SEARCH-EFFECTIVENESS MEASURES FOR SYMBOLIC MUSIC QUERIES IN VERY LARGE DATABASES

Craig Stuart Sapp
CCRMA and CCARH
Dept. of Music
Stanford University
Stanford, CA 94305-3076, USA

Yi-Wen Liu
CCRMA
Dept. of Electrical Engineering
Stanford University
Stanford, CA 94305, USA

Eleanor Selfridge-Field
CCARH
Depts. of Music, Symbolic Systems
Stanford University
Stanford, CA 94305-3076, USA

ABSTRACT

In the interest of establishing robust benchmarks for search efficiency, we conducted a series of tests on symbolic databases of musical incipits and themes taken from several diverse repertories. The results we report differ from existing studies in four respects: (1) the data quantity is much larger (*c*. 100,000 entries); (2) the levels of melodic and rhythmic precision are more refined; (3) anchored and unanchored searches were differentiated; and (4) results from joint pitch-and-rhythm searches were compared with those for pitch-only searches.

The search results were evaluated using a theoretical approach which seeks to rank the number of symbols required to achieve "sufficient uniqueness". How far into a melody must a search go in order to find an item which is unmatched by any other of the available items? How much does the answer depend on the specificity of the query? How much does anchoring the query matter? How much does the result depend on the nature of the repertory? We offer experimental results for these questions.

1. REPERTORY

The musical data used for analyses in this paper are derived from *Themefinder* [1] which contains a family of databases encoded in the Humdrum format. [2] Unlike MIDI data in which some pitch-data compared to graphical notation is ambiguous, this format provides explicit pitch and rhythm descriptions for all notes. This enables evaluation of the importance of distinguishing between enharmonic spellings (*e.g.,* G♯ vs. A♭) as well as the importance of a note's metric position.

The main purpose of the *Themefinder* website is to enable trained musicians to identify works by their melodies as remembered. Users are assumed to be notationally literate, since they are most likely to seek a work-title. That is, they are seeking textual metadata from a symbolic-data search. Results are viewable in notation and playable as corresponding MIDI files. The length of the queries is at the discretion of the user. The constituent collections (some publicly searchable, others limited to licensed use) each represent a different kind of music (Table 1).

	Dataset	Genre	Orig. Code	# Incipits
•	US RISM A/II	Instrumental, Vocal (17th–18th cents.)	Plaine & Easie	55,490
•	Renaissance (Italy)*	Motets (16th cent.)	DARMS	18,946
•	Classical*	Instr., Vocal	MIDI	10,718
•	Essen European*	Folksongs	EsAC	6,232
•	Polish religious monophony	Devotional songs, 16th, 19th cents.	EsAC	6,060
•	Essen Asian*	Folksongs (China)	EsAC	2,241
•	Luxembourg*	Folksongs	EsAC	612
	Total			100,299

Table 1. Constituent databases used for analysis. Starred items are publicly searchable.

The repertories vary substantially by musical mode. Some collections are tonal, some modal, and one is pentatonic. Within the tonal collections, significant range can be found with respect to diatonic, chromatic, and (occasionally) enharmonic usage. The Essen-European, Classical, and RISM datasets are overwhelmingly tonal. The Renaissance Italian database employs modes of the period; the Polish data contains two subsets and is almost evenly divided between modal and tonal monophony. The Essen-Asian dataset is pentatonic. Pentatonicism (the use of five tones per octave) makes scale-degree searches ambiguous, since mapping five-tone profiles onto seven-tone grids yields inevitable differences in scale-degree usage. Since we were interested in comparing procedures and their effectiveness in different repertories, we did not attempt to correct for this distortion. The Renaissance reper-

[1] http://www.themefinder.org
[2] http://dactyl.som.ohio-state.edu/Humdrum

tories subscribe to different systems of rhythmic organization than what is conventional in common music notation. This complicates the investigation of multiple tiers of precision in rhythmic definition. The classical dataset is overwhelmingly instrumental, while the Polish data is exclusively vocal.

None of the component datasets in *Themefinder* originated as Humdrum data, which represents pitch, duration, barring, and the global variables of notated music (meter signature, key signature, clef, *etc*.). However, All the repertories were originally encoded at a level of detail sufficient to support translation into Humdrum, or were significantly manually edited in the case of MIDI translations. The total number of records used in this study was 100,299.

The use of musical incipits raises fundamental questions of musical identity. All incipits are monophonic as used in *Themefinder*, but some of the underlying repertories are polyphonic. In relation to polyphonic music, Lincoln [6] gives one incipit for each voice (typically five). RISM gives the incipit (usually from the highest-pitched instrument or voice, *e.g.*, Violin 1 or Soprano) but is generous in giving separate incipits for linked portions of a single movement (*e.g.*, instrumental ritornello and aria).

A qualitative difference distinguishes incipits from themes. *Incipits* introduce a song, work, or movement. They serve well for short works that are uncomplicated. *Themes* represent a piece of music (usually a longer, more complex one) in some more essentially cognitive way. The mental extrapolation of themes is a human task and therefore vulnerable to subjective variation. As best we know, there is no study of the variation that might be found by multiple subjects in the identification of thematic material.

