Universitat Politècnica de Catalunya

Intelligent System Project

Project Definition

Master in Artificial Intelligence

Authors:

Gerard Caravaca Ibáñez
Benjamí Parellada Calderer
Armando Rodriguez Ramos

Supervisor:

Miquel Sànchez Marrè

Fall Term 2023/2024

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

FIB

Contents

1 Introduction

The focus of this project is on developing an Intelligent System to convert audio into sheet music automatically. This initiative draws on multiple disciplines, such as Signal Processing, Deep Learning, and Music Theory. The system will be designed to listen to audio clips of musical performances and convert them into sheet music, the standard written form for music that musicians across genres and instruments recognize and use.

1.1 Definition of the problem

Transcribing music is a meticulous and intricate task often brimming with complexities, especially for human musicians and composers. For novices, it is integral for understanding musical notations and structures, serving as a foundational learning tool. For experienced musicians, especially when learning new pieces, transcriptions are indispensable, offering detailed insights into compositional structures and facilitating the learning process and future performances. While human transcriptions are often lauded for their precision and finesse, automatic transcriptions frequently lack this degree of refinement and cleanliness.

A significant contributor to the disparity in quality between human and automatic transcriptions is the prevalent approach of directly translating from *MIDI* (Musical Instrument Digital Interface) to transcription. *MIDI* [1] is a protocol that allows musical instruments and computers to communicate with each other; it encodes music information in a digital format that can be easily interpreted and manipulated, making it a valuable tool in music production. A rendition of a song can be written down in *MIDI* to later be transcribed to a sheet of music.

However, one of the main challenges with utilizing *MIDI* for transcriptions is its precision in reflecting the musician's performance. *MIDI* captures every nuance, deviation, and subtlety executed by the artist, making it a near-perfect mirror of the musician's interpretation. While this level of detail is valuable in certain contexts, it poses significant difficulties when converting to transcriptions. The resulting transcriptions, when translated directly from such precise *MIDI* renditions, can be 'too exact', embodying every minor fluctuation in tempo or subtle variance in pitch inherent to a live performance. This over-fidelity can make the transcription excessively complex and often deviates from a standardized representation of the music piece, hence losing the generalizability and accessibility usually associated with musical sheets.

Nevertheless, most of the time, the user does not have the *MIDI* performance of the song, but just the audio. Thus, a process to convert from audio to *MIDI* must be done. However, this method often experiences artifacts arising from noise and harmonics, leading to the inclusion of ghost notes—notes that are not present in the original piece—thus obscuring the clarity and accuracy of the transcription. Despite its utility, the direct translation from *MIDI* can, therefore, introduce inaccuracies and impede the reliability of automatic transcriptions compared to their human counterparts.

For example, comparing the original sheet music versus the transcribed rendition of a pianist, as seen in Figure 1, illustrates the striking discrepancies between the two. Particularly, the anacrusis (a note or sequence of notes which precedes the first downbeat in a bar) and embellishments like trills, result in a more cluttered and discordant representation in the automatic version.

Additionally, even though the system used allowed for user defined information to help, such as key signature, tempo, etc., the final sheet still failed in getting the correct time signature, and has more notes than in the original.

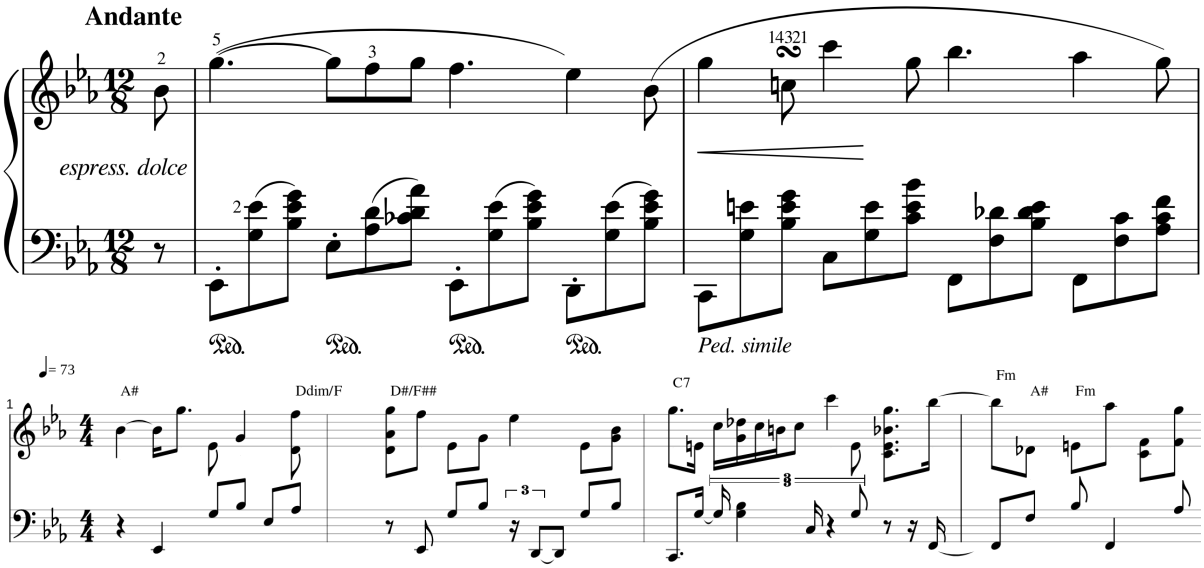

Figure 1: Sheet music for: Chopin – *Nocturne E Flat Major Op.9 No.2.* Top is the original sheet, and below is an automatic transcription (using [2]) of a rendition of the song played by pianist Valentina Lisitsa. The shown snippets represent the same notes to be played, but the top one is much easier to read. Moreover, the pedal notation, dynamics, ligatures, and staccatos are all absent in the automatic version.

