ABSTRACT

Many musics across the world are structured around multiple modes, which hold a middle ground between scales and melodies. We study whether we can classify mode in a corpus of 20,865 medieval plainchant melodies from the Cantus database. We revisit the traditional ‘textbook’ classification approach (using the final, the range and initial note) as well as the only prior computational study we are aware of, which uses pitch profiles. Both approaches work well, but largely reduce modes to scales and ignore their melodic character. Our main contribution is a model that reaches 93–95% $F_1$ score on mode classification, compared to 86–90% using traditional pitch-based musical methods. Importantly, it reaches 81–83% even when we discard all absolute pitch information and reduce a melody to its contour. The model uses tf–idf vectors and strongly depends on the choice of units: i.e., how the melody is segmented. If we borrow the syllable or word structure from the lyrics, the model outperforms all of our baselines. This suggests that, like language, music is made up of ‘natural’ units, in our case between the level of notes and complete phrases, a finding that may well be useful in other musics.

1. INTRODUCTION

In his seminal Grove entry, Harold Powers [1] points out a remarkable cross-cultural generalisation: many musics are structured around multiple modes. Modes are often associated with the major–minor distinction in Western music, but there are much richer systems of modes: examples include Indian raga, Arabic makam, Persian dastgah, pathet in Javanese gamelan music and the modes of Gregorian chant. The specifics obviously vary, but all these phenomena share properties with both scales and melodies, and are perhaps best thought of as occupying the continuum in between [1]. On the one hand, a mode is more than a scale: it might imply a hierarchy of pitch relations or favour the use of characteristic motifs. On the other hand, it is not as specific as a particular tune: a mode rather describes a melody type. Modes are of central importance to their musical tradition, both as means to classify the repertoire, and as practical guides for composition and improvisation [1]. Characterising modes computationally is therefore an important problem for computational ethnomusicology.

Several MIR studies have investigated automatic mode classification in Indian raga [2, 3], Turkish makam [4, 5] and Persian dastgah [6, 7]. These studies can roughly be divided in two groups. First, studies emphasising the scalar aspect of mode usually look at pitch distributions [2, 5, 7], similar to key detection in Western music. Second, studies emphasising the melodic aspect often use sequential models or melodic motifs [3, 4]. For example, [4] trains n-gram models for 13 Turkish makams, and then classifies melodies by their perplexity under these models. Going beyond n-grams, [3] uses motifs, characteristic phrases, extracted from raga recordings to represent every recording as a vector of motif-frequencies. They weigh counts amongst others by the inverse document frequency (see section 3.4), which balances highly frequent motifs, and favours specific ones.

In this paper, we focus on automatic mode classification in Medieval plainchant. This has only rarely been studied computationally, even though the term (if not the phenomenon) ‘mode’ originates there. At first glance, mode in plainchant is relatively clear, though certainly not entirely unambiguous. With a second glance, it has a musicological and historical depth that inspired a vast body of scholarship going back over one thousand years. The music is indeed sufficiently distant in time from most other musics, including Western classical and pop music, to provide an interesting cross-cultural comparison. And for once, data is abundant, thanks to the immense efforts of chant scholars.

Chant has mostly figured in MIR studies in optical music recognition of medieval manuscripts: the SIMSSA project, for example, has used such systems to transcribe plainchant from the Cantus database [8]. Recent ISMIR conferences have also included analyses of Byzantine plainchant [9] and Jewish Torah tropes [10], and a comparison of five Christian chant traditions using interval n-grams [11]. But, to the best of our knowledge, Huron and Veltman’s study [12] is the only computational study addressing mode classification in chant. They took a scalar perspective on mode by using pitch class profiles, an approach which was later criticised, partly for ignoring mode’s melodic character [13].

We aim to revisit this work on a larger dataset, and also to model the melodic aspect of mode. Concretely, we compare three approaches to mode classification:

1. Classical approach: based the range, final, and initial note of a chant.
2. Profile approach: uses pitch, pitch class and repetition profiles (cf. [12]).

