Deep Learning Techniques for Music Generation (1)

Jean-Pierre Briot

Jean-Pierre.Briot@lip6.fr

Laboratoire d'Informatique de Paris 6 (LIP6)

Sorbonne Université – CNRS

Programa de Pós-Graduação em Informática (PPGI)

UNIRIO

Deep Learning for Music Generation

- Survey and Analysis of various deep learning-based music generation systems
- Very active domain
  - Ex: Google Magenta Project
  - Spotify Creator Technology Research Lab (CTRL)
  - Also image and text generation
- Objective: Understand and analyze (classify) various existing approaches and challenges/issues
- Challenge for deep learning, designed for prediction and classification, and not for generation, and furthermore not for creative and controlled generation
- Joint work with Gaëtan Hadjeres (Sony CSL) and François Pachet (Spotify CTRL)

Objective

- Show that Deep Learning Techniques for Music Generation are getting Mature
- Show that Control Issues are Important and Difficult
  - and with some Partial Solutions
- Show that Creativity vs Plagiarism Issues are Important and Difficult
  - and with some Partial Solutions

Outline

- Computer Music and Algorithmic Composition
- Deep Learning and Neural Networks
- Deep Learning for Generating Music
  - Challenges
- Feedforward
- Recurrent
- Variability by Sampling
- Control by Input Manipulation
  - Ex: Deep Dream, Style Transfer, C-RBM
- Control by Reinforcement (RL-Tuner)
- Creativity (CAN)
- Interactivity (Ex: DeepBach)
- Survey/Analysis
- Conclusion

Book (Soon Available)

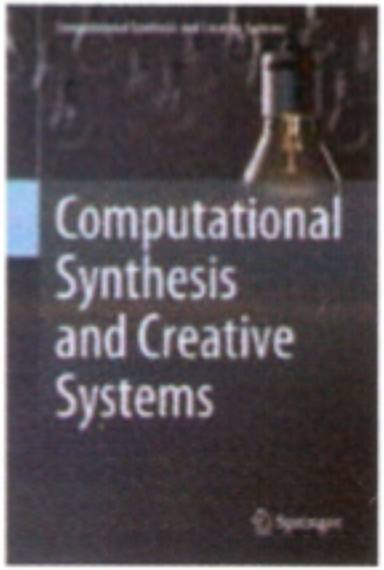
Springer

Deep Learning Techniques for Music Generation

Jean-Pierre Briot [UPMC, Paris and PUC, Rio de Janeiro]
Gaëtan Hadjeres [Spotify Creator Technology Research Lab]
François Pachet [Spotify Creator Technology Research Lab and UPMC, Paris]

ISBN: 978-3-319-70162-2; ca. 01/2018; ca. 140 pp.
(http://www.springer.com/978-3-319-70162-2)

Series: Computational Synthesis and Creative Systems
(http://www.springer.com/series/15219)



arXiv Preliminary Version (September 2017)

Deep Learning Techniques for Music Generation – A Survey

Jean-Pierre Briot, Gaëtan Hadjeres, François Pachet

(Submitted on 5 Sep 2017)

This book is a survey and an analysis of different ways of using deep learning (deep artificial neural networks) to generate musical content. At first, we propose a methodology based on four dimensions for our analysis: – objective – What musical content is to be generated? (e.g., melody, accompaniment...); – representation – What are the information formats used for the corpus and for the expected generated output? (e.g., MIDI, piano roll, text...); – architecture – What type of deep neural network is to be used? (e.g., recurrent network, autoencoder, generative adversarial networks...); – strategy – How to model and control the process of generation (e.g., direct feedforward, sampling, unit selection...). For each dimension, we conduct a comparative analysis of various models and techniques. For the strategy dimension, we propose some tentative typology of possible approaches and mechanisms. This classification is bottom-up, based on the analysis of many existing deep-learning based systems for music generation, which are described in this book. The last part of the book includes discussion and prospects.

Comments:	108 pages
Subjects:	**Sound (cs.SD)**; Learning (cs.LG)
Cite as:	arXiv:1709.01620 [cs.SD]
	(or arXiv:1709.01620v1 [cs.SD] for this version)

Computer Music

Why Using Computer for Music

- Bad Reasons (Fears)
  - Lead human musicians to unemployment
  - Lower the quality of music

- Good reasons
  - Facilitate storing, indexing, delivering and sharing of music (MIDI, MP3, Spotify…)
  - New instruments and interaction (Synthesizers, Interactive music performances…)
  - New sounds (Synthesizers and Signal processing)
  - Analysis tools and algorithms (Spectrum, Patterns…)
  - Initiation and Education (Band in the box, Garage Band…)

- Production
  - Partially automate tasks (mixing, etc.)
- Composition, Analysis and Arrangement
  - Algorithmic composition
  - Harmonization
  - Analysis

Why Using Computer for Music

- Vast Associative Memory
  - More systematic than Human memory
- Representation of Musical pieces, Style, Patterns…
- Associations and Correlations
- Knowledge (Theory, Rules, Heuristics…)

