Music segmentation analysis in Iberian folk music

Amir Abbas Orouji

Mestrado em Multimédia da Universidade do Porto

Supervisor: Gilberto Bernardes de Almeida

Co-supervisor: Nadia Carvalho

March 2025

Music segmentation analysis in Iberian folk music

Amir Abbas Orouji

Mestrado em Multimédia da Universidade do Porto

Aprovado em provas públicas pelo Júri:

Presidente: Rui Pedro Amaral Rodrigues

Vogal Externo: Filipe Lopes

Orientador: Gilberto Bernardes de Almeida

Abstract

Music segmentation is one of the fundamental tasks within music information retrieval (MIR). It aims to understand and mimic how our brain understands music and segments it into different structural elements such as beats, meters, phrases, and sections. This work focuses on the automatic phrase segmentation subtask. The models used for phrase segmentation can be divided into three major categories: rule-based, supervised, and unsupervised models. Rule-based models are based on musical rules and how humans perceive music phrases. The second and third categories handle phrase detection as a machine-learning problem and detect the musical phrases by analyzing the pieces as an annotated or unannotated data. Among many statistical and music-theory approaches, the local boundary detection model (LBDM) is a prominent method due to its accuracy and cognitive-basis explainability. It relies on musical changes to shape a novelty curve and find the candidate segments based on pitch intervals, inter-onset intervals, and rest information extracted from symbolic musical surface structure (i.e., scores). An identified limitation of the LBDM algorithm is the lack of repetition awareness in the musical surface structure. To this end, we study the enhancement of the baseline model with structural repetition information resulting from the variable Markov oracle (VMO). A finite-state automaton that can efficiently search for factors (substrings) in a string. Furthermore, we assess optimal weights for each of the surface elements information extracted, using genetic algorithm search heuristic and entropy, which is a measure of information in a desired variable change.

To evaluate our models, we adopted a novel dataset of Iberian folk music consisting of 59 Portuguese and 738 Spanish folk pieces in symbolic representation, and we used the commonly used F-measure as our reference metric to assess the effectiveness of our models. Our results show that with a good amount of good-quality data and an optimization algorithm, the LDBM, or any weight-based model, can be optimized and achieve better performance for a specific genre, region, or dataset. Our model, with sufficient training data, managed to increase the overall F-measure of the original LBDM in Spanish folk music from 60% to 67% which is even higher than what has been reported in previous literature.

Resumo

A segmentação musical é uma das tarefas fundamentais da recuperação de informação musical (MIR). O seu objetivo é compreender e imitar a forma como o nosso cérebro compreende a música e a segmenta em diferentes elementos estruturais, tais como batidas, metros, frases e secções. Este trabalho centra-se na subtarefa de segmentação automática de frases. Os modelos utilizados para a segmentação de frases podem ser divididos em três categorias principais: modelos baseados em regras, supervisionados e não supervisionados. Os modelos baseados em regras baseiam-se em regras musicais e na forma como os humanos percepcionam as frases musicais. A segunda e terceira categorias tratam a deteção de frases como um problema de aprendizagem automática e detectam as frases musicais analisando as peças como dados anotados ou não anotados. Entre muitas abordagens estatísticas e de teoria musical, o modelo de deteção de limites locais (LBDM) é um método proeminente devido à sua exatidão e explicabilidade de base cognitiva. Baseia-se em alterações musicais para moldar uma curva de novidade e encontrar os segmentos candidatos com base em intervalos de altura, intervalos de interconexão e informações de repouso extraídas da estrutura simbólica da superfície musical (ou seja, partituras). Uma limitação identificada do algoritmo LBDM é a falta de consciência da repetição na estrutura da superfície musical. Para este efeito, estudamos a melhoria do modelo de base com informação estrutural de repetição resultante do oráculo variacional de Markov (VMO). Um autómato de estado finito que pode procurar eficientemente factores (substrings) numa cadeia. Além disso, avaliamos os pesos óptimos para cada uma das informações extraídas dos elementos de superfície, utilizando a heurística de pesquisa do algoritmo genético e a entropia, que é uma medida de informação numa mudança de variável desejada.

Para avaliar os nossos modelos, adoptámos um novo conjunto de dados de canções folclóricas ibéricas, constituído por 59 canções folclóricas portuguesas e 738 espanholas em representação simbólica, e utilizámos a medida F, normalmente utilizada, como métrica de referência para avaliar a eficácia dos nossos modelos. Os nossos resultados mostram que, com uma boa quantidade de dados de boa qualidade e um algoritmo de otimização, o LDBM, ou qualquer modelo baseado em pesos, pode ser optimizado e alcançar um melhor desempenho para um género, região ou conjunto de dados específico. O nosso modelo, com dados de treino

suficientes, conseguiu aumentar a medida F global do LBDM original em canções espanholas de 60% para 67%, o que é ainda mais elevado do que o relatado na literatura anterior.

Acknowledgments

First of all, I would like to express my utmost gratitude to my advisor, Dr. Gilberto Bernardes, and his PhD student Nádia Carvalho, for their constant willingness to advise and support that made it possible for me to meet this challenging deadline. Additionally, I would like to send my warmest thanks to my family and especially my beloved sister, Shaghayegh who even though she wasn't in Portugal but has always listened to my overthinking and annoying mind and can make any journey a lot more pleasant. Moreover, the words cannot describe the amount of kindness, love, and encouragement I have received from Ana Rita until the time I have finished this work. I also want to express my gratitude toward Marisa Silva who always welcomed me with her smile and helping hand in the faculty throughout this course. Last but not least, my special thanks and appreciation to all my friends in the residence hall who share the same burden of studying in a foreign country and managed to keep a smile on my face no matter the situation I was in.

Amir Abbas Orouji

Contents

List of Figures

List of tables

Abbreviations and symbols

MIR	Music Information Retrieval
SMC	Sound and Music Computing
INESC TEC	Institute for Systems and Computer Engineering, Technology and Science
LBDM	Local Boundary detection model
VMO	Variable Markov Oracle
FO	Factor Oracle
AO	Audio Oracle
LRS	Longest Repeated Suffix
SFX	Suffix link
GA	Genetic Algorithm
RF	Random Forest
GTTM	Generative Theory of Tonal Music
GPR	Grouping Preference Rules
IR	Implication-realization
IOI	Inter Onset Interval
DOP	Data-Oriented Parsing
RBM	Restricted Boltzmann machine
DH	Digital Humanities

1. Introduction

This Chapter will elaborate on the motivation and logic behind this work and a brief methodology along with the structure of this document. Firstly the motivation and logic behind the initiation of this work are discussed and further we continue with our methodology, objective, and the rationale behind it. Lastly, we will elaborate on the timeline for achieving the related objectives.

1.1 Context and Motivation

The proposed investigation falls within the Music Information Retrieval (MIR), Sound and Music Computing (SMC) areas, and Digital Humanities (DH). Digital humanities is an interdisciplinary field that applies computational methods and tools to the study of humanities disciplines, such as literature, history, philosophy, and the arts. It explores how digital technologies can enhance research, teaching, and the creation of new knowledge within the humanities. The field focuses on using data analysis, visualization, machine learning, and other techniques to process large datasets, make new connections, and enable novel interpretations of cultural and artistic artifacts. Additionally, sound and music as an inseparable part of human life and culture can be looked at as a gigantic source of data for the following purposes. SMC is a broad field that encompasses the study and application of computational methods in music creation, analysis, and interaction with sound. It covers areas like sound synthesis, algorithmic composition, performance technologies, and music perception. MIR is a subfield that deals with techniques that are focused on retrieving, analyzing, and organizing these music-related data. This study tried to build upon the previous literature in phrase segmentation and detection which is a sub-area in the MIR techniques and tries to see the possibility of improvement in existing models by examining and optimizing them on a novel tradition-specific folk music dataset.

This dissertation work is framed in a major European project EA-Digifolk which partners with universities and industry in Portugal, Spain, and Ireland. This project intends to improve the ways we understand, access, and analyze Folk music so we can trace the footprint of different cultural influences on each other and intertwined traditions of Iberian Folk music. In this project's

scope, the Sound and Music Computing Group at FEUP-INESC TEC has been assigned the task of developing algorithms to find similarities across folk music, using both their melodic and harmonic aspects. Towards that end, the task of segmenting folk music tunes and pieces into their structural phrases is fundamental to enhancing similarity metrics and retrieval.

1.2 Folk Music

Folk music refers to traditional music that originates from the cultural heritage of communities and is often passed down orally through generations. It is deeply rooted in people's daily lives, customs, and rituals and reflects their collective experiences, emotions, and history. Unlike classical or commercial music, folk music tends to be more community-driven and less formalized, with variations depending on the region, language, and culture in which it evolved (Ben-Amos, 2020; Nettl, 2005).

Folk music can be characterized by simple melodies, repetitive structures, and a focus on storytelling through lyrics. It is often performed with acoustic instruments, such as guitars, fiddles, and flutes, and incorporates local scales and rhythms, giving it a unique sound within different regions. In many cases, folk music serves as a way for communities to preserve their identity, history, and moral values, often addressing themes of love, labor, struggle, and nature and as a result, it can manifest itself in various ways. Bruno Nettl (2005) emphasizes this concept as he tries to define a credo for ethnomusicology by stating this field is the study of all the musical manifestations of society and by all, he refers to any kind of music of different classes and minorities.

This issue becomes more complicated when he brings up the concept that not all the manifestations can be discussed classically and expressed with common Western notation and methods. Additionally, he gives an example of the Hausa of Nigeria who have no Terms for music in their dialogue but they are engaged in many musical activities as cultural phenomena. With these crucial points in mind, cultural institutions prioritize digitizing and unlocking their collections to preserve and study the rich world of Folk music, society, and ourselves. Upon these Projects and the efforts of ethnomusicologists, the Co-Poem Project was born to preserve folk music with a special focus on the younger generation. Respectively, the byproduct of this project was a new database for Portuguese, Spanish, and Italian folk music called I-Folk (Carvalho et al., 2021) which will be adopted in this study.

1.3 Objectives

Expanding upon the previous context, the Project's main goal is to develop an efficient method for computing the similarity of folk music melodies. Accordingly, this dissertation as a

part of the project tries to study the available methods of music segmentation and similarity detection in musical representation of folk music and build upon that literature to propose an improved method. After studying all the available models and approaches, specific models have been picked to work on and improve it for our dataset. Thus, it is necessary to divide the overall objective of this work into three main sections.

- **Similarity and segmentation:** Study and Review the available methods and present a more detailed comparison of the previously introduced algorithms
- **Implementation:** Focus on specific models and apply them to our selected features and assess the results
- **Optimization and evaluation:** applying the optimization algorithm on previously selected models and evaluating the corresponding results compared to baseline models

1.4 Methodology

The approach we have used in this work, considering the first objective, after studying all the common methods in the music segmentation, is to choose four different models, tailor and optimize them to our needs, and compare the results on our Iberian dataset I-Folk (Carvalho et al., 2021). Regarding the second objective, the chosen model is the local Boundary Detection model (LBDM) (Cambouropoulos, 2001) Variable Markov Oracle (VMO) (C. Wang & Mysore, 2016), Entropy and information of features (Pearce et al., 2010), and Random Forest (van Kranenburg, 2020). Firstly, we calculate a series of features using VMO and LBDM and their respective entropy and try to see the performance of finding the musical segments with their novelty curve. Secondly, as the third objective dictates, we perform the optimization on our features using a genetic algorithm (Goldberg, 1989) to make sure they are well-tailored to our dataset and finally, we use the same features to train a random forest and compare all the models results with the second objective results.

1.5 Workplan

As the context and boundaries of the work have been decided, a Gantt chart is also created to guide us throughout time. The overall timing of the work decided to be seven months since we started the research and study of available algorithms in music segmentation in August. The Coding and implementation started after we gained an overall insight and clearer picture. Various methods and algorithms were picked and implemented which later on became the foundation for further improvement and optimization. While the code was being finalized and tried on multiple datasets, we started to work on writing the rest of the dissertation namely the literature review and methodology. Following the literature review presentation in February, the previous chapters have been refined and some more models including random forest have been added. Furthermore,

the results and evaluation section of this document came into the picture, and the full dissertation was shaped.

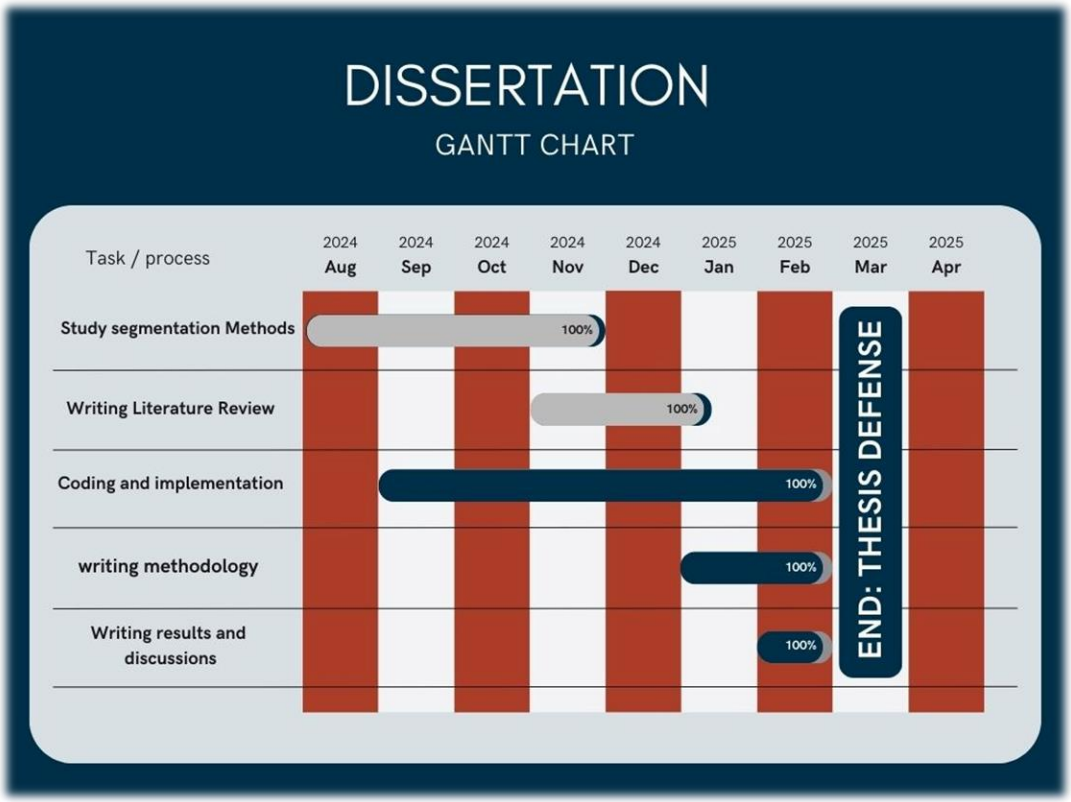

Figure 1: Dissertation Gantt chart

1.6 Dissertation structure

In addition to this chapter, this dissertation has three more chapters. In Chapter 2, we will discuss the literature review and all three main categories of symbolic music segmentation: Ruel-based, Supervised, and unsupervised. Afterward, we continue with the common evaluation techniques like F-measures and the drawbacks of a high recall rate. Following Chapter 2, in Chapter 3, we lay the detailed theoretical foundation of our work and the models that are used introducing LBDM, VMO, and the structure of the genetic algorithm used for optimization. Furthermore, we discuss the peak-picking techniques and the structure of random forests. Finally,

in Chapter 4, we conclude our work with the result and insight achieved by implementing these techniques on our dataset. We first study the structure of data and then try to evaluate the results for each model and compare them together concerning their nature.