The aforementioned issues are further amplified when transcribing pieces involving multiple instruments, as the complexity of the notation escalates exponentially. Each instrument may produce unique sounds, harmonics, and overtones, which, when played in tandem, create a layered sonic texture. Disentangling these layers and accurately attributing them to the respective instruments while maintaining the integrity of the original composition is a colossal challenge. Once disentangled, each layer could be transcribed to different sheets.

Addressing these challenges is paramount, as a more refined automatic transcription system can bridge the gap between the accessibility of sheet music and the intricate nuances of musical pieces. This will not only expedite the transcription process but also facilitate a clearer, more accurate representation of musical compositions, making them more accessible and comprehensible, especially for beginners and those without formal musical training.

By fostering an environment where musical compositions can be studied, shared, and understood with greater ease and precision, this project can potentially propel the realms of musical education and creativity forward, allowing for a more inclusive and diverse musical landscape.

1.1.1 Problem Statement

The core problem is building an AI system capable of audio transcription containing musical information into an accurate, digitally-rendered sheet of music. This task is usually called Automatic Music Transcription (AMT). The system would take an audio file as its input and produce sheet music as its output, potentially in file formats easy to edit, like MusicXML.

1.1.2 Challenges and Complexities

Automatic Music Transcription is a highly complex task with various inherent challenges due to the multifaceted nature of music. Here, we highlight the key challenges and their implications on the quality and reliability of transcriptions:

- **Audio Quality:** Poor recording quality or external noise can significantly impact the accuracy of transcriptions. For example, faint background noises might mistakenly be transcribed as soft notes, leading to inaccuracies in the resultant sheet music.
- **Polyphonic Audio and Harmonic Relations:** Polyphonic music, involving multiple simultaneous sound sources, each producing one or more musical voices with distinct pitch, loudness, and timbre, presents severe difficulties. The overlapping sound events often exhibit harmonic relations, forming small integer ratios of fundamental frequencies. This overlapping makes separating individual contributions extremely underdetermined and complex. For example, in a C major chord (C:E:G), the harmonic positions overlapped by other notes are 46.7%, 33.3%, and 60% for C, E, and G, respectively.
- **Musical Nuances and Timing:** Capturing the subtle aspects of music, such as slurs, staccatos, and grace notes, requires a deep understanding of musical context and is often difficult to automate. Moreover, the synchronization of onsets and offsets between different voices, governed by the regular metrical structure of the music, is critical and contravenes the assumption of statistical independence between sources, complicating the separation process further.
- **Annotation and Ground Truth Challenges:** Annotating ground-truth transcriptions for polyphonic music is time-consuming and demands high expertise. The scarcity of such annotations has restricted the application of powerful supervised-learning techniques to specific AMT subproblems [3, 4], such as piano transcription. Even accurate sheet music does not generally provide ideal ground-truth annotations for AMT as it is not time-aligned to the audio signal and often does not represent a performance accurately.

1.2 Literature review

This section meticulously examines the existing literature in Automatic Music Transcription (AMT), delineating the connections and divergences between prevailing works and the methodologies and transcription types relevant to the present project. The analysis is organized into four focused subsections to facilitate a layered exploration of the domain. The initial subsection delves into the various types of AMT, offering insights into their unique characteristics and functionalities. Following this, the subsequent subsection delves deeper, exploring the intricate abstraction levels within AMT and illuminating the complexities inherent in each. The third subsection succinctly highlights key state-of-the-art papers, providing a snapshot of the most innovative and influential contributions to the field. Concluding the section, a comprehensive market analysis is conducted to compare the myriad of products currently available, shedding light on their features, capabilities, and positioning in the AMT landscape.

1.2.1 Transcription Types

When exploring AMT, it's pivotal to recognize a range of transcription types, each introducing its own unique complexities and considerations. A succinct summary of these transcription types is provided below, focusing exclusively on the nuances involved in each transcription type. These categories have been extracted from [3, 4, 5].

In **Piano Solo Transcription**, the model focuses on converting piano audio into sheet music. It necessitates intricate signal processing to recognize each note and its characteristics from a solo piano performance. The model typically outputs a time-pitch representation with precise time and pitch resolution, distinguishing pitch activation, onset, and offset channel to generate *MIDI* transcription results. The major challenges in piano solo transcription involve handling a vast range of notes and ensuring accurate note detection and timing.

In contrast, **Multi-instrument Polyphonic Transcription** deals with music pieces containing multiple instruments, making it more challenging due to overlapping harmonics and timbres from different instruments. The transcription model usually supports various instrument classes and outputs multiple channels of piano rolls, each representing one class of instrument. The precision of such transcription models is crucial, especially when the instrument classes in the test music piece are unknown, adding a layer of complexity and requiring refined classification capabilities.

Drum Transcription involves the identification of percussive events within an audio clip. Given the distinct and often abrupt nature of drum sounds, this transcription requires a model capable of accurately predicting the onsets of such events and is usually developed with attention to beat information.

Vocal Transcription in a polyphonic environment targets the extraction of vocal elements from a mix of sounds, a complex task due to the overlapping and interweaving of multiple sound sources. This transcription type requires sophisticated models capable of frame-level pitch extraction and note segmentation, ensuring precise vocal line extraction from polyphonic mixes. Handling pitch extraction and note segmentation precisely is crucial for the accurate transcription of vocal components amidst instrumental sounds.

Chord Recognition focuses on analyzing the harmony within a piece of music, identifying chord changes and progressions. This involves segmenting the input into different chords and recognizing the chord progression based on the segmentation result, necessitating models with high accuracy in recognizing harmony and chord transitions within a given piece, whether the input is audio or symbolic music data.