In constructing this analysis, we gave consideration to the relationship (or lack thereof) between incipits (in monophonic contexts) and overall pieces (we did not distinguish between incipits and themes *per se*). We subsequently compared search results for incipit data compared to full-work data for the Essen folksong collection (in which central European music predominates).

2. EXISTING STUDIES

In searching for related literature, we found few studies which were systematic in nature and which addressed substantial quantities of data, although many studies touched on some aspect of this general area of inquiry. McNab *et al.* [7], Dannenberg *et al.* [2], and Rand and Birmingham [8] explored similar procedures but in relation to a query-by-singing situation.

In [8] 188 MIDI files were used as a basis for profiling durational change, pitch change, and "note-drop" in an effort to simulate the kinds of user errors anticipated in sung input. They noted that correlation coefficients and count correlations performed equally well in processing and combined them in a modified scoring metric. In [2] one database of 2,844 items was generated from a MIDI collection of Beatles songs and a second, of 8,926 themes (averaging 41 notes), was based on an encoded collection of songs. Here they compared the results of metrics derived from (1) pitch plus inter-onset intervals (IOI), (2) melodic contour matches, and (3) Hidden Markov Models. In [7] a similar study to ours using the Essen database was conducted as part of research for symbolic-music retrieval via singing.

The closest parallels with our own work in purely symbolic searching are found in [4], [9], [10], and [5]. The first two were concerned with sorting records stored in related symbolic databases into a musical equivalent of alphabetical order for bibliographical purposes (*e.g.*, finding concordances for works which are anonymous or which are attributed to multiple composers). The authors of [10] sought to determine the feasibility of the query-by-humming approach by simulating some of the known deficiencies in user input. They considered melodic representation at five levels of pitch-resolution, although it is unclear exactly how "intervals" were defined at some levels (in the categories 3, 5, 7, 9, 12) of pitch resolution. They attempted to simulate different levels of inaccuracy in sung queries. They reported results for different database sizes (of 0.6, 1.2, 2.5, and 3.6 million notes) and search-key lengths (4–8 tokens). They found that a three-interval contour required a 1.7 longer query-length than a semitone resolution. They reported that 5-state, 7-state, and 9-state representations of the underlying melodies led to similar results but in all cases produced improvement over three-state representations. The authors of [5] examined the effectiveness of using rhythm and pitch of notes in parallel for search queries.

3. DATA FEATURES

Currently *Themefinder* allows searching for music examples by several levels of precision, going from very specific (exact pitch) to very general (gross contour). However, for this study the descriptive features were expanded to include rhythm as well to examine which rhythmic features may be of benefit when searching in *Themefinder*. Fourteen symbolic features of music were examined in this study—seven for pitch and seven for rhythm. Here is a list of the seven pitch features extracted from the musical data which are ordered from specific to general:

- p1 *Enharmonic pitch class*: Pitches are named by diatonic letter (A..G) and inflection: natural (♮), sharp (♯), flat (♭), double-sharp (x), or double-flat (♭♭).
- p2 *Musical interval*: The distance between two pitches specified by melodic direction, diatonic size (3rd, 5th, etc.), and "quality": perfect, major, minor, augmented, or diminished. Table 2 gives a list of the 35 basic intervals per octave considered.
- p3 *Twelve-tone pitch class*: 12 pitches a semitone apart per octave which map one-to-one on the musical keyboard.

p4 *Twelve-tone interval*: The distance between two successive twelve-tone pitches in terms of semitones.

p5 *Scale degree*: Pitches are described by their diatonic position in a 7-tone scale (*e.g.*, major or minor).

p6 *Gross contour*: Intervals are assigned to one of three categories by melodic direction (up, down, or the same).

p7 *Refined contour*: Intervals are assigned to one of five categories: 2 "steps" (pitch changes of any type of second interval) up or down, 2 "leaps" (pitch changes of minor thirds or greater) up or down, or unchanged pitch.

name	code	name	code	name	code	name	code
C♮	0			per1	0	per8	40
C♯	1	C♭♭	39	aug1	1	dim8	39
Cx	2	C♭	38	aaug1	2	ddim8	38
.	3	Bx	37	(ddim2)	3	aaug7	37
D♭♭	4	B♯	36	dim2	4	aug7	36
D♭	5	B♮	35	min2	5	maj7	35
D♮	6	B♭	34	maj2	6	min7	34
D♯	7	B♭♭	33	aug2	7	dim7	33
Dx	8	.	32	aaug2	8	(ddim7)	32
.	9	Ax	31	(ddim3)	9	aaug6	31
E♭♭	10	A♯	30	dim3	10	aug6	30
E♭	11	A♮	29	min3	11	maj6	29
E♮	12	A♭	28	maj3	12	min6	28
E♯	13	A♭♭	27	aug3	13	dim6	27
Ex	14	.	26	aaug3	14	(ddim6)	26
F♭♭	15	Gx	25	ddim4	15	aaug5	25
F♭	16	G♯	24	dim4	16	aug5	24
F♮	17	G♮	23	per4	17	per5	23
F♯	18	G♭	22	aug4	18	dim5	22
Fx	19	G♭♭	21	aaug4	19	ddim5	21

Table 2. The 35 pitch-names per octave and musical intervals used in p1 (**pch**) & p2 (**mi**) Codes given in a *base-40* enumeration [3]. Intervals in parentheses not considered.

