Applying case-based reasoning in social computing to transform colors into music

María Navarro-Cáceres *, Sara Rodríguez, Javier Bajo, Juan Manuel Corchado

Calle Espejo, 2 - 37007 Salamanca, Spain

ARTICLE INFO

ABSTRACT

Social contexts are essential for human decision-making, providing the basis for inferences and activities planning. Computationally, this social information is incorporated to the concept of social computing, and applied in very different research fields, such as Medicine, Cyber Security or Economy. The specific aim of this work is to apply social computing in a musical context. We want humans and machines to cooperate for the purpose of generating music from the colors of a digital image; the colors of an image are used as inspiration for the generation of melodies. The social machine is implemented as a Virtual Organization (VO), which contains a case-based reasoning process to select the best combination of sounds extracted from the colors according to the social context. In order to evaluate the present approach, a small society of musical experts rate the sound produced following consonance criteria.

1. Introduction

Human beings commonly live in a social context, taking different roles in this society, cooperating and making decisions according to the surrounding social information, often comparing with previous similar experiences. This inherent social facet in daily life supposes a challenge to incorporate social context information and improve the interaction between social groups and machines. Thus, social machines emerged to solve this problem and introduce a new approach to design digital systems that makes information visible to the users (Erickson and Kellogg, 2000).

This social paradigm has been applied in various computational contexts to provide a social interaction among people (Wang, 2007), to support online communities (Deng and Tavares, 2013), to imitate a social community (Girard et al., 2013) or to serve as recommender systems (Wang et al., 2007). One of the best implementations of social computing are Virtual Organizations (VO) based on Multiagent Systems (MAS), due to their flexibility in dynamic environments, and their properties such as autonomy, social behavior or reasoning. These properties can be modeled by using different methodologies. In particular, case-based reasoning (CBR) has been applied to control the organization behavior successfully, allowing a better adaptation of the VO to a social environment based on previous social experience (Corchado et al., 2013), following a very similar process to human beings as they learn social behavior.

This paper presents a social computing model which creates cooperation between humans and machines. Our motivation is twofold. One of the aims of this paper is to musically express a pictorial work of art. We aim to compose a melodic sound cloud, that is, the generation of a melody inspired by an image. It is not expected to be a full tonal piece, although some rules of tonal music were followed. On the other hand, we want to develop a social machine that is able to learn from the context of social opinions about the quality of the generated music. For this purpose, the inclusion of a CBR architecture is essential; CBR is able to extract valuable musical knowledge from previous experiences.

The social machine allows real-time sharing of knowledge and information about most musical activities, and the ability to put people in touch with others who have an interest in the same musical contexts. The agents proposed in this architecture incorporate case-based reasoning as an inference mechanism to adapt to the changes that occur in the environment. The use of a paradigm based on virtual organizations (O'Leary et al., 1997) and social computing (Wang et al., 2007; Vassileva, 2012) makes it possible to design the multi-agent and controlled systems to emulate human organizations. These techniques will facilitate the ability to approach problems in a distributed fashion and from both a social and a computational perspectives.

In our previous work (Navarro-Cáceres et al., 2017), a social machine was developed to involve humans and machines in a creative process and transform a picture into a musical composition. The VO defines specialized roles for extracting sounds from the color pixels of

---

an image. Agents then begin an iterative process based on a swarm algorithm to create sound according to different rules about swarm intelligence and music composition. The prototype built to reproduce a social machine was evaluated by experts who rate the sounds produced by applying novelty and consonance criteria (Navarro-Cáceres et al., 2017). However, this evaluation was not applied as a feedback for the machine learning process. In the present work, we improve the design of the social machine enriching its machine learning capacities. We incorporate these results to a social machine and build a CBR architecture to permit the machine to evolve and improve the music according to general musical tastes.

The next section briefly explains the concept of social machines and CBR. The previous architecture for creativity scenarios, based on social computing is detailed in Section 3 and the improvements made are explained in Section 4. Section 5 presents the experiment carried out with the preliminary results obtained, comparing our previous system and the present one. Finally, Section 6 discusses the implications of the proposal and future work.

2. Social machines and virtual organizations of agents overview

This section gives an overview on the concept of social computing and the case-based reasoning paradigm.

2.1. Social computing

Social Computing is being applied in many research fields today. Internet has provided a new way to share social information through different platforms, data very useful to evolve new ways of computation based on different social behaviors.

