Classifying Derivative Works with Search, Text, Audio and Video Features

Jordan B. L. Smith, Masahiro Hamasaki and Masataka Goto
National Institute of Advanced Industrial Science and Technology (AIST), Japan

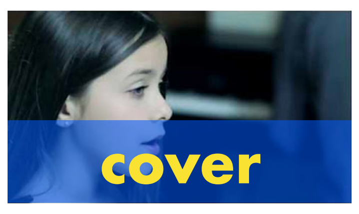
1. Motivation

Goal: to find derivative works of popular songs posted online.

Music videos and their derivatives are hugely popular on YouTube and other services.

Audio Content (AC) and **Video Content (VC)** are independent, and both important:

youtu.be/rYEDA3JcQqw
AC: Original audio
VC: Official music video



youtu.be/a7UFm6ErMPU
AC: Cover (new arrangement)
VC: Live (performance)

youtu.be/n7xoVgmQVDQ
AC: Live (perf. by orig. artist)
VC: Live (concert)

youtu.be/8WCb58e14Mo
AC: Remix (samples orig.)
VC: Still image

youtu.be/LP4kBLxW5RQ
AC: Original audio
VC: Lyrics (as slideshow)

youtu.be/oGqFexs3xkk
AC: Original audio
VC: Dance performance

Problem: text search for derivatives gives many errors: a search for "covers" often gives remixes; a search for "live" can give covers; etc.

Solution: build a system to re-classify search results based on **search** rank, **text** [i.e., video title], **audio** and **video** features.

2. Related work

Techniques exist to identify specific types of derivatives:

Fingerprinting: peaks in audio signal spectrum are robust to noise and can be used to identify exact copies of audio.

Remix detection: look for any matching fingerprints, possibly time- or frequency-distorted.

Cover song detection: characterize chord and/or melody structure, then look for matches.

Live song identification: use a variation on fingerprinting that is feasible if artist is known.

But these methods were not designed and are not guaranteed to *distinguish* between types of derivatives.

NATIONAL INSTITUTE OF
ADVANCED INDUSTRIAL SCIENCE AND TECHNOLOGY (AIST)

This work was supported in part by JST ACCEL Grant Number JPMJAC1602, Japan.

3. Proposed system

Inputs: keyword list and query song/artist

Keywords
"choreo"
"concert"
"cover"
"dance"
"guitar"
"karaoke"
"live"
"lyrics"
"piano"
"remix"
"tutorial"

Query
'Gotye Featuring Kimbra, "Somebody That I Used To Know" [keyword]'

YouTube

All search results
```
8UVNT4wvIGY, d9NF2edxy-M, DqRC5tquyU0,
6YzGOq42zLk, IAVL1JBjtoU, vOlF5YRV8Ow,
doJbORheLzw, te0Q1XwaTyo, hOKuAigsrec,
00KnR6CltzE, _g9bBO1lfu8, UzSbbzYGECQ...
```

Extract features:
- Search rank
- Text features
- Video features

"STV"
SVM classifier

Initial predictions
```
8UVNT4wvIGY, orig./orig.
d9NF2edxy-M, cover/live
DqRC5tquyU0, orig./lyric
6YzGOq42zLk, live /live
IAVL1JBjtoU, orig./dance
...          ( AC / VC )
```

AC exemplars
```
Best orig.: 8UVNT4wvIGY
Best live : 8V6241UyHOE
Best remix: -d_PZikUT2I
...
```

Extract audio features
(compare audio to class exemplars)

"STVA"
SVM classifier

Refined predictions
```
8UVNT4wvIGY, orig./orig.
d9NF2edxy-M, cover/live
DqRC5tquyU0, orig./lyric
6YzGOq42zLk, live /live
IAVL1JBjtoU, remix/dance
...          ( AC / VC )
```

Phase 1: make initial prediction of AC and VC for each video.

Treat the highest-confidence examples for each AC category as **class exemplars.**

Phase 2: compare audio files to class exemplars, and obtain **refined predictions**.