Lastly, **Beat/Downbeat Tracking** is dedicated to identifying the rhythmical elements of music, the beats, and downbeats, in symbolic music data. This requires models that can accurately predict the probability values of beat and downbeat for each time step, even on synthesized audio, paying close attention to rhythmic components to ensure precise tracking of beat and downbeat positions in music pieces.

1.2.2 Depth of Transcription in AMT

In the realm of Automatic Music Transcription (AMT), the concept of 'depth' refers to the various abstraction levels–namely frame, note, and stream–at which transcription processes operate. Each of these levels unravels distinct facets of music transcription and serves as a critical lens through which the underlying musical structures and elements are perceived and analyzed.

Each abstraction level abstraction offers distinct insights and attends to various facets of music transcription, with complexity escalating from frame to notation level. The multitude of approaches within each level exemplifies the extensive and profound research in AMT, ranging from basic frequency estimations to developing comprehensive, readable musical notations.

- **Frame Level Transcription:** at this level, the emphasis is on the estimation of the number and pitches of notes in each time frame (around 10ms) [3, 4]. Traditional Signal Processing [6], Neural Networks [7], Probabilistic [8] and Bayesian Models [9] are the primary techniques used. Traditional processing is simple, fast, and generalizes well, whereas Neural Networks offer higher accuracy for specific instruments but are instrument-specific. However, these methods often lack the conceptualization of musical notes and high-level musical structures.
- **Note Level Transcription:** here, the focus is on connecting pitch estimates over time into notes [3, 4] using techniques like Hidden Markov Models (HMM) [10] and Neural Networks for post-processing frame-level outputs. It provides a higher abstraction level, considering the onset and offset of notes. However, it often faces challenges such as ambiguity in note offsets and incorrect estimations due to harmonic sharing among notes.
- **Stream Level Transcription:** This level groups estimated pitches or notes into streams corresponding to different instruments, offering greater abstraction and consideration of timbral characteristics for stream differentiation [11]. It remains underexplored compared to note and frame-level transcription, due to its higher complexity and consideration of multiple musical structures and cues [4].

1.2.3 Prominent Studies

Several prominent studies and their methodologies are noteworthy in understanding the current state of the art in AMT:

- Hung et al. [12] utilized musical scores as supervised learning for source separation, focusing on separating sounds of different musical instruments.
- Jansson et al. [13] applied U-Net CNN, originally used in medical imaging, for source separation of audio, focusing on separating vocal and backing track in music.
- Lin et al. [14] introduced a zero-shot unified model for source separation, transcription, and synthesis allowing for implementation of separation, transcription, and synthesis of new pieces using U-Net CNN.

- Manilow et al. [15] developed a single deep learning architecture for separating and transcribing musical mixtures into single-instrument recordings and transcribe these instruments into a human-readable format at the same time.
- Hernandez-Olivan et al. [16] analyzed the impact of instrument timbre on AMT, developing models less dependent on onset strength, excelling in transcription for non-piano instruments and outperforming models like Google Magenta Onset and Frames (OaF).
- Leś et al. [17] highlighted the utility of synthesized audio data in training universal models for AMT, acting as an effective base for pretraining general-purpose models and facilitating quicker adaptation of transcription models for various instruments.
- Simonetta et al. [18] combined Deep Learning models for AMT with HMM-based alignment to achieve advanced Audio-to-score alignment (A2SA) at the note-level, exploiting large, unaligned datasets effectively.
- Maman and Bermano [19] introduced NoteEM for AMT, enabling simultaneous transcriber training and score alignment, proving versatile across various datasets, overcoming existing data collection limitations in AMT.

While some methodologies and ideas from earlier papers have been realized in open-source software, the innovative approaches and findings in the more recent publications, notably the last four described, have not yet been implemented widely and represent promising avenues for advancing the state-of-the-art in Automatic Music Transcription, offering substantial opportunities for enhancement and refinement in the field. Furthermore, there are no approaches using Large Language Models, which have been shown to generalize to other domains with fine-tuning.

1.2.4 Market Analysis

Several music transcription tools exist to facilitate the transformation of music into sheet form, each with distinct features, advantages, and shortcomings:

- ScoreCloud [2] (proprietary) excels in versatile music transcription, specializing in producing advanced musical notations and offering synchronization of tracks over the internet.
- MuseScore 4 [20] an open-source option, offers comprehensive features and a clear design, emphasizing accessibility and extensive score access. Very limited abilities in AMT task.
- AnthemScore [21] (proprietary) utilizes advanced neural networks (architecture not specified) for accurate transcription and offers customization of various musical elements.
- Noteflight [22] (proprietary) operates online, offering ease of use, with access to an active community and marketplace, no information on the AMT tech specifications.

Additionally, there are Digital Audio Workstations, like Cubase 12 [23], and other open-source solutions capable of converting music to sheet form. For example, Omnizart [5], a Tensorflow based Python library that provides a streamlined solution to automatic music transcription.

However, they, along with the previously mentioned products, often share common issues as AMT is not a solved problem. All products still present various common issues in their outputs, including octave errors, missed or extra notes, merged or fragmented ones, incorrect onsets/offsets, or misassigned streams, especially in the presence of dense chords and unseen timbres.

Many current AMT systems do not fully address these challenges, leading to inaccuracies and unreliable transcriptions. Potential solutions involve leveraging professional music performers for annotations and focusing on thorough score pre- and post-processing, but these are not without their own sets of challenges and limitations. Nevertheless, while the specific technologies used for most of these systems is not available, we can assume that they are a bit outdated with the advent of Large Language Models. Thus, the possibility of improving the performance of competing AMT systems is reasonable.

