

Music generation through text sonification

Andreea Lorena Petre

National University of Science and
Technology Politehnica Bucharest
313 Splaiul Independenței Bucharest,
Romania
Petrelorena44@gmail.com

Ștefan Trăușan-Matu

National University of Science and
Technology Politehnica Bucharest
313 Splaiul Independenței Bucharest,
Romania
and
Research Institute for Artificial Intelligence
"Mihai Drăgănescu"
and
Academy of Romanian Scientists
stefan.trausan@upb.ro

ABSTRACT

This paper presents an approach of automatic generation of music from literary texts by combining natural language processing techniques, including sentiment analysis, with music production methods. A dataset of 166 literary passages was built, covering six fundamental emotions. Two approaches to music generation were implemented and compared: a deterministic method that maps textual parameters to musical structures, and an artificial intelligence-based method using the MusicGen model in which prompts were built using details extracted from previous analyses. An interactive interface was developed to visualize and play the generated music in real time, and the results showed that the deterministic approach provides correctness from a music theory perspective, while the artificial intelligence-based approach produces more expressive compositions.

AUTHOR KEYWORDS

Sonification; sentiment analysis; named entity recognition; generative AI.

DOI: 10.37789/icusi.2026.14

INTRODUCTION

Music is considered a basis of human expression, combining emotion, culture, and innovation. It is a part of human culture that has evolved significantly over time, and its generation has evolved along with the outstanding advances in artificial intelligence (AI). Composers can now use AI-powered music generation tools, which use machine learning algorithms to learn music from datasets and generate new melodies that are similar to human-generated music. Neural network architectures such as MusicVAE and TransformerVAE and generative models such as Denoising Diffusion Probabilistic Models have been used for the creation of longer compositions.

Music and language are fundamental forms of communication throughout time, both being viewed as a way

of expressing emotions [4]. Between music and language are many connections, including evidence from brain research [11]. In music there is a set of rules that dictate how musical notes are organized to create melodies, similar to the rules used in languages to create meaningful sentences. Even raw sound is a way of communicating information, without using speech or words, in our daily activities: the use of car horns, fire alarms, mobile phone notifications, and sonification in of some physical such as Geiger counters. used for detecting and measuring ionizing radiation.

The word sonification comes from Latin “sonus” - sound and “ficatio” – composition. In this paper sonification will be used to generate musical compositions from texts, taking into account rhythmicity and emotions, using AI technics such as sentiment analysis and concept extraction.

RELATED WORK

The first musical composition made in 1956 entirely by a computer is the Illiac Suite [2]. However, even before the advent of computers, composers and musicians have been looking for ways to generate music. Wolfgang Amadeus Mozart created a game of dice in the late 18th century in which the rolls of the dice determined the sequence of short musical fragments, an example of a composition that demonstrates how logic and luck can be combined to produce music [1].

There is a variety of music generation systems, and they focus on two important concepts: composition (made up of four essential elements: melody, harmony, rhythm and timbre) and note [9]. AI in music was implemented under the symbolic or connectionist paradigms. For the first case mainly rule-based systems were used. The second paradigm is based on machine learning algorithms that aim at the ability to copy human creativity. One example is EMI (Experiments in Musical Intelligence), which uses algorithms to analyze elements of classical compositions by composers such as Bach and Mozart [12]. The system can then generate new compositions with similar styles. AI analyzes music datasets by breaking down sound into

measurable components, and then identifies patterns and relationships within a composition. Neural networks are used to analyze and generate music because of their ability to identify complex patterns, inspired by the way neurons in the human brain process information. However, artificial intelligence cannot replicate human imperfections and emotions, and the fact that it uses datasets to identify patterns does not allow for the creation of entirely new compositions.

Data sonification in music composition and other domains

Expressing data as sound instead of graphs or tables provides a secondary source of information in remote monitoring and allows data to be analysed when visualizations cannot be used. Sonification has been used in several domains: medicine, science and research, music, sports and education [5].

In the game industry, more specifically computer games, the soundtrack of games is an important element signaling information such as: how many seconds are left in the round, player's health, generated by various methods, having some similarities with programmatic music, where there are sound sequences specific to certain subject or actions. Also, sonification in games builds tension and keeps the players engaged without distracting them. So, sonification has an important role in generating content in this field [17].

In the training stages of athletes, devices are used to monitor parameters such as: blood pressure, oxygen saturation, temperature, ECG, heartbeat, being monitored using portable devices. Monitoring these parameters during activities is almost impossible, so sound is used to check if normal values are exceeded.

