Bardo: Emotion-Based Music Recommendation for Tabletop Role-Playing Games

Rafael R. Padovani
Departamento de Informática
Universidade Federal de Viçosa
Viçosa, MG, Brazil

Lucas N. Ferreira
Department of Computational Media
University of California, Santa Cruz
Santa Cruz, CA, USA

Levi H. S. Lelis
Departamento de Informática
Universidade Federal de Viçosa
Viçosa, MG, Brazil

Abstract

In this paper we introduce Bardo, a real-time intelligent system to automatically select the background music for tabletop role-playing games. Bardo uses an off-the-shelf speech recognition system to transform into text what the players say during a game session, and a supervised learning algorithm to classify the text into an emotion. Bardo then selects and plays as background music a song representing the classified emotion. We evaluate Bardo with a Dungeons and Dragons (D&D) campaign available on YouTube. Accuracy experiments show that a simple Naive Bayes classifier is able to obtain good prediction accuracy in our classification task. A user study in which people evaluated edited versions of the D&D videos suggests that Bardo's selections can be better than those used in the original videos of the campaign.

Introduction

In this paper we present Bardo, a real-time intelligent system to automatically select the background music for tabletop role-playing games (RPGs). Our goal is to enhance the players' experience through a selection of background music that accounts for the game's context (e.g., if the players are going through a suspenseful moment in the game, then Bardo should play a music to match that moment). The object of our research is Dungeons and Dragons (D&D), a tabletop RPG in which the players interpret characters, known as player characters (PC), in a story told by the dungeon master (DM). The DM tells the story and interprets all other characters in the story, which are known as the non-player characters (NPC). PCs have ability scores and skills that determine if actions performed in the game (such as attacking an opponent or picking a lock) are successful.

Aiming at enhancing their immersion, D&D players often manually select the game's background music to match their story (Bergström and Björk 2014). Unless one of the players constantly selects the background music according to the story's context, the music might not match the emotional state of the game (e.g., the PCs could be battling a dragon while a calm music is being played). Having one of the players constantly selecting the background music is not ideal, as the player might get distracted from the game.

Bardo uses supervised learning to select the background music based on what the players say in the game. This is achieved by employing an off-the-shelf speech recognition (SR) system to capture the players' sentences, which are then classified according to a model of emotion we introduce for the game of D&D. Our emotion model is categorical and includes the following emotions: Happy, Calm, Agitated, and Suspenseful. Bardo's supervised learning algorithm maps a set of sentences to one of the four emotions of our model. The system then plays as background music a song that expresses the classified emotion. We assume the songs to be already labeled according to our emotion model and to be provided as input by Bardo's users.

We evaluate Bardo with an online D&D campaign (i.e., a story played through multiple sessions of the game) available on YouTube.[1] This online campaign offers a great environment for our research because the videos allowed us to use YouTube's SR system to automatically convert voice to text (Harrenstien 2009). Moreover, the songs played as background music in the original videos offer a baseline to our experiments, although with caveats.

Two algorithms were tested with Bardo: a lexicon-based approach that considers the density of emotions in sentences (D) and a Naive Bayes classifier (NB). Experiments showed that NB outperforms D. We then conducted a user study in which we use Bardo with NB to select the background music of excerpts of the original videos of the D&D campaign. We compare Bardo's selections with those made by the video authors. Although with caveats, the results of our study show a clear preference by the participants for Bardo's selections.

To the best of our knowledge, Bardo is the first real-time intelligent system for automatic background music selection for tabletop RPG games, and the first system to perform emotion-based music selection based on people's speech. Another contribution of this paper is a labeled dataset of sentences from the D&D campaign we use in our experiments.

Bardo opens several directions for future research as the system is likely to be directly applicable to other tabletop games, specially storytelling-based games. Moreover, Bardo could be applied beyond the realm of games, in problems such as background music selection for daily events such as

[1] https://www.youtube.com/playlist?list=PLQOB_yCwC5J2m1lKNxYQfwyAhp2MRihAb

having dinner with friends or hosting a party.