2 Project Overview

2.1 Goals

In this section, we delineate the principal goals and objectives essential for the successful completion of the project. The overarching aim is to devise an intelligent system proficient in transcribing a song into sheet music for each instrument present within it. A pivotal objective is to disentangle a song, into its constituent instrumental tracks. Given the potential complexities, a selection of instruments will be prioritized to assure the acquisition of respective audio tracks.

Subsequently, the isolated audio tracks are to be converted to a more suitable format, specifically, *MIDI*. With the audio aptly structured in *MIDI* format, we can accurately identify notes and their timestamps, enabling us to realize our final major objective: the transformation of *MIDI* into coherent and readable sheet music that is musically representative of the original audio.

The core objectives of this project aim to construct a system adept in intelligent music transcription. The end-goal is to convert a song into musical sheets for each instrument involved. The goals, systematically divided into primary and sub-goals, are detailed as follows:

- **Consolidate Varied Audio Formats:**
  - Identify and consolidate multiple input audio formats to a standardized format to ensure uniform processing (for example, *MP3*).
  - Develop procedures to maintain the integrity and quality of the audio during the format conversion.
  - Test and validate the consolidation process with different audio formats to guarantee reliable conversion.
- **Separate a Song into Individual Instrument Tracks:**
  - Separate an *MP3* song into distinct audio tracks representing different instruments, acknowledging the potential limitations in handling multiple instruments.
  - Identify instances of polyphony where multiple notes are played simultaneously.

  - Distinguish notes originating from different instruments.
  - Validate the separation by cross-referencing the identified instruments and their corresponding tracks.

- **Transform Individual Audio Tracks to *MIDI* Format:**
  - Develop a profound understanding of the *MIDI* format and its representation of music.
  - Process the converted audio to eliminate unwanted artifacts.
  - Validate the conversion by reverting the process and cross-referencing with the original.

- **Transcribe *MIDI* to Musical Sheets:**
  - Research how the state-of-the-art methods usually convert *MIDI* to music sheets.
  - Transcribe comprehensive musical elements including key signatures, time signatures, and other musical notation elements.
  - Detect inconsistencies and post-process the music sheets to improve their quality.
  - Validate the music sheets with experts and original music sheets done by composers, ensuring accuracy and completeness in musical representation.

- **Develop a User Interface:**
  - Create an intuitive and user-friendly interface to enhance user interaction and product appeal.
  - Incorporate features enabling users to easily navigate through the application and access its functionalities.
  - Test the interface with potential users to ensure usability and gather feedback for further improvements.

2.2 Motivation

The manual conversion of audio into sheet music is both time-consuming and specialized work. Automating this process will significantly benefit musicians, composers, and musicologists for purposes like studying compositions, sharing musical ideas, and educational endeavors. Music has the power to inspire and connect people on a profound level. Yet, the intricate interplay of instruments in a musical composition often remains hidden, overshadowed by the collective sound. Our project is motivated by several compelling reasons:

- **Enhancing musical education**: Music students and enthusiasts often seek to understand the nuances of their favorite songs or learn to play specific instruments. Providing music sheets for individual instruments empowers them with invaluable resources for focused practice and learning.

- **Easing collaboration**: Musicians, whether amateurs or professionals, often collaborate on projects. Isolating individual instrument scores simplifies the process of collaboration, allowing future musicians to practice their parts independently before coming together to create harmonious melodies.

- **Promote creative possibilities**: Musicians and composers may find inspiration in dissecting the individual components of a song. By providing music sheets for each instrument, we encourage experimentation and creative reinterpretation of existing compositions. As well, this will let musicians quickly iterate through musical ideas by playing them and seeing instant transcriptions.

- **Diverse musical experience**: Listening to a song while following along with a specific instrument's music sheet provides a deeper appreciation for the complexities of music. It allows listeners to immerse themselves in the subtleties of each instrument's contribution.

2.3 Requirements

In this section, we will explain the functional and non-functional requirements that we consider important for this project. Even if they provide a comprehensive framework for planning and executing an Automatic Music Transcription tool, these requirements may vary a little bit through the project development. On one hand, there are the functional requirements which describe the desired end function of a system operating within normal parameters. And, on the other hand, there are the non-functional requirements which do not speak directly on the operation of the system. Otherwise, they must be taken into account from the beginning for the development of the goals set, and they have to be considered during the whole project.

2.3.1 Functional

The functional requirements are the ones below:

- **Multiple Instruments and tracks**: Support the conversion of audio for various musical instruments and voices, accommodating different pitch ranges, timbres, and instruments like piano, guitar, violin, etc. into separate music sheets for each instrument or voice.

- **Machine Learning and AI**: Incorporate machine learning and AI algorithms to achieve effortless correct conversions, hoping we obtain better results than previous tools.

- **Format Compatibility**: Ensure compatibility with standard music notation formats (e.g. MusicXML), making it easy for musicians to work with and edit the generated music sheets.

- **Feedback Mechanism**: Implement a feedback mechanism to collect user feedback and improve the system based on user experiences.

- **Documentation and Support**: Provide comprehensive documentation, tutorials, and customer support to assist users in effectively utilizing the tool.

2.3.2 Non-Functional

The non-functional requirements are the ones below:

- **Accuracy**: Ensure that the converted music sheets are as accurate as possible when compared to the original audio. This includes accurately transcribing notes, rhythms, dynamics, and other musical nuances.
- **Speed and Efficiency**: Aim for fast processing times to convert audio to music sheets, especially for longer pieces of music.
- **Robustness**: Ensure that the system can handle a variety of audio qualities, including recordings with background noise, varying levels of instrument/vocal clarity, and different recording environments.
- **Accessibility**: Make the tool accessible to individuals with disabilities, ensuring that it can be used by people with visual or hearing impairments and the usage of a user-friendly and intuitive interface for musicians and composers to upload audio files and receive music sheets.
- **Scalability**: Design the system to handle a large volume of audio conversions, especially if the project gains popularity.

