The Common Sense Disc Jockey

Luke Ouko, Arnan Sipitakiat, Carla Gomez-Monroy, Joan M. DiMicco

MIT Media Lab
20 Ames St
Cambridge, MA, 02139
+1 617-253-9326
{ouko, carlagm, arnans, joanie }@media.mit.edu

Abstract. In this paper, we introduce the Common Sense DJ (CSDJ), a technology to aid DJs in music selection. It uses ThoughtTreasure (TT) as a common sense database to aid in selecting the optimal music for a crowd, given certain parameters. ThoughtTreasure handles the reasoning aspect and has *critics* to facilitate in making better decisions. A camera captures the crowds' level of response to the music. The CSDJ responds after going into TT's database, finding the necessary common sense knowledge, and validating its usefulness through the TT Prover. Given certain goals, the application is able to learn. We demonstrate that using a common sense database such as ThoughtTreasure we can develop more intelligent applications.

1 Introduction

Being a good DJ requires vast practical experience with many factors, some of which are music, people who dance for fun, their culture, moods, and trends. All of these factors go from overtly noticeable or learnable to being extremely nuanced. Good DJs skilfully tune into all of these elements and their interplay, at least most of the time. The CSDJ is intended to act as a DJ aid, not to substitute one. It helps in reducing the over-head of juggling multiple variables in order to provide the DJ's audience with an optimum performance. The DJ is thus left to concentrate more on the creative process of selecting songs –or parts of songs– and sequencing them depending on which go best with each other, in which order, beat, and tempo and with what DJ-improvised/pre-programmed effects and comments; most importantly, the DJ needs to concentrate on exactly when to punch in the next piece. This creative process "is manifestly dependent on the DJ's personal taste in music" [1.] The CSDJ thus supplements the DJs knowledge base.

The Common Sense DJ is an application employing common sense to build a play list for a dance club environment. It uses a situation's current context to suggest the best decision given the prevalent environment. As the context changes over time, the application continually updates its suggestions, resulting in an application that is flexible enough to adapt to and learn from its prevailing context within the domain of a dance club setting. Thus the CSDJ observes the current population of the club:

"Who are the people? What do they look like? What are they doing?" And thereafter makes a suggestion about the best type of music for the crowd. Imagine the scenario of a dance club in a tourist destination, where the club management never knows who will show up on a given night. Tonight it might be a group composed of Thai teenagers, tomorrow night it might be a mix of Europeans and Americans.

Our human common sense gives us some intuition about what kind of music these different groups might like dancing to, but it becomes increasingly difficult to manually select the best songs on the fly. The CSDJ offers a solution to this problem, by first allowing a real-life DJ to supply basic observations to the system, and then by returning a suggestive play list of the songs that are most likely to get the crowd dancing. The application also has the ability to learn through the night, with a camera sensor detecting the number of people dancing. When the system's best suggestion does not get the crowd dancing, the system will adjust its common sense understanding of the crowd's music preference and suggest different songs to fit the given criteria and the new knowledge.

> *Common sense is not a simple thing. Instead, it is an immense society of hardearned practical ideas—of multitudes of life-learned rules and exceptions, dispositions and tendencies, balances and checks.*
>
> *Marvin Minsky [2.]*

The common sense knowledge and reasoning is implemented in ThoughtTreasure [3.] ThoughtTreasure (TT) is both a common sense knowledgebase and architecture for natural language processing (NLP). We use the knowledge in TT to get facts about people and music styles and then used the TT "Prover" to make inductions about different groups and their music preferences.

2 The Common Sense Disc Jockey Application

2.1 Gathering Input

Data for TT comes from the Java application, both through a feedback form and through the tracking of the number of people dancing with a video camera.

2.2 The Java Interface

The graphical user interface of the CSDJ is a Java Swing application, which in addition to providing the visual interface to the system, also manages all communication between the music database, and camera application that acts as the dance detector. There are four sections to the graphical interface, paralleling the communication with the different components of the CSDJ: the demographic input screen, the reasoning

display, the suggested play list, and the music player. Figure 1 shows a screen shot of the interface.