- Can Help Human musicians

- Human musicians rarely compose from scratch – They steal from others
  - Consciously
    - Plagiat, Citation…
  - Unconsciously
    - Influence
  - Recombinations
  - Historical Evolution
    - Modal monophonic -> Polyphonic (Counterpoint) -> Tonal Music (Harmony) -> Extended Harmony (Debussy, Jazz…)
  - Ruptures (Dodecaphonism, Free Jazz)
    - Rare and often transient

What for Using Computer for Music

- Autonomous Artificial Musicians
  - Audio music generation
  - Ex: Amper Music, Jukedeck…
  - For Commercials and Documentaries
  - Create Royalty-free or Copyright-buyable Music
  - Based on Deep learning + Samples + Sound processing techniques

  \+ Business model

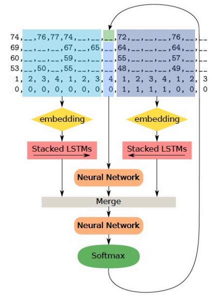
  – Musical model

What for Using Computer for Music

- Autonomous Artificial Musicians

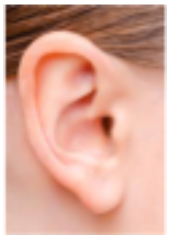
- Music Composition Turing test
  - Imitation Game Scenario
  - Designed by A. Turing to explore the question "Can Machines think?"

(A) J. S. Bach



(B) DeepBach [Hadjeres et al., 2017]

?



(C) Listener

- To evaluate artificial composers techniques
- To explore music cognition

A. M. Turing (1950) Computing Machinery and Intelligence. *Mind 49:* 433-460.

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. The Imitation Game

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous, If the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the 'imitation game." It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart front the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either "X is A and Y is B" or "X is B and Y is A." The interrogator is allowed to put questions to A and B thus:

C: Will X please tell me the length of his or her hair?

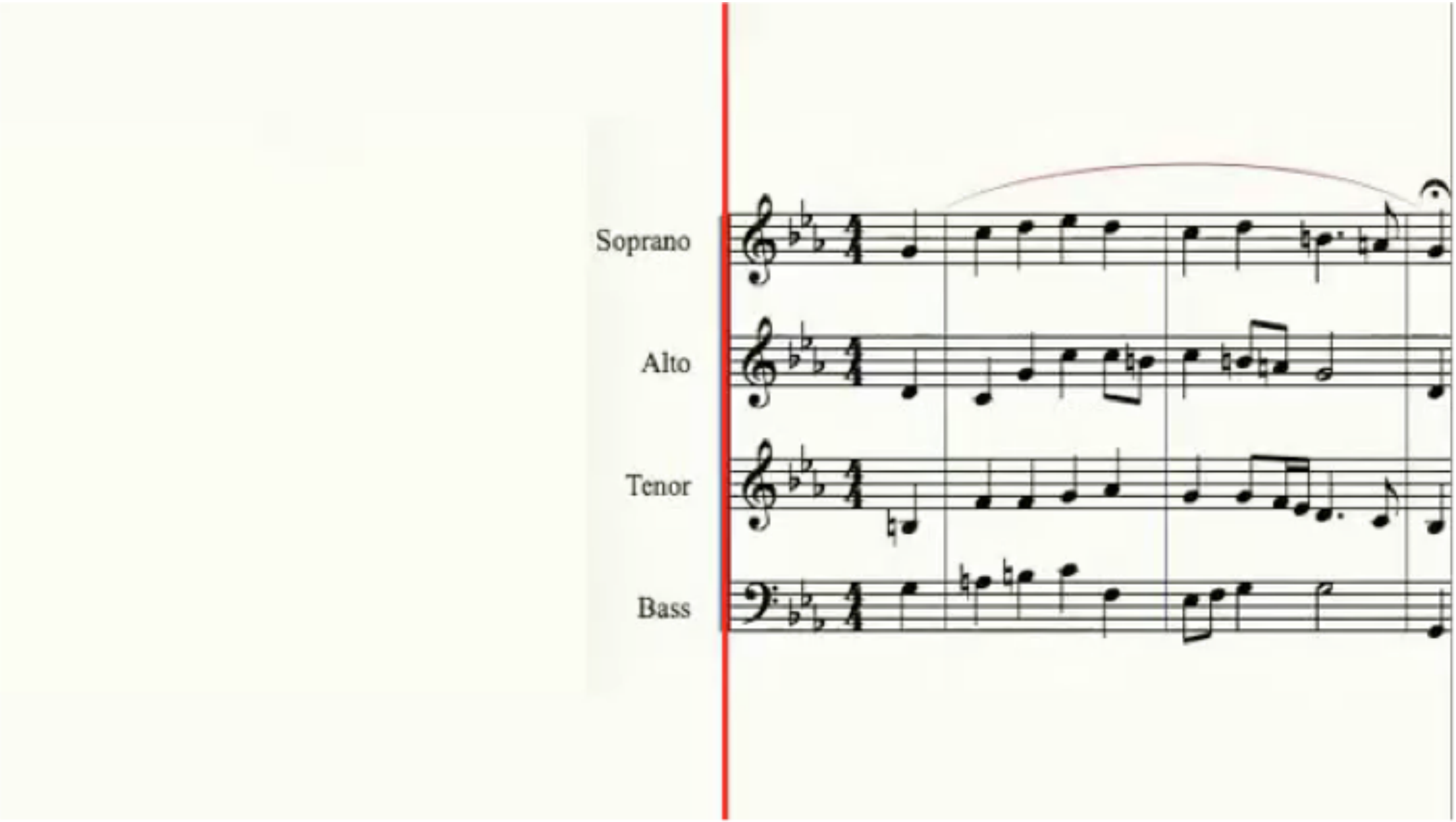
Now suppose X is actually A, then A must answer. It is A's object in the game to try and cause C to make the wrong identification. His answer might therefore be:

