Investigating Style with Scale Embeddings

Matt Chiu(✉)

University of Rochester - Eastman School of Music, Rochester, NY 14604, USA
mchiu9@u.rochester.edu

Abstract. In this paper, we use pitch-class vector *embeddings* to study scale relationships between composers. Recent research in natural language processing (NLP) has used machine learning to derive vector representations-known as embeddings—for words based on their co-occurrence. Borrowing from NLP, we use the word2vec algorithm to encode windows of pitch-classes, or *pitch-class vectors*, of music. We show that these embeddings not only replicate the well-known theoretical circle of fifths, but can also capture stylistic nuances between composers' use of scales.

Keywords: Scale theory · Embeddings · Word2vec · Vector space · Style

1 Introduction

In Natural Language Processing (NLP), the *word2vec* algorithm is a technique for deriving vector representations-known as *embeddings*-for words by iterating through a corpus [7]. Embeddings contain information about the syntactic placement and the semantic similarity of words. In symbolic music research, recent studies have explored musical embeddings for chords [6], and motivic fragments [1], as well as applications for harmonic tension [8], and music generation [2]. In this paper, we used word2vec to derive *scale embeddings* from pitch-class collections of different composers and compare embeddings between and within their styles.

2 Methodology: Word2vec, Encoding Procedure, and Training

2.1 Word2vec Algorithm

As said, the aim of the word2vec model is to generate dense vector representations (embeddings) for words based on their co-occurrence. Each unique word in a corpus is represented with a corresponding vector, with words that occur near one another in the corpus having similar embedding vectors. Since words with semantic similarity have similar syntactic placement, they also have similar embedding vectors (whether or not they not co-occur). In this paper, we used the

M. Montiel et al. (Eds.): MCM 2022, LNAI 13267, pp. 405–410, 2022.
https://doi.org/10.1007/978-3-031-07015-0_37

skip-gram version of word2vec: given a corpus W of words w with surrounding context-words c, the algorithm maximizes the likelihood of surrounding words:

$$\arg\max_{\theta} \prod_{w \in W} \prod_{c \in C} P(c|w;\theta) \tag{1}$$

where v_c and v_w are vector representations of words v and c respectively and C is the set of all contexts, the probability $P(c|w;\theta)$ is calculated with the softmax function:

$$P(c|w;\theta) = \frac{exp(v_c v_w)}{\sum_{c^{\intercal} \in C} exp(v_{c^{\intercal}} v_w)} \tag{2}$$

2.2 Encoding Procedure

As the word2vec algorithm is designed to parse words in a corpus, we needed both a corpus and a method for encoding musical objects as words-our musical vocabulary. For the corpus, we used the Yale Classical Archives Corpus, henceforth YCAC [11]. The YCAC is a spreadsheet of 13,769 midi files of works by 571 composers. Each midi file is parsed into "slices" containing the set of vertical pitches at any new pitch onset, time points T as quarter-note offsets from the beginning of the score, and other metadata.

For each piece, we extracted pitch classes using a sliding windowing procedure. Where a piece $a_t, a_{t+x}, \ldots, a_T$ consists of slices A at corresponding time points T, a window of length m is a subsequence $s_t = a_t, a_{t+x}, \ldots, a_{t \leq t+m}$. Each piece therefore contained $T - m$ windows, with each window consisting of m adjacent slices. Pitch classes in a window were then encoded as fixed-length, *pitch-class vectors* $PC = \{pc_0, pc_1, ..., pc_{11}\} \in \{0,1\}^{11}$ where the pitch class with subscript 0=C, subscript 1=C#/Db, etc. Present pitch classes in a window were represented as 1, and absent pitches were represented as 0, i.e. the C-major scale, for example, is represented as {1,0,1,0,1,1,0,1,0,1,0,1}. In NLP, this encoding method is also known as *one-hot* encodings. Note that since our methodology had no concept of tonic, pitch-class vectors represented multiple scales simultaneously: C major and A natural minor have the same collection and were represented identically. This is a significant simplification of scale and should not be taken as the end goal. Rather, the work here represents a crude proof-of-concept where future work might use weighted pitch-class vectors to distinguish pitch-class salience [3].