2. Related works

In this chapter, we will discuss the literature that was the basis and inspiration for this work and research. Firstly, we discuss music segmentation techniques with music theory approaches such as LBDM or machine learning approaches like random forest and temporal prediction error (Bassan et al., 2022). Later, we will proceed with evaluation metrics commonly used in segmentation tasks. Finally, since the importance of the dataset in all the studied works we review the commonly-used dataset in the music segmentation field and introduction of our dataset: I-Folk.

2.1 Music Segmentation

Music segmentation in symbolic notation is a critical task in computational musicology, aiming to divide a piece into meaningful segments such as beats, measures, phrases, motifs, or sections. This segmentation enhances various applications, from music retrieval to performance analysis. Over time, three main algorithmic approaches have emerged: rule-based, supervised, and unsupervised.

Rule-based methods are based on Gestalt psychology (Tenney & Polansky, 1980) and rely on predefined musical theories and heuristics, incorporating knowledge of harmony, rhythm, and structure to detect segment boundaries. Supervised approaches, on the other hand, utilize labeled data to train machine learning models that can generalize across different compositions and styles. Lastly, unsupervised methods attempt to discern patterns directly from the data without relying on annotations, often employing clustering techniques to reveal inherent musical structure. Each approach has its strengths and challenges, and a thorough understanding of these methods is essential to advancing music segmentation technologies. Finally, a comparative table is constructed to summarize the available literature for better understanding.

2.1.1 Rule-Based models

One of the fundamental works in systematizing the process of melodic segmentation is the work of Lerdahl & Jackendoff (1983) in a generative theory of Tonal Music (**GTTM**). GTTM drives from the rules defined by Gestalt psychology (Tenney & Polansky, 1980). They came up with a hierarchy of rules called Grouping Preference Rules (GPR) that governs the possibility of segmenting by an idealized listener. These fundamental rules which later shaped the foundation of rule-based music segmentation are 1. temporal proximity: GPR 2a 2. inter-onset-interval: GPR2b 3. change in register: GPR 3a 4. Dynamics: GPR 3b 5. Articulation: GPR 3c 6. Length: GPR 3d 7. Symmetry: GPR 5 and 8. motivic similarity: GPR 6. The validity of GPRs was largely supported in listening experiments (Cambouropoulos, 2001; Cenkerová, 2017; Deliege, 1987; Frankland & Cohen, 2004; Friberg et al., 1998). In addition, Frankland and Cohen (2004) used the effectiveness of these rules and proposed **quantified GPRs** 2a,2b,3a, and 3d as a standalone segmentation algorithm. However, according to Meredith (2016), even though GTTM systematizes the cognitive process of segmentation very beautifully, but also brings a lot of implementation issues with it. Meredith states the main advantage of GTTM is that it can acquire tree structures while its ambiguous definition of rules, cases of confliction among different rules, no clear explanation of feedback link, and lack of good proposed algorithm, left the main debate among computer scientists open and that why he works on proposing a series of detailed practical methods of using these rules in his books creating a more practical approach called time-span tree analyzer by a hierarchical system among different rules.

GPR	Description	n	Boundary Strength
2a	Rest		absolute length of rest (semibreve = 1.0)
2b	Attack-point	length	$\begin{cases} 1.0 - \frac{n_1+n_3}{2\times n_2} & \text{if } n_2 > n_3 \wedge n_2 > n_1 \\ \perp & \text{otherwise} \end{cases}$
3a	Register change	pitch height	$\begin{cases} 1.0 - \frac{\lvert n_1-n_2\rvert+\lvert n_3-n_4\rvert}{2\times\lvert n_2-n_3\rvert} & \text{if } n_2 \neq n_3 \wedge \lvert n_2-n_3\rvert > \lvert n_1-n_2\rvert \wedge \lvert n_2-n_3\rvert > \lvert n_3-n_4\rvert \\ \perp & \text{otherwise} \end{cases}$
3d	Length change	length	$1.0 - \begin{cases} n_1/n_3 & \text{if } n_3 \geq n_1 \\ n_3/n_1 & \text{if } n_3 < n_1 \end{cases}$

Figure 2: quantified GPR

In parallel with GTTM, there is another music cognitive theory which is proposed by Narmour (1990, 1992) known as the **Implication-realization** (IR) theory of music cognition. In contrast with GTTM, IR theory emphasizes dynamic processes rather than processing the entire

piece statically. This theory also uses gestalt principles of proximity, similarity, and continuation (Pearce et al., 2010).

The Local Boundary Detection Model (**LBDM**) introduced by Cambouropoulos (Cambouropoulos, 2001) operates on local changes in pitch, IOIs, and rests. The strength of the boundaries in each segment is based on the degree of the change between two intervals. Moreover, Cenkerová (2017) used the LBDM in MATLAB toolbox developed by Eerola and Toiviainen (2004) with a genetic algorithm to optimize the parameter of LBDM for the Essen folk database and reached a better result compared to the previous method.

The **Grouper Program** proposed by Temperley (2004) relies only on temporal information (note-to-note intervals and meter) and analyses the score as a whole, performing all possible analyses and selecting the favorites using three criteria (Pearce et al., 2010): 1. rule one (PSPR1): Sum of IOIs and OOI divided by the mean of IOI of all previous notes 2. rule two (PSPR2): a system of penalizing predicted phrase length and optimum length set by default to 8 optimized for the dataset EFSC (Temperley, 2004) and 3. rule three (PSPR3): preferring to choose successive groups at parallel points in the metrical hierarchy. The beats in Temperley's work are preliminarily analyzed by the **Meter Program**[1] which uses the division of time points into small units called "pip". The Grouper method was examined by Höthker, Thom, and Spevak (2002) and reached an F-measure of .62, and Pearce (2010) also achieved an average of 0.66. Most recently Cenkerová (2017) tried using Grouper with and without the Meter program to measure the difference between using the Meter Program to shed more light on the influence of the Meter algorithm on the performance.

One of the repeated references that were used in the comparison work of Bassan (2022) and Cenkerova (2017) was the use of GPR2a from GTTM as a standalone model named **Pause,** which is very loosely detecting boundary at each pause, irrespective of its length. He also pointed out that the effect of long inter-onset intervals (IOI) and rests (GPR2a and 2b) was stronger than pitch changes in listening experiments (Deliege, 1987; Frankland & Cohen, 2004; Peretz, 1989). Similar to the pause model is the **Meter Finder** by Toiviainen (2006) which uses autocorrelation to classify melodies into double or triple and correctly classifies them with 90% accuracy using the Essen corpus.

Cenkerová and collaborators (2017) provide a good overview and comparison of the rule-based methods but also propose a novel model of using IOI differences (**ΔIOI**) rather than IOI itself. Wherever a negative difference is detected, a boundary is placed. Additionally, since the negative IOIs are expected to occur at the phrase end, they considered only the lowest value of difference in each song. The idea of this approach comes from using one of the rules mentioned in the rule-based model proposed by Boroda (1982). Cenkerová (2017) furthermore combines all ΔIOI, meter finder, and pause altogether. This model assumes a boundary at any given location that satisfies any of these three models considering IOI, Rests, and metrical information. She

[1] https://www.link.cs.cmu.edu/melisma

finally implements a meta-classifier with all the discussed models to see which combination of all these models would perform better (**Compound model BIC**) which can be categorized as one of the unsupervised methods (Cenkerová, 2017).

2.1.2 Supervised Models

In this approach, a symbolic musical piece is considered as a string of events, and finding these possible groups of events is approached as a data science issue that relies on computational algorithms to detect patterns and identify the segments. As with most machine learning problems, feature engineering, selection, and processing is the point of difference in various works. That being said, the approach in Kranenburg's work (2020) largely is a feature engineering exercise. Kranenburg takes inspiration from what features may contribute to the establishment of a phrase boundary. Then they apply two machine learning algorithms: **RIPPER** (Cohen, 1995) and **Random Forest** (Breiman, 2001a). Since they do not take a prior theoretical basis, such as the gestalt principles, the learning algorithm explores which features are of value and in what combination.

One of the older works in this field is the work of Rens Bod (Bod, 2002) and his similar unsupervised counterpart Brent (1999) which originate from models in computational linguistics and rely on statistical information. Bod (2002) believes that music perception may be much more memory-based than previously assumed and thus he tries three memory-based parsing models: 1. treebank grammar learning techniques (Charniak, 1996) 2. Markov grammar (Collins, 1999; Seneff, 1992) and 3. Markov grammar with data-oriented parsing (**DOP**) (Bod, 1998). Bod (2002) found out that the best result (85.9% out of a set of 1000 folk pieces) will be achieved by combining Markov Grammer of colins and data-oriented parsing of Bod.

One of the more recent works due to the increasing application of neural networks and deep learning in data processing was pursued by Zhang & Xia (2021) and Guan et al. (2018) who use these algorithms in music segmentation. Zhang & Xia's work specifies that the main issue of applying **Deep learning** methods to phrase segmentation is the sparse labeling of training datasets and as a result, they propose two label engineering techniques to address the issue and they manage to prove that their method mixed with neural networks is better than the well-known approaches and any other neural networks that has been tried.

Similar to this work and based on natural language processing is the work of Hirai and Sawada (2019), which proposes **Melody2Vec** that uses the same concept of Word2Vec (Mikolov et al., 2013) to detect the similarity and segments and then he tries to offer a melodic replacement that sounds natural to the human ear.

2.1.3 Unsupervised Models

Unsupervised models for melodic phrase segmentation usually treat unannotated data and attempt to come up with strategies for detecting and separating the segments of a musical piece. One of the unsupervised models based on n-gram models (Manning & Schutze, 1999) is commonly used in statistical language modeling proposed by Pearce (2010) which uses conditional probability, information content, and entropy to detect the level of unexpectedness. The logic of this method which is called **IDyOM** is based on Narmour's (1990, 1992) theory. Pearce's method assumes that boundaries occur at the high values of unexpectedness and uncertainty of predictions. In addition to this method, Pearce constructed a logistic regression model that included Grouper, LBDM, IDyOM, and GPR2a (Pause) as predictors. Using a Bayes Information Criterion, he modifies his model to **Hybrid IDyOM** (Pearce, 2008) which achieves a better F1 score on significantly more melodies than the original IDyOM.

Latter et al (2015) developed a method based on the **restricted Boltzmann machine (RBM)** in the Essen folk data set and a subset of it called Erk. His algorithm processed melodies as n-grams and generated the conditional probability of a note given its previous predecessors.

Another good example of using neural networks in segmentation is the work of Lazzari et al (2023) which passed on the natural language processing algorithms and used harmonic information extracted from symbolic notations. He first introduces a novel **Pitchclass2Vec** chord embedding method, then he uses that with a recurrent LSTM neural network on a corpus of musical chords and compares the result with previous similar methods like FORM (De Haas et al., 2013).

Apart from his detailed comparison of all previous methods which was a very inspiring work in this thesis, Bassan et al (2022) use of a **temporal prediction error** algorithm with **ensemble learning** in music segmentation, performed very well in comparison to other methods. Table 1 shows an overview of previous methods presenting a summary of what we have discussed.

Table 1: Music segmentation models

Model	Author	Model type	Method	Performance metrics	Dataset used	Code available
LBDM for Essen data set	Eerola and Toiviainen (Eerola & Toiviainen, 2004)	Rule-based	Local changes in pitch, IOI, and rest, used with different parameters than the original one	F-score, R-value, Precision	Different Subset of Essen collection (Höthker et al., n.d.; M. T. Pearce et al., 2010)	MIDI MATLAB Toolbox(Eerola & Toiviainen, 2004)
ΔIOI	Zuzana Cenkerová (Cenkerová, 2017)	Rule-based	Difference between successive IOI	F-score, R-value, Precision	Essen folk song dataset(Helmut, 1995)	Nothing found
Grouper and groper with Meter	Zuzana Cenkerová David Temperley (Cenkerová, 2017; Temperley, 2004)	Rule-based	Grouper 3 main rules with/without 0ne of the Rules	F-score, R-value, Precision	Essen folk song dataset(Helmut, 1995)	https://www.link.cs.cmu.edu/melisma
Meter	Zuzana Cenkerová (Cenkerová, 2017)	Rule-based	Hierarchy of beats on a 0-5 scale	F-score, R-value, Precision	Essen folk song dataset(Helmut, 1995)	https://www.link.cs.cmu.edu/melisma
ΔIOI OR Meter Finder OR Pause	Zuzana Cenkerová (Cenkerová, 2017)	Rule-based	IOI, rest, and metrical info all together	F-score, R-value, Precision	Essen folk song dataset(Helmut, 1995)	Nothing found
Meter Finder	P. Toiviainen and T. Eerola(Toiviainen & Eerola, 2006)	Rule-based	used autocorrelation to classify melodies into "double meter" or "triple meter"	F-score, R-value, Precision	Tested on Essen folk(Helmut, 1995) song dataset (Cenkerová, 2017)	MIR MATLAB toolbox
Pause	Zuzana Cenkerová (Cenkerová, 2017)	Rule-based	GPR2a	F-score, R-value, Precision	Essen folk song dataset (Helmut, 1995)	Nothing found
Ensemble Temporal Prediction Errors	Shahaf Bassan (Bassan et al., 2022)	Unsupervised	temporal prediction error with ensemble learning	Precision, Recall, and F-score	Essen folk song dataset(Helmut, 1995)	Nothing found
ΔIOI Compound model(PIC)	Zuzana Cenkerová (Cenkerová, 2017)	Unsupervised	logistic regression to make a meta-classifier with all the rule-based models from the previous analysis	Precision, Recall, and F-score	Essen folk song dataset(Helmut, 1995)	Nothing found
IDyOM	Marcus T. Pearce (M. T. Pearce et al., 2010)	Unsupervised	Using information content and entropy	Precision, Recall, and F-score	Subset of the EFSC(Helmut, 1995), database Erk, containing 1705 Germanic folk melodies	https://www.marcus-pearce.com/idyom/

Restricted Boltzmann Machines (RBM)	Stefan Lattner (Lattner et al., 2015)	unsupervised	probabilistic segmentation method, based on Restricted Boltzmann Machines (RBM)	Precision, Recall, and F-score	Essen folk song dataset (Helmut, 1995) and Subset of Erk similar to IDyOM	Nothing found
Hybrid IDyOM	Marcus T. Pearce (M. T. Pearce et al., 2010)	Unsupervised	constructed a logistic regression model including Grouper, LBDM, IDyOM and GPR2a	Precision, Recall, and F- score	Essen folk song dataset (EFSC)	https://github.com/mtpearce/idyom-tutorial
Pitchclass2vec	Nicolas Lazzari (Lazzari et al., 2023)	UnSupervised (Neural Network)	Natural language processing with an LSTM neural network	Precision, Recall, *F*1 with under-segmentation,over-segmentation, and normalized cross entropy F1	ChoCo(de Berardinis et al., 2023) , a dataset of chord annotations and Billboard(Burgoyne et al., 2011), a dataset of structurally annotated tracks	no official code release yet
Ripper and Random Forest	Peter van Kranenburg (Janssen et al., 2014)	Supervised	rule-mining algorithm RIPPER as well as a Random Forest classifier on several subsets of features.	Precision, Recall, and F-score	EFSC(Helmut, 1995), Merteens tune collection (*Dutch Song Database*, n.d.) and Bach Chorales(*The Humdrum Toolkit for Computational Music Analysis | Humdrum*, n.d.) (just soprano)	https://github.com/pvankranenburg/ismir2020
Bi-LSTM-CNNs and CNN-CRFs	Yixiao Zhang, Yixing Guan (Guan et al., 2018; Zhang & Xia, 2021)	Supervised	Neural Network with Conditional Random Field	Precision, Recall, and F-score	Essen Folksong Collection (Helmut, 1995), POP909, 1000 well-known Chinese pop pieces by Yixing Guan	https://github.com/ldzhangyx/music-melody-segmentation-using-neural-CRF
Data-Oriented Parsing technique (DOP)	Rens Bod(Bod, 2002)	Supervised	the Markov grammar technique with the Data-Oriented Parsing technique(DOP)	Precision, Recall, and F-score	1000 folk pieces from the Essen (Helmut, 1995)	Nothing found
Melody2vec	Tatsunori Hirai(Hirai & Sawada, 2019)	Supervised (Neural Network)	Natural language processing.Word2vec for melodies (Unsupervised)	Similarity and accuracy	Lakh MIDI dataset (*The Lakh MIDI Dataset v0.1*, n.d.)	https://github.com/TatsuoriHirai/Melody2vec