We have observed that exact-pitch searches penalize faulty recollection and that gross-contour searches often produce an excessive number of prospective matches. We have come to believe that scale-degree searches are overall the most robust for searching for tonal music incipits. Scale-degree searches tie for the most-used search type on *Themefinder*, along with the exact-pitch search followed in third place by gross contour. [3]

Two- to three-letter abbreviations used in this paper for each pitch and rhythm feature are given in Table 3. In addition to the seven pitch features described above, seven rhythm features were also extracted from the musical data for this study. Rhythm can be categorized into two basic components of duration and meter. Three levels of duration and four levels of metric features are given in the following list:

r1 *Duration*: the duration of notes in the musical score (not the same as performance duration). Also called inter-onset interval in perceptual studies.

[3] http://ismir2003.ismir.net/tutorials/Sapp_fichiers/slide0025.htm

	Abbr.	Search type	# states
p1	**pch**	enharmonic pitch class	35
p2	**mi**	musical interval	(35)
p3	**12p**	12-tone pitch class	12
p4	**12i**	12-tone pitch interval	(12)
p5	**sd**	scale-degree (diatonic pitch class)	7
p6	**pgc**	pitch gross contour	3
p7	**prc**	pitch refined contour	5
r1	**dur**	duration	?
r2	**dgc**	duration gross contour	3
r3	**drc**	duration refined contour	5
r4	**blv**	beat level	2
r5	**mlv**	metric level	?
r6	**mgc**	metric gross contour	3
r7	**mrc**	metric refined contour	5

Table 3. Symbolic music features. Numbers in parentheses not limited to one octave in the study. States listed with "?" are not enumerable without a specific database (see Table 4).

r2 *Duration gross contour*: The following note is characterized as being (1) longer, (2) shorter or (3) equal in duration to the current note.

r3 *Duration refined contour*: A refined version of the rhythmic gross contour where *longer* and *shorter* are further split into two sub-categories: (1) next note is $> 2\times$ as long, (2) next note $\leq 2\times$ as long, (3) next note is same duration, (4) next note is $\leq 2\times$ as short, and (5) next note is $> 2\times$ as short.

r4 *Beat level*: A gross metric description of notes being either (1) on the beat, or (2) off the beat.

r5 *Metric level*: A refined metric description of the metric position (in a particular meter). For example, beats could be on the quarter-note level, 8th-note off-beats would be the 8th-note level.

r6 *Metric gross contour*: qualitative differences between metric levels of adjacent notes: (1) next note is on a stronger, (2) weaker, or (3) equivalent metrical position.

r7 *Metric refined contour*: refined qualitative differences between metric levels of adjacent notes. Metric changes placed in five categories, where the next note is in a metric position which is: (1) weaker, (2) much weaker, (3) stronger, (4) much stronger, or (5) the same as the current note.

The *Themefinder* search-engine also allows users to filter by meter-signatures, but it currently offers no searches of rhythmic features on an item-per-item basis as listed above. Figure 1 demonstrate the extraction of various feature data from a musical incipit.

In actual datasets, not all possible feature states are used. We find that the number of states encountered at higher-orders of pitch precision varies from repertory to repertory. Also, intervallic searches at higher levels of precision are computed with discrete octaves (*i.e.*, the interval of a perfect twelfth is not equated with that of a perfect fifth), so the range of actual feature states varies

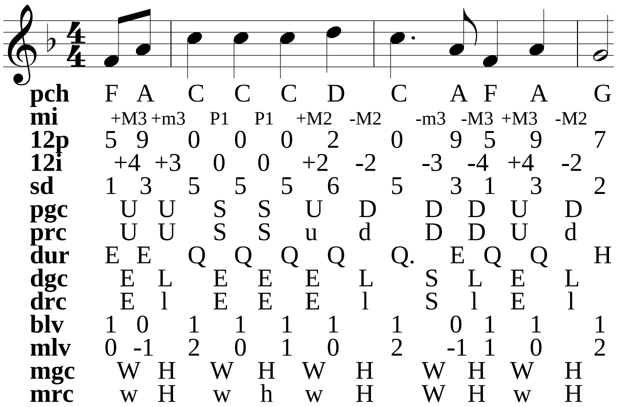


Figure 1. Example music along with extracted pitch and rhythm features.

Search Type	**Classical**	**Polish**	**All**
12i	70	40	88
pch	29	26	32
mi	95	52	109
dur	109	29	122
mlv	10	12	14

Table 4. Numbers of states for variable features.

with the database and the symbol extraction method. Table 4 shows various experimental state counts for selected repertories.