3. Distributional approach: uses tf–idf vectors based on various segmentations and representation of the melody.

2. GREGORIAN CHANT

Gregorian chant is the monophonic, Latin chant sung during services in the Roman church. It started out as an oral tradition, coexisting with several others in late Antiquity. Although the specifics are debated [14, ch. 2], from the 9th century onwards it gradually turned into a (partly) written tradition, displacing other chant traditions. Initially, only the texts of the chants were written down, as singers would know the melodies by heart. Chant is rooted in recitation, and the music and text are intimately related: “the basic unit of music-writing [was] not the note, but the syllable” [15], the smallest singable unit of text. Accordingly, the earliest notation lived between the lines of text: signs, called neumes, reminding the singer of the contour of the melody: perhaps how many notes and their direction, but not which exact pitches. The earliest melodies are therefore unknown, but later manuscripts use a pitch-specific notation by placing neumes on staff lines, preserving those melodies to the present day (see Figure 1A).

There are different chant genres for different parts of the liturgy, each with own musical characteristics [16]. Some genres consist of recitations of a sacred text mostly on a fixed pitch, with common starting and ending formulae, while others use elaborate melodies and few repeated notes. Genres also differ in their melismaticness: the number of notes per syllable (see Figure S5). In syllabic genres like antiphons, every syllable of text aligns with roughly one note. More melismatic genres like responsories align single syllables to long melismas of ten notes or more. In this paper, we focus on antiphons and responsories, two melodic and common genres.

Gregorian chant uses a distinct tonal system of eight modes, usually numbered 1–8, but sometimes named like church scales. Modes come in pairs that share the same scale (Dorian, Phrygian, Lydian or Mixolydian), but have a different range or ambitus: authentic modes move mostly above the final, plagal ones mostly around it. Mode 3 is for example also called Phrygian authentic, and melodies in this mode rarely go below the final note E. The standard way of determining the mode is to first determine the final, and then the range [16]. For the majority of the chants this will be sufficient, but one might further consider the initial note, characteristic phrases or circumstantial evidence (e.g. psalm tones). Nevertheless, the mode of some chants will remain ambiguous: the theory of eight modes was borrowed from Byzantine theory in the 8th century, and applied to an already existing chant repertoire (with its own modalities [13]). The fit between theory and practice was reasonable, but not perfect [1]. This also suggests that perfect classification accuracy is likely out of reach.

![Figure 1](image-url)
3. METHODS

The design of this study is visualized in Figure 1.

3.1 Data: the Cantus Database

We use chant transcriptions from the Cantus database [17]. This is primarily a digital index of medieval chant manuscripts, recording the chant location in the manuscript, its full text, and properties like the mode, the liturgical feast, but also links to manuscript images. Cantus currently consists of almost 150 manuscripts, containing over 450,000 chants, contributed by chant scholars from all over the world. Over 60,000 chants also contain melodic transcriptions written in Volpiano.  

\footnote{1 Volpiano is a typeface developed by David Hiley and Fabian Weber for notating plainchant. See \url{fawe.de/volpiano/}}

It sets plain text as musical notes on a five-line staff, as illustrated in Figure 1a. Volpiano also supports some accidentals, clefs, liquescents, barlines and strokes. All submissions to Cantus are subject to strict guidelines and manually checked by the Cantus editors (see also [18]). This ensures the quality and consistency of database, making it a valuable resource for computational research.

We scraped the entire database of 497,071 chants via its REST API and we have released this as the CantusCorpus.  

\footnote{2 See github.com/bacor/cantuscorspus, here we use v0.2.}

We only consider chants that have a Volpiano transcription (63,628 chants) and further filter out chants with incomplete or non-standard transcriptions, without a complete melody, without ‘simple’ mode annotation, and exact duplicates (see section S1). This resulted in 7031 responsories (966,871 notes, avg. length 138 notes) and 13,865 antiphons (825,143 notes, avg. length 60 notes). We fixed a 70/30 train/test split for all datasets and only used training data in exploratory analyses. Cantus often contains multiple variants of any particular melody, transcribed from different manuscripts (see Figure S11). One may wonder whether the simple train/test split is sufficient, or whether even more care is needed to avoid overlap between such melodic variants in the train and test sets. This is a difficult issue that also applies to other musical corpora (e.g., the Essen folk-song corpus), and for which there is no perfect solution. We tried repeating our experiments on a subset without variants and return to this issue in section 4.4.

According to the transcription guidelines, flat symbols are transcribed only once, directly before the first flattened note. We replace the first and later flattened notes by the corresponding accidental, a Volpiano character that sits at a specific staff line. In this way, flat notes are also encoded by a single Volpiano character. We discard characters like clefs and pausas, and only retain the notes, accidentals and boundaries (hyphens). The resulting string is used in our three classification experiments, which we now discuss.

