only a few summarize their main events. Automatic summarization is a good method to review stories of interactions or simulations.

When a story involves several characters, it is common for each character to have its own perspective of the story. That is to say that each character will tell a story in its own way according to his point of view of the story. It means that the same story can be told in different ways. The generation of different summaries of the same story is an interesting method to give more credibility to the characters of a story.

We propose to generate diverse summaries of a story produced by an application according to the various perspectives of that story. We identify the most important situations of an application log through an analysis of the emotional state of the characters. We assume that the important events in a story are provoked by strong emotions and, consequently, they originate strong emotions. We can choose these emotions according to each perspective of the story and so we can generate different summaries of the same story. The resultant stories serve as input to another system with the purpose of dynamically generating comics of them.

2. RELATED WORK
There are at least four methods to summarize game logs, namely, Nick Halper and Maic Masuch’s framework [5], Yun-Gyung Cheong and R. Michael Young’s system [2], Automated Creation of Movie Summaries [4] and Ariel Shamir et al.’s system [7].

The framework of Nick Halper and Maic Masuch automatically extracts action scenes for spectator modes and summaries in computer games. The system assumes that interesting events occur when the state of a game or player has significantly changed. This system is adjusted for action games and may be limited for games where actions can have long-term effects.

The system of Yun-Gyung Cheong and R. Michael Young translates a game log into a sequence of actions structured as a plan. It identifies the essential events of the plan measuring their qualitative importance which is determined by their roles in the plan. The summary of the most important events of the game is constructed by identifying essential elements of a story that are similar to those that human users would choose.

The creation of movie summaries by Doron Friedman et al. consists in the extraction of the interesting highlights of a log of all events from an interactive virtual environment. To distinguish between interesting and non-interesting events, they remove the obvious events of the story. Despite this solution being specific, limited in scope and domain dependent, the approach can be generalized to many types of virtual environments, particularly for games where actions have long-term effects.

Ariel Shamir et al.’s system summarizes a game log dividing it in scenes not only when location and time changes but also by modelling interactions between entities. The system depends on the importance of the entities and actions. This approach is applicable to many scenarios.

3. SYSTEM’S OVERVIEW
The aim of this system is to automatically generate comics from emergent narrative in order to create visual story summaries. Figure 1 illustrates all the steps of the system giving more relevance to the module presented in this paper, the summarization module.
FearNot! is an interactive program in which children deal with bullying situations. The main characters are the bully and the victim and there are usually bully assistants. The victim character suffers aggressions (which can be verbal or physical, or both) by the bully characters and then the interaction with the user occurs. The children interact with the victim of the story through a chat interface that resembles an instant messaging application such as Windows Live Messenger. This interaction is done in dialogue situations in which the children advise the victim about the best way to deal with the problem. The advice influences the victim’s behaviour but it does not interfere with his autonomy, which preserves the character’s believability.

The system uses the log generated by FearNot! and gathers the most significant events and some emotional information of the involved characters. Then, the Comics Translation Module creates comic strips descriptions using the Comic Strip Description Language (CSDL) and finally the Realizer Module generates a visual comics representation for each strip.

4. SUMMARIZATION MODULE
The Summarization Module includes three essential steps: the parsing of the log, the filtering of the important actions and the writing of the summary. The evolution of the module consists in a repetition of these steps for each received line of the log. Next, we describe these three steps.

4.1 Parsing of the log
The parsing of the log consists in reading it line by line and storing the information that was read. This information could be a property or an event. The properties are related to the objects, the characters and the user. The events are related to the selection of an episode, the narrative text and the actions of the story. There are different types of actions, as can be seen in Figure 2.