3 Initial Strategy

In this section, we will review our initial approach to solve the problem. The strategy is based mainly on the goals defined in Section 2.1, following a *Multi-instrument Polyphonic Transcription* on a *Stream-Level*, nevertheless, these are subject to change.

As mentioned in the goals, after the user inputs an audio with a song, it will be converted to a standardized format, which will then be feed into a *Splitter*. This *Splitter* will divide the song into the different audio tracks, for example in *MP3* format. This is depicted in Figure 2 with the box numbered as *1. Splitter*. Currently, we are not sure if this step will be carried by our own handcrafted model, or we will use an already existing project.

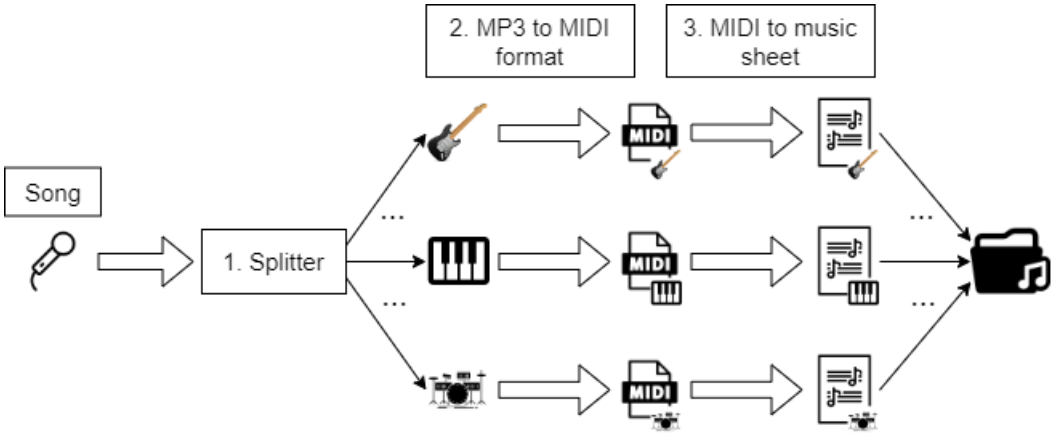

Figure 2: Summary of the initial strategy to follow.

Afterward, we will convert the independent audio tracks from *MP3* to *MIDI*, a specialized audio format that will help us to generate the music sheets for the next steps. This can be seen as the process identified as *2. MP3 to MIDI format*. It is crucial to emphasize that the quality of the conversion step will significantly influence the complexity of generating music sheets from the *MIDI* inputs. Again, we are not sure if we will use our own model or if we will use an already trained one. Regardless of the chosen approach, we intend to implement a post-processing step to ensure the conversions meet a high standard of quality.

Finally, the conversion of *MIDI* tracks to accurate music sheets—designated as process *3. MIDI to Music Sheet*—is perhaps the most crucial step. This process may take a while to complete, and many different models, of our own and existing ones, will be implemented and tested in order to achieve plausible music sheets generation. Upon completion, the ensemble of music sheets will be presented to the user who uploaded the song, enabling them to play/learn the original song faithfully. This culminates the process, and expect some kind of feedback from the user in order to tweak some parameters or do the appropriate modifications to improve the generation capabilities of our system.

We would like to note that our preliminary search for data has yielded three prospective datasets [24, 25, 26]. These will undergo thorough analysis, and are anticipated to be pivotal in the development of our models for conversion to *MIDI* and transcription to music sheet.

4 Preliminary Analysis

4.1 Sustainability Analysis

The sustainability analysis provides a detailed exploration of current environmental thinking from a variety of perspectives, including social, economical and environmental edges. This analysis recognizes that the success of our project extends beyond mere technological achievement, emphasizing our commitment to responsible practices that benefit the environment, society, and the economy.

4.1.1 Social Edges

Social sustainability is a cornerstone of our project, as we aim to create positive impacts within the music community and society at large. One key aspect is inclusivity, where our AI-generated music sheets have the potential to make music notation more accessible to everyone.

By offering free access to foundational music sheet generation tools and educational resources, we endeavor to level the playing field for musicians from varied socio-economic backgrounds. This approach aims to engender a sense of fairness within the music domain, cultivating a musical environment that is more socially sustainable and inclusive.

4.1.2 Economical Edges

On the economic sustainability of the project, we recognize the importance of conducting a comprehensive cost-benefit analysis to ensure the long-term viability of our AI-driven music sheet generation system. Through market research and analysis, we have identified a growing demand for AI-generated music sheets among music educators, composers, and students.

Additionally, we have assessed potential competitors and devised strategies to differentiate our offering, ensuring our long-term sustainability and relevance in a competitive market.

On the other hand, as our project expands, we anticipate the creation of job opportunities in various domains. This includes software developers, AI specialists, musicologists, and customer support staff. By contributing to job growth, our project not only enriches the employment landscape but also supports the local economy.

On the cost optimization side, we have developed a detailed cost analysis that foresees possible costs due to risks and unforeseen events. This gives us a certain credibility and security in the economic growth of the project. You can review this economic analysis in section 4.2.

4.1.3 Environmental Edges

Environmental sustainability is a crucial facet of our project's overarching strategy. In our commitment to reduce the ecological footprint, we diligently evaluate the energy consumption associated with training and operating the AI system.

