

КРАТКОЕ ОПИСАНИЕ

Алгоритм на основе теории музыки, преобразований Фурье и методов эвристической оптимизации позволяющий автоматически подбирать аккорды к сложным зашумленным композициям (в частности, рока), а также отдельно заданным мелодиям.

АННОТАЦИЯ

Идея этого проекта родилась из хобби — игры на гитаре. Для игры требуется знать аккорды, которые зачастую несложно найти в интернете. Однако на многие песни аккордов нет. Возникает вопрос — а нельзя ли создать скрипт, способный подбирать их автоматически?

Конечно, идея не нова. Благодаря развитию нейронных сетей существуют проекты, позволяющие за плату получить относительно качественный подбор аккордов. Доступ к готовым нейросетям получить нельзя, собрать достаточный набор данных и натренировать подобную нейросеть мы не можем. Но главное — что нейросеть является „черный ящиком“, интерпретировать «знание», полученное ей в ходе обработки „скормленной“ информации невозможно. А ведь это позволило бы глубже изучить „принцип работы“ музыки, её законы. Хороших ответов на такие вопросы до сих пор нет.

Ввиду этого мы решили подойти к проблеме с другой стороны — с позиций теории музыки. Область запутанная, а те, кто ей занимаются, подходят к ней «гуманитарно», не пытаясь понять и формально обосновать её основные законы, которые лишь декларируются. Как известно, лучший способ понять что-то до конца — научить этому компьютер.

ХОД РАБОТЫ

Изначально задача выглядела простой: достаточно всего лишь сделать преобразование Фурье над файлом с музыкой, определить основные тона, из которых и выделить аккорд. Это можно сделать небольшим количеством кода, особенно — если взять готовую библиотеку. Для этого хорошо подходит python — простой, лаконичный, есть библиотеки на любой случай жизни. Для преобразования Фурье мы взяли библиотеку `librosa`, предназначенную для анализа сложной музыки, которая „из коробки” представляла распределение по музыкальным тонам. Вдобавок хорошо умела распознавать ритм, что решало проблему разделения на такты.

Конечно, всё оказалось не так просто. Аккорды программа подбирать могла, и подбирала так, что каждый на своем такте звучал удовлетворительно, но в контексте песни образовывалась неопиcуемая какофония. Иными словами, подбор аккордов независимо к каждому такту не работает. Нужны более серьезные инструменты.

Для этого был задействовано такое понятие теории музыки, как тональность.

Wikipedia:

Тональность (фр. *tonalité*, от греч. *τόνος* — напряжение, натяжение; от лат. *tonus* тон) в учении о гармонии:

1. Принцип лада, центральной категорией которого является **тоника**. Остальные категории и функции тональности (прежде всего, **доминанта** и **субдоминанта**) прямо или косвенно связаны с тоникой. «Система функциональных отношений иерархически централизована, и тоника пронизывает всю гармоническую структуру»[1].
2. Высотное положение **мажорного** или **минорного лада**. В обозначении тональности указывается **тоника** (она же — **основной тон** тонического трезвучия) и лад (в данном случае, мажор или минор), например: C-dur (до мажор), Fis-dur (фа-диез мажор), a-moll (ля минор), es-moll (ми-бемоль минор) и т. п.

Определение для незнакомых с теорией музыки не очень информативно, поэтому следует пояснить: тональность — что-то наподобие матрицы, на которой мы строим произведение. Из 12 доступных звуковых классов выбираются семь основных, из комбинации которых и будет состоять произведение. Конечно, отклонения от структуры более чем допустимы. Структура этой матрицы задает «стиль» произведения, хотя и не является решающим фактором (можно сравнить с выбором палитры для картины).

Тональность помогает фильтровать аккорды, не допуская тотальной дисгармонии: теперь аккорды определяются для тактов не независимо, а с учётом предыдущих тактов, по которым можно с хорошей точностью и шумоустойчивостью определить тональность. Но этого мало. Получается мешанина из сносных звуков, но с песней имеющих мало общего. Песня сама по себе зашумлена, полна солирующих мелодий и т. п. Ориентироваться на доминирующие тона весьма опрометчиво. Нужно как-то определить дисгармоничность аккорда с песней, а для этого нужно такое понятие, как интервалы.

Wikipedia:

Интерва́л (от *лат.* *intervallum* — промежуток, расстояние; разница, несходство) в **музыке** — соотношение двух **музыкальных звуков** по их **высоте**^[1].

По сути своей, интервал — расстояние между двумя тонами. Но важно, что это не одно соотношение частот (или количество полутонов, что то же самое), но и расстояние в тональности, в „матрице” — сколько между ними нот, входящих в тональность. По расстоянию в тональности идет разделение на примы (сам с собой), секунды (соседние), терции, кварты и т.д. Реальное расстояние (соотношение частот) вносит такие добавки, как "малая секунда", "чистая кварта", "уменьшенная септима" и т.п. В нашем случае названия не имеют значения, но важно понимание того, что свой вклад в звучание вносит не только прямое отношение частот, но и их расположение в тональности.

Здесь мы выдвинули достаточно смелое предположение (не упоминавшееся в литературе по теории музыки): будем считать, что каждый интервал (задаваемый тональностью и реальным отношением частот) имеет свою примерно постоянную дисгармоничность. Для определения общей дисгармоничности рассматриваются попарные дисгармоничности между всеми одновременно звучащими нотами. Суммарная дисгармоничность — некая функция от них, например, сумма (с соответствующим громкости весом).