4.2 Filtering of the important actions
The filtering of the important actions is done by taking into account the characters’ emotions. Our approach is related to the approach of Doron Friedman et al. [4], which consists in removing the regular events of a story. They use a script that tags the more frequent events of a story as obvious events and these events are considered non-interesting. Their approach could not be sued in this application because FearNot! always generates bullying situations which we evidently want to filter. So, the distinction between interesting and non-interesting events is made according to the emotions of the characters. We assume that if a character has a strong emotion it is because the events that are associated with this emotion are non-obvious. We filter two types of events: the events that lead to the emotion and the events that were motivated by the emotion. We consider the non-neutral emotions that have a high intensity to filter the events. Table 1 shows the non-neutral emotions in FearNot!.

<table>
<thead>
<tr>
<th>Non-neutral emotions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Love</td>
</tr>
<tr>
<td>Hate</td>
</tr>
<tr>
<td>Hope</td>
</tr>
<tr>
<td>Fear</td>
</tr>
<tr>
<td>Satisfaction</td>
</tr>
<tr>
<td>Relief</td>
</tr>
<tr>
<td>Fears-Confirmed</td>
</tr>
<tr>
<td>Disappointment</td>
</tr>
<tr>
<td>Joy</td>
</tr>
<tr>
<td>Distress</td>
</tr>
<tr>
<td>Happy-For</td>
</tr>
<tr>
<td>Pity</td>
</tr>
<tr>
<td>Resentment</td>
</tr>
<tr>
<td>Gloating</td>
</tr>
<tr>
<td>Pride</td>
</tr>
<tr>
<td>Shame</td>
</tr>
<tr>
<td>Gratification</td>
</tr>
<tr>
<td>Remorse</td>
</tr>
<tr>
<td>Admiration</td>
</tr>
<tr>
<td>Reproach</td>
</tr>
<tr>
<td>Gratitude</td>
</tr>
<tr>
<td>Anger</td>
</tr>
</tbody>
</table>

Besides the standard produced summary, the system offers the possibility to have summaries according to perspectives. These perspectives can be described by the user in the beginning of the system. Thus, the generation of the story by the perspective of the victim and the perspective of the bully is possible. The users may choose from how many perspectives they want to
create a story. For each perspective, they have the possibility to choose the actions and emotions they want to highlight and omit those they want to ignore. If the users do not indicate the perspectives, the system generates the summaries according to the perspectives that exist by default. These perspectives were constructed by choosing only the good emotions of the victim, and the bad emotions of the bully.

4.3 Writing of the summary
The writing of the summary consists in a transformation of the actions filtered in a story. The story is represented by a story description language and is stored in an XML format. The user has the possibility to choose how he wants to see the summary. The summary can be presented in an XML file or in visual representation in comics art. The summary is constructed as the important actions are filtered.

5. EVALUATION
The objective of the evaluation was to examine two main issues: to understand if the neutral summaries are well done and to understand if the readers perceive the existence of a given perspective.

It was necessary to separate the participants in two groups in order to compare the results of the story interpretation before and after its summarization. The first group watched a video that depicted an episode occurred in FearNot!. The other group read this same episode transformed by our system into a comic strip.

The variables used to understand if the neutral summaries are well done were the identification of the important actions and the identification of predefined actions. In the former, the participants wrote the actions that they consider more important. In the latter, the actions are given and the participants had only to indicate the actions that really happened in the story. These variables are measured by the frequency of each indicated action. The results of the first variable can be seen in Figure 3.

![Figure 3. Frequency of identified actions in the Video and the Comics.](image)

The results have shown that, in a general way, people understand the comics stories as well as the video story.

We generated two stories to evaluate the perspectives’ summaries: the summarized story with the bully’s perspective and the one with the victim’s perspective. These comics stories were presented in the same way, but the different perspectives were given to different groups of participants. The group of participants who read the neutral comic read the comic representing the bully perspective.

First, we evaluated if people consider that the story presented through a certain perspective is different from the neutral story and then we tried to find out if people understood who is telling the story. The first method consisted in comparing the events that were identified by the participants in the two different perspectives. This method was evaluated through the average, median and standard deviation. The results were very different, which means that the participants considered that the stories have considerable differences. The second method consisted in analysing the answer where the participants identified the main character of the story. We assumed that this character was the most important one. Table 2 shows the obtained results.