There are various applications of sonification in the field of art, one example being the transformation of dance movements recorded using motion sensors into sound compositions. Another example is the use in music, Judy Twedt used climate data of Arctic Sea ice levels from 1979 to 2016 to express the dramatic impact of climate change through piano music [15]. Also, there is DNA-based music called "DNA Suite" that turns DNA sequences into musical notes. David Deamer and some friends composed about half an hour of music based on human insulin gene and some bacterial DNA sequences [16].

Sonification is used in medicine both for research and practical projects. An example of its use is the prediction of an organism's traits based on genetic data, an objective emphasized by the National Academies in 2010 [6]. GEMusicA (Gene Expression Music Algorithm) allows this, being a useful tool in identifying complex patterns and allowing the analysis of nucleotide sequences. The heart rate can also be sonified to provide patients with information to regulate their emotional state through breathing techniques. Another example is the sonification of the EEG to represent brain activity through sounds, being used in the detection of epileptic seizures. Also, transforming complex data from

medical images into sounds can help radiologists identify abnormalities, such as tumors or lesions, by interpreting sound variations. Complex surgical systems can generate sounds during surgery, providing surgeons with real-time information (the heartbeat used to understand the general condition of the patient during surgery). By sonifying dermatoscopic images, sounds can represent textures or other characteristics of skin lesions, helping to identify abnormalities.

As early as 1961, sound has been used to analyze earthquake movements. Research has been conducted to determine if seismometric sounds can be used to identify atomic explosions, as well as natural earthquakes, with the aim of understanding whether people can differentiate between them using sounds. The result of the experiments showed that 67.5% of people could identify a natural earthquake better. An important project in this field was carried out in the year 1994, presented by Quinn, in which the California earthquake of magnitude 6.7 was transformed into a musical composition to allow the listener to identify the shape and intensities of a seismic wave [7].

There are various examples in which data sonification has led to the production of musical compositions, one example being the piece "Atlas Eclipticalis" by John Cage in which the positions of the constellations were used to obtain the musical score. In "Music for Solo Performer", made in 1965, Alvin Lucier played a percussion ensemble, being inspired by the alpha waves of his brain. Christina Kubisch has two important works: "Dreaming of a Major Third" - which transforms sunlight into the sounds of bells and "Magnetic Nets" - in which she uses electromagnetic headphones to examine anti-theft devices and transpose them into music [10].

Text sonification can be considered a way to combine language and music by transforming texts (more precisely words or sentences) or conversations [3] into sounds used for musical compositions. Emotional states defined by words with a certain impact on the reader can be analyzed, sonification having the role of transposing emotions into musical sounds or tones.

Existing Approaches

Music composition through text sonification combines NLP, sentiment analysis, and music generation techniques. There are several projects that have combined artificial intelligence and musical concepts for music composition. Herndon created the album titled PROTO using an AI program called "Spawn". Southern also released the first album called "I AM AI", created entirely using artificial intelligence tools, specifically Amper Music [14].

TransProse was a system that automatically generates musical pieces using sentiment analysis from literary texts. It uses well-known musical concepts such as tempo and musical scale and determines note sequences that reflect emotional activity in the text [8]. It has applications in

information visualization, e-book creation, and music application development. Given a novel in an e-readable format, the system generates simple piano pieces whose notes depend on the emotional words in the text.

A film's soundtrack enhances the cinematic experience, yet its creation requires both expertise and time. ScriptTONES operates in two stages to generate music suited for film scenes. The first stage (sentiment analysis) involves encoding the emotional content of a scene. The second stage consists of music generation, where the resulting vector serves as input to generate piano music in MIDI format. Since a screenplay contains both dialogue and additional information, it is necessary to extract the relevant words that contribute to the scene's emotional component and to segment the script into individual scenes.

Conversations sonification was implemented using NLP for identifying the ideas discussed and their interanimation starting from the polyphonic model of discourse [3].

Unlike TransProse, which performs a binary sentiment classification (positive/negative), the approach presented in this paper relies on a six-emotion model, enabling a better musical mapping: not only key, but also tempo, scale and instrumentation differentiated per emotion. Compared to ScriptTONES, which uses film scripts and generates MIDI piano music from a single sentiment vector per scene, the presented approach introduces a three-segment emotional arc (and an additional concept-extraction step that extends the generation parameters beyond sentiment alone. Furthermore, this is the only one of the three systems that explicitly compares a deterministic, rule-based generation approach against a generative AI model (MusicGen) on the same corpus, and it is complemented by an interactive web interface with real-time text-music synchronization.