"My hair is shingled, and the longest strands are about nine inches long."

In order that tones of voice may not help the interrogator the answers should be written, or better still, typewritten. The ideal arrangement is to have a teleprinter communicating between the two rooms. Alternatively the question and answers can be repeated by an intermediary. The object of the game for the third player (B) is to help the interrogator. The best strategy for her is probably to give truthful answers. She can add such things as "I am the woman, don't listen to him!" to her answers, but it will avail nothing as the man can make similar remarks.

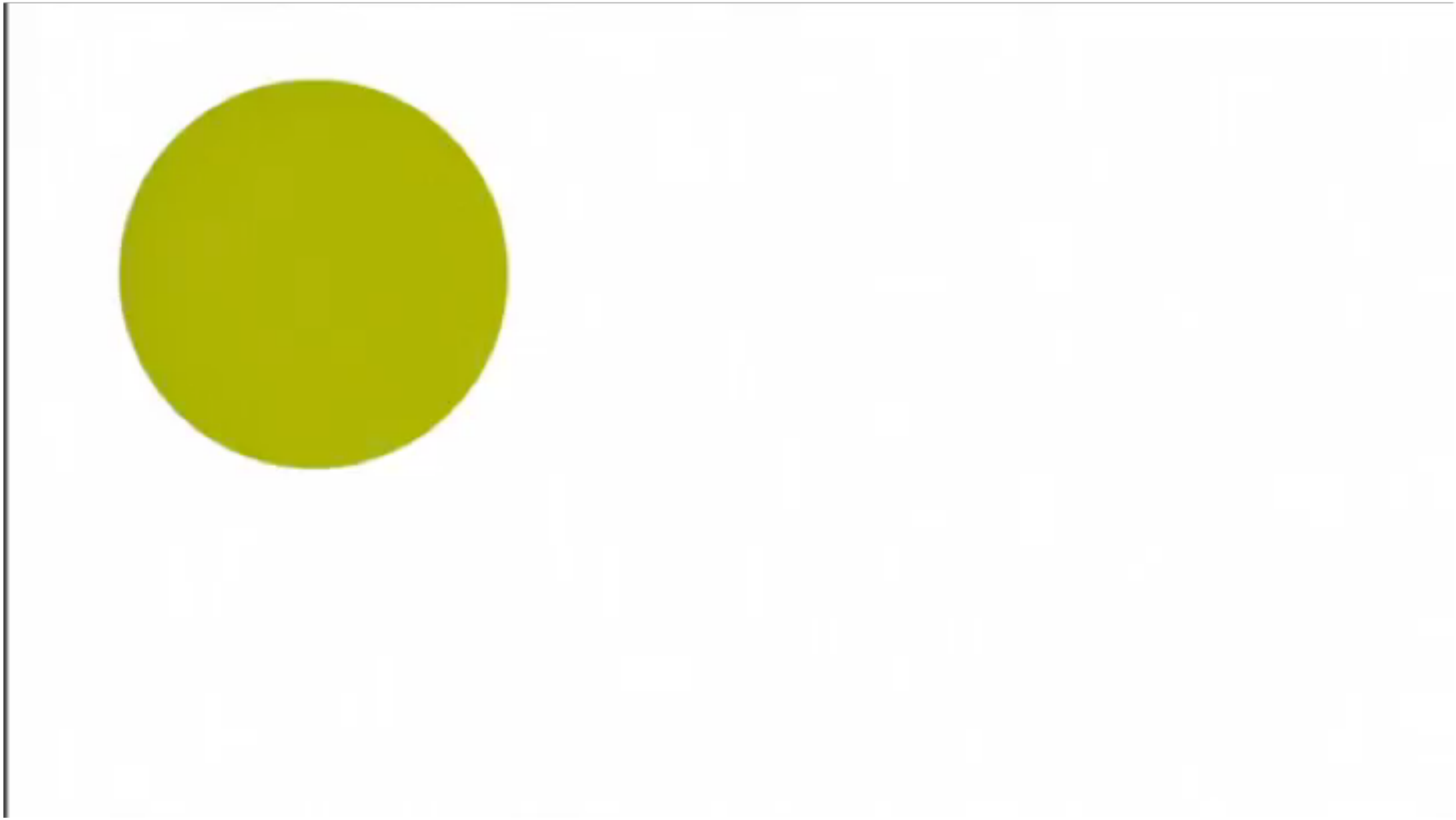
We now ask the question, "What will happen when a machine takes the part of A in this game?" Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, "Can machines think?"

