

# Reconstituted Bach: Generative Music from Hierarchical N-Gram Analysis

*Chris Thornton*

October 30, 2009

Generative music composition often applies some form of Markov analysis to examples, with output then being generated through Markov chaining. But the n-gram evaluation at the heart of this approach promotes models which better reflect low-level, note-transition structure than high-level structure. In the talk I'll describe a hierarchical adaptation of the approach which aims to be more responsive to high-level/coarse-grained structure. I'll also demo a music-generation system which shows what happens when the approach is applied to (snippets of) Bach preludes.

# Empirical music generation

Music-generative methods often use a combination of Markov modeling and Markov chaining.

But a pure Markov approach captures the low-level (e.g., note-transition) structure better than than high-level.

'Wandering melody' problem

What's the best way to capture structure at different levels?

What's the best way to capture hierarchical structure?

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively'?

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively'?

- Apply n-gram analysis to source data

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively'?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively' ?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis
- Treat second-level n-grams as constituents for a third level etc.

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively' ?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis
- Treat second-level n-grams as constituents for a third level etc.

But...

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively' ?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis
- Treat second-level n-grams as constituents for a third level etc.

But...

- Each stage generates more entities than there are to start with.

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively' ?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis
- Treat second-level n-grams as constituents for a third level etc.

But...

- Each stage generates more entities than there are to start with.
- They do not form a sequence.

# Hierarchical n-gram analysis?

Can n-gram analysis be used to identify hierarchical structure?

Use the method 'recursively' ?

- Apply n-gram analysis to source data
- Treat n-grams as constituents for a second level of analysis
- Treat second-level n-grams as constituents for a third level etc.

But...

- Each stage generates more entities than there are to start with.
- They do not form a sequence.

# N-gram encodings

Use information theory to organize n-grams as an encoding.

Following ordinary analysis, find the n-gram sequence which provides the most informative encoding of the source.

Apply next round of analysis to the encoding, then treat the derived encoding the same way, and so on.

This requires an adaptive-n approach and works better if we let the analysis use disjunctive n-grams.

# Sequence reduction

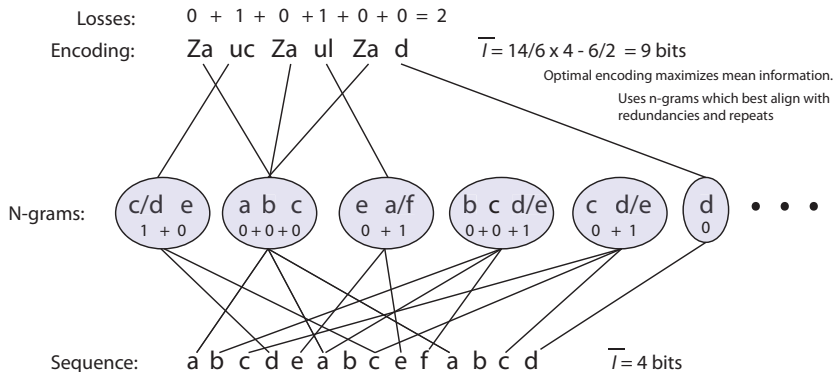


Figure:

# Information profiles

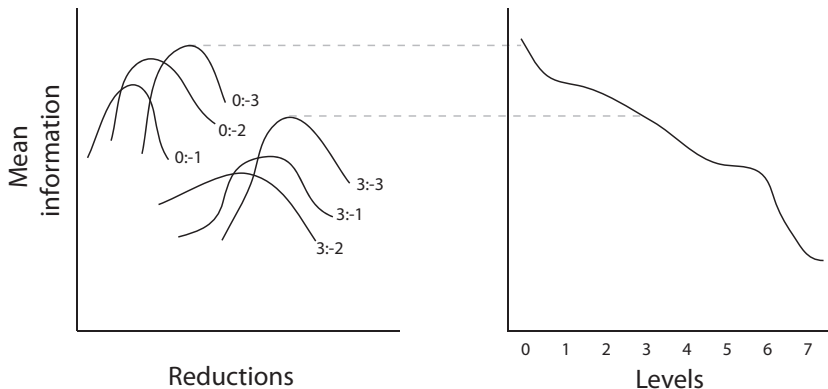


Figure:

With this approach we can't generate output through Markov chaining.

Since information hierarchies are AND/OR trees, they can be used like phrase-structure grammars for generative purposes.

This involves recursively expanding the top-level symbol, randomly selecting alternatives where there is disjunction.

Outputs generated this way are 'reconstitutions' or 'replexes' of the original sequence.

How well does this work as a generative method for music?

# Replexer applet

archive=Replexer.jar

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

- Sequence analysis?

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

- Sequence analysis?
- Data compression?

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

- Sequence analysis?
- Data compression?
- Learning?

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

- Sequence analysis?
- Data compression?
- Learning?
- Creativity?

Musical reflexes usually sound terrible. Sometimes they sound bearable. Very occasionally they sound interesting.

Is there any way to improve the process to get better results?

Also, how should the approach be viewed?

- Sequence analysis?
- Data compression?
- Learning?
- Creativity?

# Concluding remarks

Informational sequence reduction can capture and exploit some aspects of high-level structure.

Are they the aspects we want?

General question: is hierarchical structure always informationally evident?