Despite the amount of applications that involves social computing, it does not exist an uniform definition of social computing yet. Swartout et al. (2006) define Social Computing as the computational facilitation of social studies and human social dynamics as well as the design and use of information and communications technologies (ICT) that consider social context. Social Computing can be also described as any type of computing application in which software serves as an intermediary or a focus for a social relation (Schuler, 1994). For Charron et al. (2006), social computing contains a social structure in which technology puts power in individuals and communities, not institutions. Wang et al. (2007) describes the concept of social computing as the computational service of social studies and human social dynamics as well as the design and use of those tools that consider social context. Robertson and Giunchiglia (2013) indicates that the power of the social computer resides in the programmable combination of contributions from both humans and computers.

Social computing is a concept that can be implemented within diverse architectures capable of regulating social patterns and norms. One of the options here applied are Virtual Organizations (VO). According to Rodriguez et al. (2011), a VO consists of an open system that permits the collaboration of heterogeneous entities. Designing a VO involves a description of elements, incorporating norms, behavior and structural compositions. To do so, we need to follow a methodology which guides the construction of a VO. Section 3 will detail briefly the steps followed to design a VO architecture.

2.2. Case-based reasoning systems

Case-Based Reasoning (CBR) is a type of reasoning modeled by observing human behavior to solve new problems based on past experiences. If a past problem was solved using a particular solution and a certain result was also obtained, it would be logical to use that past experience to solve a new problem with similar features. Thus, if the result obtained after applying the solution in the past was good, then it seems logical to apply a similar solution to solve the current problem. On the contrary, if the result obtained in the past was not good, then the logical choice would be to modify the solution for a better result.

We must take into account two aspects of the reasoning model based on experience. First, the model is based on the idea that similar problems have similar solutions. However, a lack of similar problems does not mean that the system is not able to propose good results, as the reuse of past memories can become a creative process to learn about the new experience. Secondly, a value judgment is always needed to determine whether a solution is good or not. This judgment can be issued by an expert on the problem solved to increase the learning ability (Riesbeck and Schank, 2013).

The working flow of a CBR system is known as the life cycle, which consists of four sequential processes (Aamodt and Plaza, 1994): Retrieve, Reuse, Review and Retain. These processes are based on the concept of case, defined as a portion of contextualized knowledge that represents an experience, and makes it possible to achieve the solution for a current problem. Thus, a case consists of a description of the problem, the solution to that problem and the results obtained after applying the solution. We can use a mathematical notation in which a case is represented by a 3-tuples, $\langle Problem, Solution, Result \rangle$.

The four stages previously named are applied as follows. When a new problem is presented to a CBR system it uses past experience to get the best solution. The first step followed is the recovery phase. In this stage, the system searches cases with a similar description of the current problem. The next phase executed is the adaptation phase, which works with solutions corresponding to the most similar cases recovered in the previous phase. The result of the adaptation phase is a solution to the current problem. The proposed solution is evaluated in the review phase, which checks its validity. Finally, in the learning phase, the system stores the new experience and learns from it. As the reader can appreciate, there can be multiple possibilities for running each of the steps of CBR cycle. For example, to retrieve similar cases, different techniques and similar algorithms can be applied.

The CBR can be implemented by different kind of agents. The model followed here is the Case-Based Planning (CBP) agent. A CBP agent is a particular type of CBR agent, which uses a CBR system (Glez-Bedia et al., 2002) to generate plans from cases. There are various works applying a CBP in dynamic contexts. SHOMAS (Bajo et al., 2009) is a guidance platform in shopping centers that uses a CBP based on BDI architecture. Chamoso et al. (2016) proposes a CBP to control electric power networks. Shen et al. (2015) platform aids clinical decision based on MAS and CBR along with medical knowledge. The referenced works demonstrate that agents and CBR are successfully applied in different fields to solve many problems. We will try here to use a CBR to generate creative musical content.

3. Social machine

We aim to develop a social machine capable of generating music from images provided by users. In a previous work (Navarro-Cáceres et al., 2017), the social interaction consisted of uploading images and rating the final results. However, the present proposal additionally incorporates feedback mechanisms to consider humans' opinion to improve the musical results in a CBR architecture.

Fig. 1 represents a social model where humans (providers and experts) collaborate with the intelligent system performing human tasks that cannot be carried out by a computer. Providers upload the digital image to create the composition, while Experts evaluate the quality of the musical result.

The overall work flow is shown in Fig. 1. Initially, a provider interacts with the machine by providing a digital image, essential for our system to work. This picture is considered the input of the machine component in the social machine. The machine components extract the color from the image in HSL (Hue, Saturation, Luminosity) codification. This information about the extracted colors is given to the planner module, which analyzes whether there are previous cases with similar features (similar color composition in the image) to be retrieved and adapted to this new case. These cases are stored in a database that can