LitSens: An Improved Architecture for Adaptive Music Using Text Input and Sentiment Analysis

Manuel López Ibáñez[1], Nahum Álvarez[2], and Federico Peinado[1]

[1] Department of Software Engineering and Artificial Intelligence
Complutense University of Madrid
c/Profesor José García Santesmases 9, 28040 Madrid, Spain
manuel.lopez.ibanez@ucm.es, email@federicopeinado.com
[2] Artificial Intelligence Research Center
National Institute of Advanced Industrial Science and Technology
2-3-26 Aomi, Koto-ku, Tokyo 135-0064, Japan
nahum.alvarez@aist.go.jp

Abstract LitSens aims to be a sound system which takes into account the contingencies of real-time decision making in video games. Through this article, we present several improvements for an earlier version of our system architecture, which consisted of an emotion manager attached to predesigned, interactive texts. The main additions to said system are the possibility for users to input custom text to a dialogue manager and the automatic tagging of that text by using real time sentiment analysis, thus improving the knowledge base. We conclude that, though lacking validation due to its early development stage, LitSens can constitute an efficient method for generating adaptive music in real time, suitable for interactive experiences like video games.

Keywords: Affective Computing · Sound Design · Natural Language Processing · Video Games · Procedural Content Generation

1 Introduction

Though it is pretty common to use sentiment analysis for the task of processing big text corpora, such as opinions given on social media by thousands of individuals [1], few are the cases where this field of Natural Language Processing is used to create real-time responses to brief text inputs.

Through this article we aim to present a revision of a previously designed audio system architecture [2], to which we added the ability to input text, as well as a real-time sentiment analyser, based on Synesketch, by Uroš Krčadinac [3, 4] (lexicon-based). After a brief introduction to the current state of the art, our system architecture will be presented in detail, with special focus on the newest additions, and we will end with a discussion on the potential of an adaptive sound system for soundtrack creation, and some suggestions for future work.

2 Adaptive sound and music

The concept of *adaptive music* has changed profusely in recent years. Classic approaches to dynamic and adaptive audio systems [5] consisted mostly of smooth transitions between scripted events, predesigned narrative interactions or cinematic sequences. However, the increasing complexity of interactive experiences of our days demands a very different approach. Current adaptive music must consider player reactions and adapt to them in real time, thus generating an adequate emotional atmosphere which is unique to each experience. Here is where procedural music generation comes into play. However, in spite of the efforts of authors like Jewell, Nixon & Prügel-Bennett [6], which add semantic information to the automatic music generation process, improving its results and bringing them closer to commercial soundtracks, available technology still constitutes a big impediment to real time music generation. Producing realistic music in a current computer (even a powerful one) is a demanding process, which requires enormous processing capacities, as well as high read and write speeds from a hard disk drive.

As Collins states [7], available technology is one of the main issues when generating procedural music, and this problem is not going to be solved soon, as current soundtracks can take minutes to generate when using virtual instruments (VSTi), and the ideal latency should be of a few milliseconds (current gaming monitors claim latencies of a mere 1 to 5 ms).

3 The LitSens architecture

Our software architecture is designed to work in Unity, a popular game engine[3], and functions as follows (see Figure 1):

- LitSens works as an interactive experience in which there is narrative content in the form of text.
- For the design of our system, we take into account three main components: a dialogue system, an audio system and an emotion manager.
- At the beginning of the experience, a text with narrative content is shown in a text box. The user can then use a text field to input a response, which will produce an output consisting of the next sentence of the narrative.
- The input is then processed by a sentiment analyser, which provides several values. The three values with more weight are then normalized and stored by the Emotion Manager.
- The Audio System reads those values from the Emotion Manager, and selects a maximum of three music fragments, which start playing through the game engine after a simple mix process. As shown in Figure 2, all music fragments are previously designed by a human, tagged with an emotion and stored in a database.