DeepBach [Hadjeres et al., 2017] – Demo



https://www.youtube.com/watch?v=QiBM7-5hA6o

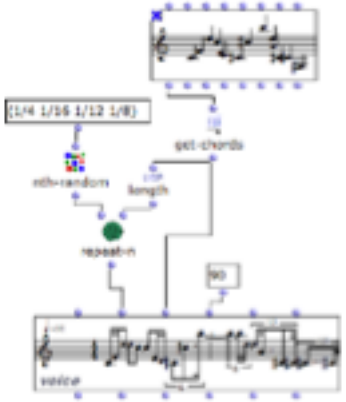
Turing Bach Test



https://www.youtube.com/watch?v=DPNHbtiXGEM

What for Using Computer for Music

- Expert Musician Algorithmic Composition or/and Real-Time Production Environment
- Ex: OpenMusic, Max/MSP…
  - Musical Concepts
  - Algorithmic Composition
  - Tool-Box
  - Composition vs Interaction
  - Symbolic vs Signal Processing



- Average Musician Music Production and Composition Environments
- Ex: GarageBand, Band-in-a-Box…
  - Little Memory
  - Little Knowledge

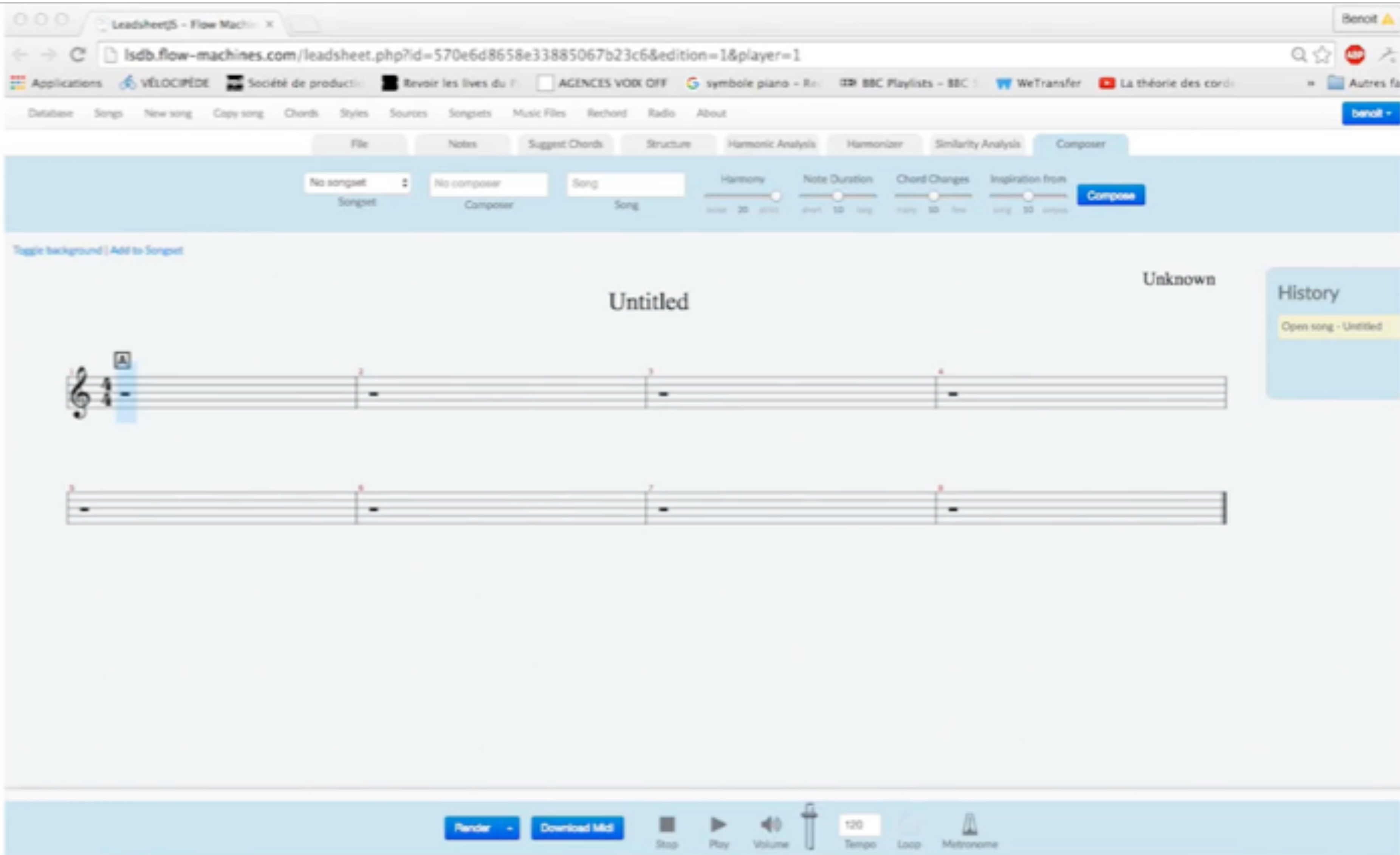
What for Using Computer for Music

- Human Musician Composition and Production Environment Assistants
- Ex: FlowComposer…

  - Propose
  - Complete
  - Refine
  - Analyze
  - Harmonize
  - Transpose
  - Adapt
  - Produce

  - Wide Memory (FlowComposer: more than 12.000 songs/lead sheets)
  - Harmony knowledge (ex: Hamonic analysis, Harmonization, Chord substitution…)

FlowComposer – Demo



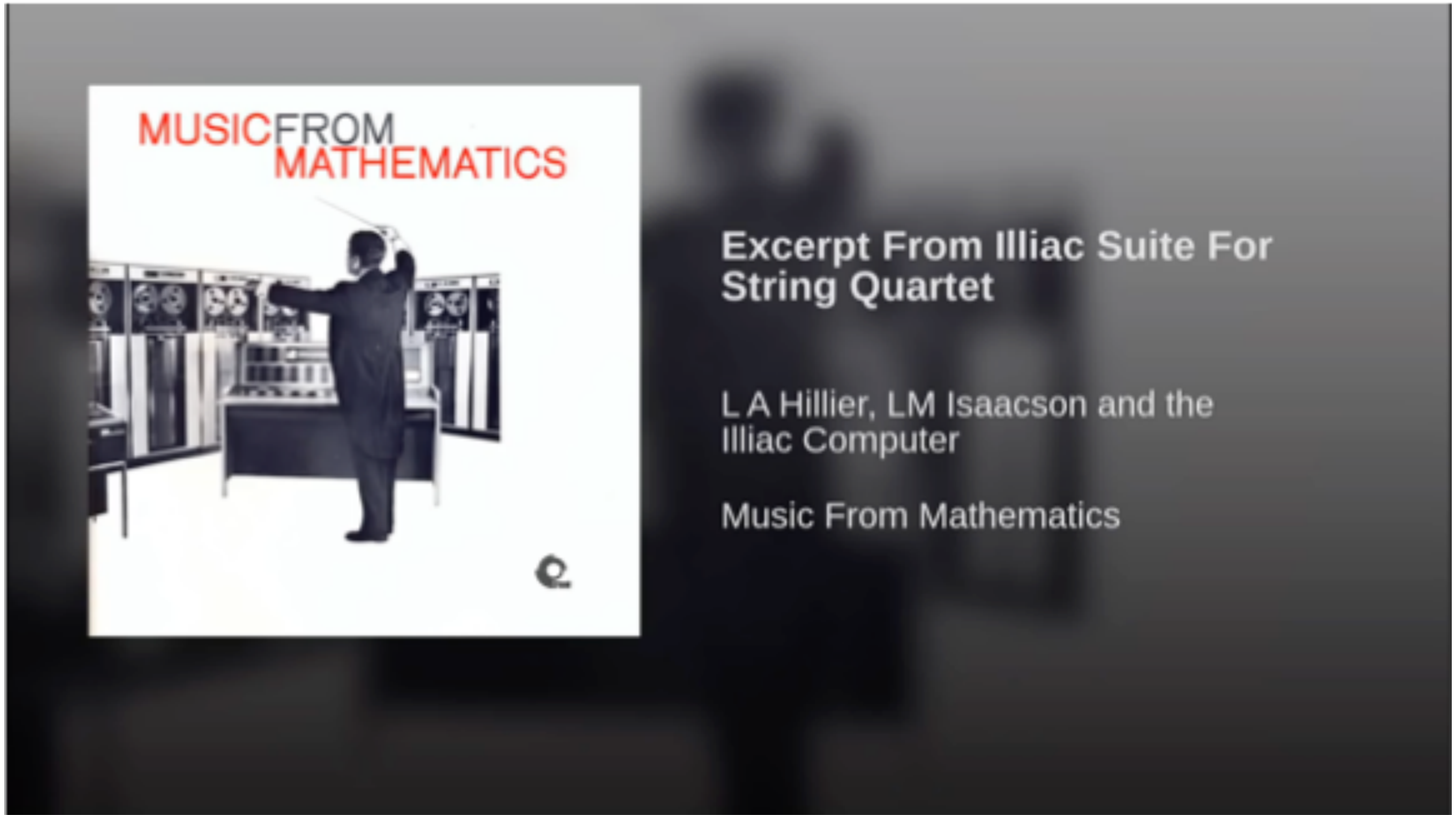
https://www.youtube.com/watch?time_continue=5&v=SDnkX8v8caY