2.2 Performance Evaluation

When it comes to the evaluation of the segmentation algorithms, there are a series of common methods that are used by most of the available works in the field, namely Precision(P), Recall(R), and F1-score (Bod, 2002; Cenkerová, 2017; Kreuk et al., 2020; Michel et al., 2017). Räsänen et al (2009) give a very detailed overview of all the methods used to Evaluate general segmentation algorithms:

$HR = \frac{N_{hit}}{N_{ref}} * 100 \quad (1)$	$OS = (\frac{N_f}{N_{ref}} - 1) * 100 \quad (2)$
$PRC = \frac{N_{hit}}{N_f} \quad (3)$	$RCL = \frac{N_{hit}}{N_{ref}} \quad (4)$
$F = \frac{2.0 * PRC * RCL}{PRC + RCL} \quad (5)$	

Figure 3: Five common evaluation metrics according to (Janssen et al., 2014): Nhit: number of boundaries correctly detected Nf: total detected boundaries and Nref: number of boundaries in the reference

Precision (equation 3 in Figure 3) describes the likelihood of how often the algorithm identifies a correct boundary. HR or hit rate (equation 1 in Figure 3) is Recall (equation 4 in Figure 3) in percent. Describing the performance of an algorithm with one scalar value brings us to the definition of F-value (equation 5 in Figure 3) which can be computed from precision and recall (Ajmera et al., 2004). False-positives rates and miss rates are also sometimes used and can be calculated based on these parameters (Esposito & Aversano, 2005). Räsänen (2009) also specifies the ambiguity in all these traditional factors and models and how one can easily increase the hit rate by over-segmentation. If we have an algorithm that produces a segment each second, we will probably reach the recall of 100 percent while this doesn't mean the algorithm segments the piece reliably unless we can tolerate a high number of segments that are irrelevant or we find a way to disregard them afterward. Räsänen implements a random segmentation algorithm and shows how a stochastic process can produce a very convenient Hit rate by just over-segmentation and that's why he offers his new performance metric called R-value which is also sensitive to over-segmentation compared to the F1 factor (Räsänen et al., 2009).

$$r_1 = \sqrt{(100 - HR)^2 + (OS)^2} \quad (6)$$

$$r_2 = \frac{-OS + HR - 100}{\sqrt{2}} \quad (7)$$

$$R = 1 - \frac{abs(r_1) + abs(r_2)}{200} \quad (8)$$

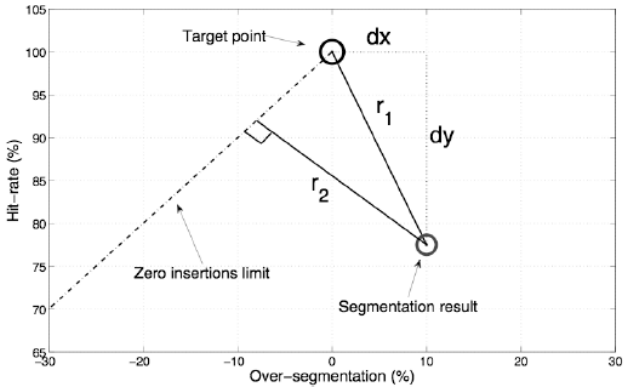

Figure 4: R-value definition from (Räsänen et al., 2009)

Moreover, In the previously mentioned work of Lazzari (Lazzari, 2023), we can see the usage of more performance measurement techniques for over-segmentation (S_o), under-segmentation (S_U), and Normalized Cros entropy (S_{F1}) using the MIR evaluation library mir_eval (Raffel et al., 2014). These methods are discussed in detail by Lukashevich (2008) According to the author when a method has high over-segmentation, we properly face a false segmentation and, on the contrary, when we have a high under-segmentation factor, it means ground-truth segments in our prediction are merged. Lukashevich also indicated that the common methods do not take into account the effect of over and under-segmentation and she uses a series of concepts from a field like speaker clustering and Image segmentation to use entropy to introduce a series of new factors and examine them to see which one proved a better insight toward the quality of segmentation. She also states that the higher the states of segmentations are, the better these factors describe the quality of the segmentation, and the less they are, it is more probable for a random segmentation to achieve high values.

Finally, As the last section of the literature review, we will provide you with previously introduced studies of Shahaf (2022) and Cenkerová (2017, 2018) which also evaluated all the most commonly available methods and compared their Precision, Recall, F-value, and R-value with Essen Folk dataset.

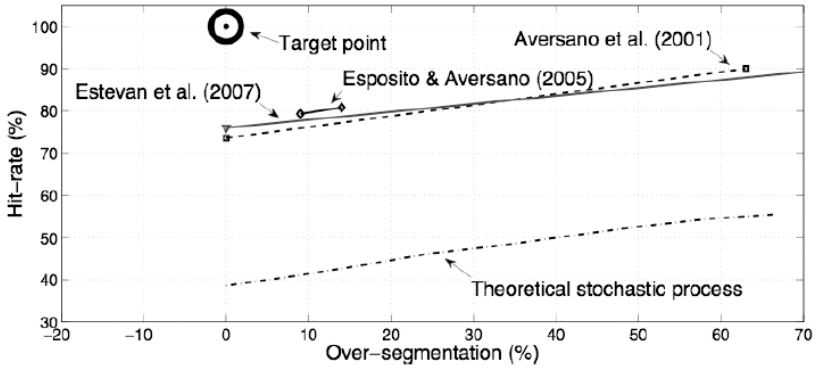

Figure 5: The effect of over-segmentation on hit rate (Räsänen et al., 2009)

	Model	Percision	Recall	F-Measure	R-Value
Rule-based models	ΔIOI	79	54	58	67
	Pause	98	48	60	63
	Meter	59	70	61	65
	Meter Finder	70	64	64	72
	LBDM	81	60	65	71
	Grouper	77	73	74	78
	ΔIOI, Meter finder, Pause	64	81	68	63
Unsupervised	IdyOM	76	50	58	64
	Restricted Boltzmann Machines	83	50	60	64
	Hybrid IdyOM	87	56	66	69
	ΔIOI Compound(BIC)	92	68	75	77
	Temporal Prediction Error	77	81	77	82
Supervised	Ripper	78	63	69	73
	Random Forest	83	69	76	77
	DOP-Markov	77	86	81	81
	CNN-CRF	_	_	82	_
	BI-LSTM-CRF	_	_	84	_

Table 2: Performance summary of different models (Bassan et al., 2022)

2.3 Dataset

The choice of dataset and how well it has been annotated and organized can affect the overall performance of a method. In this section, we will discuss the commonly used datasets in folk music.

2.3.1 Irish Traditional Music Archive (ITMA)

Irish Traditional Music Archive (ITMA) [2] is a publicly accessible digital archive that documents, preserves, and provides access to Irish traditional music and dance (Harkin, 2022). It contains a vast collection of music recordings, transcriptions, and other resources, with a focus on fostering both research and practice within Irish music traditions. ITMA was established in 1987 by Dr. Nicholas Carolan, who was instrumental in building the archive. ITMA is now run by a team of musicologists, ethnomusicologists, and archivists with support from the Irish Arts Council. This dataset holds over 40,000 audio recordings and 15,000 transcriptions of Irish traditional music, making it one of the largest collections of its kind. The dataset covers a variety of forms like ballads, jigs, reels, airs, and hornpipes, making it rich for stylistic analysis. The format of the data is mainly audio recordings, alongside symbolic (ABC or MIDI) transcription. The annotated phrase boundaries for selected tracks have rich metadata including song title, origin, performance style, and historical context.

2.3.2 RWC Folk Song Database

The RWC (Real World Computing) Music Database [3] is a copyright-cleared music database that is available to researchers as a common foundation for research (Goto et al., 2003). It was built by the RWC Music Database Sub-Working Group of the Real-World Computing Partnership (RWCP) of Japan. The RWC Music Database contains subsets for different musical genres with a maximum of 100 pieces, and the folk music subset has pieces segmented into phrases. This dataset uses WAV and MIDI formats for the pieces and TXT for segmentation. Moreover, they included the metadata with song title, region, and artist in all their songs.

[2] https://www.itma.ie/ [Access data: 01.03.25]
[3] https://staff.aist.go.jp/m.goto/RWC-MDB/ [Access data: 10.03.25]

2.3.3 Turkish Makam Music Dataset (TMD)

The Turkish Makam Music Dataset (Karaosmanoğlu et al., 2014) is a collection of symbolic and audio data focusing on traditional Turkish makam music. Developed by researchers from the Music Information Retrieval (MIR) Group at Bogazici University in collaboration with the Turkish Music State Conservatory. Dr. Baris Bozkurt and his colleagues have been primary contributors to this dataset. It contains over 500 symbolic pieces and several hours of audio data. Each piece is tagged with information about the makam, which defines the scale and modal structure, and saved in both audio (WAV/MP3) and symbolic (MIDI, ABC) formats. Their metadata includes information on the performer, the composer, makam, usul (rhythmic cycle), and recording details.

2.3.4 The Essen Folk Song Collection (EFSC)

The Essen folk song collection (Helmut, 1995) is a set of 6,236 mostly Germanic folk music in symbolic format, with phrases annotated by music experts. This most used collection is arranged as a series of sections according to geographical region -- beginning with one of four continent designations (Africa, America, Asia, Europe) followed by the country or region name. Features that are available in this dataset are as follows:

- Melodic contour: The pieces are transcribed in symbolic formats (like Kern), representing the pitch and rhythm of each note.
- Duration and rhythm: Notes are represented with rhythmic values (whole notes, half notes, etc.).
- Pitch class and scale: The pitch information is encoded in numerical values representing pitches (e.g., MIDI numbers), along with key and mode information (e.g., major, minor, Dorian, etc.).
- Metadata: Basic information like title, origin, and source (often where or by whom the song was collected) is also included.
- Folk motifs: Many music scores are indexed by traditional folk motifs or melodic formulas used in folk music studies.

2.3.5 Meertens's Dutch song database

The Dutch Song Database is a digital repository documenting Dutch song culture throughout the ages. The database was initiated in the early 1990s by the Dutch musicologist Louis P. Grijp (1954–2016), who continued to lead the development of the database until 2015. During these

years, many research and documentation projects have been carried out, and gradually an enormous amount of high-quality data has been collected (van Kranenburg et al., 2019).

At the moment, the database contains metadata on 173 thousand occurrences of Dutch songs in a variety of sources dating from the twelfth century up to the present day. A large number of song texts and melodies have been digitized and are currently accessible within the collection. The database documents many kinds of (folk) music, including love songs, satirical songs, beggar songs, psalms, other religious songs, children's songs, St. Nicholas songs, and Christmas songs. The main sources in which these songs were found are songbooks, song sheets (broadsides), song manuscripts, and fieldwork recordings. The cataloged record for each song contains information about the source in which the text and/or the melody occurs. In most cases, direct links are provided to the complete song text, to a scan of the source, to the notated sheet music, or to a recording of an individual song.

The Dutch Song Database is hosted by the Meertens Institute of the Royal Netherlands Academy of Arts and Sciences (KNAW) in Amsterdam and is maintained and developed further by the Oral Culture Group of the institute, in cooperation with several partners. Each song entry typically includes metadata such as the first line, first line of the refrain, songwriter, and source information. For many entries, full song texts, music notations, and audio recordings are available. Additionally, the database provides links to related songs sharing the same melody or stanza form, facilitating comprehensive research and exploration. The most recent version of this database has a collection of 360 songs from Dutch folklore in various formats, including MIDI and the already mentioned **kern format.

2.3.6 I-Folk

The data set used for this work will be the I-Folk (Carvalho et al., 2021) which consists of a set of more than 500 Iberian folk melodies encoded in MIDI symbolic representation and including metadata. The information present in this database is such as:

- Title of the Song (if missing, the first line of the song’s lyrics (i.e., incipit) is used instead);
- Author of the Song (if missing it is annotated "Anonymous");
- Title and subtitle of source;
- Compiler and copier of source, if existent;
- Date of compilation;
- Geographic Information: country, local, and district from which the song is characteristic;
- Genre;
- Meter;
- Tempo;
- Key and mode of the song;
- Time signature(s);
- Pitch range;

- Rhythmic pattern;
- Phrases types (start and endnotes and cadence type).

Despite all this information made available by the dataset, the most critical aspect to consider will be the actual melody of each song, which will be parsed by Tonal Interval Space (TIS) (Bernardes et al., 2016).

3. Enhancing LBDM Segmentation: Methods and Foundations

In this work, we focus on the accuracy enhancement of the rule-based method behind the LBDM (Cambouropoulos, 2001), notably by exposing the different traditions and assessing the importance of each feature on a traditional and individual piece basis. Although LBDM shows an overall good result in segmentation and has a straightforward algorithm, Cambouropoulos (2001) states the performance can vary depending on different pieces and annotation methods since it doesn't take into account any similarity detection algorithm or dynamic characteristics of a piece. What Cambouropoulos (2001) additionally points out because LBDM can be applied to a variety of musical features of a piece, is the probability of contradiction among some of these features in detecting the candidates of the segment. For instance, a good example of this can be "slur" in a performance of a piece that might be different due to the taste of the conductor and performer and create a different feel of segments to a musical ear and might contradict if we just follow the rhythm or pitch changes. As stated, since LBDM is not sensitive to repetition, one of the approaches for enhancement is to add a mechanism to take the repetition and similarity into consideration. We use Variable Markov Oracle (VMO) to take into account the repetition of segments in each tradition. Then the enhanced tradition-specific features will be weighted by two methods: entropy and optimized weights using a genetic algorithm. The first is applied on an individual piece basis, assuming that the structure of each piece weights differently the mechanics to produce salient phrase segmentation points and the second assumes that those mechanics are somehow shared among traditions. Secondly, we also used the same optimization to come up with a tradition-specific peak-picking algorithm and compare the results for each stage with the original LBDM and previous works.

This chapter will detail all the foundational knowledge used in our work and the novelty introduced by our method. We start by describing the baseline LBDM algorithm and the VMO as the basis of our feature selection and segmentation method. Furthermore, we propose two strategies for weighting features including the entropy method and optimization via a genetic algorithm. Finally, we discuss the peak-picking phase of the method and explain the rationale behind choosing a specific algorithm and a tradition-based optimization.