[3] www.unity3d.com

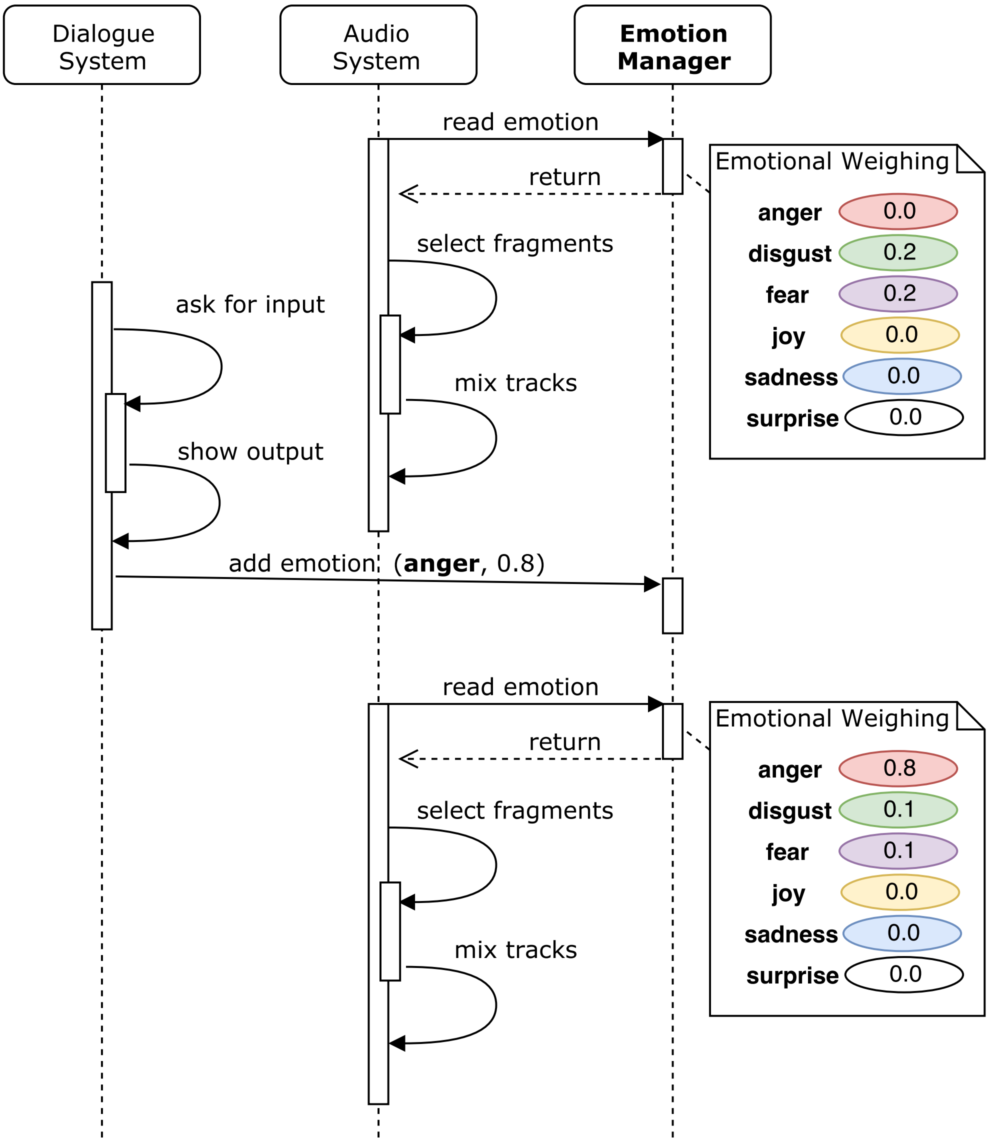

Figure 1. Software architecture of LitSens.

- Following inputs will modify the state of the Emotion Manager, producing different fragment combinations and causing them to change in real time.
- A total of 6 emotions are considered, following Ekman's taxonomy of basic emotions [8]: happiness, sadness, fear, surprise, disgust and anger.
- We have chosen a 3-layer system so as to be able to represent multiple emotions at the same time. Jørgensen [9] states that game audio has a dual role: it supports the general feeling of an environment, but also gives vital information during gameplay. By having three layers, we can include an atmospheric track based on past feelings and common dual emotions like "happiness-sadness", "fear-disgust" or "disgust-anger".

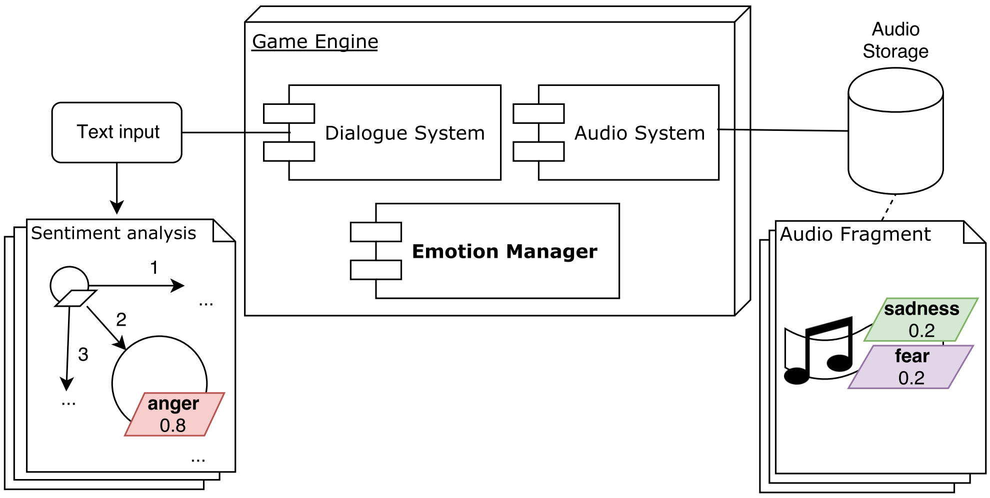

Figure 2. Detail of how LitSens' dialogue and audio systems fit in the game engine.