3.1. Searching Procedures

We used the *Themefinder* data collections directly to avoid restrictions of the web interface and to process data more quickly with a specialized searching program written in C which stored the data in memory between searches. Anchored searches of the database (searches that match from the start of an incipit only) could be done at a rate of 500/second with 100,000 incipits on a 1.5 GHz computer. Fully examining a particular musical feature's search charactistics on the entire database requires about one million database queries, taking about 1/2 of an hour to complete.

Unanchored searches (searching starting at any position in the incipit as well as the beginning) required longer search times of about 5 searches/second on 1.8 million notes in the combined database, with search statistics taking almost three days per feature to collect. Anchored searches are suitable for incipits. Unanchored searches may be more useful for finding themes in complete works.

We ran search tests on each of the various pitch and rhythm feature sets listed in Table 3. With those figures in hand, we then ran a set of joint pitch-rhythm feature search tests to use for comparison. We also tested the results for incipits against the results for full-score searches in the Essen dataset. Our purpose was to determine the extent to which the tests we ran on incipits would be valid for complete works (Table 6).

4. EFFECTIVENESS MEASURES

Three related concepts were developed for analyzing search characteristics: (1) Match Count Profiles, (2) Time-to-Uniqueness and to Sufficiency, and (3) various entropy measurements. These concepts are described in detail in the each of the following subsections.

4.1. Match Count Profiles

Performing a search with a short query sequence yields more matches in the database than longer query sequences. Therefore, a *Match Count Profile* was developed to examine the behavior of the match counts as the query length is increased. For example, Table 5 shows the anchored and unanchored Match Count Profiles for the music in Figure 1 using scale-degree features.

Query Length	1	2	3	4	5	6	7	8	9	10	11
Anchored	2493	566	288	126	64	18	12	5	4	2	1
Unanchored	8399	4834	1772	626	259	56	36	10	6	2	1
Theoretical	1643	304	64	14	3	1					

Table 5. Match Count Profiles for music in Figure 1.

In Table 5, the query length indicates the number of elements used in the search starting from the first note in the music. For example, the length-four query is "1 3 5 5". This query resulted in 126 total songs starting with the exact same pattern, and 626 songs which contain the exact same pattern anywhere in the song.

The "theoretical" line at the bottom of the table indicates a prediction of how many anchored matches the query is expected to generate given the probability distribution of symbols in the database which will be examined in the entropy section in further detail. Table 5 also demonstrates that the query length needed to find a unique match is the same whether the search was anchored at the start of the music or allowed to start anywhere in a short monophonic song.

After generating individual feature profiles for all themes in the database, the similar-feature profiles were averaged and can be seen in Figure 2. This figure shows profiles for each of the 14 pitch and rhythm features extracted from music in the database. Succinct feature sets display steeper slopes on the curves. All pitch features find unique matches in the database by about 12-length queries; however rhythm features require much longer sequences to uniquely find matches in the database.

The two rhymic features, **blv** and **mgc** are exceptionally bad at finding unique matches. The **mgc** feature has an unusually gradual slope when compared to **mrc** but it still seems to be correct after careful double-checking of the data and may just be a peculiarity of the folksong dataset.

4.2. TTU and TTS

We now define two experimental measures of importance in the Match Count Profiles:

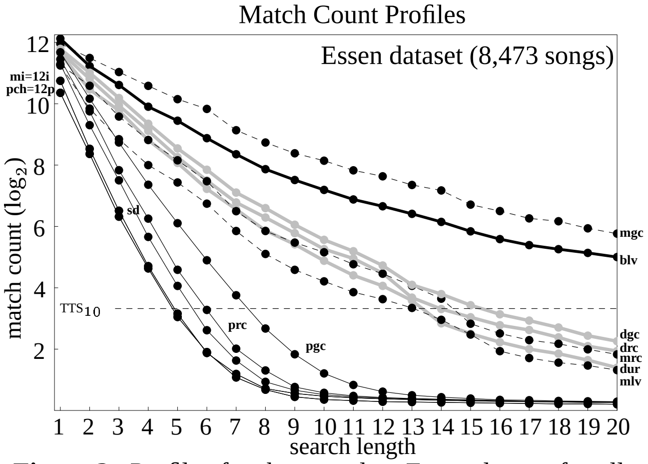


Figure 2. Profiles for the complete Essen dataset for all 14 pitch and rhythm features. Gray = duration features; dashed = metric features; thin solid = pitch features.

		Incipit only	Full work
Anchored	TTU (**mi**)	6.87619	8.74826
seach	Failure rate	0.669%	0.0354%
Unanchored	TTU (**mi**)	7.926	10.2858
search	Failure rate	1.07%	0.0472%

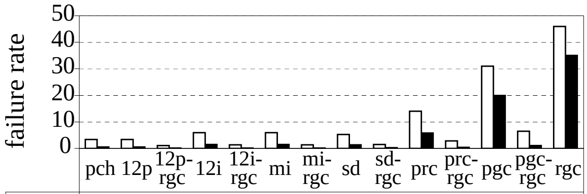
Table 6. Comparison of TTU for incipits and for the full works from which they originate (based on Essen dataset).