<table>
<thead>
<tr>
<th></th>
<th>Victim Story</th>
<th>Bully Story</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average</strong></td>
<td>5</td>
<td>1.1</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td><strong>Standard Deviation</strong></td>
<td>0.00</td>
<td>0.31</td>
</tr>
</tbody>
</table>

In the case of participants that read the comics representing the victim’s perspective, they should consider that the victim is the main character of the story and this is verified. In the case of participants who read the comics of the bully’s perspective, the answers should indicate the bully as the main character and this is not observed.

The results of the stories told through perspectives show that people have the notion of perspective, but they do not recognize to whom the perspective belongs.

6. RESULTS
Next, we present an example of the transformation of a line of a log into the stories description language and the respective depiction as a comic panel.

The log line that represents an event has the following format: 

If the action presented in this line is filtered, it is transformed into an action represented in the stories description language as follows:

```xml
<action name="say">
  <character>
    <name>Luke</name>
    <pose>standing</pose>
    <emotion>Gloating</emotion>
  </character>
  <target type="character">John</target>
  <speech>Olhem todos, apresento-vos o Joao, o bebe chorao!</speech>
</action>
```
7. CONCLUSIONS AND FUTURE WORK

We proposed to solve the challenge of automatically generating summaries of emergent narratives. Additionally, we aimed at giving a notion of perspective with the intent of generating richer summaries and giving more credibility to the characters of the story. These summaries are used to automatically generate comics. We took advantage of both these features to create a module of a system that is capable of depicting these summaries in comics art form.

We chose an approach identical to the approach that was presented in the work of Doron Friedman et al. [4]. They remove the obvious events of a story by tagging the more frequent events as obvious, those that have become routine. In our approach, we filter the non-obvious events in a bullying scenario. For that, we look at the non-neutral emotions of the characters and we choose the actions that are related to these emotions.

Through the evaluation of the chosen approach, we conclude that people often identify the same important actions in the original story source as the ones that are included in the summary produced by our module. Thus, we can conclude that we successfully demonstrated our hypothesis. The results were satisfactory in what concerns the notion of perspective, but people do not always recognize to whom the perspective belongs. It is possible to conclude that, while not being totally efficient, the system provides promising results.

The aspects that we believe are relevant as future work are described in the following items.

- **More information in the Stories Description Language**: the Stories Description Language can be extended in order to include more information that can be constructive for the story. For example, in the beginning of each episode the characters that are present in the respective scenery could be included in the story description. The duration of each episode could also be indicated.

- **Add the concept of dramatic arc [6] to the story structure**: Michael Mateas considers that stories usually have a structure of the dramatic arc. In this structure the story has five different stages with different importance. Through the characters’ emotions, in the end of the log, we are capable of identifying these different stages in our story. The idea is to add a least one action of each stage into the story. This would be essential for people to understand the main plot of the story.

- **Generate the story through the characters’ autobiographic memory**: in FearNot!, the autobiographic memory of the characters consists in their capacity to memorize the essential events that occurred along the development of the story [3]. It is possible that, at the end of each episode, this information is reported in the log and the system filters all this information, producing a summary. This would be a very interesting extension to make the comparison between the perspectives’ stories chosen by users and the story generated by the characters possible.

- **Integration with different applications**: this module could be integrated with other applications that generate logs in the same format of the ones from FearNot’s. Teatrix [7] is an example of an application that could generate logs in this format.

- **Story’s output**: the story could be presented in a common story format such as a narrative. It would be necessary to define connectors to link the actions and to introduce a new episode, for instance. These connectors could indicate time, such as “Then” or “Later”, or they could join events with words such as “and” or “moreover”.

8. ACKNOWLEDGMENTS

This work was partially supported by European Community (EC) and is currently funded by the eCIRCUS project IST-4-027656-STP. The authors are solely responsible for the content of this publication. It does not represent the opinion of the EC, and the EC is not responsible for any use that might be made of data appearing therein.

9. REFERENCES