4 Discussion and future work

An audio system like the one we present here has, in our opinion, more potential for creating truly adaptive soundtracks for interactive experiences than totally procedural content. As all fragments in the audio database must be created by a human, they can sound credible and realistic, while maintaining enough flexibility to adapt to a variety of player responses. This, however, means there must exist some human work behind the scenes for the system to work. Still, as most user reactions are difficult to plan or design in advance, our approximation is to allow for complex player decisions while producing a fully adaptive soundtrack at a relatively low cost.

The next natural step for our system would be to validate its functioning with real subjects in an interactive experiment resembling a commercial, narrative video game. During said experiment, we should be able to recover two fundamental pieces of information. On one hand, we intend to retrieve data regarding the sort of emotions users feel while playing our game, by means of a tool like the Self-Assessment Manikin Test (SAM) [10]. On the other, we will evaluate how having an adaptive music system influences presence [11–14].

Acknowledgements

This article was funded by the Complutense University of Madrid (grant CT27/16-CT28/16 for predoctoral research), in collaboration with Santander Bank and NIL research group.

References

1. A. Pak and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," *In Proceedings of the Seventh Conference on International Language Resources and Evaluation*, pp. 1320–1326, 2010.
2. M. López Ibáñez, N. Álvarez, and F. Peinado, "Towards an Emotion-Driven Adaptive System for Video Game Music," in *Proceedings of the 14th International Conference on Advances in Computer Entertainment Technology (Pending publication)*, 2017.
3. U. Krcadinac, P. Pasquier, J. Jovanovic, and V. Devedzic, "Synesketch: An Open Source Library for Sentence-Based Emotion Recognition," *IEEE Transactions on Affective Computing*, vol. 4, no. 3, pp. 312–325, 2013.
4. U. Krcadinac, J. Jovanovic, V. Devedzic, and P. Pasquier, "Textual Affect Communication and Evocation Using Abstract Generative Visuals," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 3, pp. 370–379, 2016.
5. W. Strank, "The Legacy of iMuse: Interactive Video Game Music in the 1990s," in *Music and Game: Perspectives on a popular alliance*, pp. 81–91, Wiesbaden: Springer Fachmedien Wiesbaden, 2013.
6. M. O. Jewell, M. S. Nixon, and A. Prugel-Bennett, "CBS: A concept-based sequencer for soundtrack composition," in *Proceedings - 3rd International Conference on WEB Delivering of Music, WEDELMUSIC 2003*, pp. 105–108, 2003.
7. K. Collins, "An Introduction to Procedural Music in Video Games," *Contemporary Music Review*, vol. 28, pp. 5–15, feb 2009.
8. P. Ekman, "An argument for basic emotions," *Cognition & Emotion*, vol. 6, no. 3, pp. 169–200, 1992.
9. K. Jørgensen, *"What Are These Grunts and Growls Over There?" Computer Game Audio and Player Action.* PhD thesis, 2007.
10. M. M. Bradley and P. J. Lang, "Measuring emotion: the Self-Assessment Manikin and the Semantic Differential.," *Journal of behavior therapy and experimental psychiatry*, vol. 25, no. 1, pp. 49–59, 1994.
11. W. Barfield and S. Weghorst, "The sense of presence within virtual environments: A conceptual framework," *Advances in Human Factors Ergonomics*, vol. 19, p. 699, 1993.

12. B. G. Witmer and M. J. Singer, "Measuring presence in virtual environments: A presence questionnaire," *Presence: Teleoperators and Virtual Environments*, vol. 7, no. 3, pp. 225–240, 1998.
13. M. Lombard, T. Ditton, and L. Weinstein, "Measuring Presence: The Temple Presence Inventory," in *Proceedings of the 12th Annual International Workshop on Presence*, pp. 1–15, 2009.
14. J. Lessiter, J. Freeman, E. Keogh, and J. Davidoff, "A cross-media presence questionnaire: The ITC-Sense of Presence Inventory," *Presence: Teleoperators and Virtual Environments*, 2001.