Symbolic vs Audio

- Symbolic
  - Conceptual Level
  - Composition Level
  - Symbolic Representations
  - Knowledge Representation (Symbolic AI)
  - Note, Chord, Rest…
  - Harmony, Modes…
  - Digital
- Audio
  - Signal Level
  - Signal Processing
  - Spectrum, Fourier Transform, MFCC…
  - Analogic
- Complementarity
- Integration challenge

Computer Music

- Started in the 50's
- ILLIAC Suite [Hiller & Isaacson, 1957]



https://www.youtube.com/watch?v=v5z3t2qz4go

An example of Stochastic Algorithmic Music

- Generate and Test
- Random Generation
- Handcrafted Filtering Rules

- Generation of Pieces of Music Matching the Rules (Constraints)
- (Inefficient Constraint Solving Strategy)

Also (Ex):
- Zenakis Stochastic Compositions

- Musikalisches Würfelspiel (Musical Dice Game) by Mozart
  - Stochastic Combination of pre-Existing 2 measure-long Musical Segments
  - Fixed Style (Austrian Waltz) and Tonality/Key
  - Expertise of the Composer

https://mozart.qvwx.de/

Mozart Dice Music (Ex of Piece Generated)

Pre-Existing Models (Examples)

- Elements
  - Combination (ex: Mozart Dice Music)
- Rules
  - Application
  - Filtering (ILIAC Suite), Generation, Harmonic Analysis…
- Generative Grammars
  - Valid Sentences generated by the Grammar
  - Harmonic Cadences Construction, Substitutions…
- Constraints
  - Constraint Solving Problem
  - Generation (ILIAC Suite), Accompaniment: Harmonization, Counterpoint…
- Markov Chains
  - Generation (Random Walk, Constrained)
  - Style Imitation

Pre-Existing Models (Examples)

- Cellular Automata
  - Evolution
- Genetic Algorithms
  - Selection
- Case-based Reasoning
  - Similarity and Adaptation
- Planning
  - Path (Melody, Chord Sequence…) Construction
- …

Importance of Randomness (Stochasticity)

- Randomness/Stochasticity
  - Governed by Probabibilities
- To avoid Determinism
  - Otherwise Infinitely Repeats the same Generation
- To be able to Generate Various Musical Pieces from a single Generative Model

- Ex: Mozart Dice-Music, Xenakis…
- Incorporated in most of Algorithmic Compositions

Reorchestration of Ode of Joy Using Various Techniques (Flow Machines)

Ode to Joy in several styles

https://www.youtube.com/watch?v=buXqNqBFd6E

(Simplistic) Example of Generative Grammar for Blues

8-Measures	->	First-4-Measures Second-4-Measures
First-4-Measures	->	Opening-Cadence Opening-Cadence
	->	Opening-Cadence’ Opening-Cadence
Second-4-Measures	->	Middle-Cadence Opening-Cadence
Opening-Cadence	->	| I | I |
	->	| I | V |
Opening-Cadence’	->	| I | III |
	->	| I | IV |
Middle-Cadence	->	| I | IV |
	->	| I | V |
	->	| IV | I |

[John-Laird, 1991]

Ex:

| I | I | I | I | IV | I | I | V |

| I | IV | I | V | IV | I | I | I |

Acquiring Models

- Handcrafted
  - Tedious
  - Error-Prone

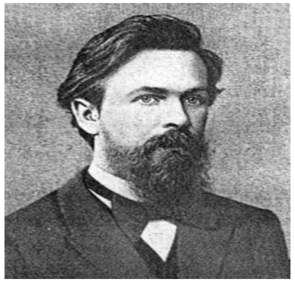
- Automatically Learnt (Induction)
- Ex:
- Markov models
- Neural models

- Style Automatic Learned from a Corpus (Composer, Form, Genre…)

- Machine Learning Techniques
  - Neural Networks, Deep Learning, Reinforcement Learning (and other models/techniques)