METHODOLOGY

Several pre-trained classification models were tested to implement sentiment analysis. The first approach used the `distilbert-base-uncased-finetuned-sst-2-english` model, a version of BERT fine-tuned on the Stanford Sentiment Treebank (SST-2) dataset. The model classifies texts into two categories: positive and negative, returning a probability score for the positive class. Applied to an initial subset of the Gutenberg corpus (<https://www.gutenberg.org/>), the binary classification does not distinguish between emotions such as fear, sadness, or disgust, all of which are labeled as "negative".

This limitation made it impossible to generate music differentiated by specific emotions, so the `cardiffnlp/twitter-roberta-base-sentiment-latest` model was introduced, which adds a third class (neutral) and provides separate scores for negative, neutral, and positive. In addition, the `cardiffnlp/twitter-roberta-base-irony` model was tested for irony detection. Both models were later eliminated due to a limitation: they were trained on tweets, a type of text

characterized by informal language, abbreviations, completely different from literary prose.

To avoid models trained on Twitter, the `nlptown/bert-base-multilingual-uncased-sentiment` model was tested, trained on product and service reviews in several languages. It returns a prediction on a scale from 1 to 5 stars, which was manually mapped to the categories negative (1–2 stars), neutral (3 stars), and positive (4–5 stars). Although more domain-appropriate than Twitter models, this approach introduced a problem: the mapping from stars to emotions has no semantic justification and is not suitable for identifying the emotional tone of works.

The model chosen for the final version of the implemented solution is `j-hartmann/emotion-english-distilroberta-base`, based on the DistilRoBERTa architecture and fine-tuned to classify six fundamental emotions: joy, sadness, fear, anger, surprise and disgust. The choice of this model was motivated by the direct classification of the emotional character of the texts, eliminating the need for any manual mapping. The model returns probability scores for all six emotions simultaneously, which allowed the calculation of the emotional arc on three segments of the text, the emotional complexity and intensity by the L2 norm of the score vector. Applied to a corpus of 532 literary passages extracted from Project Gutenberg, the `j-hartmann` model led to the results presented in Figure 1. Positive emotions such as joy and surprise were quite rare, which led to an improved way of extracting literary passages, where the source books were predefined taking into account the author and the literary genre that he uses most often.

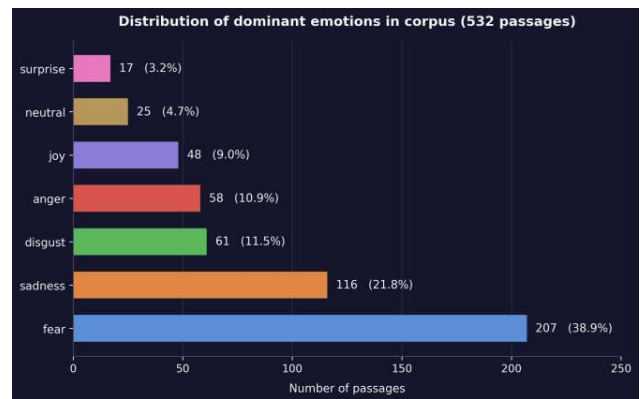


Figure 1. Distributions of dominant emotions in the corpus (532 passages)

In terms of emotional complexity, 83.5% of the passages presented a clear dominant emotion (simple complexity), while 16.4% were classified as mixed with two simultaneous significant emotions. Emotional intensity was very high in 74.6% of cases, confirming that passages selected by the confidence threshold ≥ 0.60 tend to contain strongly expressed emotions, suitable for the music generation process.

IMPLEMENTATION DETAILS

Dataset Creation

A Python script was used for automating the process of downloading and processing books from Project Gutenberg, with a focus on texts from English literature. Three different approaches were tested to obtain a dataset suitable for generating musical content.

In the first version, books were filtered by genre, downloaded and cleaned of formatting issues, and four paragraphs were extracted from the middle of each book, since the beginning and end often contain prefaces or epilogues unrepresentative of the author's style. The limitations of this version were identified during the sentiment analysis stage: the final distribution contained 278 passages classified as having a neutral tone, 27 negative and only 5 positive, which shows us that the random extraction from the middle of the books produced a dataset with texts without emotional content relevant for music generation.

Starting from the identified limitations, the next version introduced an intelligent system for identifying and extracting fragments with high emotional content to improve the quality of the data, ensuring a dataset in which the extracted passages contain narrative material that expresses feelings, essential for the generation of expressive musical compositions. The first improvement was the narrowing of the list of accepted genres, focusing on genres with high emotional potential. Genres such as "American", "English", "Classical", "General" or "Philosophy" were removed because they did not generate emotional content. The second update was the emotional scoring system, absent in the previous version. The script defines a list of emotional keywords, including positive emotions such as "love", "joy", "happiness", "delight", but also negative emotions such as "hate", "fear", "anger", "sadness", "grief", etc. The emotional score function goes through each text and counts the number of times these keywords appear, thus generating a numerical indicator of emotional intensity. This system allows not only to identify emotional passages, but also to facilitate the automatic selection of the most expressive fragments in each book.

