DISENTANGLED MULTIDIMENSIONAL METRIC LEARNING FOR MUSIC SIMILARITY

概要

- 楽曲を、その類似度に着目して埋め込みたい。
- 類似度には様々な指標がある。（ジャンル、ムード、テンポ、etc）これらを埋め込んだ中間表現の次元の一部に対応させるように学習させる。（disentanglement）
- 今回はジャンルとムードとテンポと使用楽器の4つ(?)に着目し、楽曲をある一点に埋め込む関数を学習する。
- データは人手のラベルと、テンポについては既存の自動ムード割り当てマシーンを使う。
- track-regularization：どの類似度指標においても良い埋め込みを行うための技術？

導入

- モチベーションは楽曲検索システムの利便性の向上（あとは似たような楽曲に置換したい場合も。）
- 強調フィルタリングとかが多い
- 先行研究は、ベクトル量子化 [4]、線形メトリック学習 [5, 6, 7]、さらに最近では、人間の類似度ラベル [11]、アーティストラベル [12]、トラックラベル [13]、ゼロショット学習 [14] の文脈でのタグを使用したディープメトリック学習 [8, 9, 10] などがある。
- ここでは、複数の類似度指標に基づく楽曲の埋め込みを行う。データの教師は人手のラベル（ジャンルなど）と自動で付与されたラベル（テンポなど）の両方を用いる。

学習モデル

- triplet loss 使うよ！
- 条件付き類似度ネットワーク(Conditional Similarity Networks: CSN) [15]
  - サイズ の埋め込み空間に適用されるマスキング関数　　を導入
  - は類似度指標(コンディション？)を表す。（c.f.　　→ジャンルの類似度）
  - 中間表現　　にマスク　をかけるとあるコンディション の類似度をを表現する次元のみ抽出できるので、これを使ってtriplet-lossを設計する。

- track-regularization
  - 同じ曲の別のフレームから作成されたサンプルを用いて、「anchorに対し同じ曲のフレームならpositive、別の曲からのフレームならnegative」と言うtripletデータを作成し、そのLossを　　を「track-gegularization」と呼ぶ。
  - データセットのラベル（ジャンルなど）から作成された通常のtriplet-loss　　にこのtrack-regularizationを正則化項として加える。

- あとマスクをしないで学習させちゃうって方法も考えられるヨネ！

実験

データセット

- Million Song Dataset (MSD) [16]を使用。
- コンディション（類似度指標）はジャンル、ムード、楽器、テンポにの四つに特定。
- テンポ以外のラベルはついている。テンポはMadmom Pythonライブラリ[18, 19]を使用
- track-regularization項作成時の楽曲の分割をする時は、サンプルどうしが50%以上重ならないように。
- 一つの曲のうちの３秒を一つのサンプルとして、入力はメル周波数表現に変換した行列としている。

モデル構造

- ニューラルネットの構造は画像処理でよくある構造（conv-batchnorm-maxpool ブロック、ResNet [23]、Squeez-and-Excitation [24]、Inception [25]）を採用。
- 今回の中間表現は256次元で、４つのコンディションをそれぞれ64次元づつ対応させる。
  - mask自体も学習対象にしてみたが、効果はなかった。

評価

- anchorとpositive間の埋め込み距離がanchorとnegative間の距離よりも小さいか(スコア1)、または大きいか(スコア0)をテストし、テストデータ全体で平均したもの。
- 人手でtripletのポジネガを付け直して判定。（本当に直感にあってるかを確かめた）
  - 割と真摯なデータセットを作ったみたい。

ベースライン

- 類似性に基づく音楽検索と自動タグ付けの両方に使用されているベクトル量子化手法

結果

- 意外とtrack-regularizationをすると個々のコンティションの距離学習もよりうまくいっているみたいだった。

Point

- 中間表現zに対してマスクをかけ、一部の次元のみに対してある一つの類似度の指標（例えばジャンル）の距離学習を行う→disentanglement

掘りどころ

- 暗黙の了解で、一つの曲はずーっと同じジャンル、ムード、etcを保っているとしている？
    - 楽曲の状態の遷移を考慮できたら良いよね。
- 入力MFCCじゃなくてそのまま楽曲データにして、フィルタを学習させてみたらどうでしょ。