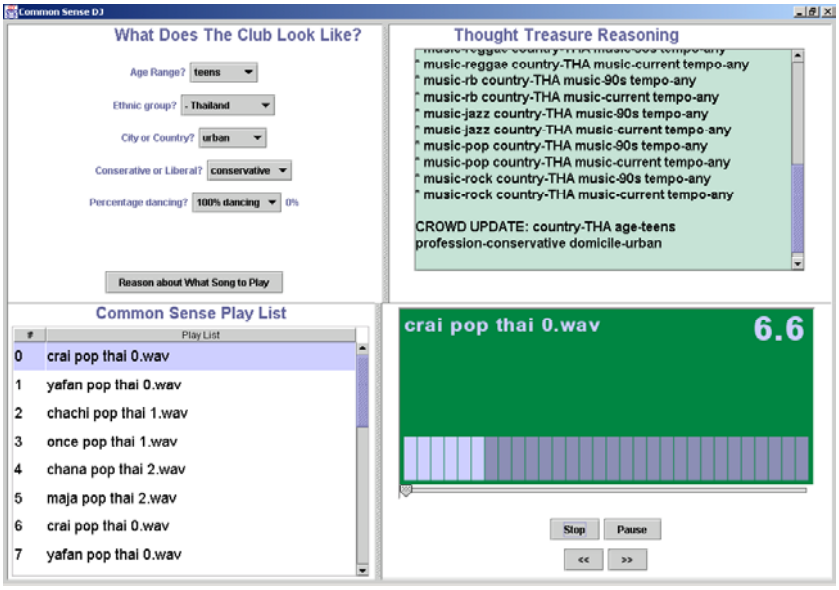

Fig. 1. This is the CSDJ Graphical User Interface.

The demographic input screen sends the dance club's current state to TT through TT's Java API, including the type of people in the club and the percentage of people dancing (this value is obtained and sent from the Dancing Detector). The reasoning display on the upper right outputs the TT reasoning information sent to the interface. This information is then used to filter down the application's music database of wav files to a suggested play list. This play list is displayed in the lower left corner of the interface. When the songs are played, the current song being played is displayed in the lower right corner.

2.3 The Dancing Detector

To detect the percentage of people dancing in the room, we modified an existing application built by the MIT Media Lab's House_n research group. The application uses a top-down camera tracker (specifically, an Intel web-cam) for input on the total number of people in a room and the number of those that are dancing. The application uses a real time multi-person/object tracking algorithm that has multiple hypotheses reasoning to enforce multi person match constrains. Reasoning is achieved by clustering the pixel quantities at a given location and determining how the cluster changes over time. (Figure 2). Once the number of people and the number of those dancing is calculated, the camera application sends the percentage of people dancing to the

Common Sense DJ's Java application layer. The Java application then updates its current knowledge about the dance club's state and sends it to ThoughtTreasure for suggestions.

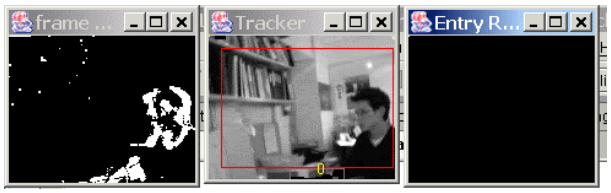

Fig. 2. Image Capture and Processing of Camera Application.

3 ThoughtTreasure

ThoughtTreasure (TT) is the core of our application: it manages the collection and storage of common sense knowledge and performs all the reasoning on this knowledge before sending suggested music types to the Java interface. TT uses a hierarchical storage structure and further categorizes its knowledge into three categories: goals, facts, and rules. This structure simplifies the process of reasoning, by enforcing the structure that facts and rules must be combined to achieve goals. TT also has a natural language processing component, which could for example, be used to receive open-text suggestions from people in the dance club.