The script also analyzes each paragraph and counts the occurrences of quotation marks, thus identifying passages that contain conversations between characters. This functionality is based on the observation that literary dialogues often contain emotions expressed by the speech of characters in moments of conflict, love, fear or joy. Passages identified as dialogue receive a 2-point bonus to the total emotional score, reflecting their increased value for sentiment analysis. Three passages per book were extracted instead of one, each scored independently, and only passages above a 2-point threshold were kept. Passage length was bounded to 80–200 words, long enough for context but short enough to keep a single clear emotion. This version produced 532 passages with a richer emotional distribution: 267 positive, 250 negative.

The final version eliminated the random extraction strategy, choosing a different approach: emotion-based selection. Instead of extracting texts and subsequently applying the sentiment analysis stage, books with a high probability of containing a target emotion were pre-selected. This strategy eliminated the need for emotional scoring as a filtering mechanism, replacing it with prior knowledge of the dominant tone of each work. Poems from the merv/poetry dataset on HuggingFace were added, with separate filtering criteria for prose (100–150 words, at least 2 complete sentences with a minimum of 5 words each, the text must not be written entirely in capital letters) and poetry (6–60 lines, 60–220 words). The acceptance criterion has also been updated: a fragment is accepted only if the dominant emotion coincides with the target emotion of the respective book, the confidence score ≥ 0.60 and the score of the dominant emotion must be at least 1.5 times higher than the second score. The result consists of 166 passages (97 poems, 69 prose fragments), evenly distributed across 6 emotions: joy (44), anger (36), fear (30), sadness (25), disgust (20), surprise (11).

Sentiment and emotion analysis

The second stage of the approach consists in the creation of a script that allows the identification of the emotions conveyed in the selected literary passages, an essential aspect for the generation of music based on the texts. The original script used `cardiffnlp/twitter-roberta-base-sentiment-latest` for sentiment analysis, a model trained on tweets that is not suitable for the structure of literary texts. This model has been replaced by `nlptown/bert-base-multilingual-uncased-sentiment`, a model trained on reviews, providing ratings on a scale of 1 to 5 stars instead of simple positive/negative/neutral classifications. This change allowed the identification of sentimental nuances specific to literature, where many passages are not only positive or negative, but contain mixed tones and ambiguities. Also, the initial script included a third model, `cardiffnlp/twitter-roberta-base-irony` for detecting irony and sarcasm, later removed because literary irony is different from irony in tweets, and the model did not provide relevant results for classical texts.

Initially each passage was divided into paragraphs, analyzed separately and combined by calculating averages. The later version analyzes the passage as a whole, avoiding excessive splitting that could remove the emotional character of the text, an approach more suitable for the 80–200 word fragments already selected based on the emotional score. A `detect_literary_tone` function was added, which combines the results of sentiment analysis with keyword-based lexical analysis to detect tones such as gothic, romantic, tragic, suspenseful, melancholic, joyful, intense, contemplative, and lyrical. For the gothic tone, the function looks for the words "darkness", "shadow", "blood", "death", "horror", "grave", "ghost" and simultaneously checks if the fear score is high. The romantic tone is identified by "love", "heart", "passion",

"kiss", "embrace" combined with a high joy score. Melancholic tone is detected when sadness is dominant but anger is low, while tragic tone requires both high sadness and the presence of elements of anger or conflict. Suspenseful tone is identified either by words related to mystery or by the combination of moderate fear and surprise. For poetry, the tone is automatically classified as lyrical.

The emotional intensity is classified into four levels (very intense, intense, moderate, weak), based on either a single strong emotion or several moderate emotions combined. Emotional complexity measures how many emotions are simultaneously significant: simple (one dominant emotion), mixed (two), or complex (three or more). For example, a romantic breakup scene can simultaneously feature sadness, love, and fear of the future, resulting in high complexity.

Results obtained on the initial dataset of 532 passages showed that 267 passages were positive, 250 negative and 15 neutral with the average rating of 3.12 out of 5 stars. The small percentage of neutral passages confirms that the selection method succeeded in eliminating descriptive texts. Fear predominates in 207 passages (38.9%), followed by sadness (116), disgust (61), anger (58), joy (48) and surprise (17), reflecting the genres selected initially. The most common literary tones were narrative, melancholic and gothic, a distribution that later versions aimed to balance further.