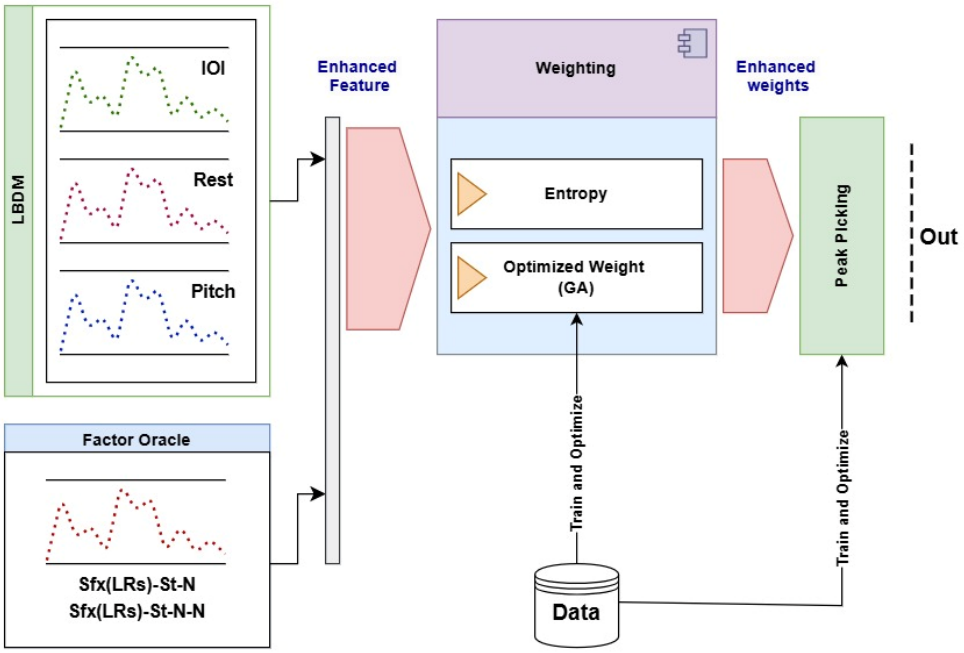

Figure 6: Diagram of our method

3.1 Feature enhancement

Tradition-based LBDM is the core of our work but Since the LBDM is insensitive to repetition, we decided to use VMO in the repetition detection of string of symbols. This section will shed light upon these two algorithms in detail and how we create an enhanced version of LBDM features that is region-specific and repetition-sensitive.

3.1.1 LBDM

The LBDM is a segmentation method that uses a novelty curve where peaks indicate segment boundary candidates. The model first calculates the degree of change of three specific features: pitch, inter-onset interval (IOI), and rests, from which the strength of the segment boundary candidates is computed. Prior to the computation of segments, by detecting peaks in the novelty function, the features' individual functions are combined with possible weighting. Let's presume P_k is a parametric profile as a sequence of n intervals of size x_i.

$$P_k = [x_1, x_2, \dots, x_n] \qquad where: k \in \{pitch, IOI, rest\}\ and\ i \in \{1,2, \dots, n\} \tag{2.1}$$

Then the degree of change r between two successive intervals of x_i and x_{i+1} can be calculated as follows:

$$\begin{cases} r_{i,i+1} = \dfrac{|x_i - x_{i+1}|}{x_i + x_{i+1}} & if\ x_i + x_{i+1} \neq 0\ and\ x_i, x_{i+1} \geq 0 \\ r_{i,i+1} = 0 & if\ x_i = x_{i+1} = 0 \end{cases} \tag{2.2}$$

The strength of the boundary S_i for the interval x_i is affected by both degrees of the change to the preceding and following intervals.

$$S_i = x_i . \left(r_{i-1,i} + r_{i,i+1}\right) \tag{2.3}$$

Following the strength sequence for each feature, we apply the specific weights to each feature to achieve the final novelty curve and, respectively, the peaks of the curve are the candidates for segment boundaries. The weights that are recommended by the original LBDM algorithm and tried by various researchers on the Essen Dataset are 0.25, 0.5, and 0.25 for the pitch, IOI, and rest across the piece. Our work questions these fixed weights per piece and tradition, aiming to improve this method for Portuguese and Spanish traditions. We add a repetition-sensitive algorithm based on VMO which will be discussed in the next section to these three LBDM features. Consequently, the mixture of new features with tradition-specific and piece-specific weights and proper peak-picking will improve the performance of the method. The peak-picking threshold and averaging system for detecting the maximum of the novelty curve will be discussed further.

3.1.2 Variable Markov Oracle (VMO)

The Variable Markov Oracle (VMO) (Wang & Dubnov, 2015) is a finite state automation that can efficiently search for factors and substrings in a body of symbols which is also used for audio similarity detection based on structural repetition and variation (Wang & Mysore, 2016). It builds on concepts from Markov processes and factor oracles, which are often used in machine learning and computational musicology to model music structure, predict patterns, and identify recurring motifs in a sequence. Similar to the Markov model which represents a sequence where the probability of transitioning from one state to another depends only on the current state (the "memoryless" property), VMO used suffix links to the previously repeated state to keep track of repetition and subsequently the pattern in a piece. This method can be applied to symbolic and

audio representation (Wang & Dubnov, 2015; Wang & Mysore, 2016) of a music signal known as Factor Oracle(FO) (Allauzen et al., 1999) and Audio Oracle(AO) however since our focus is the symbolic representation, we mainly refer to Factor Oracle while discussing the algorithm but it is still worth mentioning that even in the symbolic representation and specifically in our algorithm there is a slight difference between VMO and FO:

Factor Oracle (FO): A factor oracle is a compact automaton used to identify repeated patterns or sub-sequences in a string (in this case, a musical sequence). It's designed to efficiently find all repeated subsequences and offer a representation of how these repeats are structured.

Variable Markov Oracle (VMO): The VMO is an extension of these ideas, combining the strengths of Markov processes and factor oracles to segment music based on repeating patterns while accounting for variations. Essentially, it identifies "similarities" in a musical sequence rather than exact repetitions.

The VMO builds a graph-like structure where each node represents a segment of the music sequence (such as a note or a chord), and edges between nodes represent potential transitions (either identical or similar segments). The process involves two main stages: First, the sequence is processed to identify exact repetitions using the Factor Oracle. The FO identifies all possible repeated patterns (e.g., identical motifs or themes) and constructs a structure that reflects these repetitions. The VMO then goes beyond exact repetition and tries to capture variations. This is important because, in music, motifs often repeat with small modifications (e.g., in pitch, rhythm, or instrumentation). The VMO handles this by linking nodes (segments) that are not the same but "similar enough" based on some defined distance metric. Once the oracle is constructed, the next step is to segment the sequence. This is done by identifying clusters of similar patterns that represent structural sections in the music. These clusters form the boundaries of segments, allowing the VMO to break the piece down into meaningful sections.

Let’s consider a simple melody like: C-D-E-C-D-E-F-G. The VMO would identify that the sequence C-D-E repeats exactly twice (once at the beginning and once in the middle). While in VMO, in case of a slight variation, the algorithm would recognize this as a variation rather than a completely new segment. For instance, if the second repeat was C-D-F instead of C-D-E, the algorithm would group these sections as similar, marking a segment boundary.

The two major elements that the VMO algorithm uses to trace the repetition during a piece or any sequence of symbols are the suffix link and the longest repeated suffix (LRS). Suffix links in VMO refer to the previous state in a sequence or series of symbols was occurred While LRS refers to the length of the longest sequence that is repeated before. Thus, with the VMO algorithm, we have access to the length of the previous longest repetition and also their location which will help us to track and find segments throughout the piece or any sequence of symbols.

For instance, in Figure 6, if we have a sequence of “ABCDEFAB” the suffix link for the sequence “AB” that occurred at the end of the string will refer to the first B in the string, and LRS for the last B would also be equal to 2 due to the fact that VMO algorithm marks the end of the

sequence or symbol as its reference and this gives us the motivation of defining more features to be able to find segment more clearly throughout the piece.

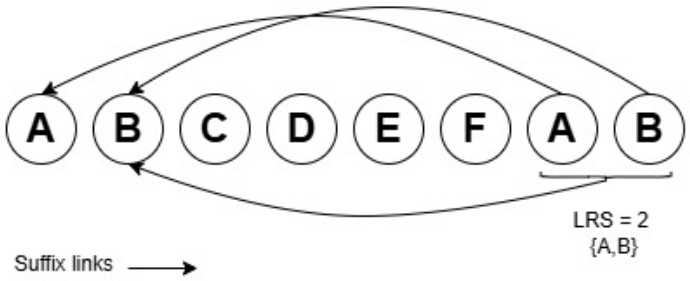

Figure 7: LRS and Suffix link in VMO

In this work, we use the syvmo[4] (Carvalho & Bernardes, 2021) based on the work of Wang & Mysore (2016). Using this package, we calculate the VMO for each piece and have access to the suffix links (Sfx) which refers to the previous repeated symbol or state, and the Longest repeated Suffix (LRS) which indicates the size of the longest repeated sequence throughout the piece. Additionally, based on the code written by Carvalho & Bernardes (2021), we have introduced some novel additional features such as a derivative of the LRS to be able to detect the strength of the LRS changes throughout the piece. Here are the details of the features we take into consideration in regards to VMO:

1. Sfx: Suffix links for each note directly from VMO
2. LRS: Longest repeated suffix directly from VMO
3. Deriv (LRS): Derivative of Current notes LRS and the next note. Similar to LRS, it emphasizes the end of the repeated sequence and can be regarded as the repetition strength
4. S.Sfx: the difference between Sfx and Lrs: this feature indicates how much deviation is between the suffix link and the longest repeated suffix for each node. If S.Sfx = 0, it means the suffix link perfectly matches the longest repeated suffix and if S.Sfx is positive or negative, it indicates some kind of variation or distance between the suffix and the repeated pattern. Larger values might indicate a larger variation or a more distant reference. S.Sfx can shed more light on the stable repeating patterns rather than variation and transition to the new patterns.
5. Sfx chain: this keeps the record of previous Sfx related to the specific note/sequence and by looking at it we can always have access to the repeated segments throughout the piece
6. Sfx (LRS): Cumulative derivative for each chain of the repeated sequence where each entry represents the largest derivative value propagated through the suffix chain from the

[4] https://github.com/NadiaCarvalho/VMO [Access data: 31.03.25]

corresponding node. This feature is calculated with a loop that goes through all the elements in the Sfx chain for each note and assigns the maximum of Deriv (LRS) to all the affected notes that the chain refers to:

$$\text{Sfx(LRS)}[Sfx\ chain[t]] = \text{Max (Deriv(LRS)[j])} \quad j \in Sfx\ chain[t] \tag{2.4}$$

As stated, any time a higher derivative is detected in the chain, it will update the array in a way that all the segments that the Sfx chain referring to also have the same number for Sfx (LRS). Considering this, Since VMO uses the last note of each segment to calculate LRS and Suffix, by this feature we have access to the end of the segments and their highest derivatives which indicates the possibility of an actual segment. Additionally, since we used the normalized version of this feature, we will refer to it as Sfx (LRS)-N.

7. Sfx (LRS)-St: This feature is a variation of the previous feature to take into account the beginning of the segment. In this feature, we still look for the maximum derivative for every element in the Sfx chain (not considering the beginning of the score), but we don't substitute the end of the segment but the beginning of the segment by going back to the amount of LRS. Here is the formula for this feature:

$$Sfx(LRS)\_st[Sfx\ chain[t] - LRS(Sfx\ chain\ [t-1])] = Max(Deriv\ LRS) \quad t > 0\ , not\ in\ the\ start \tag{2.5}$$

$$Sfx(LRS)\_st[Sfx\ chain[t] - LRS(Sfx\ chain\ [t])] = Max(Deriv\ LRS) \quad t = 0\ , not\ in\ the\ start \tag{2.6}$$

Considering the formula, we go back by the LRS of the previous suffix in the Sfx chain, and similar to the previous feature, all the equivalent notes in the Sfx chain will be affected and replaced by this maximum of the Deriv LRS. For better comparison with the previous feature, we normalize this feature and refer to it throughout the document with Sfx (LRS)-St-N.

9. Sfx (LRS)-St-N-N: this feature is the reverse normalized version of feature 6

Using our personalized parser, we have access to the pitch, IOI, rest, and overall strength profile for each note. From the VMO features above, we use Sfx (LRS)-N and Sfx (LRS)-St-N. Both of these features are based on the changes in the LRS and the strength of these changes among the repetition of a sequence throughout a piece while the former's emphasis is on the end of each sequence and the latter marks the beginning of each sequence and help us in tracking the whole sequence similar repetitions across each piece.

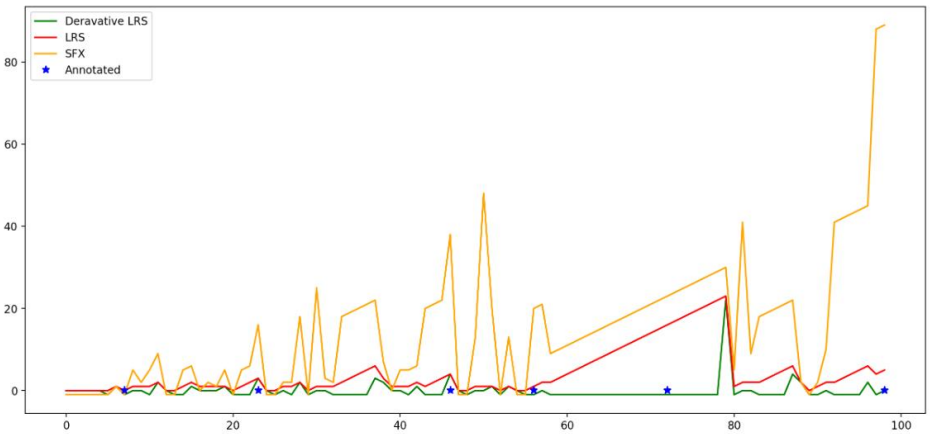

Figure 8: derivative (LRS), LRS and Sfx of a random song with annotated segments

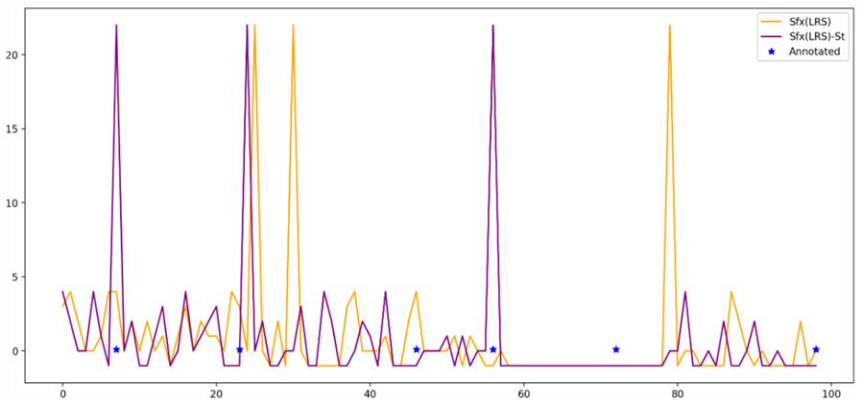

Figure 9: Sfx (LRS) and Sfx (LRS)-St of the same piece with the annotated segments

	midipitch	duration	Sfx	LRS	Deriv(LRS)	S.Sfx	Sfx Chain	Sfx(LRS)	Sfx(LRS)-St	Sfx(LRS)-N	Sfx(LRS)-St-N	Sfx(Mix)-N	Sfx(LRS)-St-N-N
0	71	2.0	-1	0	0	0	[0, -1]	3	4	0.173913	0.217391	0.217391	0.086957
1	73	1.0	-1	0	0	0	[1, -1]	4	2	0.217391	0.130435	0.217391	0.000000
2	71	1.0	-1	0	0	0	[2, -1]	2	0	0.130435	0.043478	0.130435	0.043478
3	74	2.0	-1	0	0	0	[3, -1]	0	0	0.043478	0.043478	0.043478	0.000000
4	74	0.0	-1	0	0	0	[4, -1]	0	4	0.043478	0.217391	0.217391	0.173913

Figure 10: A sample of all the VMO features for a random song

3.2 Assessing Tradition and Piecewise Weights

As illustrated in Figure 10, After careful selection of our specific features, we focus on the second part of the LBDM model which is the weighting of each feature. As we observed in the work of Cenkerova (2018), a proper weight selection could lead to improved F-measure thus, we will now discuss the two main approaches we study to optimize the method's accuracy.