Related Work

Bardo relates to works on text-based emotion recognition (Strapparava and Mihalcea 2008; Alm, Roth, and Sproat 2005) which develop computational techniques to recognize emotions (e.g., anger, sadness, and surprise) from text. To model emotions these techniques can use either categorical or dimensional approaches. Categorical models use discrete labels to describe affective responses (Dalgleish and Power 2000) and dimensional ones attempt to model an affective phenomenon as a set of coordinates in a low-dimensional space (Posner, Russell, and Peterson 2005). Text-based emotion recognition systems are usually lexicon-based or machine learned-based, and they are often applied to problems such as sentiment analysis (Pang, Lee, and others 2008), computer assisted creativity (Davis and Mohammad 2014) and text-to-speech generation (Alm 2008).

Davis and Mohammad (2014) used a lexicon-based approach to emotion recognition similar to the one we use in this paper. Their approach was used to classify emotions in novels and later generate music for them. Strapparava and Mihalcea (2008) used a similar lexicon-based method to classify the emotions in newspaper headlines and a Naive Bayes classifier to detect Ekman's emotions (Ekman 1999) in blog posts.

Balabantaray et al. (2012) presented an emotion classifier that is able to determine the emotion of a person's writing; their approach is based on Support Vector Machines. Suttles and Ide (2013) used a Distant Supervision approach (Mintz et al. 2009) to classify emotions considering the eight bipolar emotions defined by Plutchik (1980). This allowed them to treat the multi-class problem of emotion classification as a binary problem for opposing emotion pairs.

Bardo also relates to emotion-based music recommendation systems, which recommend music to match the physiological state of the user (Song, Dixon, and Pearce 2012). Cai et al. (2007) proposed an emotion-based music recommendation approach called MusicSense to automatically suggest music when users read Web documents such as Weblogs. Andjelkovic et al. (2016) proposed an interactive system called MoodPlay to model the user profile based on a set of artists selected by the user. Deng and Leung (2012) considered the user's historical playlist and employed Conditional Random Fields (Lafferty et al. 2001) to predict the user's emotional state. Bardo differs from previous approaches because it performs emotion-based music recommendation considering the user's speech as input.

Bardo

Bardo uses the sentences provided by an off-the-shelf SR system to classify the emotion of the game's state according to the four emotions of our model. In contrast with many works that deal with text classification, sentences provided by the SR system are often grammatically incorrect and lack structure. This is because the SR system might not be able to properly translate what is being said by the players. Also, the system could add to the same sentence words said by multiple players. For example, one player could say *"I unsheathe my longsword and move toward the dragon."*, while another player says *"I run away looking for shelter."*. The SR system could capture parts of the two sentences, providing to Bardo a mixture of the two, e.g., *"I run away looking move toward the dragon."*. That is why we classify the emotion of the game based only on the bag of words returned by the SR system, we do not try to use the structure of the sentences, and whenever referring to a sentence, we are referring to a bag of words returned by the SR system. Also, since we use YouTube's system to generate subtitles, what we refer as a sentence is in fact a subtitle, i.e., a bag of words that appears at a given moment on a YouTube video.

Bardo's SR system generates text from what is being said by the players and the text is grouped into sentences. Each sentence is mapped by a supervised learning algorithm to an emotion in our categorical emotion model, which considers the emotions: Happy, Calm, Agitated, and Suspenseful. This model is a simplified version of Ekman's model (Ekman 1999), which considers the following emotions: anger, disgust, fear, happiness, sadness, and surprise. We believe our simplified model to encompass most of the emotions that appear in D&D narratives. Moreover, our model simplifies the supervised learning classification task by accounting for a reduced number of classes.

Since we assume the emotional state of the game to always be classified into one of the four emotions of our model, we can treat the problem of emotion classification in tabletop games as the problem of deciding when to change states in a finite state machine (FSM). In our FSM each emotion is represented by a state and a transition from a state to any other state is possible. We assume the Calm state to be the starting state in all D&D sessions. The state changes whenever the supervised learning model classifies a sentence to be of a state different than the current. Since the emotional state of D&D sessions do not change very frequently, some of the supervised learning algorithms we employ will only change states in the FSM when there is a "clear sign" of change—whenever in "doubt", Bardo will not change states. Tabletop games frequently have clear signs of emotional state changes, which are usually related to the game's mechanics. For example, whenever the DM asks for the PCs to *roll their initiative*, it means that a combat is about to start, indicating a possible change from the current emotional state to the Agitated state (if not already in that state).