The final version brought three significant changes. The first was the complete replacement of the nlptown model with j-hartmann/emotion-english-distilroberta-base, eliminating the need for mapping from stars to emotions and directly producing scores for the six basic emotions. The second change was the introduction of the emotional arc. Each passage is divided into three equal segments and each segment is classified individually with j-hartmann. The method of division differs depending on the type of content: for prose, the division is done by sentences detected using the punctuation marks .!?!; for poetry, the division is done by lines, respecting the structure of the texts. The result is an arc of the form fear-fear-sadness or disgust-sadness-joy, which contributes to the generation of music: the transition from negative to positive feeling produces `minor_to_major`, and from positive to negative produces `major_to_minor`, otherwise it generates `same_mode_shift`. The third change was the improvement of the existing parameters: intensity was recalculated as the L2 norm of the score vector (the root of the sum of squares) instead of the simple sum; emotional complexity received a fourth category, `very_complex`, for passages with four or more significant simultaneous emotions; the acceptance criterion was modified by adding a minimum ratio of 1.5 times between the dominant emotion and the second emotion, eliminating passages with ambiguous emotions.

The statistics obtained on the final dataset of 166 passages (69 prose + 97 poetry) highlight differences compared to previous versions:

- Emotional intensity: 140 passages (84.3%) have very high intensity and 26 high intensity, no passages with moderate or weak intensity, compared to 54 in the previous version
- Emotional complexity: 135 passages (81.3%) are simple, with a single dominant emotion and 31 (18.7%) mixed. The absence of the "complex" category compared to the previous version reflects the intentional selection of passages with clear emotion, more suitable for music generation
- Literary tones: the distribution has changed significantly compared to the previous version where narrative (205) and gothic (103) dominated. In the final version, intense (32), joyful (29), romantic (28) and melancholic (28) are the most frequent, reflecting the better balance between positive and negative emotions achieved by pre-selection of books
- Emotional arcs: the most frequent arc is joy-joy-joy (11 passages), followed by disgust-disgust-disgust (7) and several arcs with transitions such as fear-sadness-sadness (5) or joy-sadness-joy (5). Of the 166 passages, 103 will change key in the music, distributed as follows: none (63), `same_mode_shift` (58), `major_to_minor` (27), `minor_to_major` (18).
- Prose vs. poetry comparison: the average intensity is almost identical: 0.837 for prose versus 0.836 for poetry

Concept and topic extraction

To extract concepts and topics from literary texts, a script using BERTopic, improved for literary texts, was initially implemented. The system used SentenceTransformer for generating embeddings, UMAP for dimensionality reduction in semantic spaces, HDBSCAN for clustering and a TF-IDF vectorizer. This approach allowed simultaneous processing of texts in several languages such as: English, French and German, since removing texts in other languages led to reducing the dataset.

One of the important aspects of the implementation was the preprocessing, adapted to handle various languages using langdetect. Extremely common stopwords and words with a length of less than four characters were eliminated. The system detected and reassigned topics with very few documents to larger and more stable topics, improving the results and reducing noise.

Analyzing the results, nine distinct literary topics were identified from a dataset of 331 documents. Compared to this approach, the final version completely eliminated BERTopic in favor of direct analysis through spaCy with the `en_core_web_lg` model. This change was necessary because topic modeling through clustering identifies common themes across the entire corpus, losing the specificity of each individual passage; a text about a medieval knight fighting for his beloved would have been forced to choose between a "war" or "love" topic, when in fact it contained both.

The final version script uses the large `en_core_web_lg` model instead of the small or medium variants, justified by the need

for high accuracy in processing classical literature that frequently uses archaic vocabulary.

In addition to linguistic analysis with spaCy, the final version introduced a generation of an ambient sounds vocabulary using WordNet (<http://wordnet.princeton.edu>) Starting from representative anchor words per category, using WordNet a vocabulary of 3,584 unique words distributed in 12 categories was generated. A particular case highlighted a limitation of the use of WordNet. The crowd category generated incorrect terms: "rallying", "year", "world", leading to the detection of crowd in 106 out of 166 passages, an incorrect result. The solution was to manually define a list of 17 terms, reducing the occurrences to 7 passages (4.2%). Applied to the 166 passages, the constructed vocabulary allowed the detection of ambient sounds in 107 passages for nature categories. The most frequent categories detected: water (65 passages), night (64), sea_ship (53), wind (41), forest (36), battle (35), church (31), fire (30), birds (22), horse (13), thunder (8), crowd (7).