3.2.1 Entropy

The first approach we studied for defining weights was piece-specific, i.e., we calculated different weights per piece. To this end, we use the information theory measure of entropy. Prior to novelty curve computation, we assess which features contain a great degree of information, thus higher entropy. Our rationale is that each piece may adopt different structural cues to create salient segmentation boundaries. Furthermore, it assumes that within each tradition, the pieces may have distinct structures.

We compute the entropy of each feature, *H,* for every song such that:

$$h\left(e_i \middle| e_1^{i-1}\right) = \log_2 \frac{1}{p\left(e_i \middle| e_i^{i-1}\right)} \tag{2.7}$$

$$H\left(e_1^{i-1}\right) = \sum_{e \in \varepsilon} p\left(e_i \middle| e_1^{i-1}\right) h\left(e_i \middle| e_1^{i-1}\right) \tag{2.8}$$

Where $p\left(e_i \middle| e_i^{i-1}\right)$ is the conditional probability of an element at index i in the sequence given the preceding element in the sequence and $h\left(e_i \middle| e_1^{i-1}\right)$ is the information content of the element. $H\left(e_1^{i-1}\right)$ or entropy dictates the higher the information content and the lower the probability, the higher the entropy. Our features benefit from the Scipy Library to compute the entropy with two scenarios in mind:

1. The low entropy means less information as a result more common and a sign of a repetitive pattern so we reversed the entropy so the feature with the lowest entropy gets the highest weight. After that, we used natural logarithms to make the entropy changes smoother, calculated the weighted summation of the features, and compared it to the annotated data.

2. The high entropy means more information and thus it shows the abnormality in the segment and the possibility of a new segment thus, we give the highest weight to the highest entropy and calculate the Sum of all features and LBDM features with and without weights and evaluated them with ground truth data.

3.2.2 Genetic Algorithm (GA)

The second weight system that is implemented in this work is using the Genetic Algorithm (GA) to find optimized tradition-based weights for the Spanish and Portuguese datasets separately. The genetic algorithm (Goldberg, 1989) is a search heuristic inspired by the process of natural selection. It's used for solving optimization problems where the goal is to find the best possible solution from a population of possible solutions. GA is especially useful for problems where traditional optimization methods may struggle, such as when the search space is large, complex, or non-differentiable.

A genetic algorithm works by simulating the process of biological evolution: selection, crossover, mutation, and survival of the fittest. It generates a population of candidate solutions and evolves them over generations to find an optimal or near-optimal solution to a problem. Genetic Algorithms and any other evolutionary algorithm work well in real-life problems which are, most of the time, complex problems. This approach escapes the drawback of getting stuck in local optima and is most likely to converge at a globally optimal or near-globally optimal solution.

3.2.3 Components of a Genetic Algorithm

Population: A group of possible solutions (called individuals or chromosomes) to the problem. Each individual represents a set of parameters (or genes) that we want to optimize.

Fitness Function: The objective function we want to optimize (minimize or maximize). The fitness of an individual determines how well it performs at solving the problem.

Selection: Based on fitness, some individuals are selected to pass their genes to the next generation. Fitter individuals have a higher chance of being selected.

Crossover (Recombination): Selected individuals are combined (crossed over) to create new offspring. The offspring inherit genes from both parents, which introduces new combinations of parameters.

Mutation: Random changes are introduced to some individuals to ensure diversity and avoid premature convergence to a suboptimal solution.

Survival of the Fittest: After several generations, the algorithm tends to evolve toward the best solutions. The fittest individuals are more likely to survive and reproduce, while weaker individuals are removed from the population.

In single-objective optimization, we have a single fitness function to optimize. The goal is to either maximize or minimize this function. For example, the function could be something like $f(x_1, x_2, \dots, x_n)$ where $x_1, x_2, \dots, x_n$ are the parameters that need to be optimized.

Bound constraints are simply limits placed on the values that the parameters can take. For instance, if we are optimizing a parameter x, we might have a constraint:

$$x_{min} \leq x \leq x_{max}$$

The genetic algorithm must respect these bounds when searching for an optimal solution. During the mutation or crossover processes, if an updated gene (parameter) falls outside the allowed bounds, it is either clipped or mapped back into the permissible range. The algorithm that we used to implement this algorithm in Python is the DEAP Library (De Rainville et al., 2012).

Finally, as illustrated in Figure 5, for calculating the envelope of strength related to enhanced weighted features a normalization and peak picking algorithm is used that will also be optimized in this work and will be discussed further. Additionally, although all Introduced VMO features are used for the entropy-based weighting, the VMO-related features that are used for the optimization algorithm are only limited to the "Sfx (LRS)-St-N" and "Sfx (LRS)-St-N-N".

3.2.3.1 DEAP: A Python Framework for GA

DEAP (Distributed Evolutionary Algorithms in Python) is a novel evolutionary computation framework for rapid prototyping and testing of ideas. With this framework, we can define and modify our GA and all its different layers from population to mutation. Its design departs from most other existing frameworks in that it seeks to make algorithms explicit and data structures transparent, as opposed to the more common black box type of frameworks. DEAP's core consists of three modules: Base, Creator, and Tools.

- **Base:** contains objects and data structures that are not implemented in the Python standard library. This module has classes like generic fitness and toolbox. The Toolbox is responsible for the operators that we are using in our evolutionary algorithm and the user can choose between available operators such as the mutation algorithm provided by the library and modify it whenever required.
- **Creator:** This module allows the creation of new classes both data and functions
- **Tools:** frequently used evolutionary algorithms operators and also objects related to additional analysis and statistical processing

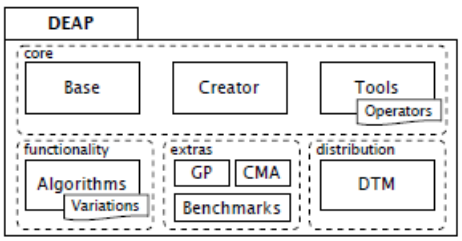

Figure 11- Deap framework inner architecture

3.3 Peak-picking optimization

As the last block of the LBDM algorithm and after the calculation of the novelty curve, the peak-picking needs to be tailored to our enhanced and traditional-specific model. In the context of music structure analysis, the peak positions of a novelty function were used to derive segment boundaries between musical parts (Böck et al., 2012; Nieto, 2015; Nieto & Bello, 2016). In the case that the novelty function has a clear peak structure with impulse-like and well-separated peaks, the selection of peaks seems a simple problem. However, in practice, one often has to deal with rather noisy novelty functions that have many spurious peaks. In such situations, the strategy used for peak-picking typically has a substantial influence on the quality of the final detection or segmentation result.

Often, simple smoothing operations may help to reduce the effect of noise-like fluctuations in the novelty function. Also, adaptive thresholding strategies, where a peak is only selected when its value exceeds a local average of the novelty function, can be applied. To further reduce the number of spurious peaks, another strategy is to impose a constraint on the minimal distance between two subsequent peak positions.

In the following, we introduce the common peak-picking strategies that we tried on our dataset which are based on various heuristics and depend on several different parameters. Additionally, the adaptive approach is optimized using the Genetic algorithm to ensure the efficiency of the parameters for the I-folk dataset.

The simplest approach that initially comes to mind is to locate the local maximum by searching for a negative derivative after a positive derivative in the final envelope of our boundary strength. However, this method can easily fail when it comes to noisy environments and results, and that is when **adaptive thresholding** (Nieto, 2015) comes into play. In this mechanism, we first apply a smoothing filter to the novelty function and we decide on the peaks based on the local average of the neighboring data. The Music Structure Analysis Framework (MSAF) (Nieto, 2015)Python Package has been built in a way to include this algorithm but we decided to use

Scipy library[5] which can do the same thing by defining a smoothing filter to its height variable. Furthermore, the well-known Librosa Library[6] also has a peak-picking function which is implemented by considering a minimum distance between each peak which seems very useful for periodic applications. Even though this function takes into account three conditions to detect a peak and it mainly works well with periodic segments, the interplay of these various conditions can lead to the rejection of some prominent peaks. Last but not least, while the comparison of these models and optimization results is further discussed in the evaluation chapter, Figure 11 shows the difference between each model and their results which is done by Muller (2021).

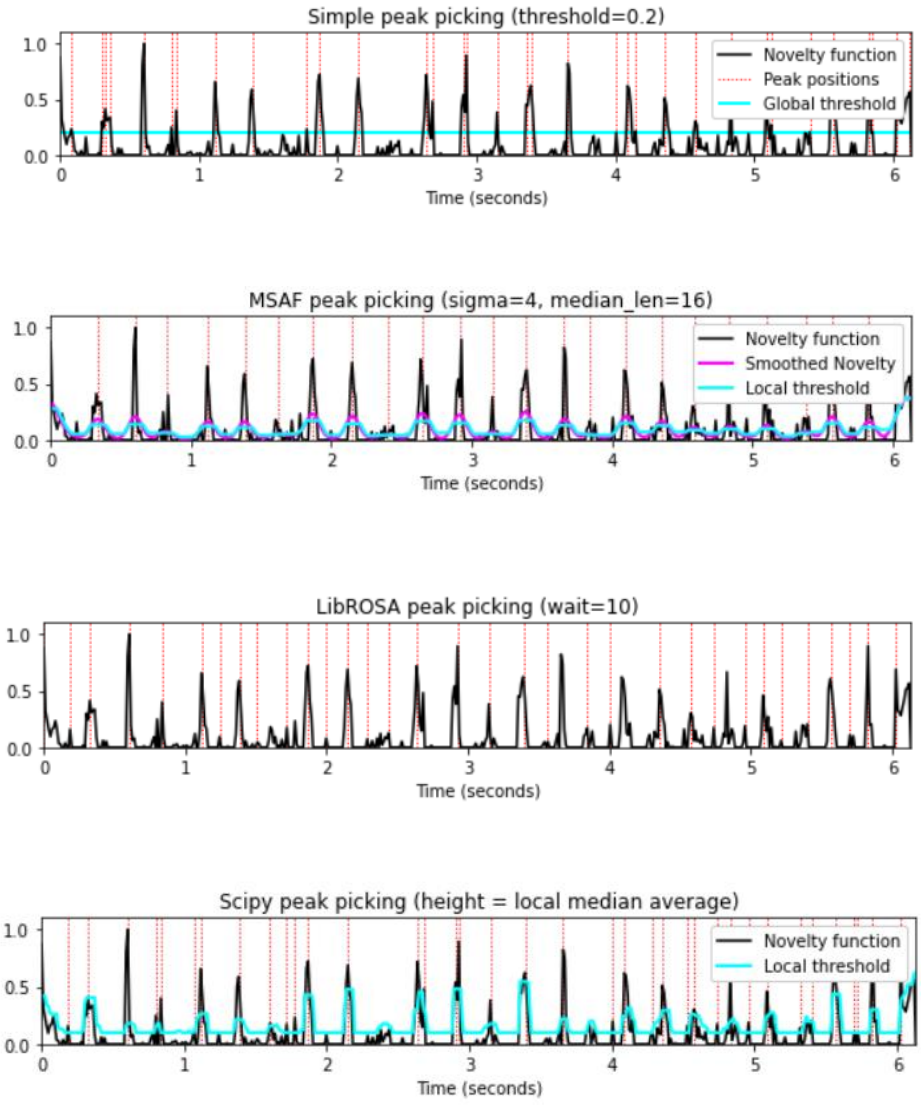

Figure 12: Comparison of peak-picking algorithms(Müller, 2021)

[5] https://scipy.org/ [Access data: 01.03.25]
[6] https://librosa.org/ [Access data: 01.03.25]

4. Results and Evaluation

In this chapter, we will review the results obtained from our method and run our algorithm on the dataset. The first section of this chapter will be a detailed assessment of the dataset and try to shed light on how the data is scattered. After that, we continue with the result of the weight enhancement on the original features of LBDM and also our added features based on the VMO model. Later, we will discuss the result of peak-picking optimization and its effect on the new features and weights. Lastly, we discuss the results obtained from random forest which we also tried as the first step of further development.

4.1 Preliminary assessment of the dataset

The first step before any processing of the data is to gain a clear understanding of the structure of the data. Thus, in this section, we will try to see the distribution of our dataset. The focus of our work is on Portuguese and Spanish folk music and we disregard the rest of the pieces. Since all our datasets are annotated by specialists, we tried to gather an overall understanding of how these segments are shaped and distributed.

In this work, we use the parser written by Carvalho et al (2021) to access the Portuguese and Spanish data sets. This parser allows us to access different parts of the data such as lyrics, partiture, and symbolic descriptors separately. With this parser, we could easily have access to the annotated segments, each note and rest position and duration, IOIs and offsets, and even the LBDM strength which minimizes the computation time and makes us focus on calculating and adding VMO features to our feature set.

In the total number of 59 Portuguese folk music, we have a number of 566 segments none of which had a rest as their previous event but 150 of them had a rest after the segment or boundary note. These numbers are drastically different for Spain because we have substantially more Spanish pieces in our data set. There are 738 Spanish music scores with a total segment of 3334. Among this number, 41 segments had a rest before and 1835 of them had rest after the boundary note. Figure 12 elaborates on these numbers in comparison and illustrates how many segments had a rest or note before and after them normalized by the number of segments.

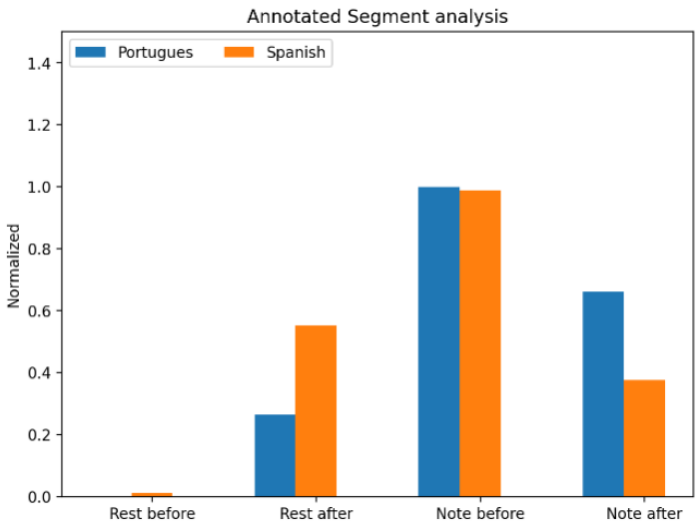

Figure 13: annotated segments general analysis

Additionally, We also studied the number of notes per each annotated segment. As can be seen in Figure 13, most of the segments in our ground truth data have 10-20 notes per segment while in Spanish music, we can see these numbers extend up to 40 notes per segment which is an indicator of longer pieces, segments, and more variety in Spanish data. Additionally, since all 566 Portuguese segments had a note before the beginning-of-the-segment note and 375 of them also had a note after the segment start, we have examined the distance of the start-of-segment note with the note before and after it in semi-tones.