Bardo requires as input a set of songs expressing each of the four emotions. Bardo plays a song provided as input for a given emotional state, and it changes the music being played whenever an emotional state change occurs.

Note that one could use other aspects of the speech such as vocal behavior as part of the input for selecting background songs for tabletop games. This is an interesting and challenging direction of future work. The use of vocal behavior for background music selection in games is challenging because often the players demonstrate an emotion in their voice that is different from the emotion of the scene being played. For example, the players of the D&D campaign we use in this paper often demonstrate happiness even when the

PCs are going through suspenseful or agitated scenes.

The Dataset Annotation Process

Our dataset includes the first 9 of the 12 episodes of a D&D campaign named *Call of the Wild* (CotW), which is available on YouTube. We did not use the last 3 episodes because they did not have the automatically generated subtitles available. The first 9 episodes have 5,892 sentences and 45,247 words, which result in 4 hours, 39 minutes, and 24 seconds of D&D gameplay—each episode is approximately 30 minutes long. CotW is played with D&D's 5th edition, and in addition to the DM, the campaign is played by 3 PCs, all players are male. CotW was the first D&D campaign for one of the PCs, all the other players had experience playing the game.

YouTube generates captions automatically by combining Google's automatic SR technology with its caption system (Harrenstien 2009). In our annotation process we label each sentence generated by YouTube's caption system according to our four-class emotion model.

The process of emotion annotation is hard in general due to its subjectivity. For example, the κ metric for inter-annotator agreement ranges from 0.24 to 0.51 and the percentage annotation overlap ranges from 45 to 64% for children's fairy tales (Alm, Roth, and Sproat 2005). We hypothesize that emotion annotation for tabletop games is easier as the games' mechanics might offer signs of the player's emotions. Moreover, our emotion model is simpler than the model used by Alm, Roth, and Sproat (2005) for fairy tales—while they use 7 emotions, we use 4. In order to test our hypothesis we enlisted three annotators to label the CotW campaign according to our emotion model. In addition to testing our hypothesis, we expect to produce a more accurate dataset by having three instead of one annotator.

Each annotator watched all 9 episodes and labeled all sentences. The annotation process was done by assuming the PCs' perspective in the story, as there could be moments that PCs and NPCs could experience different emotions. Should two annotators agree on the label of a sentence s, then s is labeled according to the two annotators. One of the annotators watched the videos again to break the ties (sentences that each annotator attributed a distinct label for).

The inter-agreement κ metric of our three annotators was 0.60 and the percentage overlap of the annotations was 0.79. These numbers are higher than what is usually observed in the emotion annotation literature (e.g., emotions in children's fairy tales). This result support our hypothesis that emotion annotation in tabletop games might be easier due to the clues offered by the game mechanics and to the simpler emotion model. Figure 1 shows details of the annotated emotions of all sentences in the first episode of CotW. The x-axis shows the sentences of the episode, ordered by appearance, and the y-axis the four emotions. Each color shows the emotion attributed by one of the annotators to a sentence. For example, in the beginning of the episode all annotators agreed that the emotion Calm represents well that moment of the story. A disagreement between the three annotators is observed around sentence 400, where one annotator believes it to be an Agitated moment of the story, while another believes it to be Suspenseful and the last believes it to be Calm.

#	Sentence	Emotion
s_{30}	word on the street is that there are	Calm
s_{31}	some people who want your blood just	Suspense
s_{32}	like you wanted theirs in the last part	Suspense
...	...	...
s_{131}	it again but hold on rorey shot once oh	Agitated
s_{132}	okay I get two shots another 28 to it	Agitated
s_{133}	yeah it's 12 inch the arrow second arrow	Agitated
s_{134}	hits him and he falls crumples hits a	Agitated
s_{135}	spacing it's a stone it dies on the spot	Agitated

Table 1: An excerpt of our D&D dataset.

After analyzing the video again, the tie-breaker annotator decided that the moment around sentence 400 is Agitated.

The dataset contains 2,005 Agitated sentences, 2,493 Suspenseful, 38 Happy, and 1,356 Calm. Although Happy sentences are rare in CotW, the emotion may appear more frequently in other D&D campaigns. For example, it is common for the story to start with the PCs meeting each other in a tavern, and such scenes might require Happy songs.