To minimize environmental impact, we will explore the adoption of energy-efficient hardware and cloud computing solutions, harnessing technology that maximizes computational efficiency while minimizing power usage. For this, we will use wherever possible *Google* services to train and analyze our Deep Learning models, and perhaps try to avoid using these computationally-expensive models whenever possible. This approach not only lowers energy consumption but also aligns with our commitment to environmental responsibility.

In terms of carbon emissions, we plan to calculate the project's carbon footprint. For instance, during the model training phase, we will measure the electricity consumption and compute the associated carbon emissions. By doing so, we can demonstrate our commitment to carbon neutrality and contribute to global efforts in combating climate change.

Ultimately, we envision our project as a catalyst for the digitization of music resources, a transformation that holds the potential to make a substantial dent in the rampant paper production within the music industry.

4.2 Economic Analysis

An important part of project planning is the economic study, which consists on identifying and estimating the costs associated with this work. In order to be able to correctly identify the costs that the project will entail, we must take into account both human resources and hardware and software resources. In addition, it is also necessary to take into account contingencies and costs related to possible unforeseen events. In this section, a complete study of the economic viability of the project will be carried out.

4.2.1 Human resources

The staff profiles involved in this project consist of a project manager and three developers, each with their assigned roles: a Team Leader (Benjamí Parellada), responsible for overseeing the progress and coordination of development tasks; a Solutions Architect (Armando Rodriguez), who focuses on designing robust and scalable solutions and addressing technical challenges; and a Software Developer (Gerard Caravaca), tasked with implementing and testing the designed

solutions in accordance with project requirements. The project manager will be played by the professor Miquel Sànchez. Table 1 shows the detailed hourly wage breakdown for each role.

Role	Gross salary	Social security	Remuneration
Director (D)	21.65 €\h	6.50 €\h	28.15 €\h
Developer (P)	12.69 €\h	3.81 €\h	16.50 €\h

Table 1: Summary of staff costs. In order to calculate the estimated salaries for each role, estimates provided by the *Indeed* [27] website have been used. We assume that the different roles do not impact the salary of the developers, as they are all three Junior Software Developers.

Once the individual cost of each staff profile have been established, the total cost can be estimated, taking into account the estimated hours in the syllabus of the subject. In Table 2 can be seen the profiles involved in the project and the total cost of each of them.

Role	Estimated hours	Total cost
Director (D)	50 h	1,400 €
Developer (P1, P2, P3)	150 h	2,475 €
Total	650 h	8,825 €

Table 2: Total staff cost estimate.

4.2.2 General expenses

In this section, the generic costs are calculated, which have to be computed in a general way because they do not depend on the tasks of the project. We will consider as general costs depreciation, working space and electricity and internet consumption.

- **Software:** all software used is expected to be open source. For this reason, the estimated amortization is 0 €.
- **Hardware:** the required hardware for this project consists of two mid-range laptop computers, i.e. at a cost of 650 €, and a high-range computer, i.e. at a cost of 1,000 €. Assuming that, on average, the life span of a computer of these characteristics is about five years, the amortization of these resources is 5/60 * 650 + 5/60 * 1000 = 191.65 €.
- **Work space:** the space needed to develop the project is a room to work in and a room for meetings. In this case, we will use the room in our houses and the FIB classroom. The cost of a room similar to ours in Barcelona is around 400 €/month and the cost of a meeting room like the one at the FIB is around 16 €/hour. In total the work space costs are 400*5*3 + 16*35 = 6,560 €, taking into account that the course has 5 months and we estimate around 35 hours of meetings and co-working.
- **Internet bill:** according to personal bills, the internet fee is around 43 €/month. With a working day of 2 hours a day the internet expenditure would be a total of 5*40*4/24 * 4 = 133.33 €.
- **Electricity consumption:** given that the current electricity price is about 0.2437 €/kWh and taking into account that the consumption of a laptop in 150 hours is 41.25 kWh. We can conclude that the electricity cost would be 0.2437*41.25*3 = 30.16 €.

Once the generic costs have been broken down, we can calculate the total cost of this section. This calculation is shown in Table 3.

Concept	**Cost**
Software	0.00 €
Hardware	191.65 €
Work space	6,560.00 €
Internet bill	133,33 €
Electricity consumption	30,16 €
Total	6,915.14 €

Table 3: Total general cost estimate.

4.2.3 Total cost

The table shows the final budget needed to carry out the project according to the estimates. A contingency cost of 20% of the total has been added to prevent additional costs due to possible risks.

Concept	**Cost**
Personal cost	8,825.00 €
General costs	6,915.14 €
Contingency	3,148.00 €
Total	18,888.17 €

Table 4: Table of final estimated costs for the project.

4.2.4 Economic management control

During the course of the work, possible deviations in the budget will be monitored. The objective is to ensure that the difference between actual and estimated costs is kept to a minimum. The following metrics will be used to carry out this control:

$$\mathbf{CD} = (AC - CV) \cdot AHR \tag{1}$$

$$\mathbf{CV} = (ECH - AHC) \cdot EC \tag{2}$$

where **CD** represents the Cost Deviation, calculated as the difference between the Actual Cost (AC) and Consumption Variance (CV) multiplied by the Actual Hours Rate (AHR). Here, **CV**, the Consumption Variance, is found by subtracting the Actual Hours Consumption (AHC) from the Estimated Consumption of Hours (ECH), with the result then being multiplied by the Estimated Cost (EC). In summary, **CD** and **CV** serve to quantify discrepancies in cost and consumption, respectively, utilizing actual and estimated values of cost and hours consumed in their calculations.

5 Project Management

Project management is the central piece that must ensure that the work is always being done in the right direction. To ensure this, a very specific working methodology will be used, which will be carried out with tools that will allow us to advance successfully in our tasks.