3.1 ThoughtTreasure Facts

Facts in TT are statements about relations between concepts and objects that hold true. For the purpose of our application, facts contain knowledge that is currently true about the environment. Thus facts can be continuously changing, and this is done through the interaction of the Java interface and TT.

We added numerous new facts into ThoughtTreasure's common sense database, supplementing its knowledge relating to people, cultures, and music. Examples of these facts:

- Culture (each continent and country within)
- Ages (Teens, 20s, 30s, 40s, 50s, 60s)
- Professions (Conservative or Liberal)
- Patrons' Domicile (Rural or Urban)

As earlier mentioned, the data in TT is stored in a structured format. Thus each fact follows a particular syntax. Examples of some of the facts added are below:

```
[the-people-of crowd1 country-USA]

[age-of crowd1 age-20s]

[profession-of crowd1 profession-conservative]

[domicile-of crowd1 unknown]
```

3.2 ThoughtTreasure Rules

Rules in TT generate all possibilities and intersects of the provided facts, drawing ties together such as if the dance club crowd has Thai teenagers, then these teenagers are also from Asia and people from Asia like different types of music from Asia, such as Thai pop music.

Sample TT Code:

```
[ifthen [and
          [ profession-of ?crowd ?profession]
          [ ne ?profession unknown]
          [ music-category-of ?music-genre ?profession]
         ]
  [music-genre-by-profession-of-crowd ?music-genre]
]
```

The second line of the above rule pulls the fact about crowd profession from the Facts file. Referring to the facts given in 3.1, the variable ?crowd and ?profession would have the values of "crowd1" and "profession-conservative" respectively. The third line makes sure that the value of ?profession is not "unknown." The fourth line pulls out music genres that match the given profession as defined in ThoughtTreasure's database. The final line generates the results. In this case, it returns all the music genres that match the rules.

3.3 ThoughtTreasure Goal/s

The "goals" defined within TT drive the reasoning, by defining the goal of the reasoning process. In our application, the goal is to find the best music suggestion for the current crowd.

Sample TT Code:

```
[music-genre-of-crowd ?music-genre ?country ?music-era ]
```

The Prover returns results in forms of statements that match the goals. The rules take all the facts in ThoughtTreasure's database as its input and generate all possible combinations of statements. These statements are then compared to the goal statements. All statements that match will be returned as results.

To the best of our knowledge this application is the first instance of actual application of the TT Prover.

4 ThoughtTreasure Critics

ThoughtTreasure can quickly reason through facts and rules to achieve goals, but frequently this results in too many suggestions due to the absence of "Higher Level Common Sense." An example is that TT might find that Chinese people like Chinese samba. While within the structure of TT's knowledge of music and culture this is logical, it is not common sense because there is no such genre of music. Because of the possibility of non-common sense results like these being generated and because of situations where TT returns too many suggestions to be of any practical use, we built an enhancement to ThoughtTreasure referred to as "C*ritics."* [4.]

We use critics to make validations of the data and to reaffirm certain common sense rules. By checking TT's suggestions and reapplying certain rules, we were able to refine TT's suggestions to the Java interface. Here are two rules that have been implemented:

4.1 Re-evaluation rules:

The critic tries to identify a cause for the conflicts in the results by looking at the input variables. If there is a match the critic will then apply an alternate solution for music selection. For example, if culture of the crowd is unknown then music suggestion is likely to be unusable. Thus, it is better to play available cross-culture music instead.

4.2 Hypothesize rules:

In some situations the critic will try to guess the music preference of the crowd. The feedback from the camera will then be used to evaluate the hypothesis. For example, if domicile of the crowd is unknown and too many suggestions have been created then the critic will guess that domicile is urban. If this guess increases the crowd's activity (this can be determined by the feedback from the camera) then the critic as-

sumes it has made the correct assumptions and assign urban to the domicile attribute. If the guess fails to increase crowd activity then the critic tries the opposite. If none of the experiments are successful then the critic terminates the experiment and moves on to the next critic rule.