We then windowed pieces for four well-represented composers in the corpus: Mozart, Liszt, Saint-Saens, and Debussy (Table 1). To find an appropriate window length, we approximated the standard 7-diatonic-pitch scale by averaging the number of notes per window for windows of length 2–10 quarter notes (Fig. 1). Reflecting the historical narrative that chromaticism progressively increased over time, Mozart and Liszt use fewer pitches per window than Debussy and Saint-Saëns. Given the results of Fig. 1, window size m was set to 6 quarter-note beats for Mozart and Liszt and 5 for Saint-Saëns and Debussy.

Table 2 shows the total number of windows for each composer and the percent of major/natural minor and harmonic minor collections for each. 27% of windows

Table 1. Sampled yale classical archives corpus data.

Composer	No. of pieces in YCAC	No. of notes	Avg. notes per piece
Mozart	882	3,865,439	4,382.58
Liszt	125	806,025	6,448.2
Saint-Saëns	72	504,663	6,913.19
Debussy	39	170,773	4,378.79

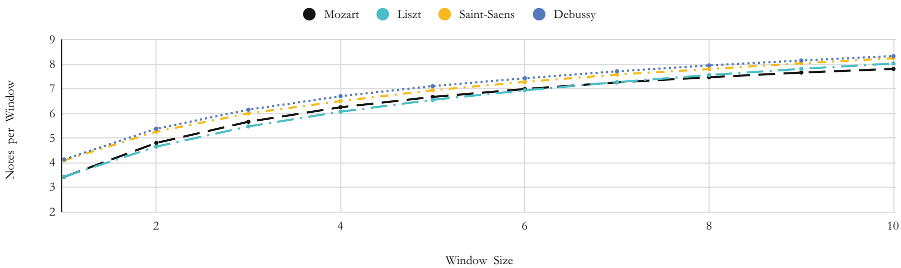

Fig. 1. Average number of notes with different window sizes, where window size is in quarter-note beats.

in the Mozart model were categorized as diatonic collections, whereas this was between 10%–14% for the other three composers, reflecting a larger variety of pitch-class collections for Liszt, Saint-Saëns, and Debussy. For a closer look, we looked at the top 5 most frequent pitch-class vectors for each composer (Table 3). Diatonic collections occupied the majority of positions in the table. For Liszt and Debussy, the most frequent pitch-class vector was the aggregate (all 12 pitches in a vector). In fact, the aggregate pitch-class vector for Liszt was more frequent than the next four diatonic sets combined. For Saint-Saëns, the most common pitch-class collection was an empty vector. These results show that Liszt and Debussy cycle through pitch classes at a faster notated rate than the others and that Saint-Saëns often has longer durations with no new pitch classes introduced.

Table 2. Frequency and percent of diatonic collections in windows.

Composer	Total no. of windows	Total major	Total harmonic minor	% diatonic
Mozart	464,229	98,490	11,806	27%
Liszt	82,179	6,442	2,282	11%
Saint-Saëns	40,643	4,645	1,093	14%
Debussy	13,923	1,931	125	15%

2.3 Model Parameters

Using the Gensim Python library, word2vec was then used to find embeddings for the resulting set of pitch-class vectors, mapping pitch-class vectors onto scale

Table 3. Top 5 frequent pitch-class vectors and their respective frequency sorted by composer.