Markov Chains

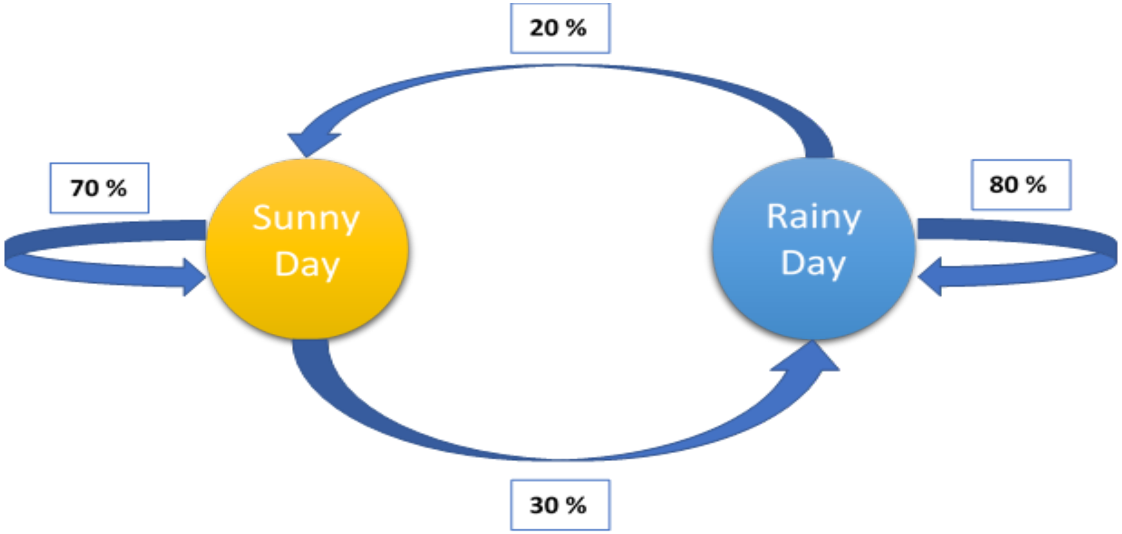
- Hypothesis (Markov)
- Past is unnecessary to model the future



- $P(s_{t+1} \mid s_1, s_2 \ldots, s_t) = P(s_{t+1} \mid s_t)$

- States
- Probabilities for transition to other (or same) State

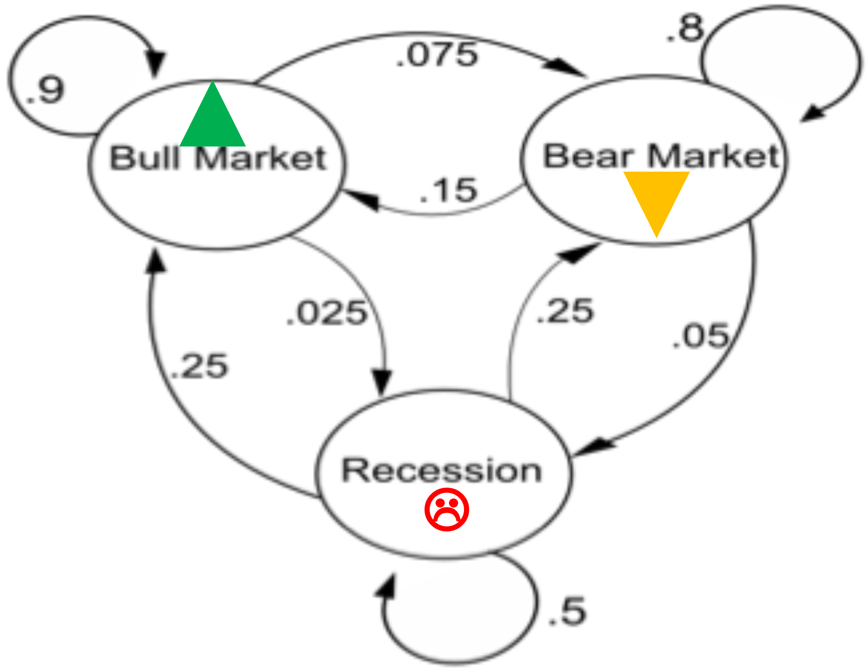
- Ex: Weather Modeling



Markov Chains

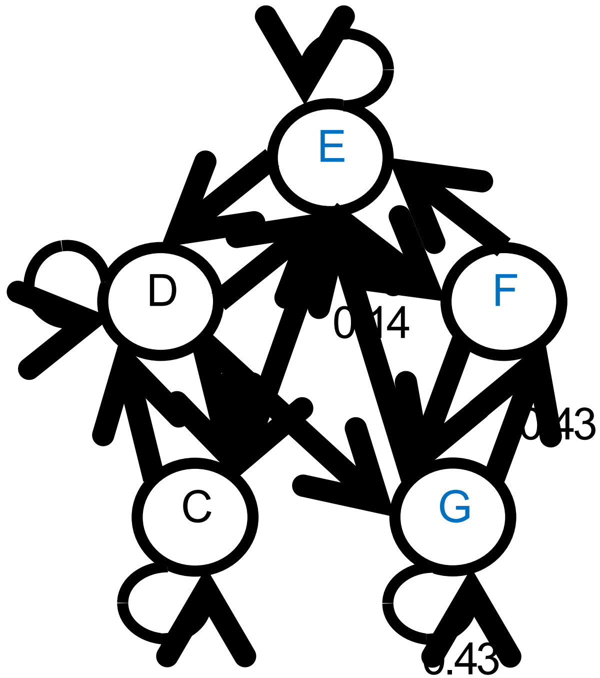
- Other Example:

- Modeling of Market Evolution ▲▼



Modeling of Beethoven’s Ode of Joy

- Beethoven "Ode to Joy" (9th Symphony)