Table 1 shows an excerpt of our dataset, from sentence s_{30} to s_{32} and from sentence s_{131} to s_{135} of an episode (see column "#"). We use the subscript of a sentence to denote the index in which the sentence appears in an episode. For example, s_{134} occurs immediately after s_{133}. The second column of the table shows the sentences and the third the sentences' emotion according to the annotators. We highlight the lack of grammatical structure and meaning in some of the sentences, e.g., *it again but hold on rorey shot once oh*.

Density-Based Classifier (D)

Our Density-based method (D) classifies a sentence s by counting the number of words associated with a given emotion—the emotion with largest count is chosen as the emotion of s. The associations between words and emotions are provided by the NRC Emotion Lexicon (Mohammad and Turney 2013), which has approximately 14,000 English words annotated with eight different emotions: Anticipation, Anger, Joy, Fear, Disgust, Sadness, Surprise and Trust. In order to use the NRC lexicon we need to find a mapping between NRC's emotions and Bardo's. For example, a mapping would count all words in the Joy, Surprise, and Trust emotions of NRC as words of Bardo's Happy class.

We devised a method to choose the emotion mapping to be used. We divide our labeled dataset into test and training data: 1 episode in the test set and the remaining 8 in the training set. We perform a leave-one-episode-out cross validation procedure in the training set to measure the classification precision of all possible mappings. Since we need to map 7 NRC emotions into 4 Bardo emotions, there are $4^7 = 16,384$ different mappings to be tested. We use a brute-force search procedure to find the mapping that performs best in the cross-validation procedure in the training set to classify the sentences in the test set. This search procedure is D's supervised learning phase.

In addition to the automatic mapping of NRC emotions into Bardo's emotions, another enhancement we use is a

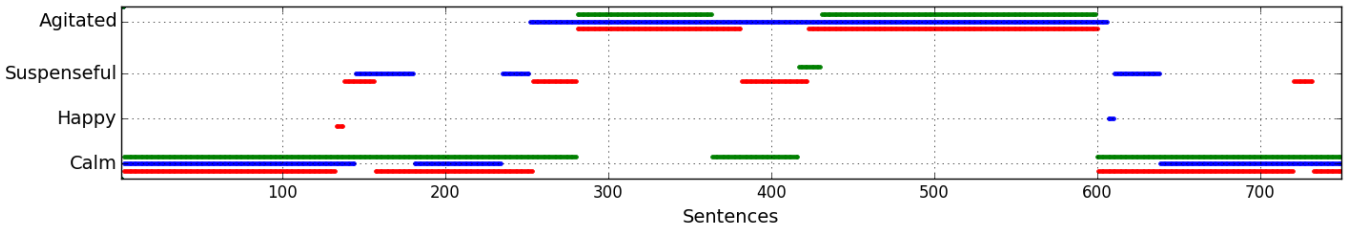

Figure 1: Inter-annotator agreement of the first episode of CotW.

density threshold d_t. Bardo only switches states in its FSM when D's emotion with the largest count exceeds d_t. This is because emotion transitions are somewhat rare in D&D sessions, thus ideally Bardo will switch states only when there is a clear sign of emotion change—we expect the density threshold d_t to avoid unnecessary and incorrect state changes. The value of d_t is also determined altogether with the emotion mapping in the leave-one-episode-out cross-validation procedure described above.

Since the number of words in a sentence is usually small (frequently smaller than 10), there is not much information within a sentence to accurately predict its emotion. Thus, to calculate the emotion density of sentence s_i, we use a sliding window of size k that is composed by all the sentences from s_{i-k-1} to s_i. The sliding window increases the amount of data accounted for D by considering previous sentences in its classification task. Note that sentences from s_0 to s_{k-1} will be classified with a sliding window of size smaller than k. As an example, consider the task of predicting the emotion of sentence s_{135}, shown at the bottom of Table 1. Instead of performing D's counting procedure in sentence s_{135} alone, if using a sliding window of size 3, we would perform the counting procedure in the union of words of sentences s_{135}, s_{134}, and s_{133}.