- Time-to-uniqueness (TTU): the query length required to find a unique match in the database.
- Time-to-sufficiency (TTS): the query length required to find an upper limit to the number of resulting matches.

Figure 2 marks the TTS level for the upper-limit of match counts when it is set to 10 matches, as we do so arbitrarily throughout this study. As it turns out, TTS is preferable to TTU when calculating the entropy in the following section where the initial slope of the profile is used to estimate the entropy rate. To verify this observation, examine the pitch feature curves in Figure 2. The start of the curves are nearly linear on a logarithmic scale, but when approaching the TTU point they significantly flatten out.

The convergence pattern shows in Figure 2 is representative of what we found in all of the databases, although there are small differences in the details from repertory to repertory. The average TTS, given the greatest precision in pitch resolution and an anchored search, was found to be less than 6 [data tokens], but the performance differences between the four most precise levels were minimal. This confirms and refines previous results reported by Schlichte [9] and Howard [4].

No TTU could be retrieved for highly generic themes, but the details vary by dataset size and feature set. (For example, for **pgc** in the classical set, 11% did not achieve TTU within the full incipit size.) Because certain incipits did not have a unique match, their TTUs were excluded from averages. Figure 3 gives the TTU and TTS failure rates for the entire dataset for various musical features. Less specific features yield higher failure rates. It is important to note that TTU was given a special definition for this study to prevent false failures: if a single match was not found by searching with an entire incipit, then the search was still determined to reach a unique end if the match count was the same for the entire incipit minus the last note.



	pch	12p	12p-rgc	12i	12i-rgc	mi	mi-rgc	sd	sd-rgc	prc	prc-rgc	pgc	pgc-rgc	rgc
□TTU	3.4	3.4	1.1	5.9	1.4	5.9	1.4	5.3	1.5	14	2.8	31	6.5	46
■TTS	.53	.53	.11	1.5	.15	1.5	.16	1.3	.21	5.8	.38	20	1.1	35

Figure 3. Search failure rates for all datasets according to search feature. For example, when searching by **pch** 3.4% of the incipits are not found "uniquely" even by the end of the incipit (with an average length of 18 notes). Similarly, 0.53% of incipits cannot be found with 10 or fewer matches by searching for the entire length of the incipit.

4.3. Entropy

Entropy is a measure of diversity used in comparisons of information. In mathematical terms, entropy, $H(X)$, evaluates the randomness of a variable in terms of how widely spread the probability distribution, $P(X)$, is. Mathematically, it can be shown that $2^{H(X)}$ is never larger than the actual number of states. $H(X)$ is measured in bits and can be called the first-order entropy. Equation 1 gives the definition of first-order entropy [1],

$$H(X) = -\sum_{i=1}^{I} P_i \log_2 P_i \qquad (1)$$

where P is the normalized probability distribution (the sum of all P_i is 1) of the random variable X, and I is the number of states which X may posses. As an example, Figure 4 shows the probability distribution for **pch** and **12p** features in all datasets which yields first-order entropies of 3.40 and 3.33 bits/symbol respectively.

On the other hand, the entropy rate, $G(X^n)$, describes the unpredictability of a random process X^n. A random process, X^n, with a large first-order entropy does not necessarily have a large entropy rate since the sequence of elements in the process may not be purely random. Mathematically, it can be shown that $G(X^n) \leq H(X)$. $G(X^n)$ is measured in bits/symbol.

If X varies in time and forms an n-length sequence, then its entropy rate, $G(X^n)$, is defined as

$$G(X^n) = \frac{H(X^n)}{n} \qquad (2)$$

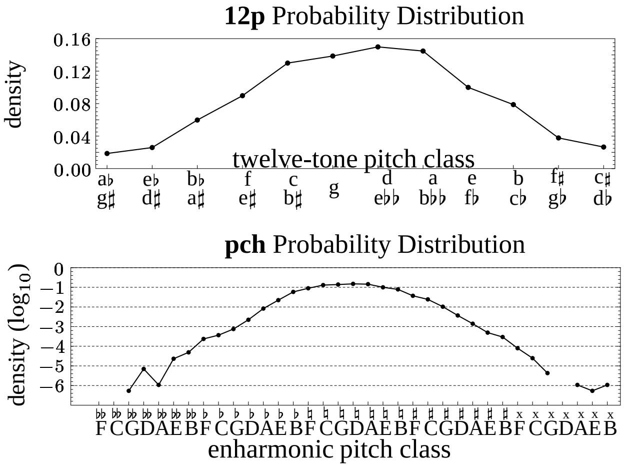


Figure 4. Example probability distributions for **pch** and **12p** features (all datasets). The pitch E♮ occurs in **pch** about once in every ten notes, while A𝄫 occurs about once in a million notes.

where X^n denotes the random vector $(X_1, X_2, \ldots, X_n)$, and its entropy can be calculated by summing over all possible sequence probabilities:

$$H(X^n) = -\sum_{i_1}^{I}\sum_{i_2}^{I}\cdots\sum_{i_n}^{I} P_{i_1,i_2,\ldots,i_n} \log_2 P_{i_1,i_2,\ldots,i_n} \quad (3)$$

The total number of terms in Equation 3 is I^n, which is the total number of possible n-length sequences X^n may realize. $H(X^n)$ can be called the n-th order entropy. In other words, the entropy rate is the average number of bits per symbol needed to encode a certain set of X^n, while the n-th order entropy is the total number of bits required to encode the same set of X^n.

