

MIR_EVAL: A TRANPARENT IMPLEMENTATION OF COMMON MIR METRICS MARL



Colin Raffel, Brian McFee, Eric J. Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, and Daniel P. W. Ellis

mir_eval is a Python module which includes metrics for the following tasks: Beat detection, chord estimation, pattern discovery, structural segmentation, melody extraction, and onset detection. It's easy to use in Python:

```
import mir_eval
# Load in beat annotations
reference_beats = mir_eval.io.load_events('ref_beats.txt')
estimated_beats = mir_eval.io.load_events('est_beats.txt')
# scores will be a dictionary where the key is the metric name and the value is the score achieved
scores = mir_eval.beat.evaluate(reference_beats, estimated_beats)
```

or without Python, using the included evaluator scripts:

```
> ./beat_eval.py ref_beats.txt est_beats.txt -o scores.json
> cat scores.json
 {"F-measure": 0.6216216216216, "Cemgil": 0.36267669947376, "Cemgil Best Metric Level": ...
```

We also quantitatively compared it to MIREX with some intriguing results!