Composer	#1 frequent	#2 frequent	#3 frequent	#4 frequent	#5 frequent
Mozart	C maj: 18,347	Bb maj: 17,580	D maj: 17,395	G maj: 16,189	Eb maj: 15,055
Liszt	All PCs: 3,187	F# maj: 927	C maj: 798	E maj: 726	A maj: 620
Saint-Saëns	No PCs: 1,067	C maj: 921	Eb maj: 853	All PCs: 689	E maj: 588
Debussy	All PCs: 487	E maj: 352	C maj: 258	A maj: 207	F maj: 186

embeddings. The scale embeddings here were set to $2^5 = 32$ dimensions. If a pitch-class vector c was within 6 beats of pitch-class vector w, it was included as a context pitch-class vector for w, notated as "Context Windows" and "Target Windows." For each target window, Word2vec maximizes the probability (using negative sampling) of context windows within 6 beats. We trained four models, one on each of the four composers, where each model iterated over the composer-corpus 20 times.

3 Properties of Embeddings

3.1 The Circle of Fifths According to the Mozart Model

Given the high dimensionality of embeddings, we used *t-Distributed Stochastic Neighbor Embedding* (t-SNE) to visualize the embeddings in a 2-dimensional space [10]. The left side of Fig. 2 shows the Mozart model's scale embeddings plotted in 2 dimensions. This figure resonates with music-theoretical claims about the circle of fifths (COF). The COF is a metric for scale distance: the further a scale on the COF, the more distant it is [5]. However, not all fifth-adjacent scales are equidistant. For example, A major is much closer to D major than it is to E major. This was likely the effect of absolute key: Mozart wrote more frequently in D major than in A major, and since pieces often modulated to their dominant in the classical period, A was drawn closer to D.

Including harmonic minor scales and clustering embeddings with euclidean distances still captured the COF (right side of Fig. 2). Each color in the figure are quadrants on the circle of fifths. There is only one minor scale–C# minor (marked with an asterisk)–located far away from its relative major.

3.2 Composer Embeddings Correlated with the Circle of Fifths

Calculating the cosine distance from C major to each other major scale for each of the four models resulted in Fig. 3. Beside the embedding distances, we plotted the distances according to the COF (normalized). COF distances were calculated based on a unit circle with 12 equidistant points from the center, measured with both angular and Euclidean metrics. Notably, each model in Fig. 3 makes an arch, signifying that distance from C gets further around the COF until reaching its diametrically opposed point (F#/Gb). The models roughly approximate the

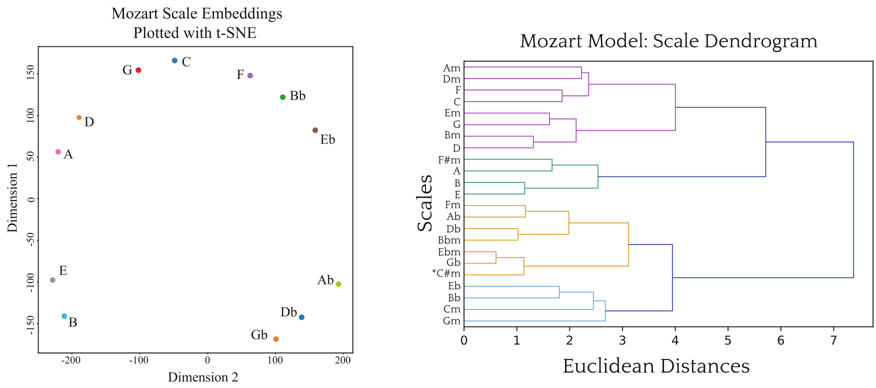

Fig. 2. Mozart model: major scale embeddings clustered with t-SNE (left) and major/minor scale embeddings clustered k-hierarchical clustering (right).

COF distances, and correlations also values also verify this claim. Correlations with the angular and Euclidean COF distances are, respectively, Mozart(.84, .9), Liszt(.78, .86), Saint-Saëns(.89, .94), and Debussy(.94, .96).