Calculating $H(X^n)$ is complicated and not useful for our purposes, especially since incipit lengths vary, so we use TTS to estimate the entropy rate, G:

$$G = \frac{\log_2 \frac{M}{K}}{\mathrm{TTS}_K} \quad (4)$$

where M is the total number of unique incipits in a dataset, K defines the cutoff number of match for sufficiency ($K = 10$ arbitrarily for our purposes), and TTS is the average time-to-sufficiency of all anchored searches. TTS determines entropy rate more accurately than TTU does, provided that the search-decay rate remains exponential up to the TTS point.

When combining two features in a search, several other information-theory concepts are useful. Joint entropy, $H(a, b)$, refers to the combined entropy of the features. The joint entropy is always less than or equal to the sum of the individual features. Mutual information, $I(a; b)$, is that portion of the joint entropy which is shared by both features (see Figure 5).

We attempted to determine how "complex" each of the repertories is. Figure 6 compares the first-order entropy

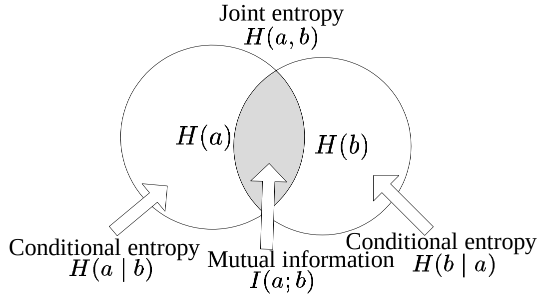


Figure 5. Venn diagram showing conceptual relationships between entropy, joint entropy, conditional entropy, and mutual information.

with the experimentally measured entropy rate for the various datasets. The first-order entropy indicates the maximum expected complexity of the music, and the entropy rate quantifies the actual complexity. Figure 6 demonstrates that the classical-theme dataset is the most "complex" from the perspective of informational feature (here **12p**) variety, while the Polish religious-song corpus is the least "complex." The Renaissance dataset has the lowest first-order entropy. However, its entropy-rate is higher with respect to its entropy than any other dataset. This means that even though the Renaissance incipits use fewer twelve-tone pitches than the classical theme set, the incipits are just as "complex."

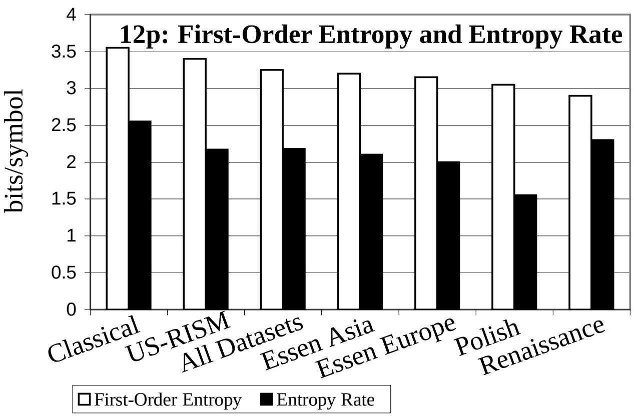


Figure 6. Twelve-tone pitch (**12p**) entropy per repertory, sorted by first-order entropy. Light bars show first-order entropy. Dark bars show entropy rate.

Note that the entropy rates in Figure 6 are always less than the entropy for each dataset. This indicates that melodic sequences are not purely random based on the feature probability distributions and explains why the theoretical match counts in Table 5 are too low.

5. JOINT SEARCH FEATURES

With the measuring techniques described in the previous sections, it is useful to now consider the effects of combining pitch and rhythm features into a single search. Entropy

Entropy Type	Definition	Value
pgc first-order entropy	$H(\mathbf{pgc})$	1.5325
dgc first-order entropy	$H(\mathbf{dgc})$	1.4643
Joint first-order entropy	$H(\mathbf{pgc}, \mathbf{dgc})$	2.9900
Conditional pitch entropy (given rhythm)	$H(\mathbf{pgc}\vert\mathbf{dgc})$	1.5256
Conditional rhythm entropy (given pitch)	$H(\mathbf{dgc}\vert\mathbf{pgc})$	1.4575
Mutual information	$H(\mathbf{pgc}) - H(\mathbf{pgc}\vert\mathbf{dgc})$	0.0068

Table 7. Calculation of mutual information from first-order pitch and rhythm entropies (values based on Essen datasets).