The extracted concepts influenced the musical parameters through several mechanisms. Tempo adjustments per emotion calculated from the text concepts produced variations: anger (+14.9 BPM compared to the emotion base), joy (+14.7 BPM), surprise (+12.5 BPM), fear (+8.0 BPM), disgust (+2.8 BPM) and sadness (-2.0 BPM). The complexity resulting from the number of abstract concepts was distributed as follows: simple (58 pieces), moderate (54), complex (32), very_complex (22).

An important result of this stage is the construction of prompts for MusicGen, each passage receiving a unique prompt built from all the extracted parameters. Of the 166 generated prompts, 153 are unique (92.2%), and 13 are duplicates, cases in which different texts produced the same combination of parameters.

The music generation stage transforms the data obtained in the previous stages into musical structures in JSON format, directly interpretable by the web interface. The script implements a deterministic mapping system that translates the characteristics of literary texts: sentiment, emotions, into specific musical parameters: tonality, scale, tempo, harmonic progressions, melodic lines and instruments. This approach is based on principles from music theory where certain sound structures are associated with certain emotional states.

Music generation mapping

The music generation stage translates the extracted characteristics into musical parameters, based on principles from music theory.

- **Tonality:** combines information about sentiment, dominant emotion, confidence and literary tone to select a tonality using a SENTIMENT_TO_KEYS table. For example, the combination of negative sentiment with fear is mapped to minor tonalities (C#m, F#m, G#m), psychologically perceived as tense; the combination of positive with joy to

major tonalities (C, G, D, F, A, E). The selection from the possible list depends on confidence: > 0.80 chooses the first option, > 0.60 the second, below 0.60 it is chosen randomly. Final keys: Cm (34 pieces), G (26), Dm (26), D (20), Am (18), Gm (16).

- **Scale:** determined by a table of literary_tone and emotional_complexity combinations

- **Tempo:** calculated from the emotional intensity (very_intense: 145 BPM, intense: 120, moderate: 95, weak: 70), adjusted with the length of the passage and the type of poem. The final result is limited to [60, 180] BPM. Final distribution: minimum 85, maximum 180, average 144.3 BPM.

- **Instruments:** mapped by literary tone: gothic: organ, cello, violin; melancholic: piano, cello; joyful: piano, flute, guitar; intense: percussion instruments; narrative: piano, flute.

- **Seed melody:** 16 algorithmically generated notes, with different melodic movement per emotion: joy ascends the scale, sadness descends, fear makes 3-step jumps for instability. Durations follow fixed patterns: sadness with long notes (2n, 1n), anger with short notes (16n, 8n), fear with sudden pauses.

- **Emotional arc:** when the emotional arc involves a change of register, the musical structure receives different parameters per section. Out of 166 tracks, 103 (62%) require modulation.

Processing the 166 passages produced 18 distinct keys, 10 scale types, tempos between 85 and 180 BPM, and 153 unique MusicGen prompts out of 166. After implementing the deterministic approach, Magenta MusicRNN, a recurrent neural network model trained on MIDI sequences, was explored. Since 2016, various models and libraries have been developed to provide functions for creating, manipulating, and generating music and art. Magenta supports multiple formats such as: MIDI and MusicXML, enabling integration with other music software. However, Magenta has certain limitations: it requires a certain level of technical knowledge to use and understand [13].

Unlike the deterministic approach that generates fixed notes through mathematical rules, Magenta took the deterministically generated seed melody as a starting point and generated a continuation based on patterns learned from real scores.

The final version introduces an artificial intelligence model specialized in generating music based on a text prompt, more precisely facebook/musicgen-small, to compare the quality of the results with the deterministic approach. Unlike the deterministic approach that applied mathematical rules, MusicGen produces realistic audio content of the instruments. Each piece received a unique prompt built from all the previously extracted parameters: literary tone, tempo, instruments, composition type, "no vocals, instrumental", modulation (if any), ambient sound, abstract concept/key

object, "theme of [Main character]". The audio files were saved as separate files for the two types of texts.

WEB INTERFACE

An interactive web interface was developed to visualize and play the results, initially implemented using HTML, CSS, JavaScript, Web Audio API and specialized libraries for music generation. The application offers three modes of music generation: deterministic mapping, generation via Magenta AI, and a hybrid mode that combines both approaches, allowing the user to explore different sound interpretations of the same literary text. The functionality for loading music data starts by reading the JSON file with all the information generated in the previous stages. The Soundfont Player library was used to load and play samples for realistic instruments, unlike the simple oscillators in the initial version. The application loads eight distinct instruments from the FluidR3_GM library: acoustic piano, organ, cello, string ensemble, violin, flute, pad and acoustic guitar. The interface rewrite in React was motivated by the need for complex state management and updates synchronized with audio playback. The architecture uses React hooks: useState for interface state (current song, volume, generation mode, playback state), useEffect for side effects (audio initialization, volume update), useRef. The interface offers the ability to select any passage from the dataset, choose between deterministic, Magenta AI or hybrid mode, and visually track the text synchronized with the music playback.