The size of the sliding window is also found in the leave-one-episode-out cross validation procedure we perform in the training set. We test $16,384$ different emotion mappings, 20 threshold values, and 5 sliding window sizes, resulting in $1,638,400$ parameter values tested during D's training.

Naive Bayes Classifier (NB)

Another algorithm we consider for classifying sentences for Bardo is Naive Bayes (NB) (McCallum and Nigam 1998). In NB one computes the probability of a sentence s being of an emotion e, for all possible e. NB returns for s the e with highest probability. Let E be the set of all four emotions considered by Bardo. Instead of using the probability of a sentence s being of emotion $e \in E$, denoted $P(e|s)$, NB uses uses a value that is proportional to $P(e|s)$, which is computed as,

$$P(e|s) \propto \log P(e) + \sum_{w \in s} \log P(w|e)\,.$$

Here, $P(e)$ is the probability of encountering a sentence of emotion e in a D&D session. $P(e)$ can be approximated by $\frac{N_e}{N}$, where N_e is the total number of sentences of emotion e in our training set, and N is the total number of sentences in our training set. $P(w|e)$ is the probability of, given a sentence of emotion e, encountering the word w in that sentence. $P(w|e)$ is approximated by $\frac{count(w,e)}{count(e)}$, where $count(w,e)$ is the total number of times the word w appears in sentences of emotion e in our training set, and $count(e)$ is the total number of words that appears in sentences of emotion e in our training set. We use the add-one smoothing scheme to avoid over-fitting and the sum of logs to avoid underflow issues (Manning et al. 2008).

We also use a threshold parameter with NB. That is, Bardo only switches states in the FSM if NB's classification exceeds a threshold. Also, similarly to the D classifier, NB also uses a sliding window. We find both the threshold and the sliding window values in a leave-one-episode-out cross validation procedure in the training set, as described for the D classifier. We test 5 sliding window sizes and 26 threshold values, resulting in a total of 130 parameter values tested. After training, NB and D are able to instantaneously classify a sentence provided by Bardo's SR system.

Alg.	Episodes									Avg.
	1	2	3	4	5	6	7	8	9	
D	**39**	38	30	42	26	24	**35**	39	35	34
DS	30	**63**	46	**57**	**63**	**53**	25	**47**	**57**	**49**
DT	34	17	35	40	38	06	04	40	35	27
DST	34	**63**	**49**	**57**	62	22	26	46	54	46
NB	46	29	61	28	24	23	0	42	46	33
NS	**64**	**62**	**76**	71	54	**69**	**44**	**59**	**79**	**64**
NT	41	48	44	49	46	50	34	44	52	45
NST	61	61	**76**	**72**	**56**	61	43	58	**79**	63

Table 2: Prediction accuracy in % of variants of NB and D.

Classification Evaluation

In this section we present the classification results of the following variants of D and NB: D with no enhancements (denoted as D), D with sliding window (DS), D with threshold (DT), D with sliding window and threshold (DST), NB with no enhancements (denoted as NB), NB with sliding window (NS), NB with threshold (NT), and NB with sliding window and threshold (NST). We separate each episode to be tested and train the algorithms on the other episodes (e.g., when testing an algorithm on episode 1, we train it with episodes 2–9 and the resulting model is applied to episode 1).

Table 2 presents the accuracy obtained by the algorithms ("Alg.") in each episode in terms of the percentage of sentences correctly classified by the algorithms. The "Avg." column shows the algorithm's average percentage accuracy. Numbers highlighted in bold indicate the best performing variant amongst all D-based or all NB-based algorithms in a given episode. For example, row D has bold numbers for episodes 1 and 7, indicating that amongst all D-based variants, D performed best in these two episodes. We highlight the background of a cell if the number in the cell represents the best result across all algorithms. For example, NS and NST were the best-performing algorithms in episode 3.

Discussion

Overall the NB approaches perform better than the D approaches. NS performs best on average, with NST being slightly less accurate, which suggests that the threshold does not improve the accuracy of the method when the sliding window is used. Similar result is observed with DS and DST, where the former is slightly more accurate than the latter.