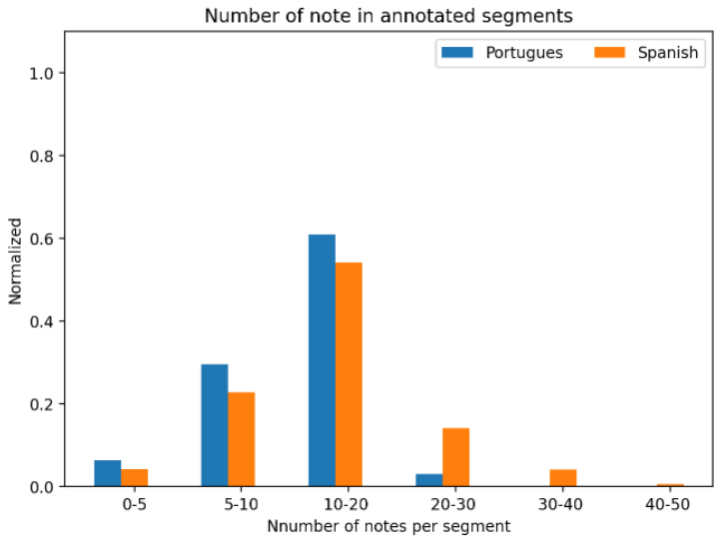

Figure 14: Number of notes per annotated segment

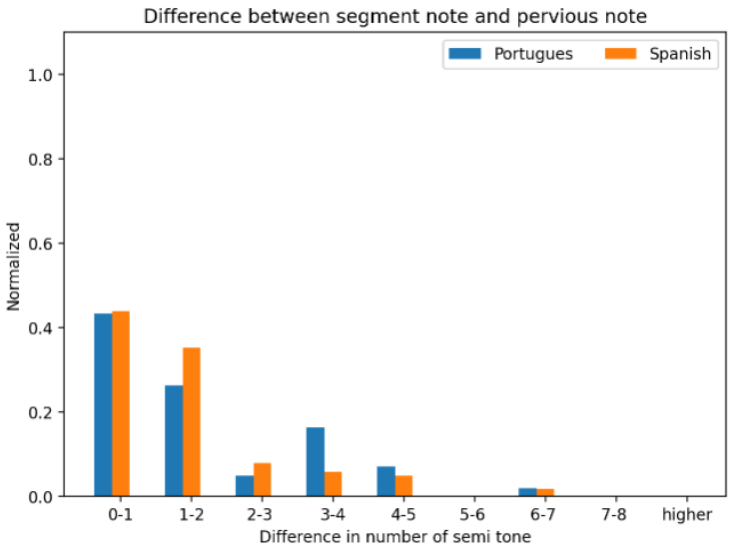

Figure 15: Difference between the segment note and previous note

Considering Figure 14, we can see that most of the musical difference between the segment note and the previous note in 566 Portuguese segments and 3293 Spanish segments, is between 0 to 2 semi-tones (unison and a second in musical interval) while Portuguese have a higher percentage of segments with up to for 4 semitones (the equivalent of a third in music intervals) and Spanish have more variety of intervals across the x-axis.

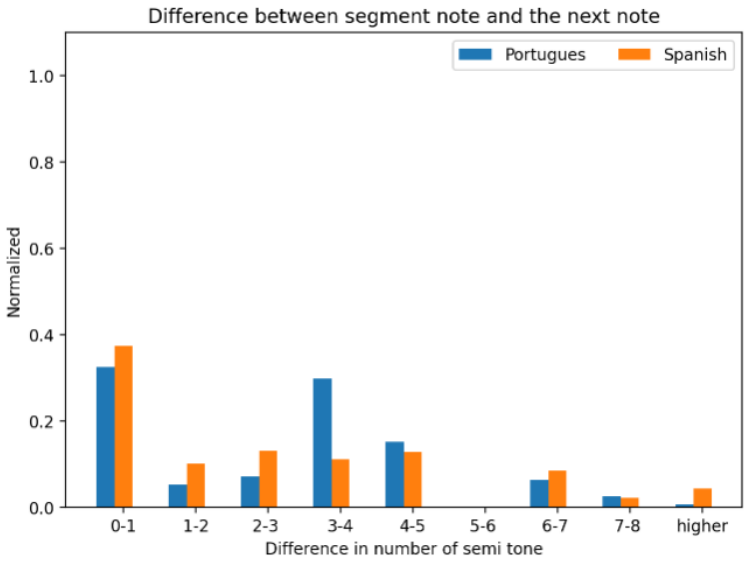

Figure 16: Difference between the segment note and second note in the segment

However, these figures change when we consider the note after each start-of-segment note, looking at Figure 15, among 1251 of the Spanish segments and 150 segments of Portugues folk music that have a note after them, the difference varies between 1 to 4 semitones (unison, second and third) whereas this distance has its highest peak on 0-1 semitone for both datasets and more evenly distributed between 1 to 5 semitones for Spanish music. It is worth mentioning that both regions do not have much content when it comes to a difference of 5-6 semitones.

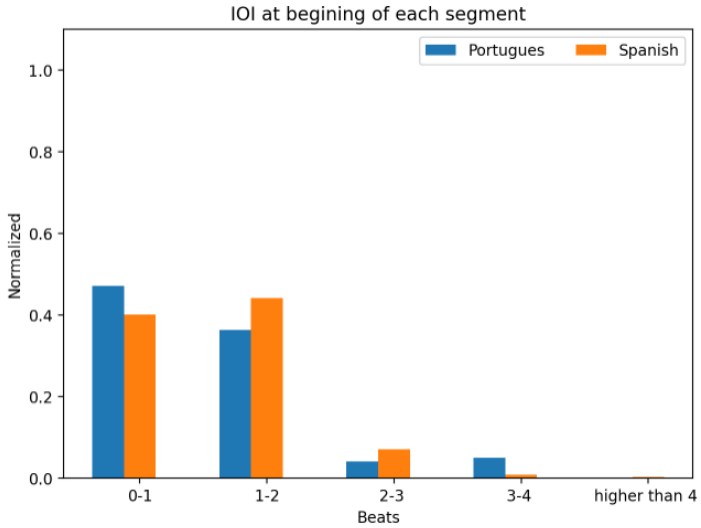

Figure 17: IOI difference at the beginning of segment

Lastly, we have also calculated the IOI of the segment note and previous note. We have used music21[7] to calculate the offset difference of the start-of-segment note with the previous note and Figure 16 is the result of this calculation for Portuguese and Spanish folk music. In terms of musical timing, this shows that mainly the Portuguese segments are highly likely to start on the 1st and 2nd beat while for Spanish this likelihood is reversed.

4.2 Weights enhancement: Entropy

After gathering a better insight into our dataset, we now proceed with the first approach which was calculating the entropy of VMO features discussed in chapter three and LBDM strength (Rest, IOI, and Pitch) per song. We calculate the overall performance of the segmentation based on the novelty curve considering three variations: 1. Without any weights for any feature 2. taking into account the maximum weight for the lowest entropy 3. Weighing the features with the maximum weight for the highest entropy. We calculate all these three variations for each song, extract the final envelope of the novelty curve, use the Scipy peak-picking to detect possible segment starts and evaluate them with recall, precision, F, and R measures.

[7] https://www.music21.org/music21docs/index.html [Access data: 31.03.25]

The features used to calculate all these three variations of weighting are LBDM (spitch), LBDM (sioi), LBDM (srest), and 'Sfx (LRS)-N' from the VMO features to introduce the repetition to our model. With these features, we managed to reach the F-measure of 0.655 in the mixture of Portuguese and Spanish data (797 folk pieces). As illustrated in Table 3, It is worth pointing out that we also take into account the LBDM separately as our reference point. Regarding the table, "Entropy" refers to weights that take into account the lower entropy as the higher importance (Variation 2), "Inv Entropy" similarly refers to high weights when there is a higher entropy (Variation 3), and finally, "No Weights" means pure features are used for the segmentation of each song (variation 1).

Table 3: the entropy-weighted and non-weighted comparison of features. All features - Entropy: LBDM and VMO features weighted (Variation 2) All features - Inv Entropy: LBDM and VMO features weighted differently (Variation 3) All features - no weights: VMO and LBDM features without weights LBDM - Entropy: LBDM features weighted by entropy (variation2) LBDM - Inv Entropy: LBDM features weighted by entropy (variation 3) LBDM: classic model with its original weights

	Precision	**recall**	**F measure**	**R measure**
All features - Entropy	0.623	0.764	0.641	**0.328**
All features - Inv Entropy	0.594	0.770	0.624	0.238
All features - No weights	0.623	0.764	0.641	**0.328**
LBDM - Entropy	0.406	0.736	0.490	-0.251
LBDM - Inv Entropy	0.419	0.761	0.505	-0.264
LBDM	**0.654**	**0.770**	**0.655**	0.310

As illustrated in Table 3, you can see even though we have used the piece-specific segmentation for each song in the dataset, still, the LBDM with its classic weights has the best performance except for the R-measures which are the highest for both entropy and inverse entropy applied on all features. This translates to the fact that this entropy-based system with these features, does not add any Improvement over the traditional method. Additionally, using entropy as an indicator of an unexpected event and possibly a segment ("LBDM - Inv Entropy" in the table), Improves the LBDM more than the normal weighting system ("LBDM – Entropy") while it has a reverse effect when Sfx (LRS)-N feature is added ("All features - Entropy" and All features - Inv Entropy).

In the next trial (Table 4), the feature 'Sfx (LRS)-St-N' from the VMO is also added to our feature list. And again, the same algorithm ran on all datasets. Since this feature as explained in the theory, keeps track of the start of the segments with the maximum derivative of LRS, we hoped it would improve the results.

Table 4: Performance comparison with Sfx (LRS)-St-N added to features contributing to the novelty curve

	Precision	recall	F measure	R measure
All features - Entropy	0.634	0.772	**0.653**	0.364
All features - Inv Entropy	0.594	0.770	0.624	0.238
All features - No weights	0.634	**0.772**	0.653	0.364
LBDM - Entropy	0.435	0.747	0.515	-0.145
LBDM - Inv Entropy	0.421	0.766	0.508	-0.265
LBDM	0.654	0.770	0.655	0.311

As indicated in this table, the major difference compared to the previous case, is the slightly improved recall for "All features - no weights" which means by adding this feature we managed to find more correct segments but the difference is very small. However, by adding this feature the overall F- measure seems to be increasing for the All "features – Entropy". We also examined another feature of VMO in the theory chapter by the name of 'Sfx (LRS)-St-N-N' but the result was similar to the previous one so we didn't include the full table for it here.

4.3 Weights enhancement: Optimisation

Since the results for song-specific entropy weighting didn't provide us with much improvement, we resorted to genetic algorithm optimization to enhance the LBDM weights and maximize the F-measure for each region. Three default LBDM weights were originally 0.25,0.5, and 0.25 for rest, IOI, and pitch with a threshold of 0.2. We ran the genetic algorithm to optimize these three weights on 60% of Training data on Portuguese and Spanish folk pieces and achieved an optimum weight of 0.52 for rest, 0.66 for IOI, and 0.25 for pitch for Portugues and also 0.47 for rest, 0.48 for IOI and 0.33 for pitch in Spanish pieces. Afterward, we ran the optimized LBDM with the new weights for I-Folk on 40% of the data (23 Portuguese and 269 Spanish pieces) to measure the overall performance. We ran this algorithm **four** times with training and reached the following average results:

Table 5: Optimized-weight LBDM performance for I-folk

	Precision	recall	F measure	R measure
Optimized weight LBDM - Portugues	0.642	0.682	**0.622**	0.422
Original LBDM - Portugues	0.625	0.69	0.61	0.372
Optimized weight LBDM - Spanish	0.727	0.705	**0.655**	0.485
Original LBDM - Spanish	0.62	0.75	0.607	0.26

Because of this improvement in the overall performances in Table 5, we decided to examine a similar approach like the entropy section by adding those features of VMO namely 'Sfx (LRS)-St-N-N' and 'Sfx (LRS)-St-N'. We trained and tested the optimization four times again with the new features and measured the performance.

The new optimized weights of the new features were 0.65, 0.51, 0.23, 0.18, 0.15 for Rest, IOI, Pitch, Sfx (LRS)-St-N, Sfx (LRS)-St-N-N for Portugues and 0.52, 0.27, 0.17, 0.17 and 0.12 for Spanish. The achieved result is reported in Table 8.

Table 6: Optimized-weight LBDM with VMO's Sfx (LRS)-St-N and Sfx (LRS)-St-N-N features

	Precision	recall	F measure	R measure
Optimized LBDM+VMO - Portuguese	0.64	0.69	**0.633**	0.443
Optimized LBDM+VMO - Spanish	0.656	0.743	0.646	0.38

Regarding Table 6, although Portugues F-measure increased by 1%, It appears that adding the new VMO features to LBDM has no significant effect on the previous algorithm in Spanish music.

4.4 Peak-Picking Optimisation

As the last stage of our enhancement, we focus on the peak-picking algorithm. Firstly, we ran the three algorithms discussed in Chapter 3 on a mixture of Portuguese and Spanish data to have an overall comparison between the two algorithms. As you can see in Table 7, among 237 folk music from two regions, The Scipy library gives us the best F-measure. As illustrated Librosa, can achieve good precision and R score since most of the found segments are correct but also due to its greedy implementation, it misses a lot of segments substantially (low recall) compared to Scipy. Due to this preliminary examination, we decided to continue with the Scipy algorithm. As we explain further, we also have used the genetic algorithm to optimize two key

factor parameters of the Scipy's peak-picking algorithm namely kernel size and offset to make sure they are fully adjusted to our dataset.

Table 7: Peak-picking methods comparison

	Precision	Recall	F-Measure	R-Score
MSAF	0.747	0.577	0.608	0.310
Scipy	0.580	**0.801**	**0.629**	0.181
Librosa	**0.886**	0.541	0.619	**0.617**

We here discuss the result we have obtained by Optimizing the peak-picking algorithm using the genetic algorithm. As explained previously we have implemented our GA algorithm using Deap Framework in Python to optimize the kernel and offset of the Scipy Peak-picking algorithm. Kernel and offset have been optimized on 40% of the data for Portuguese and 50% for Spanish (less computation time). After the optimization is finished, the best kernel choice since the kernel needs to be an odd integer, rounded up to 5 for both regions while the offset is 0.40 for Portugues and 0.50 for Spanish. On the original LBDM, we have also used 5 for kernel and 0.25 for offset in both regions which we brought in table 10 for comparison. Consequently, the new parameter substituted for the default value of kernel and offset, and the LBDM with optimized weights ran on the rest of the data.

Table 8: LBDM performance comparison with optimized peak-picking for each region (K for kernel and O for offset)

	Precision	recall	F measure	R measure
Optimized LBDM - Portugues (K=5, O=0.4)	0.70	0.69	**0.645**	**0.56**
LBDM - Portugues	0.63	0.74	0.63	0.3
Optimized LBDM - Spanish (K=5, O=0.5)	0.85	0.59	**0.64**	**0.58**
LBDM – Spanish	0.565	0.785	0.61	0.16

We ran the whole algorithm **twice** and the results illustrated in Table 8 are the average of these two examinations. As illustrated, even though, The Portuguese results are very close to the standard LBDM with default weights (kernel size of 5 and offset of 0.25), it still shows a sign of improvement and this might also be due to the lack of enough data to train the genetic algorithm compared to Spanish music. The major difference here is in R measure that increased with lower

recall which suggests the algorithm manages to give us a more realistic segmentation by avoiding overfitting. Whereas, in Spanish data, we face a promising increase in F and R measures especially due to the fact that in this section, we used what is known as static weights meaning we did apply the same weight for LBDM to all training and test data which was different from the previous section that we used unique weights for each song.