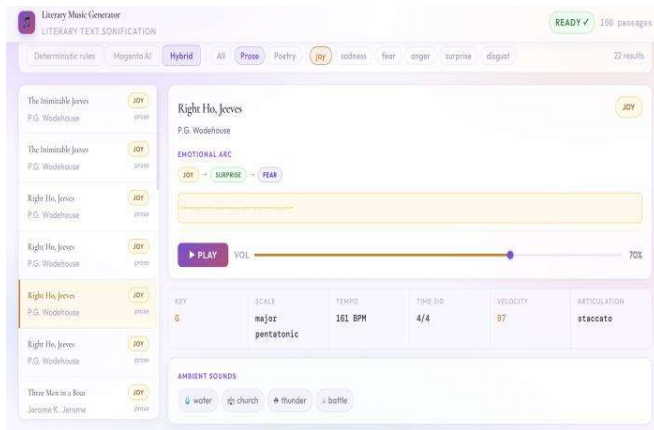


Figure 2. Web interface displaying the selected passage, its emotional arc, and the generated musical parameters (hybrid mode)

The hybrid mode combines the two generation strategies: the deterministically generated seed melody (16 notes, produced by the rule-based mapping described above) is used as input for Magenta MusicRNN, which extends it into a longer composition based on patterns learned. To resolve the text–audio mismatch identified during development: a listener has no way of knowing which words produced a given musical phrase without additional context, the interface highlights the source text in real time. A comparison panel displays, side

by side, the deterministic mapping parameters (key, scale, tempo, instruments) and the MusicGen prompt constructed from the same passage, allowing the user to directly inspect how the two approaches translate the same emotional and conceptual content into music. Figures 2 and 3 show the interface in deterministic, Magenta, and hybrid modes.

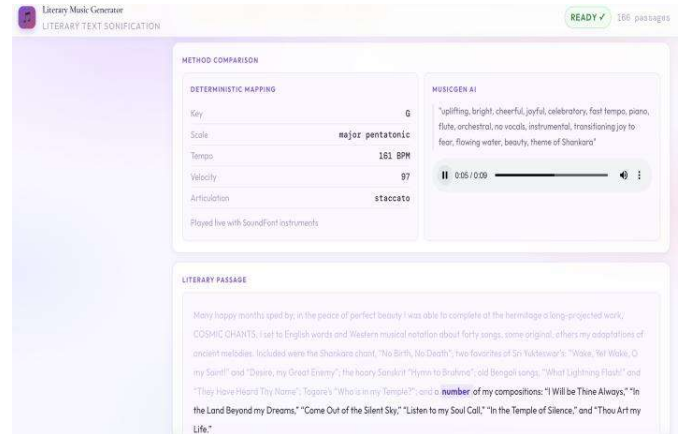


Figure 3. Comparison panel showing the deterministic mapping parameters alongside the MusicGen-generated prompt for the same passage

DISCUSSION

The presented approach of composing music by sonifying literary texts demonstrated that automatic text analysis can be used to generate musical compositions, combining techniques from natural language processing with principles from music theory and generative artificial intelligence models. In the process of generating the dataset, it was demonstrated that the quality of the dataset is more important than its size. Going from 532 randomly collected passages to 166 passages selected according to a target emotion produced a dataset more useful for music generation, confirming that the relevance of the input data directly influences the quality of the results in any NLP-based system. Exploring several sentiment classification models showed that the choice of model must take into account the domain of the processed texts. Models trained on tweets or reviews produced less accurate results on classical literary texts, while j-hartmann/emotion-english-distilroberta-base, specifically trained on emotions, proved to be the most suitable for the purpose of the approach, eliminating the need for any mapping. Concept extraction illustrated some limitations of automatic lexical resources. WordNet allowed the construction of a vocabulary of 3,584 words for ambient sounds without significant manual effort, but the case of the crowd category showed that automatic expansion requires validation for domains with high semantic ambiguity.