Ideally we would store the critic rules in a database where new rules can be added or learned in the same way as facts. ThoughtTreasure itself can be used to store these critic rules, but the rules must be treated as temporal facts that can change over time. The rules in this implementation of CSDJ are built into the critic.

5 Machine Learning

The aim of having machine learning in this application was to enable the creation of new rules that are understandable i.e. creating new common sense statements. This task would be achieved by learning actively, from prior knowledge as well as incrementally.

Our independent variables include: The demographics of the audience, the number of people dancing, the music preference of the people dancing, the details of the song playing as well as details on location, time, event, season, etc.

Our dependent variable would be the goal to be achieved, *e.g.,* given the crowd, what is the best genre of music to play given their demographics or prevailing profile of people to ensure dancers > 60%.

We have created two instances of machine learning in CSDJ. First, when the dependant variable drops below the desired threshold (*i.e.* when more than n% of the crowd are not dancing) the machine learning agent will conclude that the current music selection does not work and will add a new rule to the database preventing it from producing this same suggestion again. Similarly, if the dependant variable becomes unsatisfactory and the learning agent detects that the critic is executing a hypothesized rule (*i.e.* guessing that the crowd is liberal) then it informs the critic of the existence of a false hypothesis.

The application can be enhanced to use data mining techniques to learn new rules from the available data.

6 An example scenario of the CSDJ at work

When the system knows everything about its audience, commonsense does not play a big role in the suggestion process, as it is relatively straightforward. In reality, however, the crowd's profile is not always accessible right from the beginning. Here is where commonsense plays a major role in the suggestion process. Consider the following example:

In the beginning of a party, the crowd appears to be young Asians, but the system does not know for certain which country they are from. Their profession and domicile are also unknown.

In this case the system would conclude that it is best to play cross-cultural music. Thus, the system would suggest international pop singers such as Britney Spires, Madonna, or Back Street Boys. Here's how the conclusion is derived:

- The system generates all possible songs types from the information it has about the crowd. Since there are many countries in Asia, the possible music combinations are very high.
- The CSDJ critic detects this situation and decides to re-evaluate its suggestions.
- The critic searches its knowledge database and sees that there are too many countries in Asia to be specific.
- The critic suggests cross-cultural music.

7 Preliminary Usability Studies

While the application is specifically meant for DJs and to enhance their performance, it is ultimately supposed to benefit the audience. User testing thus needs to be carried out on both the DJs and their audiences.

The Common Sense DJ usability study is yet to be carried out. All the usage so far has been limited to our own in-house experimental testing of the application. We have only recently completed building the application.

We hope to test the application in different kinds of settings with a view to collecting data for evaluation and future fine tuning purposes. We will thus initially give it out to a few DJs (4-5) to use it with the simulated audience profiles that are available off the application. We would hope to capture at this stage the major problem areas with the system at this stage. Thereafter we can take it to different dance clubs and special events that will allow us to observe the different setting-bound reactions and responses.

We can then collect feedback at given sessions from DJs, dancers, club staff, and invited observers. The feedback can be of two kinds, objective and subjective. Objective feedback will be obtained through measuring devices such as the Dancing Detector, a video camera application that maps Dancer/Non-Dancer ratios. Subjective Feedback can be gathered by existing surveying techniques or by creating ad hoc mechanisms. The mere fact that the audience is having fun does not negate the possibility of gaining useful information from them. Persuasive surveyors or giving discounts, free passes, samples, and free drinks may be ways to go around this problem.

8 Other Potential Users

Our current application allows the usage of dummy crowds or audiences. This would be very useful for trainee DJs and inexperienced professionals. By performing to these audiences the DJs can learn how to best deal with different circumstances as well as what really works best in certain situations. All this happens in a simulated

environment without the added pressure of a real crowd. Even expert DJs could use it by simulating conditions that they may not have experienced and how they would deal with them.

9 Other Applications for DJs

Most applications that are currently available for DJs are restricted to blending or mixing music with the underling assumption that the profile of the crowd is known or is a given and that their preferences can be predetermined. Certain applications help in the automation of play lists or the organization and management of music.