Lastly, since we already have the Optimized peak-picking and weights which showed a promising result, we added our VMO features as well and gathered the results for optimized LBDM+VMO which is illustrated in Table 9.

Table 9: Optimized-weight LBDM with VMO and optimized peak-picking

	Precision	recall	F measure	R measure
LBDM+VMO - Portuguese	0.64	0.63	**0.60**	0.55
LBDM+VMO - Spanish	0.75	0.69	**0.67**	0.57

During the multiple trials, it appears the use of enhanced weight, feature, and peak-picking optimization Improved the Spanish datasets, but it reached a lower result compared to previous models in Portuguese folk music. Thus, although this might be caused because of the lack of enough data for Portuguese compared to Spanish, it is difficult to reach a conclusion related to the effectiveness of this combination for Portuguese pieces. Additionally comparing our results with the work of Cenkerova (2017, 2018) in Table 10 which was our inspiration, we managed to improve the overall result of LBDM for the Spanish part of our dataset with these two optimizations.

Table 10: LBDM results from Cenkerova (2017,2018)

Model	Precision	Recall	F-measure
LBDM	0.81	0.60	**0.65**

4.5 Random Forest

Inspired by the work of Kranenburg (2020), we decided to also try the random forest classification (Breiman, 2001) on our dataset as well. Random Forest is a powerful ensemble learning technique widely used for classification tasks, including music segmentation. It is an extension of decision tree classifiers and is particularly suited for scenarios where patterns are complex. In the context of music segmentation, Random Forest helps in identifying boundaries between distinct musical phrases or segments, leveraging both the musical features and the underlying patterns of the data. This algorithm constructs an ensemble of decision trees during training, each of which predicts the class (segment or not) of a given data point (the musical notes in the score and their features). The final classification is based on the majority vote from these decision trees. Instead of relying on a single decision tree, Random Forest aggregates the predictions from multiple trees to improve accuracy and avoid overfitting.

Music segmentation involves identifying boundaries between sections within a song. Given the complexity of music data, Random Forest provides the advantage of detecting non-linear relationships between features and segments or boundaries, and also due to the aggregating output of multiple trees, this method reduces the risk of overfitting that might occur in less diverse datasets.

As the last part of this work and also as a guide for future development and continuation of our work, we tried this supervised model as well. This algorithm ran on an 80-20 basis for training and test data (5 folds). But we also ran the prior experiment with Portuguese folk music with 3 and 6-fold to make sure 5 is the best choice.

For running the algorithm, we unified all the pieces in one big data frame for each region and ran it for 5 folds which is the average of these five trials for different features illustrated in Table 10.

Table 11: Random forest results

Features-region	Precision	recall	F measure
LBDM features-Portugues	0.458	0.419	0.430
LBDM+VMO features- Portugues	0.526	0.437	0.464
LBDM+VMO+ Other features - Portugues	0.636	0.498	**0.548**
LBDM features -Spanish	0.510	0.316	0.389
LBDM+VMO features - Spanish	0.533	0.338	0.413
LBDM+VMO+ Other features - Spanish	0.708	0.545	**0.616**

The features used for random forest are three LBDM features (rest, pitch, and IOI strength), Sfx (LRS)-St-N and Sfx (LRS)-St-N-N from VMO and pitch (in MIDI), IOI, duration of the note,

duration of the rest which is shown as “Other” in table 10. As illustrated in the following tables, adding more features improves the performance of our classification algorithm. Although adding the features mentioned as “Other” improves it substantially, the maximum F-measure that we could have reached with this algorithm would be around 61% for Spanish folk music which is still lower than what Kranenburg (2020) reached with Meerten’s dataset (MTC in table 12) which is still lower than the rest of our model which is what has been suspected due to the fact that statistical methods usually have a loser performance considering the rule-based methods.

Table 12: Random Forest from Karnenburg (2020)

Dataset	Random forest (F-measure)
MTC	**0.68**
EFSC	0.76
CHOR	0.89

4.5.1 Worst cases

In this subsection, we will look at a few of the pieces that resulted in the worst F-measures after we applied the optimized weights and peak-picking.

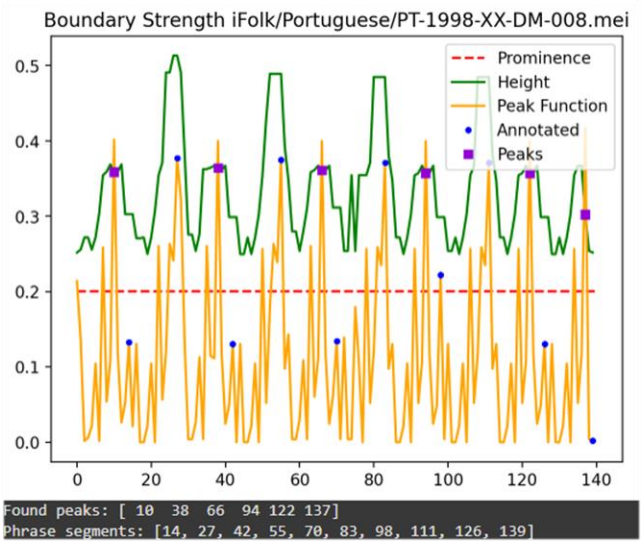

Figure 18: The output of segmentation for one of the Portuguese folk music with lowest F-measure of 0.12

Many of the pieces that have relatively low F-measures had a very similar peak graph, meaning the minimum and maximum of the boundary strength graph or novelty curve (orange graph) is high and as you can see in Figure 18, for the smoothed curve (green graph) to be able to detect maximums, it has to disregard a lot of potential segments that are actual boundaries. The Prominence (red line) in the Scipy peak-picking which is equivalent to the threshold in LBDM, is also examined, and even though it causes some changes in some pieces the overall performance stays the same.

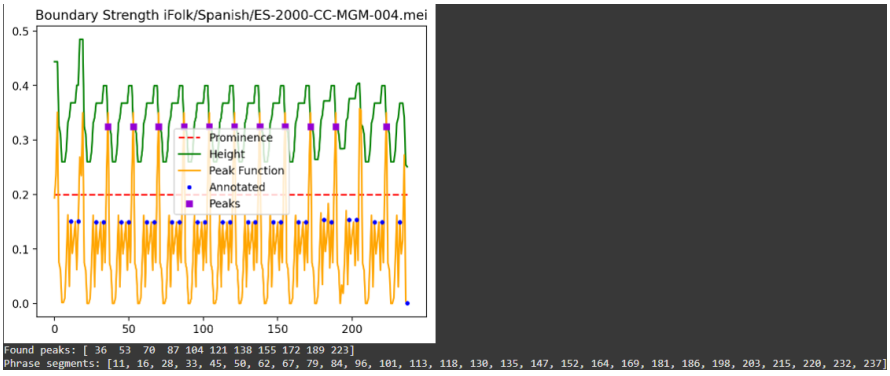

Figure 19: The worst segmentation in Spanish pieces

Another good example of this is shown in Figure 19 among the Spanish pieces. As illustrated, the difference between the lowest and highest picks is more than 0.4 with a lot of local maxima which shows even though we have good IOI, Rest and Pitch jumps on each boundary that the algorithm has found, we also have inter-boundary segments that are having similar characteristics and can be counted as standalone segments. This shows the weakness of this rule-based model and how we need a system to be able to detect the intricate patterns within the more obvious segments.

5. Conclusion

In this chapter, the conclusion of our work is discussed and our plan for this work is laid out. Firstly, We review the overall methodology and discuss the main points of our results. Furthermore, we lay the foundation for the next step of this project and the future of our music segmentation model.

5.1 Discussion

In this work, we tried to explore the ways of improving classical segmentation models for the I-folk dataset using genetic algorithm optimization and with a slight lean toward the statistical approaches. In comparison, the Rule-based models reach a better segmentation compared to supervised and non-supervised models and that's the main reason we chose one of the common models like LBDM and tried to tailor it to our specific dataset and improve it while we also tried other models in search of finding the best segmentation model for our dataset.

Studying the data, we realized the rest and IOI has a significant effect on boundary detection since 25 percent of segments in Portugues folk music and 50 percent of Spanish pieces had a rest after the start of a segment and the IOI difference for each segment can vary in 0.5 to 2 beats. Additionally, most of the segments start with unison or major/minor second in Portuguese and major/minor third for Spanish folk music which demonstrates itself in the weight for the pitch in LBDM.

The weights for the optimized LBDM were 0.52, 0.66, and 0.25 (rest, IOI, and pitch) for Portugues and 0.47, 0.48, and 0.33(rest, IOI, and pitch) for Spanish. In classic LBDM, we also observe the importance of IOI as the highest weight which is also repeated here but what is different in our LBDM is the weight for the Rest. The Rest's weight also has an important role in the overall segmentation which is in cohesion with the fact that more than 25 to 50 percent of our segment have a rest afterwards.

Moreover, When the VMO features are added to the picture, and due to the fact that VMO features are created based on the Oracle that takes into account the pitch, duration, and the rest, we see a decrease in all the weights contributions and interestingly the weight related to rest becomes the highest among all weights due to the correlation on rests and IOI with segment detection.

These phenomena, strengthen the foundation for our theory which is if the improvement of LBDM or any other weight system is of importance, the weights shall be personalized for each dataset, and using the same old tradition cannot be the best case. Cenkerova (2018) reached an overall performance of 65 percent and we started this work with the motivation of improving it even more for the I-folk dataset.

We first tried to implement a method that does not rely on annotated data to improve the segmentation, so we used the entropy of features related to LBDM and VMO to extract a weighted novelty curve and find possible segments. The respective results didn't outperform the LBDM. Consequently, we changed our approach by trying to optimize the LBDM with more complicated algorithms. We managed to find the optimized parameter by trying a genetic algorithm on 60 percent of annotated data. Since the tendency to difference and signs of improvement appeared (1% for Portuguese and 5% increase for Spanish), we followed the same path by optimizing the peak-picking algorithm in LBDM and Adding more features to it. Adding the VMO features, increased another percent in the Portuguese pieces but showed a one-percent decrease for Spanish data. Also, using optimized peak-picking, lowered our optimized LBDM for Portuguese folk music and caused no changes in Spanish ones. Despite all these, when we added Optimized LBDM with VMO features and also optimized peak-picking we managed to increase the F-measure for Spanish to 67% compared to the original performance of 60% for classic LBDM and higher than Cenkerova's work while a substantial decrease to classical LBDM in Portugues folk music. These fluctuations in performance results for Portuguese data in different algorithms are expected to root in firstly the lack of enough data for the algorithm to train well and secondly the quality-of-the-data-driven nature of the genetic algorithm.

It is worth mentioning, that regarding the peak-picking optimization, we had to split the test and train data 50-50, due to the fact that the optimization algorithm was a very time-consuming algorithm and the number of Spanish data was more than 700 pieces. For instance, for LBDM+VMO which used 5 features in total, each generation of parameters consists of 500 variables for each iteration of algorithm for each song and this was one of the main challenges of this work.

With this in mind and as the beginning step to supervised models, we tried the random forest which is a very well-known model in classification. Observing the different folds we run, it is understood that the more features we add, the better the results will get. Even though these improvements for VMO are not very significant compared to adding features like IOI, pitch, or duration themselves (labeled other in the related table).

5.2 Future work

This work has been done on a tight schedule due to PhD scholarship deadline and it is meant to be a preliminary step for a bigger project. We plan to expand this work to other models and other regions. The improvement and future work can be categorized as follows:

- Increasing the number of Portugues folk pieces and re-apply the current models to check the validity
- Examine the supervised, unsupervised, and more advanced learning models on Portuguese and Spanish folk music and compare the results
- Annotating the Italian, Mexican, and Galician datasets that are a part of I-folk and measure the current algorithms on them
- Implementing parallel processing and using GPU to increase the speed of optimization and as a result less computational power
- Adding the folk music of other countries such as Iran to the dataset and processing the results

6. Refferences

Ajmera, J., McCowan, I., processing, H. B.-I. signal, & 2004, undefined. (n.d.). Robust speaker change detection. *Ieeexplore.Ieee.OrgJ Ajmera, I McCowan, H BourlardIEEE Signal Processing Letters, 2004•ieeexplore.Ieee.Org*. Retrieved November 17, 2024, from https://ieeexplore.ieee.org/abstract/document/1316876/

Allauzen, C., Crochemore, M., & Raffinot, M. (1999). Factor oracle: A new structure for pattern matching. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *1725*, 295–310. https://doi.org/10.1007/3-540-47849-3_18

Bassan, S., Adi, Y., & Rosenschein, J. S. (2022). Unsupervised Symbolic Music Segmentation using Ensemble Temporal Prediction Errors. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, *2022-September*, 2423–2427. https://doi.org/10.21437/Interspeech.2022-10379

Ben-Amos, D. (2020). *Folklore concepts: Histories and critiques*. Indiana University Press.

Bernardes, G., Cocharro, D., Caetano, M., Guedes, C., & Davies, M. E. P. (2016). A multi-level tonal interval space for modelling pitch relatedness and musical consonance. *Journal of New Music Research*, *45*(4), 281–294. https://doi.org/10.1080/09298215.2016.1182192

Böck, S., Krebs, F., & Schedl, M. (2012). Evaluating the Online Capabilities of Onset Detection Methods. *ISMIR*. https://archives.ismir.net/ismir2012/paper/000049.pdf

Bod, R. (1998). *Beyond grammar: An experience-based theory of language*. https://pure.mpg.de/pubman/faces/ViewItemOverviewPage.jsp?itemId=item_407836

Bod, R. (2002). Memory-based models of melodic analysis: Challenging the gestalt principles. *International Journal of Phytoremediation*, *21*(1), 27–36. https://doi.org/10.1076/jnmr.31.1.27.8106

Boroda, M. (1982). Zur Bestimmung einer phrasenänlichen melodischen Informationseinheit in der Musik. *Sprache, Text, Kunst. Quantita Tive Analysen*, 222–230.

Breiman, L. (2001a). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Breiman, L. (2001b). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Burgoyne, J., Wild, J., & Fujinaga, I. (2011). An Expert Ground Truth Set for Audio Chord Recognition and Music Analysis. *ISMIR*. https://ismir2011.ismir.net/papers/OS8-1.pdf

Cambouropoulos, E. (2001). The local boundary detection model (LBDM) and its application in the study of expressive timing. *Citeseer*. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=3a4cd69b46b30fef9eef63569120491ab69737ce

Carvalho, N., & Bernardes, G. (2021). SyVMO: Synchronous Variable Markov Oracle for Modeling and Predicting Multi-part Musical Structures. *10th International Conference on Artificial Intelligence in Music, Sound, Art and Design*, *12693 LNCS*, 37–51. https://doi.org/10.1007/978-3-030-72914-1_3

Carvalho, N., Gonzalez-Gutierrez, S., Merchan Sanchez-Jara, J., Bernardes, G., Navarro-Caceres, M., & Navarro, M. (2021). Encoding, analysing and modeling i-folk: A new database of iberian folk music. *Proceedings of the 8th International Conference on Digital Libraries for …, 2021*, 75–83. https://doi.org/10.1145/3469013.3469023

Cenkerová, Z. (2017). Melodic segmentation: Structure, cognition, algorithms. *Musicologica Brunensia*, *52*(1), 53–61. https://doi.org/10.5817/MB2017-1-5

Cenkerová, Z., Hartmann, M., & Toiviainen, P. (2018a). CROSSING PHRASE BOUNDARIES IN MUSIC. *Sound and Music Computing Conferences*.