In some cases the parameters used with the D approaches do not generalize well. For example, D is more accurate than DT in episodes 2 and 7. This is because the threshold value returned by our cross-validation procedure for DT does not generalize well to episodes 2 and 7. In fact, episode 7 is difficult to all algorithms, with NS being only 44% accurate in that episode and NB being wrong in all its prediction attempts. In episode 7 the PCs talk to different groups of NPCs trying to make an alliance for an imminent war. This episode contrasts with the others because the DM and PCs role play their characters without using the game mechanics. By contrast, the other episodes have scenes in which the PCs are stealthy or aggressive, which require the use of the game mechanics. The NB-based algorithms often rely on identifying the use of game rules to make accurate predictions. For example, episodes 3, 4, and 9 have a mixture combat and stealthy scenes and NS is able to accurately detect them, as can be observed by the algorithm's accuracy in those episodes. We conjecture that our classifiers would be more accurate in episode 7 if we had other episodes that were similar to episode 7 in the training set.

We inspected the set of most common words of the NB-based classifiers for each emotion and found that words related to game mechanics are discriminative in the sense that they appear often in sentences of one of the emotions and rarely in sentences of the other emotions. For example, the words *check*, *perception*, and *stealth* appear frequently in sentences of the Suspenseful emotion and rarely in sentences of the other emotions. Similarly, the words *damage* and *initiative* are discriminative words for the Agitated emotion. These are all words related to the mechanics of D&D.

User Studies Evaluation

The results presented in Table 2 show how accurate different learning models employed by Bardo can be, but it offers no insights on how humans perceive the selections made by the algorithm—for that we conduct a user study.

Empirical Methodology

We perform our study online with edited videos from the CotW campaign. Namely, we replace the original background songs with the songs selected by Bardo employing NS. We edit 5 excerpts of the CotW campaign with Bardo. NS is trained with episodes different than the one from which the excerpt is extracted. For example, if an excerpt is extracted from episode 1, then NS is trained with all episodes but 1. Each excerpt is approximately 2 minutes long and contains representative scenes of a typical D&D session. Also, we selected the excerpts for which NS's accuracy varied considerably. This way we would be able to relate the participants preferences with NS's accuracy. Finally, to avoid possible biases in our excerpt selection, the average accuracy of NS in the 5 excerpts is similar to NS's average accuracy in the entire dataset—approximately 64%.

We use as baseline the background music selection made by the video authors while editing their CotW videos. However, instead of using the videos with their original background music, to perform a fair comparison, we edit the videos to use the same set of songs Bardo uses. That is, we replace what we consider to be the Calm songs in the original videos by our Calm song, and so on. We use a set of songs different than the ones used in the original videos because we do not have easy access to the original songs. The songs we use are similar to the originals in the sense that they all fit in the D&D's fantasy genre. Also, the emotions in our songs are clearly differentiable so the participants can easily notice the background music changes.

One caveat of using the original videos edited with our songs as baseline is that the video authors might have selected different emotions in their editing process if they had used our songs instead of theirs. Another caveat of this approach is that it requires us to label the emotions of the original songs so they can be replaced by our songs. The labeling of the original songs was performed by two independent annotators and only one disagreement occurred, which was resolved by a third independent annotator.

Note that the authors of the videos had much more information than Bardo to make their background music selection. This is because the video authors selected the background music as a post-processing step, after the videos were recorded. As such, they could spend as much time as needed, and they knew what was going to happen in the story and could use such information to bias their decisions. By contrast, Bardo did not know what was going to happen in the story, and it had to make its decisions in real time.

The video excerpts we use have no sentences of the Happy emotion, thus we use three songs in our experiment, one for the Agitated scenes, one for the Suspenseful, and one for the Calm. Each participant listened to excerpts of all three songs after answering our consent form and before evaluating the videos. The participants were told that they would evaluate the background music used in several video excerpts of a D&D campaign, and those three songs were the only songs that would be used as background music. We believe that we reduce the chances of a participant evaluating the quality of the songs instead of the song selection procedure by telling them which songs will be used as background music.