4. Features

To estimate the AC and VC of videos, we use an SVM with a multi-modal set of four features:

Search: a video can appear in several keyword searches. We use a video's rank in each search. E.g., a dance video might have ranks:

Keyword:	main	choreo	concert	cover	dance	guitar	..	tutorial
Rank:	22	1	361	–	3	–	..	–

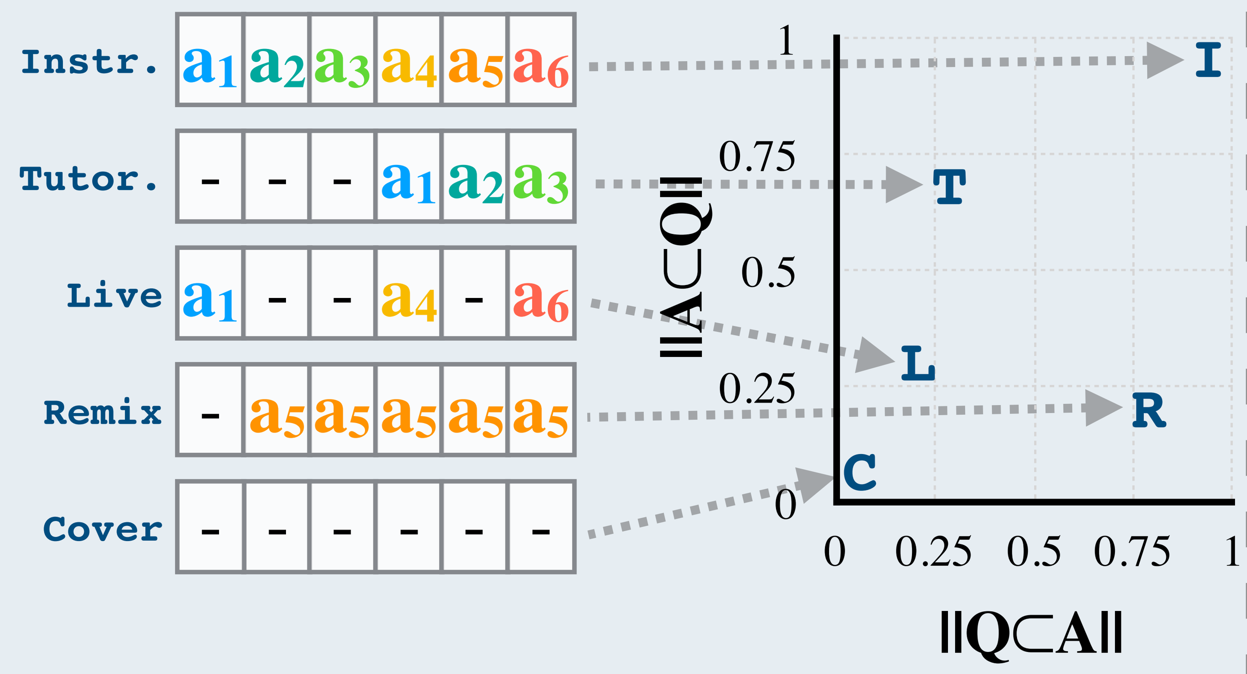
Text: words that appear in many titles are likely to relate to AC or VC: e.g., "acoustic"→`cover`, "tour"→`live`, "vs"→`remix`, etc. From an unlabelled set of ~150,000 video titles, we learn latent topics. We can then convert each title to a "topic strength" vector. To help the learning process, we detect terms from dictionaries of places, names, instruments and genres.

Video: common video features (e.g., brightness, colour variance, optical flow). To detect lyrics, we do text-recognition using *Tesseract*.