Some other applications for DJs are tools for automatic generation of specific music play lists that use adaptive search [5.]

The only related application for DJs that takes cognizance of the audience and therefore determines performance of an action given the affective reaction of the dancers known to us is the Affective DJ. [6.]

CSDJ not only aids in the selection of Music it also observes the reaction of the crowd to the music and keeps a record of the lessons learned.

Joe Paradiso of the MIT Media Lab and his group (Responsive Environments) have developed a set of very low-cost, wireless, wearable sensors that enable a large group of people (e.g., hundreds or thousands) to participate in an interactive musical performance. The sensors themselves are simple piezoelectric accelerometers that detect extremes of limb motion, upon which they transmit a narrow RF pulse.

Although one can distinguish between sensors on the upper and lower body by using a different carrier frequency, and zone the locations of activity roughly through carrier strength, they do not plan to independently ID each performer, but instead to measure and react to the characteristics of ensemble behavior. They have built this system and have developed algorithms that use this data to explore techniques of mapping large-group, real-time musical interaction. [7.]

10 Future Work

There are many future directions we would like to take with this work. On a technical implementation level, we would like to continue working to refine and enhance ThoughtTreasure's reasoning ability. One way to do this will be to further refine the critics, which evaluate the reasoning conclusions TT makes. Another enhancement would be to add the ability for more robust data mining in TT. It would aid in creating better rules as well as providing for a better base from which critics would work.

To further demonstrate our goal of building an application using common sense to adapt to a changing environment, we would like to incorporate other sources of common sense into the application, such as the Open Mind common sense data [8.]

11 Conclusion

In conclusion, the Common Sense Disc Jockey demonstrates how an application utilizing common sense can build an understanding of an environment, reason about the best decision to make, and produce an adapting list of suggested actions. Within the domain of music selection, the CSDJ is a tool for DJs to automatically filter music with common sense knowledge of the type of music different types of people like. An additional feature of the CSDJ application that is particularly applicable to a dance club setting is the ability to learn from the dancers more about their preferences for different types of music. This refinement of the learning allows the system to develop more common sense over the course the night.

A talented DJ incorporates artistic and creative expression into his/her music selection and between-song transitions. The CSDJ does not attempt to eliminate or replace the talent of a real-world DJ, but rather provides a tool for the DJ to work with in crafting his/her creative expression. By filtering the DJ's music collection down to what the current crowd prefers, the CSDJ allows the DJ to focus on the artistic nature of the craft.

Acknowledgements

We thank Henry Lieberman, Push Singh, and Hugo Liu for providing invaluable design suggestions for this project. Lastly, our project could not have been completed without the tireless efforts of Eric Mueller, creator and designer of ThoughtTreasure.

References

1. Cliff, D.: Hang the DJ: Automatic Sequencing and Seamless Mixing of Dance-Music Tracks. HP Laboratories Bristol. August, 2000. http://www.hpl.hp.com/techreports/2000/HPL-2000-104.pdf.
2. Minsky, M.: The Society of Mind. New York: Simon & Schuster. 1985.
3. Mueller, E.T.: Natural language processing with ThoughtTreasure. New York: Signiform. (1998) Available: http://www.signiform.com/tt/book/.
4. Minsky, M.: The Emotion Machine. December 2002. http://www.media.mit.edu/~minsky
5. Aucouturier, J.J, & Pachet, F.: Scaling Up Music Playlist Generation. In IEEE International Conference on Multimedia and Expo (ICME 02), Lauzanne. August 2002. Available: http://www.csl.sony.fr/downloads/papers/2002/pachet02d.pdf.
6. Healey, J., Picard R. and Dabek F. The Affective DJ: Music Selection with the New Affective Wearable. Available: http://vismod.www.media.mit.edu/tech-reports/TR-478/node4.html.

7. Joe Paradiso., Inexpensive Wearable Sensors for Large-Crowd Interaction., Available www.media.mit.edu/resenv/disposables.
8. Open Mind., Available: www.openmind.org.