Method	Video Excerpts				
	V1	V2	V3	V4	V5
Bardo	60.60	55.73	14.75	73.77	62.30
Baseline	16.42	22.95	60.65	11.48	9.84
Tie+	16.42	9.84	21.32	6.55	13.11
Tie-	6.56	11.48	3.28	8.20	14.75
NS	80.55	20.00	52.63	100.00	86.66

Table 3: Comparison between accuracy and user preference

After listening to the songs each participant watched two versions of the same video excerpt, one with the video authors' emotion selection of background music and another with Bardo's. The order in which the videos appeared was random to prevent ordering biases. We included a brief sentence providing context to the participant, to ensure they would understand to story being told in each excerpt. The participants could watch each video as many times as they wanted before answering the question: "Which video has the most appropriate background music according to the context of the story?". The participant could choose one of the options: "Video 1", "Video 2", "The background music used in both videos are equally appropriate", and "The background music used in both videos are equally inappropriate". After marking their answer, the participants would evaluate another pair of excerpts. The order in which the pairs of excerpts appeared was also random. The participants answered a demographic questionnaire after evaluating all excerpts.

Our experiment was advertised in D&D communities in the social media. In total we had 66 participants, 57 males, 8 females, and 1 other, with average age of 26. All participants had at least some experience playing RPGs. We have removed the answers of 3 participants who reported to have only basic proficiency in English (the language used in the videos and in the study), and of 2 participants who reported problems with their audio system during the experiment. We report the results of the remaining 61 participants, which resulted in 305 answers (5 pairs of videos for each participant).

User Study Results

The videos edited by Bardo were preferred 163 times by the participants, while the Baseline was preferred 74 times, and the approaches tied 68 times. The difference between Bardo and Baseline is significant according to a two-sided binomial test ($p = 7.325 \times 10^{-9}$). Table 3 shows the detailed results for all 5 excerpts used in our study. The upper part of the table shows the percentage of times the participants chose the videos edited by Bardo, by the Baseline, and the percentage of times the participants thought the videos to be equally appropriate (Tie+), and equally inappropriate (Tie-). For example, for the second excerpt (V2), the participants preferred Bardo's video in 55.73% of the answers, and for the third excerpt (V3) the participants preferred the Baseline video in 60.65% of the answers. The last row of the table shows NS's accuracy in each excerpt. The highlighted cells show the best performing approach (Bardo or Baseline).

Discussion

The results of our user study show a clear preference for the video editing provided by Bardo over that provided by the video authors. Despite the caveats of our Baseline, these results demonstrate the potential of Bardo for enhancing the players' experience in tabletop games. The results shown in Table 3 allow us to better understand the participants preferences for the music selection made by our system. We observe that there is some correlation between NS's accuracy with people's preferences. For example, NS has 100% accuracy in V4, which is also the excerpt Bardo performed best: it was preferred in 73.77% of the answers for V4. If we add Tie+ to this percentage, we obtain a positive answer in 80.32% of the cases for Bardo. On the other hand, NS has an accuracy of only 20% and Bardo a preference of 55.73% in V2, while NS has an accuracy of 52.63% and Bardo a preference of only 14.75% in V3.

The results for V2 and V3 suggest that factors other than NS's accuracy can affect the participants preferences. In V2 Bardo mistakenly chooses Agitated instead of Suspenseful for most of the excerpt. However, since the excerpt depicts a Suspenseful scene with some action, most of the participants were fine with the Agitated song selected by Bardo. The excerpt V3 depicts a scene in which a group of barbarians (PCs) go out on a hunt for food, and after killing an elk, a bear sneaks up upon them. Bardo selects the Agitated emotion due to the action of hunting the elk, and it eventually switches to Suspenseful due to the mechanics of the game used before the bear appears (the DM asked the PCs to *roll their perception*, which is usually related to suspenseful moments). Although a brief switch from Agitated to Suspenseful is correct according to the dataset, Bardo's timing is not good and it only switches to Suspenseful after the PCs had engaged in combat with the bear, which is wrong, as the combat indicates another Agitated moment. The Baseline appears to use its knowledge of what is going to happen in the story and selects Calm for the combat with the elk and switches to Agitated once the bear appears—the Baseline raises the stress of the background music as the DM raises the danger the PCs face. Since Bardo has no knowledge of the future, it has no means of using similar technique.

We had a serendipity moment when analyzing the results of excerpt V1. V1 starts at sentence 352 and finishes at sentence 388 of episode 1. Figure 1 shows that there was a disagreement amongst the annotators during the labeling process of V1. It was eventually decided that all sentences in V1 are Agitated. However, Bardo made a transition from Agitated to Suspenseful, as originally suggested by one of the annotators. After watching the video edited by Bardo, all annotators were convinced that the system made a better selection than the annotators themselves.