5.1 Project Management Methodology

In choosing a project management methodology, we have decided to adopt the Agile approach. Agile methodologies, such as Scrum or Kanban, emphasize flexibility and iterative development. This choice aligns with our project's dynamic nature, where requirements may evolve. Agile allows us to adapt to changing circumstances, promotes teamwork, and facilitates a more efficient response to emerging issues.

Specifically, we have decided to apply the **Scrum** [28] methodology for this project. Scrum is a methodology that aims to achieve the best results by coordinating the team in order to be highly productive. The idea is that with Scrum, regular and partial handovers or sharing of the work carried out are carried out as a priority and according to the benefits that each one brings to the recipients of the project. For this reason, it is a methodology that is particularly suitable for complex projects with multiple requirements and more than one participant, as is the case with this work.

The Scrum methodology goes through the following different phases, shown in Figure 3:

- **Planning:** this is the phase in which the priority tasks are established and where brief and detailed information about the project to be developed is obtained. It is necessary to be able to start with the first sprint, it allows you to change and grow as many times as necessary depending on the learning acquired in the development of the product.
- **Execution (Sprint):** this is defined as a subproject in which the work team focuses on the development of tasks to achieve the previously defined objective. Each sprint is no longer than one month, and most commonly will lasts two weeks.
- **Control:** this is the phase in which the progress of the project is measured and assessed periodically, in daily meetings.

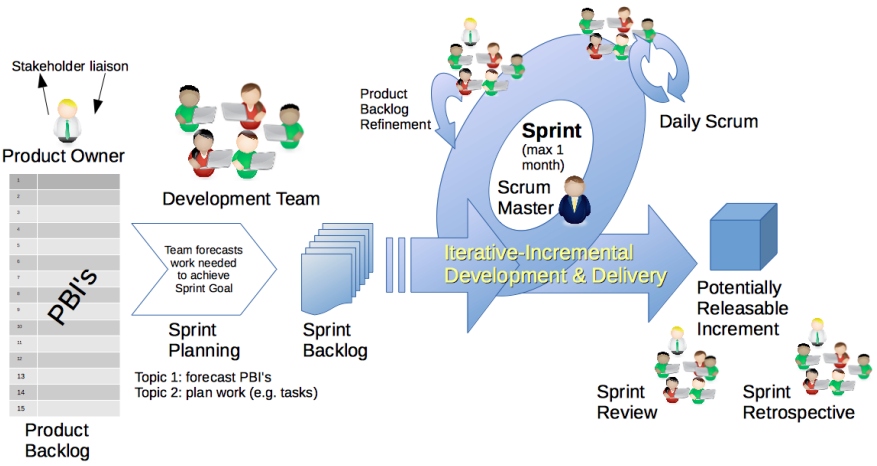

Figure 3: Scrum methodology phases diagram. [28]

We believe that Scrum is the most suitable methodology for our project, as it allows us to start working on the application at the same time as we are studying the subject theoretically. Moreover, the fact that it is a cyclical process gives us a certain flexibility in the event of errors.

5.2 Project Management Tools

To support our Agile project management approach, we will employ a comprehensive suite of digital tools carefully selected to align with our project's unique requirements and objectives. These tools encompass a wide range of functionalities, ensuring that every aspect of project management and collaboration is optimized for efficiency and effectiveness.

Jira [29] will serve as the central hub for our project management activities. This versatile tool will enable us to track tasks, issues, and user stories, facilitating sprint planning, backlog management, and progress monitoring. Its flexibility and scalability make it an ideal choice to adapt to our evolving project needs. In conjunction with Jira, **Confluence** [30] will play a pivotal role in knowledge sharing and documentation. This collaborative platform will empower our team to create, edit, and organize project documentation, user stories, and meeting notes. The seamless integration with Jira ensures that our documentation remains up-to-date, providing a single source of truth for project information.

Version control and source code management are paramount in our software development efforts. **Github** [31] will enable us to maintain a well-structured codebase, supporting collaborative coding, code reviews, and branching strategies. In addition to this, we will also use **Google Colab** [32] to share the training and evaluation code of the Deep Learning models implemented during the project. This approach ensures code quality and enables us to respond swiftly to changing requirements.

Finally, in the branch of the meeting's management, **Discord** [33] will be our chosen platform for communication and team coordination. Its voice and text chat capabilities, along with the ability to create dedicated servers and channels, will streamline communication and foster a collaborative environment. Furthermore, we will leverage **Google Workspace** [34] for cloud-based document collaboration, email communication, and calendar management. This suite of tools will enhance our productivity, making it easy to share documents, schedule meetings, and maintain a synchronized team calendar.

6 Initial Risks Identification

As in any professional project of this magnitude, unforeseen events and problems can arise. Therefore, it is always important to make a list of the main setbacks that we may encounter during the course of the work:

- **Time management:** in this case, we are talking about a poor forecast of the time it will take to complete a task. It is due to unrealistic forecasts in the planning or successions of consecutive risks.
- **Unforeseen results:** one of the inherent challenges in deep learning projects is the unpredictability of results. The model's performance might not meet the desired expectations, or it may exhibit unexpected behaviors. These issues could arise due to various factors, including inadequate training data, model architecture, or hyperparameter choices. Addressing unforeseen results may require iterative model refinement, which can impact project timelines and resource allocation.

- **Hardware and infrastructure limitations:** Deep learning often demands significant computational resources. Hardware failures, resource bottlenecks, or scalability issues could disrupt the workflow. Adequate provisioning and monitoring of hardware and infrastructure are essential to ensure the project runs smoothly.

- **Design failures:** these types of errors are due to the fact that at some point in the design phase of the project we made a mistake. Therefore, they cause us to modify the initial design, and may include errors in efficiency, scalability and data acquisition.