Search Features	TTS	TTU
pgc	7.4	12.1
dgc	11.2	17.0
pgc + **dgc**	4.6	8.4

Table 8. TTU and TTS values for Essen data for separate and joint **prc** and **dgc** features.

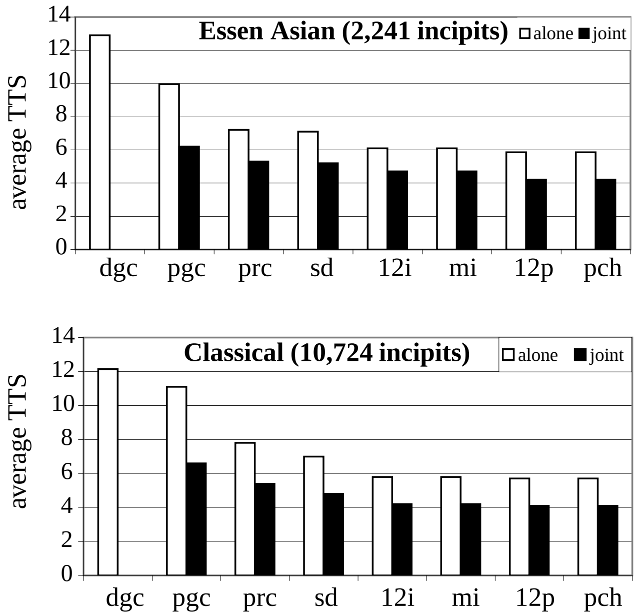


Figure 7. Pitch-only compared to joint searches with **dgc** for two different datasets. Light bars are pitch features search alone. Dark bars show joint TTS values when a pitch feature is combined with **dgc**.

calculations of mutual information show that pitch and rhythm features are very independent (Table 7, so searching with both at the same time is not a wasteful endeavor.

To implement joint searches in practical terms, we enumerated both the pitch and rhythm features in series onto the sequence of prime numbers, and then multiplied the pitch and rhythm features together to get a one-dimensional search sequence which could be searched in the same manner as the pitch or rhythm features alone. If necessary, this set can then be decomposed again into the original two feature sets without an additional lookup table.

For example, the combined **pgc**-**dgc** search employs 9 states. **pgc** states can be assigned to the primes 2, 3, and 5. **dgc** states can then be assigned to the next three primes: 7, 11, 13. Then the possible states for the joint search **pgc**-**dgc** would be: 14, 21, 22, 26, 33, 35, 39, 55, and 65.

Searches combining pitch and rhythmic features significantly reduced the TTS and TTU. For example, Table 8 shows that both TTS and TTU query lengths were reduced about 30% for **pgc**-**dgc** joint-feature searches.

Comparing all pitch features jointly searched with **dgc**, a similar reduction in TTS length is noticed in all repertories (Figure 7). These tests gave useful insights into the results of the more extensive series of measures we derived from coupling **dgc** with each level of pitch precision (Figure 2).

Figure 8 shows Match Count Profiles for two pitch features and two rhythmic features along with their joint-feature profiles. Two interesting things are shown in this figure: (1) the low-detail **pgc** feature plus a rhythmic feature performs searches as good as or better than **12i** features alone, and (2) more specific rhythmic features generate lower TTS values than less specific features. However, chosing different rhythmic features to combine with a particular pitch feature does not greatly improve the TTS values as opposed to choosing a different rhythm feature. Note in particular in Figure 8 that the 6-state **pgc**-**blv** joint feature is almost identical in entropy-rate to **12i** alone which has about 20 states.

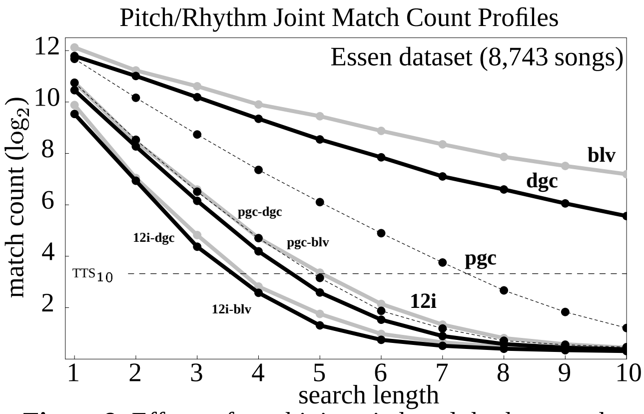


Figure 8. Effects of combining pitch and rhythm searches on match count profiles.

6. CONCLUSIONS

The three search-effectiveness measuring techniques described in this paper are useful in quantitatively evaulating the search properties of the multitude of features that can be extracted from musical data. In particular, applying these metrics to an analysis of joint pitch-rhythm features clearly sumarizes the improvements in search times which are possible under realistic search conditions.

