РАЗДЕЛ IV. МОДЕЛИРОВАНИЕ

Мельников В.Е.

Алгоритмы генерации музыкальных произведений в заданном пользователем жанре

*Сибирский государственный университет науки и технологий имени академика
М.Ф.Решетнева
(Россия, Красноярск)*

doi:10.18411/sciencepublic-12-08-2020-08

idsp: sciencepublic-12-08-2020-08

Вопросом автоматизации сочинения музыки занимались еще в средневековье. Так, например, Кеплер превращал траектории движения планет в музыку, а Моцарт и его современники изобрели игру в «музыкальные кости», суть которой заключалась в том, что при помощи бросков кубиков из большой таблицы выбирались такты, а затем из этих тактов составлялись менуэты. Но настоящее развитие алгоритмическая генерация музыки получила только с появлением компьютеров.

Однако, до сих пор нету разумных научных теорий, почему одни последовательности нот воспринимаются как музыка, а другие нет. Философским вопросом является и то, являются ли эти зависимости естественными, а не сформировавшимися в ходе исторического процесса. Так или иначе, в музыке содержатся некие закономерности, интуитивно понятные всем людям, математически строго сформулировать или смоделировать которые пока не удалось.

Самыми используемыми в последние несколько десятилетий подходами к решению задачи алгоритмической композиции можно назвать несколько техник – L-системы (системы, основанные на знаниях), клеточные автоматы, генетические алгоритмы, нейронные сети (обучающиеся системы), кроме того, существует подход генерации музыки с использованием контекстно-зависимых грамматик. Однако из вышеперечисленных подходов музыку в заданном пользователем жанре позволяют генерировать лишь L-системы, нейронные сети и контекстно-зависимые грамматики.

L-системы.

В основе работы L-систем лежит набор правил замещения, рекурсивно применяющийся на начальную строку символов и интерпретирующий конечную строку, как структурные элементы организма. Правила замещения определяют как каждый конкретный символ в текущем поколении должен быть перемещен. Базовое описание контекстно-независимой системы заключается в формуле:

$$G = |A, P, \alpha|,$$

где A – алфавит системы (набор всех символов включая пустой символ, ϵ), P – это конечный набор правил замещения (определенный в виде входящий символ – выходящий символ, выходящий символ p), и α – символ или строка из алфавита, используемая в качестве начального состояния. Также в работе алгоритма участвуют A^* – набор всех возможных символов строк из A , A^+ – набор всех возможных символьных строк без пустого символа. L-системы не имеют конечных символов, хотя конечный символ может быть эмулирован путем замены текущего символа на самого себя.

Так как чаще всего L-системы используются для моделирования роста растений, раковин моллюсков, пчелиных сот и т.д., основным представлением работы L-системы является графическое. Чаще всего для этого используется так называемый «черепаший язык». Черепаха находится на координатной плоскости и может передвигаться по ней только вперед, но может также поворачиваться на месте. При

движении черепаха способна чертить линию карандашом, или двигаться, не оставляя за собой след. Есть возможность заменять карандаши и тем самым управлять толщиной линии и её цветом. «Черепаший язык» является основой языка программирования LOGO.

Применение L-систем для генерации алгоритмических композиций подразумевает использование вместо символов определенных музыкальных параметров. Например, профессор Португальского университета Педро Пестана назначает алфавит из семи символов — нот, входящих в гамму до мажор, а правила замещения определяет исходя из матрицы переходных вероятностей.

Основным недостатком данной модели является очень большой объем вычислений. Для сети, состоящей из N узлов, существует 2^N различных состояний сети. Для расчетов по данной модели необходимо построить переходную матрицу. Ее размер будет составлять 2^{2N} элементов. Поскольку для каждого элемента матрицы при вычислении вероятности перехода происходит обход всех узлов сети, для каждого из которых также происходит обход всех узлов сети, сложность алгоритма построения матрицы переходных вероятностей составляет $O(N^2 2^{2N})$.

Кроме того, существует сложность определения самой матрицы переходов в рамках целого жанра, так как для разных музыкальных произведений данные переходы могут отличаться.

Контекстно-зависимая грамматика.

Алгоритм основан на построении контекстно-зависимой грамматики по исходной мелодии, а затем генерировании на основе этой грамматики новой мелодии по заданной начальной последовательности.

Для того, чтобы создать мелодию, требуются некоторые правила: например, какие ноты могут идти после текущей последовательности нот.