- **Ethical considerations:** As with any AI project, ethical concerns must be addressed. This includes issues related to bias in data or algorithms, the potential for misuse, and unintended consequences. To prevent this, an in-depth study must be carried out so that the model ensures a good performance regardless of the musical style or the source of the input song.

- **External factors:** External factors like economic conditions, technological advancements, or changes in regulations can impact the project. These external risks may require adaptability and contingency planning (Section 4.2) to ensure the project's resilience in the face of changing circumstances.

While unforeseen challenges are inevitable in any complex project, this risk analysis, constant vigilance, and a dedicated team can mitigate these risks and keep the project on course. Our core strategies will involve routine risk evaluations and adaptability to the evolving circumstances.

7 References

[1] *The MIDI Association - Official MIDI Specifications*. URL: https://www.midi.org/specifications/file-format-specifications/standard-midi-files.

[2] *ScoreCloud - Play your music – ScoreCloud notates!* 2023. URL: https://scorecloud.com/.

[3] Emmanouil Benetos et al. "Automatic music transcription: challenges and future directions". In: *Journal of Intelligent Information Systems* 41 (2013), pp. 407–434.

[4] Emmanouil Benetos et al. "Automatic music transcription: An overview". In: *IEEE Signal Processing Magazine* 36.1 (2018), pp. 20–30.

[5] Yu-Te Wu et al. "Omnizart: A General Toolbox for Automatic Music Transcription". In: *Journal of Open Source Software* 6 (Dec. 2021), p. 3391. DOI: 10.21105/joss.03391.

[6] Li Su and Yi-Hsuan Yang. "Combining spectral and temporal representations for multipitch estimation of polyphonic music". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.10 (2015), pp. 1600–1612.

[7] Rainer Kelz et al. "On the potential of simple framewise approaches to piano transcription". In: *arXiv preprint arXiv:1612.05153* (2016).

[8] Zhiyao Duan, Bryan Pardo, and Changshui Zhang. "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions". In: *IEEE Transactions on Audio, Speech, and Language Processing* 18.8 (2010), pp. 2121–2133.

[9] Paul H Peeling, A Taylan Cemgil, and Simon J Godsill. "Generative spectrogram factorization models for polyphonic piano transcription". In: *IEEE transactions on audio, speech, and language processing* 18.3 (2009), pp. 519–527.

[10] Juhan Nam et al. "A Classification-Based Polyphonic Piano Transcription Approach Using Learned Feature Representations." In: *Ismir*. Citeseer. 2011, pp. 175–180.

[11] Zhiyao Duan, Jinyu Han, and Bryan Pardo. "Multi-pitch streaming of harmonic sound mixtures". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.1 (2013), pp. 138–150.

[12] Y.-N. Hung, G. Wichern, and J. Le Roux. "Transcription is all you need: Learning to separate musical mixtures with score as supervision". In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 46–50.

[13] A. Jansson et al. "Singing voice separation with deep u-net convolutional networks". In: *ISMIR* (2017).

[14] L. Lin et al. "A unified model for zero-shot music source separation, transcription, and synthesis". In: *arXiv preprint arXiv:2108.03456* (2021).

[15] E. Manilow, P. Seetharaman, and B. Pardo. "Simultaneous separation and transcription of mixtures with multiple polyphonic and percussive instruments". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 771–775.

[16] Carlos Hernandez-Olivan et al. "A comparison of deep learning methods for timbre analysis in polyphonic automatic music transcription". In: *Electronics* 10.7 (2021), p. 810.

[17] Michał Leś and Michał Woźniak. "Transfer of knowledge among instruments in automatic music transcription". In: *arXiv preprint arXiv:2305.00426* (2023).

[18] Federico Simonetta, Stavros Ntalampiras, and Federico Avanzini. "Audio-to-score alignment using deep automatic music transcription". In: *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2021, pp. 1–6.

[19] Ben Maman and Amit H Bermano. "Unaligned Supervision for Automatic Music Transcription in The Wild". In: *Proceedings of the 39th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 14918–14934. URL: https://proceedings.mlr.press/v162/maman22a.html.

[20] *MuseScore - The world's most popular notation app*. URL: https://musescore.org/es.

[21] *AnthemScore 4 - Music AI for Your PC*. URL: https://www.lunaverus.com/.

[22] *Noteflight*. URL: https://www.noteflight.com/.

[23] *CUBASE 12*. URL: https://o.steinberg.net/index.php?id=15262&L=1.

[24] John Thickstun, Zaid Harchaoui, and Sham Kakade. *Learning Features of Music from Scratch*. 2017. arXiv: 1611.09827 [stat.ML].

[25] Adrien Ycart and Emmanouil Benetos. *A-MAPS: AUGMENTED MAPS DATASET WITH RHYTHM AND KEY ANNOTATIONS*. 2018. URL: http://qmro.qmul.ac.uk/xmlui/handle/123456789/45985.

[26] Curtis Hawthorne et al. *Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset*. 2019. arXiv: 1810.12247 [cs.SD].

[27] Indeed. *Indeed salary estimator*. https://es.indeed.com/career/. 2023.

[28] Atlassian. *Jira blog about Scrum methodology*. https://www.atlassian.com/es/agile/scrum. 2023.

[29] Atlassian. *Jira product website*. https://www.atlassian.com/es/software/jira. 2023.

[30] Atlassian. *Confluence product website*. https://www.atlassian.com/es/software/confluence. 2023.

[31] Github. *Github official website*. https://github.com/. 2023.

[32] Google. *Google Colab introductory tutorial*. https://colab.research.google.com/?hl=es. 2023.

[33] Discord Inc. *Discord official website*. https://www.discord.com/. 2023.

[34] Google. *Google Workspace blog*. https://workspace.google.com/. 2023.