Музыканты не задавались вопросом, насколько это предположение корректно в общем случае, и, соответственно, не дают на него ответ. Однако, основываясь на нём для упрощения, удалось получить неплохой результат. Возможно, пары интервалов действительно можно рассматривать независимо.

В этом случае задача сводится к минимизации общей дисгармоничности интервалов, возникающих от сочетания подбираемых аккордов со звуками песни. Для "чистых" заданных мелодий этого достаточно, однако для „зашумленных” композиций, заданных файлом с музыкой, аккорды определить сложнее. Так или иначе, погрешность будет большой, учитывая обилие инструментов, мелодий, шума, голосов, обертонов... Требуется дальнейшее упрощение.

Следующее используемая концепция — TSD (тоника, субдоминанта, доминанта).

В тональности каждая нота имеет номер — „ступень”. И если мы начнем строить тональность от разных высот (например, транспонировать мелодию), то ноты поменяются, а общее звучание останется. И именно за это „общее звучание” и отвечают ступени. Самые важные — первая, четвертая и пятая, также называемые **тоника**, **субдоминанта** и **доминанта**. Аккорды, построенные на них, будут теми самыми „три блатных аккорда”, на которых можно сыграть что угодно.

Тоника, основа тональности, дает достаточно спокойное течение мелодии, без лишней „энергии”. *Субдоминанта* отвечает за резкий подъем нестабильности, напряженности, который затем разрешается в *доминанту*, мощную, стабильную концовку. Иногда субдоминанта опускается, иногда мелодия из субдоминанты переходит обратно в тонику, но суть остается прежней:

доминанта почти никогда не переходит в субдоминанту. Описанный механизм действует почти везде, хотя вместо самих аккордов, построенных на TSD, могут быть „аккорды-заменители” — каждому аккорду из TSD соответствует часть из оставшихся, принадлежащих тональности и в обратную сторону. Соответствие таково, что замена а идеальном подборе одного из «заменителей» на соответствующий ему из {T, S, D} почти не уменьшит суммарную дисгармоничность и логику звучания.

Итак, вместо сложных аккордов можно подбирать только основные. Конечно, такой подбор будет звучать очень «топорно», но зато надёжно. Уловить суть мелодии, ее движение программным способом и есть основная проблема.

Напоследок остается „ерунда”. Для подбора предлагается использовать интервалы, минимизируя их дисгармоничность. Но значения дисгармоний остаются неизвестными. Таблиц с точными значениями до нас никто не составлял — в открытом доступе их нет. Попытка взять „на слух” не приводит к хорошему результату: даже из трех аккордов подбирается нечто несуразное. Для определения значений можно использовать эвристическую оптимизацию (в данном случае — **метод симуляции отжига**). Максимизируя совпадение подобранных аккордов с заранее известными, меняем дисгармоничность тех или иных интервалов, пока не получим хороший результат.

Связывая все созданные структуры воедино, получаем хороший результат и для песен, не использовавшихся в оптимизации, что, несомненно, является успехом.

НЕКОТОРЫЕ МОМЕНТЫ РЕАЛИЗАЦИИ

ПОДБОР ТОНАЛЬНОСТИ

В случае песни (музыкального файла) нужно выделить основные используемые ноты, и найти "главные". Для этого достаточно выбрать из 36 (12 нот · 3 основных используемых типа) тональностей ту, в которой максимальна доля нот тональности, в особенности TSD (с большим весом, тоника обладает максимальным весом). Задача решается перебором: для каждой тоники и типа рассчитывается `Tikelyhood`, выбирается лучшая тональность.

В случае мелодии задача усложняется, так как большого массива нот нет. Зато есть четкая последовательность нот, по музыкальным разрешениям которых можно угадать характерность той или иной тональности. Аналогично предыдущему варианту, достаточно перебрать тональности.

ОПРЕДЕЛЕНИЕ ДИСГАРМОНИЧНОСТИ ИНТЕРВАЛОВ

С учетом медленной скорости определения аккордов и невысокой требуемой точности, здесь достаточно использовать упрощенную вариацию отжига. Случайно изменяются значения дисгармонии, соответствующие каждой паре (количество ступеней; количество полутонов), а также параметров дисгармонии интервалов с не входящими в тональность нотами и DS-, DT- и т.п. переходами.

ВОСПРОИЗВЕДЕНИЕ

Как ни странно, несмотря на обилие библиотек на `python`, для быстрой и качественной оценки подбора пришлось создавать собственный низкоуровневый проигрыватель для удобной работы со звуком. Качеством похвастаться сложно, однако без проигрывателя невозможно представить этот проект.

ВЫВОДЫ

Нам удалось добиться адекватного качества подбора. Конечно, таким алгоритмам далеко до нейронных сетей, но они дают гораздо более глубокое понимание происходящего. Например, коэффициенты дисгармоничности интервалов (не приводим подробный анализ, так как пока есть сомнения в их точности и правильности, хотя уже есть интересные наблюдения). Несомненно, впереди еще много работы, но такие идеи позволяют если не догнать нейронные сети, то хотя бы получить существенные количественные знания в, казалось бы, полностью гуманитарных областях.

АВТОРЫ

Андрей Ситников — общая идея, теория музыки, идея реализации.

Владимир Латыпов — общая идея, программная реализация.