What note follows a G ? $P(n_{t+1}=X \mid n_t=G)$

- G -> E 1 times : 0.14
- G -> F 3 times : 0.43
- G -> G 3 times : 0.43

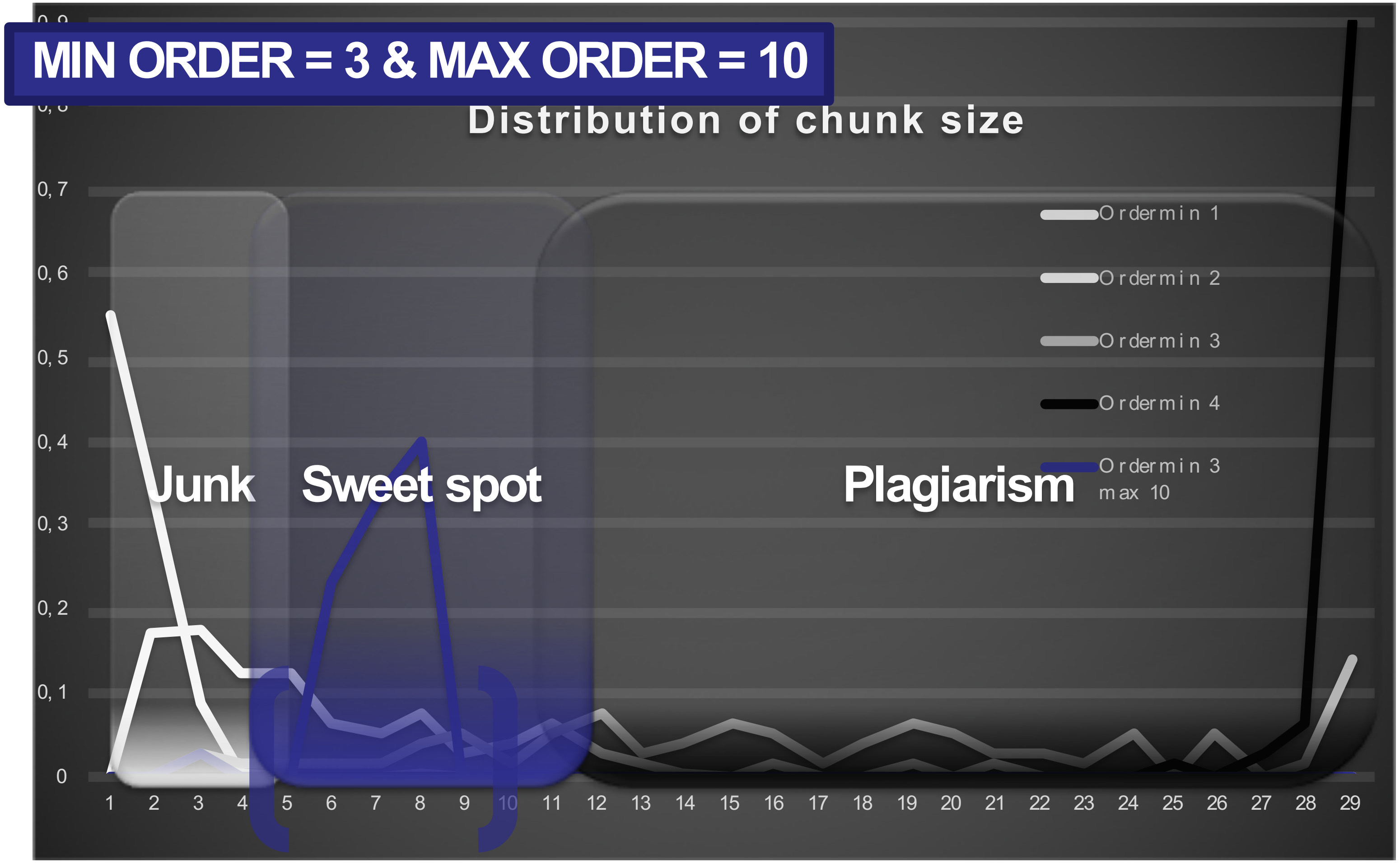
Higher-Order Markov

- Order 1: $P(s_{t+1} \mid s_1, s_2 \ldots, s_t) = P(s_{t+1} \mid s_t)$
- Order 2: $P(s_{t+1} \mid s_1, s_2 \ldots, s_t) = P(s_{t+1} \mid s_{t-1}, s_t)$
- Order 3: $P(s_{t+1} \mid s_1, s_2 \ldots, s_t) = P(s_{t+1} \mid s_{t-2}, s_{t-1}, s_t)$
- …

- Variable Order

- Limit: Plagiat

Higher-Order Markov



MaxOrder Constraint

[Roy and Pachet, 2017]

Example of Interactive Markov Chain System: Continuator [Pachet, 2002]

https://www.youtube.com/watch?v=ynPWOMzossI

Pianist: Albert Veenendaal