Conclusions

We introduced Bardo, a real-time intelligent system to automatically select the background music for tabletop RPG games. Bardo was evaluated with a real-world online campaign of D&D. We evaluated the accuracy of two classifiers for our emotion classification task. Our results showed

that a simple Naive Bayes variant is able to obtain good classification accuracy. Second, we conducted a user study in which people evaluated the background music selections performed by Bardo in the D&D campaign. We compared Bardo's selections with those performed by humans, and our results showed that the participants had a clear preference for the selections made by Bardo. Another contribution of our work is a labeled dataset of sentences captured from almost 5 hours of D&D gameplay. In the future we intend to evaluate Bardo in live sessions of D&D.

Acknowledgments

This research was supported by CAPES, CNPq (200367/2015-3), and FAPEMIG. We thank the participants of our user study, in special those from the group D&D Next on Facebook. We thank the Node YouTube channel for allowing us to use their videos in our research. Finally, we thank Davi Lelis for providing invaluable feedback on this research.

References

Alm, C. O.; Roth, D.; and Sproat, R. 2005. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, 579–586. Association for Computational Linguistics.

Alm, E. C. O. 2008. *Affect in text and speech*. University of Illinois at Urbana-Champaign.

Andjelkovic, I.; Parra, D.; and O'Donovan, J. 2016. Moodplay: Interactive mood-based music discovery and recommendation. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, 275–279. ACM.

Balabantaray, R. C.; Mohammad, M.; and Sharma, N. 2012. Multi-class twitter emotion classification: A new approach. *International Journal of Applied Information Systems* 4(1):48–53.

Bergström, K., and Björk, S. 2014. The case for computer-augmented games. *Transactions of the Digital Games Research Association* 1(3).

Cai, R.; Zhang, C.; Wang, C.; Zhang, L.; and Ma, W.-Y. 2007. Musicsense: contextual music recommendation using emotional allocation modeling. In *Proceedings of the 15th ACM international conference on Multimedia*, 553–556. ACM.

Dalgleish, T., and Power, M. 2000. *Handbook of cognition and emotion*. John Wiley & Sons.

Davis, H., and Mohammad, S. M. 2014. Generating music from literature. *arXiv preprint arXiv:1403.2124*.

Deng, J. J., and Leung, C. 2012. Emotion-based music recommendation using audio features and user playlist. In *Information Science and Service Science and Data Mining (ISSDM), 2012 6th International Conference on New Trends in*, 796–801. IEEE.

Ekman, P. 1999. Basic emotions. In Dalgleish, T., and Power, M. J., eds., *The Handbook of Cognition and Emotion*. Sussex, U.K.: John Wiley & Sons. 45–60.

Harrenstien, K. 2009. Automatic captions in youtube. https://googleblog.blogspot.com.br/2009/11/automatic-captions-in-youtube.html.

Lafferty, J.; McCallum, A.; Pereira, F.; et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, 282–289.

Manning, C. D.; Raghavan, P.; Schütze, H.; et al. 2008. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

McCallum, A., and Nigam, K. 1998. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, 41–48. AAAI Press.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, 1003–1011. Association for Computational Linguistics.

Mohammad, S. M., and Turney, P. D. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence* 29(3):436–465.

Pang, B.; Lee, L.; et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval* 2(1–2):1–135.

Plutchik, R. 1980. A general psychoevolutionary theory of emotion. *Theories of emotion* 1(3-31):4.

Posner, J.; Russell, J. A.; and Peterson, B. S. 2005. The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and psychopathology* 17(03):715–734.

Song, Y.; Dixon, S.; and Pearce, M. 2012. A survey of music recommendation systems and future perspectives. In *9th International Symposium on Computer Music Modeling and Retrieval*.

Strapparava, C., and Mihalcea, R. 2008. Learning to identify emotions in text. In *Proceedings of the 2008 ACM symposium on Applied computing*, 1556–1560. ACM.

Suttles, J., and Ide, N. 2013. *Distant Supervision for Emotion Classification with Discrete Binary Values*. Berlin, Heidelberg: Springer Berlin Heidelberg. 121–136.