3.2 Classical Approach: Final, Range, Initial

The first approach is motivated by the classical procedure for mode classification. We extract three features from every chant: the final pitch, the range (lowest and highest pitches)
and the initial pitch. Theory suggests that the final alone
should give an accuracy of roughly 50%, and adding the
range should further increase that by roughly 50%, if there
is no ambiguity. Figure 2 shows the feature distributions
for all modes. It suggests that there is some ambiguity, and so
numbers will be a little lower. For this task we use random
forest classifiers [19], which aggregate multiple decision
trees. Training details of all models are discussed below.

### 3.3 Profile Approach: Pitch (Class) Profiles

The second approach is inspired by Huron & Veltman [12].
Using 97 chants from the Liber Usualis, they compute av-
average pitch class profiles (the relative frequency of each
pitch class) for each of the modes and then classified chants
to closest profile. We take a similar approach and use k-
nearest neighbour classification, where $k$ is tuned (see sec-
tion 3.5). In a commentary, Wiering [13] argued for using
actual pitches rather than pitch classes, as the pitches an
octave above the final have a very different role than those
an octave below it. We follow that suggestion by also
computing pitch profiles (Figure 3). Finally, we propose a
repetition profile aiming to describe which notes function
like a recitation tone. For every Volpiano pitch $q$ we com-
pute a repetition score $r(q)$, which is the relative frequency
of direct repetitions, and collect these to get a repetition
profile. Formally, if a chant has pitches $p_1, \ldots, p_N$, then
$r(q) = \#\{i: p_i = q \text{ and } p_{i+1} = q\}/(N-1)$ since there are
$N-1$ possible repetitions.

### 3.4 Distributional Approach: tf–idf Vectors

Our third approach aims to capture the melodic aspect of
mode. In short, we use a bag of ‘words’ model (cf. [3]) and
tweak two parameters: the segmentation (which melodic
units to use as ‘words’) and the representation (pitches, in-
tervals and contours). The idea is to discard more and more
information about the scale, and see if we can nevertheless
determine the mode.

First, the units. For chant, three natural segmentations
suggest themselves: one can segment the melody (1) at
neume boundaries, but also wherever we find (2) a syllable
or (3) a word boundary in the lyrics. Given the close relation
between text and music in chant, there is some reason to
believe that these are meaningful units. Conveniently, all
of these boundaries are explicitly encoded in Volpiano, by
a single, double and triple dash respectively. Note that
these natural units are nested: neumes never cross syllable
boundaries. We compare the natural units to two types of
baselines. The first is an $n$-gram baseline where we slice
the melody after every $n$ notes, for $n = 1, \ldots, 16$. The
second is a random, variable-length baseline. Here the
melody is segmented randomly, but in such a way that the
segment length is approximately Poisson distributed with a
mean length of 3, 5, or 7. We stress that all these units are
proper segmentations: units do not overlap. In particular,
we choose not to use a higher-order model (using $n$-grams
of units), because we are only interested in comparing different
segmentations.

Second, the representation. We represent melodies in
three ways: as a sequence of pitches, intervals (the num-
ber of semitones between successive notes) and contours
(the contour between successive notes: up, down or level).
There is one complication when segmenting sequences of
intervals or contours: we introduce dependencies between
the units. All units would, for example, start with the in-
terval from the previous unit. We call this a dependent
segmentation. Alternatively, you could discard the intervals
between units to obtain an independent version. This effect-
ively makes every unit one interval shorter. We analyse both
independent and dependent versions, but in the independent
one we found it convenient to start all units (including the
first) with a dot to keep the segmentation identical across
representations. You can think of the dot as marking the
omitted interval to the previous unit.

Third, the model. Given a segmentation, we represent
every chant by a vector of unit frequencies, but weighted
to favour frequent, yet specific units: units that do not oc-
cur in too many chants. A standard way of doing this in
textual information retrieval is using term-frequency inverse-
document-frequency (tf–idf) scores, which multiply the fre-
cquency of a term in a document (tf) by the inverse document
frequency (idf): the inverse of the number of documents
containing the term. We use +1 smoothing for the idf, at
most 5000 features, and found it was important not to set
a minimum or maximum document frequency. We train a
linear support vector machine to classify mode using the
resulting tf–idf vectors.