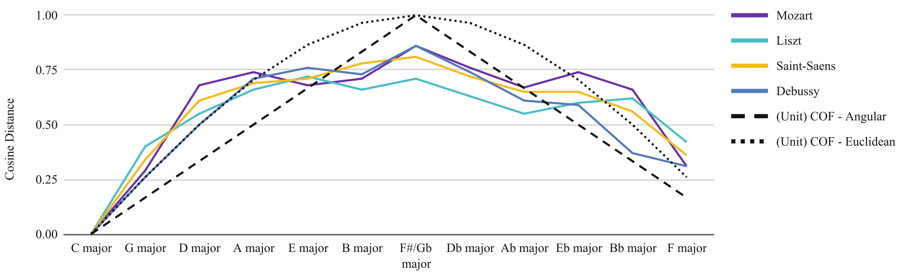

Fig. 3. Cosine distances from C-major scale embedding.

Examining Fig. 3 further reveals stylistic differences between composers. The Mozart model is relatively flat after two steps around the COF–approximately around .68. If we were to generalize this to other keys, this restates a well-known intuition: in the style of Mozart, fifths surrounding the tonic key are the most likely to be modulation goals. Despite having commensurate note-average-per-window values (Fig. 1), the Liszt model correlated less with the COF than the Mozart model (or any of the models, for that matter). This reveals his stylistic tendency to modulate to third-related keys [4,9]: Fig. 3 shows that Ab major–a scale 4 flats away on the COF–is closer to C than any other scale besides scales with keys with a single flat (F) or sharp (G).

Surprisingly, the Debussy model had a higher correlation with the COF (angular and Euclidean) than any of the other models. This could represent the disentangling of the tonic-dominant key-relationship dichotomy: whereas other composers consistently modulate to fifth-related scales, drawing their fifth-related embeddings close together, the dominant's relationship is weighted less in Debussy's music.

4 Conclusion

We have shown that scale embeddings, encoded as pitch-class vectors, capture style-specific musical intuition about scale relationships within common-practice art music. The composers modeled here treat scales differently, resulting in nuanced distances between embeddings. Future avenues for research should first more accurately encode pitch-classes to correspond with scale-degree salience, and might then study the relationship between chord and scale embeddings, and, perhaps, how this interaction changes between and within composers over time.

References

1. Alvarez, A.A., Gómez-Martin, F.: Distributed vector representations of folksong motifs. In: Montiel, M., Gómez-Martin, F., Agustín-Aquino, O.A. (eds.) International Conference on Mathematics and Computation in Music, pp. 325–332. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-030-21392-3_26
2. Brunner, G., Wang, Y., Wattenhofer, R., Wiesendanger, J.: JamBot: Music theory aware chord based generation of polyphonic music with LSTMs. In: 2017 IEEE 29th International Conference on Tools with Artificial Intelligence, pp. 519–526. IEEE (2017)
3. Chiu, M.: Macroharmonic progressions through the discrete fourier transform: an analysis from Maurice Duruflé's requiem. Music Theory Online **27**(3) (2021)
4. Kopp, D.: Chromatic Transformations in Nineteenth-Century Music. Cambridge University Press, Cambridge (2002)
5. Krumhansl, C.L., Kessler, E.J.: Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys. Psychol. Rev. **89**(4), 334–368 (1982)
6. Madjiheurem, S., Qu, L., Walder, C.: Chord2vec: learning musical chord embeddings. In: Proceedings of the Constructive Machine Learning Workshop at 30th Conference on Neural Information Processing Systems. Barcelona, Spain (2016)
7. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word represetations in vector space. arXiv preprint arXiv:1301.3781 (2013)
8. Nikrang, A., Sears, D.R., Widmer, G.: Automatic estimation of harmonic tension by distributed representation of chords. In: Aramaki, M., Davies, M.E.P., Kronland-Martinet, R., Ystad, S. (eds.) International Symposium on Computer Music Multidisciplinary Research, pp. 23–34. Springer, Cham (2017). https://doi.org/10.1007/978-3-030-01692-0_2
9. Riemann, H.: Große Kompositionslehre, vol. I. W. Spemann, Berlin (1902)
10. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(11), 2579–2605 (2008)
11. White, C.W., Quinn, I.: The Yale-classical archives corpus. Empirical Musicology Rev. **11**(1), 50–58 (2016)