An important verification in this set of tests was that the most significant increases in search-effectiveness come from the progressions in quantization precision from pitch/gross-contour (**pgc**) to **prc**, and from scale degree (**sd**) to 12-tone pitch (**12p**). However, the improvement in performance of **sd** (7 states) over **prc** (5 states) is slight, a finding which is generally similar to [9]. Also, twelve-tone searches are preferable to enharmonic pitch-class searches because **pch**/**12p** and **mi**/**12i** match profiles were coincident to about one part in 10,000. Since the **12p** states are fewer than **pch**'s and their search characteristics are nearly identical, twelve-tone searches are preferrable for minimizing query errors.

Another useful result is the relative ordering of effectiveness for various rhythmic features can be made by examining Figure 2. All rhythmic features are less effective for searching than the most general pitch feature of **pgc**. From most specific to most general, the rhythm features are: **mlv**, **dur**, **mrc**, **drc**, **dgc**, **blv**, and **mgc**. However, these features are narrowly clustered in their efficiency, so different repertories may reorder them slightly.

Combining **dgc** and **pgc** is a particularly effective search strategy since there is little mutual information between the two features. Both features are vague by themselves, but together they are just as or more effective than pitch-only scale-degree searches. Interestingly, incorporating **dgc** with other more specific pitch features yields only marginally better TTS/TTU times (about 1.5 fewer notes required in the search query, as compared to 4.5 fewer notes with **pgc** on average).

Octave information does not significantly improve search results, since the **pch**/**12p** and **mi**/**12i** sets had nearly identical entropy rates. The **pch**/**12p** set did not encode octave, while the **mi**/**12i** set preserved octave information. Also, enharmonic spellings are not more effective than twelve-tone search types, since **12p** and **pch** have similar entropy and very similar entropy-rates.

Further analytical materials, updates and results can be requested and are posted on the internet at http://themefinder.org/analysis.

7. ACKNOWLEDGMENTS

The nucleus of the *Themefinder* database was designed by David Huron. Its original implementation on the web was designed by Andreas Kornstädt. Much of its subsequent development, including data translations, was undertaken by Craig Sapp, who usually maintains it. The bulk of the Essen data was made available originally by Helmut Schaffrath and subsequently by Ewa Dahlig. The Essen Chinese data was encoded by Baoqiang Han. The Luxembourgian addition was encoded and made available by Damien Sagrillo. The Italian Latin motet data was provided by Harry B. Lincoln [6]. RISM data was made available by the US RISM Committee centered at Harvard University. The "classical" data was provided in large part by David Huron, and was revised and expanded by CCARH staff and Stanford University students. The authors extend their cordial thanks to these contributors as well as to those not mentioned.

8. REFERENCES

[1] Cover, Thomas, and Joy A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[2] Dannenberg, Roger B., W.P. Birmingham, G. Taznetakis, C. Meek, N. Hu, B. Pardo, "The MuseArt Testbed for Query-by-Humming Evaluation," *Proc. of the 4th Annual International Symposium on Music Information Retrieval*, Baltimore, Maryland, 2003.

[3] Hewlett, Walter B. "A Base-40 Numberline Representation of Musical Pitch Notation," *Musikometrika* 4 (1992), 1-14. [4]

[4] Howard, John. "Strategies for Sorting Musical Incipits," in *Melodic Similarity: Concepts, Procedures, and Applications* (*Computing in Musicology* 11), ed. Walter B. Hewlett and E. Selfridge-Field. Cambridge, Mass.: The MIT Press, (1998), 119-128.

[5] Kim, Youngmoo E., W. Chai, R. Garcia, and B. Vercoe, "Analysis of a Counter-Based Representation for Melody," *Proc. of the 1st International Sympoisum on Music Information Retrieval*; Plymouth, Mass., 2000. [5]

[6] Lincoln, Harry. *The Latin Motet: Indexes to Printed Collections, 1500-1600*. Institute of Mediaeval Music; Ottawa, 1993.

[7] McNab, Robert, L.A. Smith, I.H. Witten, C.L. Henderson and S.J. Cunningham. "Towards the Digital Music Library: Tune Retrieval from Acoustic Input". *Proc. of the 1st ACM International Conference on Digital libraries*, Bethesda, Maryland; 1996. [6]

[8] Rand, William, and William Birmingham, "Statistical Analysis in Music Information Retrieval," *Proc. of the 2nd Annual International Symposium on Music Information Retrieval*, Bloomington, Indiana, 2001.

[9] Schlichte, Joachim. "Der automatische Vergleich von 83,243 Musikincipits aus der RISM-Datenbank: Ergebnisse - Nutzen - Perspektiven," *Fontes artis musicae* 37 (1990), 35-46.

[10] Sorsa, Timo, and Jyri Huopaniemi, "Melodic Resolution in Music Retrieval," *Proc. of the 2nd Annual International Symposium on Music Information Retrieval*, Bloomington, Indiana, 2001. [7]

[4] http://www.ccarh.org/publications/reprints/base40
[5] http://web.media.it.edu/˜chaiwei/papers/MusicIR_melody.pdf
[6] http://portal.acm.org/citation.cfm?id=226934
[7] http://music-ir.org/gsdl/ismir2001/posters/sorsa.pdf