Cenkerová, Z., Hartmann, M., & Toiviainen, P. (2018b). Crossing phrase boundaries in music. *Sound and Music Computing Conferences*. https://jyx.jyu.fi/jyx/Record/jyx_123456789_59860

Charniak, E., & Magerman, D. M. (1996). *Statistical language learning*. The MIT Press. https://books.google.com/books?hl=en&lr=&id=ps3mqZANrHUC&oi=fnd&pg=PA1&dq=Charniak,+E.+(1993).+Statistical+Language+Learning,+Cambridge:+The+MIT+Press.&ots=XyLbiploCA&sig=W8d70WQY0MEIlj8lWdzxR4Wmm14

Cohen, W. (1995). Fast effective rule induction. *Proceedings of the Twelfth International Conference on Machine Learning, 1995*. https://books.google.com/books?hl=en&lr=&id=akijBQAAQBAJ&oi=fnd&pg=PA115&d

q=W.+W.+Cohen,+%E2%80%9CFast+effective+rule+induction,%E2%80%9D+in+Proceedings+of+the+Twelfth+International+Conference+on+Machine+Learning,+1995,+pp.+115%E2%80%93123.&ots=MMXBx1ALxq&sig=jy4F5gDDgypR0BE--youaO2Nozc

Collins, M. (1999). Head-driven statistical models for natural language parsing. *Computational Linguistics, 2003*. https://direct.mit.edu/coli/article-abstract/29/4/589/1822

David, M. (2016). Computational music analysis. *Computational Music Analysis*, 1–480. https://doi.org/10.1007/978-3-319-25931-4/COVER

de Berardinis, J., Meroño-Peñuela, A., Poltronieri, A., & Presutti, V. (2023). Choco: a chord corpus and a data transformation workflow for musical harmony knowledge graphs. *Scientific Data*, *10*(1), 641. https://www.nature.com/articles/s41597-023-02410-w

de Haas, W., Volk, A., & Wiering, F. (2013). Structural segmentation of music based on repeated harmonies. *IEEE International Symposium on Multimedia*. https://ieeexplore.ieee.org/abstract/document/6746800/

De Rainville, F. M., Fortin, F. A., Gardner, M. A., Parizeau, M., & Gagné, C. (2012). DEAP: A Python framework for Evolutionary Algorithms. *Proceedings of the 14th International Conference on Genetic and Evolutionary Computation Companion*, 85–92. https://doi.org/10.1145/2330784.2330799

Deliege, I. (1987). Grouping Conditions in Listening to Music: An Approach to Lerdahl & Jackendoff' s Grouping Preference Rules. *Music Perception*, *4*(4), 325–359. https://doi.org/10.2307/40285378

Dutch Song Database. (n.d.). Retrieved November 28, 2024, from https://www.liederenbank.nl/index.php?lan=en

Eerola, T., & Toiviainen, P. (2004). MIDI toolbox: MATLAB tools for music research. *ISMIR*. https://jyx.jyu.fi/bitstream/handle/123456789/49277/1/951-39-1795-9.pdf

Esposito, A., & Aversano, G. (2005). Text independent methods for speech segmentation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *3445 LNAI*, 261–290. https://doi.org/10.1007/11520153_12

Frankland, B., & Cohen, A. (2004). Parsing of melody: Quantification and testing of the local grouping rules of Lerdahl and Jackendoff's A Generative Theory of Tonal Music. *Music Perception*, *4*(21), 499–543. https://online.ucpress.edu/mp/article-abstract/21/4/499/62162

Friberg, A., Bresin, R., Frydén, L., & Sundberg, J. (1998). Musical punctuation on the microlevel: Automatic identification and performance of small melodic units. *Journal of New Music Research*, *27*(3), 271–292. https://doi.org/10.1080/09298219808570749

Goldberg, D. E., Booker, L. B., & Holland, J. H. (1989). *Goldberg, D. E. (1989). "Genetic Algorithms in Search,... - Google Scholar*. Artificial Intelligence. https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=Goldberg%2C+D.+E.+%281989%29.+%22Genetic+Algorithms+in+Search%2C+Optimization%2C+and+Machine+Learning.%22&btnG=

Goto, M., Hashiguchi, H., Nishimura, T., & Oka, R. (2003). *RWC music database: Music genre database and musical instrument sound database*. https://jscholarship.library.jhu.edu/bitstream/1774.2/36/1/paper.pdf

Guan, Y., Zhao, J., Qiu, Y., Zhang, Z., & Xia, G. (2018). Melodic Phrase Segmentation By Deep Neural Networks. *ArXiv Preprint ArXiv:1811.05688*. http://arxiv.org/abs/1811.05688

Harkin, T. (2022). Creating a Linked Data thesaurus for Irish traditional music. *AI and Society*, *37*(3), 967–974. https://doi.org/10.1007/S00146-021-01366-Y/FIGURES/2

Helmut, S. (1995). The Essen Folksong Collection in Kern Format [computer database]. In *D. Huron (ed.). Menlo Park, CA: Center for Computer Assisted Research in the Humanities*. Menlo Park, CA. Center for Computer Assisted Research in the Humanities. Available at: http://ota. ahds. ac. uk/headers/1038. xml [Accessed January 2009].

Hirai, T., & Sawada, S. (2019). Melody2Vec: Distributed representations of melodic phrases based on melody segmentation. *Journal of Information Processing*, *27*, 278–286. https://doi.org/10.2197/ipsjjip.27.278

Höthker, K., Thom, B., & Spevak, C. (n.d.). *Melodic segmentation: evaluating the performance of algorithms and musical experts*. www.ira.

Höthker, K., Thom, B., & Spevak, C. (2002). *Melodic segmentation: evaluating the performance of algorithms and musical experts*. Univ., Fak. für Informatik, Bibliothek. https://scholar.archive.org/work/5pwifgwhrja45ltl5bri4jonem/access/wayback/https://publikationen.bibliothek.kit.edu/20982002/1604

Janssen, B., de Haas, W. B., Volk, A., & van Kranenburg, P. (2014). Finding repeated patterns in music: State of knowledge, challenges, perspectives. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8905*, 277–297. https://doi.org/10.1007/978-3-319-12976-1_18

Karaosmanoğlu, M. K., Bozkurt, B., Holzapfel, A., & Dişiaçık, N. D. (2014). A symbolic dataset of Turkish makam music phrases. *Proceedings of 4th International Workshop on Folk Music Analysis (FMA 2014)*, 10–14. https://www.academia.edu/download/51550845/Turkish_Folk_Music_Phonetic_Notation_System_CantOvation_Sing___See.pdf#page=17

Kreuk, F., Keshet, J., & Adi, Y. (2020). Self-supervised contrastive learning for unsupervised phoneme segmentation. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, *2020-October*, 3700–3704. https://doi.org/10.21437/Interspeech.2020-2398

Lattner, S., Grachten, M., Agres, K., & Chacón, C. E. C. (2015). Probabilistic segmentation of musical sequences using restricted boltzmann machines. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9110*, 323–334. https://doi.org/10.1007/978-3-319-20603-5_33

Refferences

Lazzari, N., Poltronieri, A., & Presutti, V. (2023). Pitchclass2vec: Symbolic Music Structure Segmentation with Chord Embeddings. *ArXiv Preprint ArXiv:2303.15306*. http://arxiv.org/abs/2303.15306

Learning, M. B.-M., & 1999, undefined. (1999). An efficient, probabilistically sound algorithm for segmentation and word discovery. *SpringerMR BrentMachine Learning, 1999•Springer*. https://link.springer.com/article/10.1023/A:1007541817488

Lerdahl, Fred., & Jackendoff, Ray. (1983). *A generative theory of tonal music*. MIT Press.

Lukashevich, H. (2008). Towards Quantitative Measures of Evaluating Song Segmentation. *ISMIR, 2008*. https://books.google.com/books?hl=en&lr=&id=OHp3sRnZD-oC&oi=fnd&pg=PA375&dq=TOWARDS+QUANTITATIVE+MEASURES+OF+EVALUATING+SONG+SEGMENTATION&ots=oHNKtFjD91&sig=KIH4DaVvYoz8WEbahfhjChUY1ZI

Manning, C., & Schutze, H. (1999). *Foundations of statistical natural language processing*. https://books.google.com/books?hl=en&lr=&id=YiFDxbEX3SUC&oi=fnd&pg=PR16&dq=Manning,+C.D.,+Sch%C2%A8utze,+H.:+Foundations+of+Statistical+Natural+Language+Processing&ots=v0thnCiEUL&sig=or3GKVvC5yLNHXbugLiYJ0MCLi8

Michel, P., Rasanen, O., Thiollière, R., & Dupoux, E. (2017). Blind phoneme segmentation with temporal prediction errors. *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop*, 62–68. https://doi.org/10.18653/v1/P17-3011

Mikolov, T., Sutskever, I., … K. C.-A. in neural, & 2013, undefined. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*. https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html

Müller, M. (2021). Fundamentals of music processing: Using Python and Jupyter notebooks. *Fundamentals of Music Processing: Using Python and Jupyter Notebooks*, 1–495. https://doi.org/10.1007/978-3-030-69808-9/COVER

Narmour, E. (1990). *The analysis and cognition of basic melodic structures: The implication-realization model.* https://psycnet.apa.org/record/1991-97492-000

Narmour, E. (1992). *The analysis and cognition of melodic complexity: The implication-realization model*. https://books.google.com/books?hl=en&lr=&id=vfbwkoJFxQUC&oi=fnd&pg=PP13&dq=Narmour,+E.:+The+Analysis+and+Cognition+of+Melodic+Complexity:+The+Implication realisation+Model&ots=iYFnBBqMYx&sig=zmN1pjgtdsyvuyy44SgUj6nQHTM

Nettl, B. (2005). *The study of ethnomusicology: Thirty-one issues and concepts*. https://books.google.com/books?hl=en&lr=&id=hXPDDA6H5GsC&oi=fnd&pg=PP1&dq=the+study+of+ethnomusicology,+bruno+nettle&ots=yoeO6rR7e9&sig=nFnn246PHGbnlREhjU-SHpjIXZI

Nieto, O. (2015). *Discovering structure in music: Automatic approaches and perceptual evaluations*. https://search.proquest.com/openview/09f67403121bcbc7d2ee431985bf0568/1?pq-origsite=gscholar&cbl=18750

Nieto, O., & Bello, J. (2016). Systematic exploration of computational music structure research. *ISMIR*. https://ccrma.stanford.edu/~urinieto/MARL/publications/NietoBello-ISMIR16-slides.pdf

Pearce, M., Müllensiefen, D., ISMIR, G. W.-, & 2008, undefined. (n.d.). A Comparison of Statistical and Rule-Based Models of Melodic Segmentation. *Books.Google.ComMT Pearce, D Müllensiefen, GA WigginsISMIR, 2008•books.Google.Com*. Retrieved November 21, 2024, from https://books.google.com/books?hl=en&lr=&id=OHp3sRnZD-oC&oi=fnd&pg=PA89&dq=A+COMPARISON+OF+STATISTICAL+AND+RULE-BASED+MODELS+OF+MELODIC+SEGMENTATION&ots=oHMLpLiEa3&sig=RCW0O0-DzgrPQYsvmLqqdcMvhhs

Pearce, M. T., Müllensiefen, D., & Wiggins, G. A. (2010). Melodic grouping in music information retrieval: New methods and applications. *Studies in Computational Intelligence*, *274*, 365–389. https://doi.org/10.1007/978-3-642-11674-2_16

Peretz, I. (1989). CLUSTERING IN MUSIC: AN APPRAISAL OF TASK FACTORS. *International Journal of Psychology*, *24*(1–5), 157–178. https://doi.org/10.1080/00207594.1989.10600040

Raffel, C., Mcfee, B., Humphrey, E. J., Salamon, J., Nieto, O., Liang, D., & Ellis, D. P. W. (2014). mir_eval: A TRANSPARENT IMPLEMENTATION OF COMMON MIR METRICS. *ISMIR*.

Räsänen, O. J., Laine, U. K., & Altosaar, T. (2009). An improved speech segmentation quality measure: The R-value. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 1851–1854. https://doi.org/10.21437/interspeech.2009-538

Sciences, M. B.-T. in C., & 1999, undefined. (1999). Speech segmentation and word discovery: A computational perspective. *Cell.ComMR BrentTrends in Cognitive Sciences, 1999•cell.Com*. https://www.cell.com/AJHG/fulltext/S1364-6613(99)01350-9

Seneff, S. (1992). TINA: A natural language system for spoken language applications. *Aclanthology.OrgS SeneffComputational Linguistics, 1992•aclanthology.Org*. https://aclanthology.org/J92-1004.pdf

Temperley, D. (2004). *The cognition of basic musical structures*. https://books.google.com/books?hl=en&lr=&id=IDoLEvTQuewC&oi=fnd&pg=PR9&dq=D.+Temperley,+The+Cognition+of+Basic+Musical+Structures.+MIT+Press,+2001.&ots=wN24v9rrBR&sig=vu7zF9rFOVR8ErDiBetj5PXe7qM

Tenney, J., & Polansky, L. (1980). Temporal gestalt perception in music. *Journal of Music Theory, JSTORJ*. https://www.jstor.org/stable/843503

Refferences

The Humdrum Toolkit for Computational Music Analysis | Humdrum. (n.d.). Retrieved November 28, 2024, from https://www.humdrum.org/index.html

The Lakh MIDI Dataset v0.1. (n.d.). Retrieved November 28, 2024, from https://colinraffel.com/projects/lmd/

Toiviainen, P., & Eerola, T. (2006). Autocorrelation in meter induction: The role of accent structure. *The Journal of the Acoustical Society of America*, *119*(2), 1164–1170. https://pubs.aip.org/asa/jasa/article/119/2/1164/830063

van Kranenburg, P. (2020). Rule mining for local boundary detection in melodies. *Proceedings of the 21st International Society for Music, 2020*. https://program.ismir2020.net/static/final_papers/226.pdf

van Kranenburg, P., de Bruin, M., & Volk, A. (2019). Documenting a song culture: the Dutch Song Database as a resource for musicological research. *International Journal on Digital Libraries*, *20*(1), 13–23. https://doi.org/10.1007/S00799-017-0228-4

Wang, C. I., & Dubnov, S. (2015). Pattern discovery from audio recordings by Variable Markov Oracle: A music information dynamics approach. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *2015-August*, 683–687. https://doi.org/10.1109/ICASSP.2015.7178056

Wang, C., & Mysore, G. (2016). Structural segmentation with the variable markov oracle and boundary adjustment. *IEEE International Conference on Acoustics, Speech and Signal*. https://ieeexplore.ieee.org/abstract/document/7471683/

Wang, C.-I., & Mysore, G. J. (2016). Structural segmentation with the variable Markov oracle and boundary adjustment. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Zhang, Y., & Xia, G. (2021). Symbolic Melody Phrase Segmentation Using Neural Network with Conditional Random Field. *Lecture Notes in Electrical Engineering*, *761 LNEE*, 55–65. https://doi.org/10.1007/978-981-16-1649-5_5