Все эти правила образуют собой грамматику. Как левая, так и правая часть правила могут быть окружены контекстом из терминальных и нетерминальных символов, поэтому она и называется контекстно-зависимой.

После построения правил встает следующая проблема: если генерировать новую последовательность строго по заданным правилам, то получится в точности исходная мелодия, а требуется создать новую. Поэтому в ряде случаев нужно не доходить до конечного правила, а взять промежуточное.

Плюс построения новой мелодии на основе уже существующей в том, что не нужно самим придумывать правила построения, так как они формируются из мелодии. Кроме того, объем вычислений при использовании данного подхода не такой большой по сравнению с использованием L-систем.

Минус данного подхода заключается в следующем: поскольку правила формируются на основе существующей мелодии, то обучение модели в рамках целого жанра представляется сложным и требует дополнительных мер, так как правила построения в различных мелодиях будут отличаться.

Нейронные сети.

Рекуррентные нейронные сети. Простейшей моделью памяти для нейронных сетей является передача во времени информации о выходах нейронов в её скрытых слоях. В отличие от обычных сетей прямого распространения, нейроны рекуррентной сети получают на вход не только выходы предыдущего слоя, но и собственный выход в предыдущий момент времени.

Для обучения параметров применяется алгоритм Back Propagation Through Time (BPTT) Обычно последовательности делят на фрагменты некоторой фиксированной продолжительности, и на очередном этапе обучения полагают, что в начале этого фрагмента скрытое состояние равнялось 0; градиентами за пределы фрагмента пренебрегают.

Из преимуществ данной нейросети является то, что они способны использовать полученную в прошлом информацию для текущих задач, что оказывается полезным при генерации музыки, так как при добавлении в музыку следующего такта, требуется учитывать ранее сгенерированную последовательность нот.

Однако, рекуррентные сети подвержены проблеме долговременной зависимости, которая заключается в том, что совсем не редко промежуток между необходимой информацией и местом, где она нужна, становится очень большим, и к сожалению, по мере роста промежутка, РНС становятся неспособны научиться соединять информацию.

Модель Long Short-Term Memory(LSTM).

Модель LSTM (Long Short-Term Memory) – одна из самых распространённых моделей памяти. Главная идея LSTM – клеточное состояние. Это напоминает ленту конвейера. Она движется прямо вдоль всей цепи только лишь с небольшими линейными взаимодействиями. Информация может просто течь по ней без изменений. LSTM имеет способность удалять или добавлять информацию к клеточному состоянию, однако эта способность тщательно регулируется структурами, называемыми вентилями.

Вентили – это способ избирательно пропускать информацию. Они составлены из сигмоидного слоя НС и операции поточечного умножения (pointwise multiplication).

LSTM имеет три таких вентиля, чтобы защищать и контролировать клеточное состояние.

Экспериментально показано, что модель значительно успешнее справляется с задачами счёта, что существенно в поиске музыкальных зависимостей. Кроме того, преимущество LSTM сети в том, что оно способно избежать проблему долговременной зависимости.

На сегодняшний день, практически все модели для генерации музыки используют LSTM-сети. Существует много вариаций LSTM. Было проведено сравнение самых популярных LSTM сетей, которое показало, что все они приблизительно одинаковы.

Таким образом, более предпочтительным подходом к генерации музыки в заданном пользователем жанре является использование нейронных сетей.

1. Ф.М. Гафаров, Искусственные нейронные сети и их приложения: учеб. пособие /Ф.М. Гафаров, А.Ф. Галимянов. – Казань: Изд-во Казан. ун-та, 2018. – 121 с.
2. Антон Николаевич Шве́ц – Perl. Примеры программ [электронный ресурс] – 2008 – 2012 – Режим доступа: <http://mech.math.msu.su/~shvetz/54/inf/perl-problems/chLSystems.xhtml>https://algorithmiccomposition.ru/article_entry_l-systems.html
3. Генерация музыки на основе заданного стиля [электронный ресурс] – 2009 Режим доступа: <http://habrahabr.ru/post/69985/>
4. LSTM – сети долгой краткосрочной памяти [электронный ресурс] – 2017 Режим доступа: <https://habr.com/ru/company/wunderfund/blog/331310/>
5. Composing Music With Recurrent Neural Networks [электронный ресурс] – 2015 Режим доступа: <http://www.hexahedria.com/2015/08/03/composing-music-with-recurrent-neural-networks/>
6. Генерация классической музыки с помощью рекуррентной нейросети [электронный ресурс] – 2015 Режим доступа: <https://habr.com/ru/post/382711/>