Replacing BERTopic with spaCy resolved an important limitation: topic modeling through clustering lost the specificity of each individual passage, while direct linguistic analysis produces unique parameters per text. The

comparison between the deterministic approach and MusicGen highlighted a difference between the two methods. The deterministic approach can correctly capture the theoretical musical parameters: key, scale, tempo, chord progression, but MusicGen produces more expressive results because the model uses musical patterns from real music composed by humans. The React interface that uses automatic text highlighting at the time of music playback solved a limitation of sonification: the text-music connection is not easy to understand for a listener without visual context, regardless of the generation method. The simultaneous display of text synchronized with the music playback transforms the experience from a simple audio playback to a presentation in which the user can follow in real time the words that generated the music. The comparison panel between the deterministic approach and the MusicGen prompt explains the text-music transformation process for each passage. Based on the information and limitations identified, several development directions could improve the system. A first improvement could be to test the prompts generated using the SA and NER steps together with the original literary text. Comparing the results obtained with a simple prompt versus an enhanced prompt with the original text would provide insight into the extent to which text-to-music models can process and reflect literary content in musical compositions. Also, evaluating the quality of the generated music through a user study could provide information on the accuracy of the system. In the current version, it is not known whether a listener perceives the difference between a piece generated for a text of fear versus one for joy, or whether they correctly associate the music with the text that generated it. By testing the larger variants of MusicGen: musicgen-medium and musicgen-large on the same dataset, we can identify whether the additional parameters of the model produce an improvement over the musicgen-small used in the current version

CONCLUSION

The approach demonstrates that sonification of literary texts is technically feasible and produces relevant musical results, but there is also a limitation: no matter how complex the translation rules or the generated prompts, a listener who does not know the source text will perceive the music as a composition with a certain mood, not as a representation of a specific text.

REFERENCES

1. The Music of the Future: How AI Is Composing Melodies. (2025). (n.p.): Recorded Books.
2. Funk, T. (2018). A musical suite composed by an electronic brain: Reexamining the Illiac Suite and the legacy of Lejaren A. Hiller Jr., *Leonardo Music Journal*, Vol. 28, pp. 19-24
3. Trausan-Matu S. (2018) Chat Sonification Starting from the Polyphonic Model of Natural Language Discourse, *Tehnologii informatice și de comunicație în domeniul muzical*, Ed. Media Musica, ISSN 2067-9408, Vol. IX, nr. 1, pp.79-85.
4. <https://www.languagesunlimited.com/the-role-of-music-in-the-evolution-of-languages/>, Last accessed: 6 January 2025
5. Mihalaș, G. I. (2022). Sonification – A bridge between science and arts. *Music Cognition*, 5, 128–153
6. Braun, R., Tfirm, M., & Ford, R. M. (2024). Listening to life: Sonification for enhancing discovery in biological research. *Biotechnology and Bioengineering*, 121, 3009–3019. <https://doi.org/10.1002/bit.28729>
7. Ramaswamy, L., Hutchinson, T. C., & Kuester, F. (2004, July). Towards enhancing earthquake response interpretation using sonification. *Proceedings of ICAD 04 – Tenth Meeting of the International Conference on Auditory Display*, Sydney, Australia
8. Davis, H., & Mohammad, S. M. (2014). Generating music from literature. arXiv preprint arXiv:1403.2124.
9. Herremans, D., Chuan, C. H., & Chew, E. (2017). A functional taxonomy of music generation systems. *ACM Computing Surveys (CSUR)*, 50(5), 1-30.
10. Schoon, A., & Dombois, F. (2009). Sonification in music. Paper presented at the 15th International Conference on Auditory Display, Copenhagen, Denmark, May 18–22, 2009.
11. Trausan-Matu, S. (2020) Connections between human music and natural language, in Milena Slavcheva (ed.) *Knowledge, Language & Models*, INCOMA Ltd. Shoumen, pp. 302-317.
12. Wei, L., Yu, Y., Qin, Y., & Zhang, S. (2025). From Tools to Creators: A Review on the Development and Application of Artificial Intelligence Music Generation. *Information*, 16(8), 656.
13. Zhu, Y., Baca, J., Rekabdar, B., & Rawassizadeh, R. (2023). A Survey of AI Music Generation Tools and Models. arXiv preprint arXiv:2308.12982.
14. Kadam, A. V. (2023). Music and AI: How artists can leverage ai to deepen fan engagement and boost their creativity. *Int. J. Sci. Res*, 11, 1589-1594.
15. Strack, F. (2025). Sounds of Climate Change: A Conversation between Sound Art, Data, and Alfred N. Whitehead. *Theory & Event*, 28(4), 555-586.
16. <https://www.smithsonianmag.com/science-nature/why-scientists-are-turning-molecules-into-music-180980022/> Last accessed: 10 January 2026
17. Cardinale, S., Cook, M., & Colton, S. (2022). AI-Driven Sonification of Automatically Designed Games. The Experimental AI and Games Workshop at AIIDE