In sum, we analyse 22 segmentations (3 natural ones,
16 $n$-grams, 3 random) and 5 representations (pitch and
dependent/independent interval/contour), giving a total of
110 conditions.

### 3.5 Training

We tune every model using a randomised hyperparameter
search with 5-fold stratified cross-validation. That is to
say that we randomly sample hyperparameters from a suit-
able grid (determined by extensive manual analyses) and
determine their performance using 5-fold cross-validation
on the training set, where we ensure the class frequencies
are similar in all folds. We use the hyperparameters yield-
ing the highest cross-validation test accuracy to train the
final model. All models were implemented in Python using
scikit-learn [20] and data and code are available online.\(^3\)

### 4. RESULTS

Figure 5 gives support-weighted\(^4\) averages of $F_1$-scores
obtained on the full test sets for all three approaches. The
scores are averages of five independent runs of the exper-
iment, using different train/test-splits. Standard deviations
were small and are included in figure S12. We now compare
the three approaches and then discuss the effect of repres-
entation and segmentation on the distributional approach.

\(^3\) See github.com/bacor/ismir2020

\(^4\) The retrieval scores for all classes (modes) are averaged, weighted by
the number of instances in each class.
4.1 Approaches: Distributional Approach Works Best

First of all, we report the highest classification scores with our distributional approach using pitch representations: an $F_1$-score of 93% for responsories and 95% for antiphons. This corresponds of an error reduction of 30–60% compared to the classical approach (90% and 86%). The classical approach confirms the rule of thumb: the range and final are very informative features. Using only these, we obtain $F_1$-scores of 89% and 79%, which are further increased by also adding the initial. The profile approach outperforms the classical approach for antiphons (90% vs. 86%), but is outperformed for responsories (88% vs. 90%). Our results support Wiering’s [13] intuition that pitch profiles more accurately describe mode than pitch class profiles, but the effect is small: it increases $F_1$ scores by 2–3%. Repetition profiles appear to be less useful for both genres.

In broad strokes, our results validate the classical and profile approach, both of which peak around a 90% $F_1$-score, using simple features. The distributional approach improves this, up to 95% using complex features. Importantly, we now show that the distributional approach maintains high performance when using interval or contour representations.

4.2 Representations: Contours are Sufficient

We find that the classification task gets harder when the representation gets cruder, from those based on pitch, to intervals and finally to contours (figure 5C, horizontally). This was anticipated: cruder representations are obtained when moving from pitch to independent contour representation. At that point it performs at majority baseline (a 7% $F_1$-score for three approaches to mode classification, using two chant genres: responsories and antiphons). Scores are averages of five independent runs of the experiment. The classical approach (A) using the final, range and initial reaches $F_1$-scores of 90% and 85%. The profile approach (B) works better for antiphons (90% vs. 86%) and somewhat worse for responsories (88% vs. 90%). As [13] suspected, pitch profiles outperform pitch performance when using interval or contour representations. Importantly, we now show that the distributional approach maintains high performance when using interval or contour representations. The distributional approach confirms the rule of thumb: the range and final (95%) beats the classical approach (90% and 86%). The classical approach works better for antiphons (90% vs. 86%) and somewhat worse for responsories (88% vs. 90%). As [13] suspected, pitch profiles outperform pitch profiles by a small margin. The distributional approach (C) reaches the highest $F_1$ scores of 95% on both responsories and antiphons. The choice of segmentation (vertically) is crucial: classification is improved by using ‘natural’ units, word-based units in particular, rather than $n$-grams. As the representation (horizontally) becomes cruder, from pitches to intervals and finally to contours, the task becomes much harder. But, when using word-based segmentation, performance remains high.
only 3% below the classical baseline using the highly im-
poverished independent contour representation. In contrast
to the other representations, the contours do not carry any
information about the scale: the same contour can be repro-
duced in any scale. Apparently, we can discard the scalar
aspect of mode, and still classify it: contours alone contain
sufficient information for mode classification. The success
of pitch-based methods might obscure the fact that mode is
as much a melodic phenomenon as a scalar one.

It is interesting to note that the earliest chant notation
used unpitched neumes that mainly described the contour
of the melody—not the exact pitches. Our results reinforce
the idea that contour is highly informative—so informative
that given a mode, text and contour, an experienced singer
could reconstruct the chant melody.