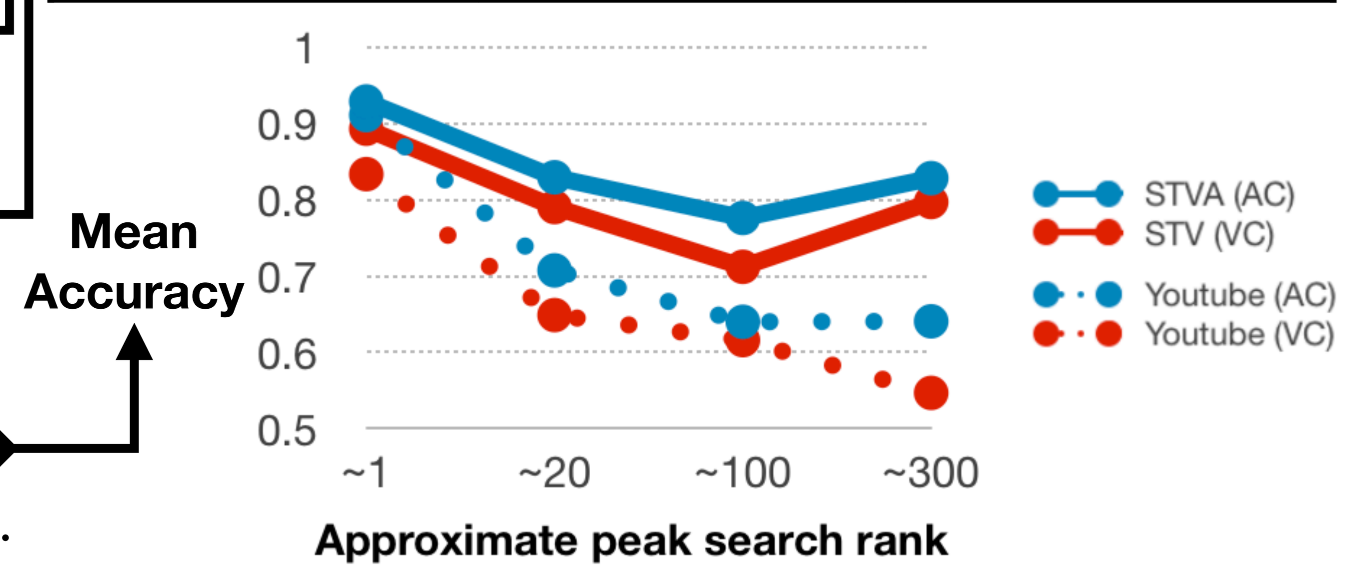
Audio: we use audio fingerprinting to match 10-second snippets to the original song. The distribution and quality of matches found is different for different types of derivatives.

Original audio A, *cut into chunks:* $a_1\ a_2\ a_3\ a_4\ a_5\ a_6$…

We characterize each song Q with two values:
- the fraction of snippets in song Q ($q_1q_2q_3q_4$...) that have a match in the original, A: $\|Q\subset A\|$
- the fraction of snippets in song A ($a_1a_2a_3a_4$...) that have a match in song Q: $\|A\subset Q\|$



Features:	AC accuracy	VC accuracy
S(earch)	0.699	**0.705**
T(ext)	**0.781**	0.690
V(ideo)	0.416	0.505
A(udio)	0.623	0.552
ST	0.822	0.767
STV	0.815	**0.804**
STVA	**0.847**	0.781
YouTube Baseline	0.746	0.685



5. Evaluation

Data: We discovered 160,000 derivatives of the BillBoard Top 100 songs of 2012. We labeled 562 videos related to 10 unique songs, and used the rest (titles & search ranks) for unsupervised training.

Ground truth: manually annotated **AC** and **VC**:

- **6 AC categories**: `official, cover, instrumental, live, remix, tutorial`.
- **9 VC categories**: `official, dance, karaoke, live, lyrics, slideshow, still image, tutorial, other`.

Baseline: a decision tree (DT) using peak search rank from YouTube searches. E.g., if video has the highest rank for the "remix" keyword search, then **AC→`remix`** and **VC→`still`**.

Results:

- <u>Text features</u> alone match the performance of the YouTube baseline.
- Combining all features (<u>STVA</u>) in two-phase approach, we surpass baseline by 10%.
- Our system is also **more robust than baseline**: classification of deep search hits still accurate.