4.3 Segments: Natural Units Work Best.

Our most important result is that among all the representa-
tions we considered, natural units (neume, syllables, and
words) yield the highest classification performance. The 4-
and 6-gram baselines also reach top $F_1$-scores in antiphons,
but only when we use representations that include informa-
tion about pitch. Furthermore, the success of natural units
cannot be explained solely by their length. In responsor-
ies, neumes, syllables and words are on average 2.3, 3.0
and 7.1 notes long, respectively (see table S6), and yet the
performance of these natural units is consistently higher
than $n$-grams of comparable length. The performance of
the natural units is also consistently higher than that of the
variable-length Poisson baselines, which are intended to
mimic the overall distribution of natural lengths but ignore
musical and textual semantics.

A few other observations merit discussion. Firstly, al-
though neume and syllable segmentations behave differ-
etly for responsories, they behave similarly to each other
for antiphons. The reason may be that in antiphons, neumes
and syllables more often coincide. Antiphons are less melis-
matic than responsories (i.e., they use fewer notes per syl-
lable, 1.5 to be precise). Secondly, both the $n$-grams and
the Poisson baseline perform better on antiphons than on
responsories, possibly because the $n$-grams are more likely
to end up being coincidentally aligned with the natural units
the less melismatic the genre.

4.4 Controlling for Melodic Variants

We repeated all experiments on a subset of the data from
which we removed melody variants (see supplement S13
for details). In terms of the number of notes, this meant
a 75% and 66% reduction in data size for responsories
and antiphons respectively. The performance of all models
decreased on this subset, and for responsories more than
for antiphons. Our main findings that contours are suffi-
cient and that natural units work best across representations
stay. We do observe some reorderings: some already high-
performing $n$-grams in antiphons now for example slightly
overtake word segmentations, although only for pitch and
dependent interval representations. The distributional ap-
proach works best for antiphons regardless of including or
excluding chant variants, but for responsories, the distribu-
tional approach drops slightly below the classical approach
on the subset (where the profile approach is worst). These
findings might be explained by increased sparsity in the
smaller dataset: natural units in responsories are, after all,
longer. Exploring these issues further is left for future work.

5. DISCUSSION AND CONCLUSION

In this paper, we analyzed three approaches to mode clas-
cification in a large corpus of plainchant: (1) the classical
approach using the final, range and initial; (2) the profile
approach using pitch (class) profiles and (3) the distribu-
tional approach using a tf–idf vector model and various
segmentations and representations. We found that the distri-
butional approach performs best, and that it can main-
tain high performance on contour representations if using
the right segmentation: at word boundaries, in this case.
The main findings were largely upheld when we removed
melody variants, but the handling of variants is an issue
that deserves further investigation and that has implications
beyond this study.

Although our results are specific to one corpus of me-
dieval music and one classification task, we believe our
conclusions are of wider relevance. We often fall back on
$n$-grams because they are well understood and easy to use.
A more natural segmentation may be harder to obtain, but
if finding them can have such a large effect on a relatively
simple task like mode classification, their advantages may
be even stronger for more complex tasks.

A first next step could be to explore whether lyrics yield
equally useful units in other vocal musics. As noted, the
link between text and music in plainchant is particularly
tight. This at least suggests that the text may be useful in
other types of chant, like Byzantine chant or Torah trope.
For folk melodies designed to standard poetic meters, it
is not as obvious whether lyrics would help or hinder the
identification of useful units. This is worth investigating,
as characteristic motifs and repeated pattern are commonly
used in computational folk-song studies, in particular for
tune family identification [21, 22].

Our results raise another question: is chant indeed com-
posed by stringing together certain melodic units, much like
a sentence is composed of words? It has been suggested
(and disputed) that Gregorian chant is composed in a pro-
cess of centonization, and that a chant is a patchwork of
existing melodic chunks called centos. A recent study used
the tf–idf weighting to discover centos in Arab-Andalusian
music [23]. This raises the possibility that classification
using natural units may have been successful because they
indeed are the building blocks, the centos.

Chant is not yet commonly studied in the MIR com-
nunity, but we hope that this study shows that chant is
an interesting repertoire that can yield insights of broader
relevance. The immense efforts of chant scholars mean
that data are abundant. In short, we think chant can aid
the development of models that apply beyond Western classical
and pop music, and embrace the true diversity of musics
around the world.
6. REFERENCES


