MIREX TAGGING CONTEST: A NN-LIKE APPROACH (DRAFT)

Thierry Bertin-Mahieux, Yoshua Bengio, Douglas Eck
University of Montreal, CAN
{bertinmt,bengioy,eckdoug}@iro.umontreal.ca

ABSTRACT

We present one of our submission to MIREX 2008 audio tag classification contest. The algorithm uses euclidean distance between features in order to tag new examples.

1 INTRODUCTION

We shortly describe our submission to MIREX 2008 audio tag classification contest [4] based on euclidean distance. In this paper we go over the features we use, the classification algorithm, and how do we create the output files. Finally, we briefly discuss why we would submit such a simple algorithm to MIREX.

2 ALGORITHM

2.1 Audio Features

We compute aggregate features [1] over $3s$ segments. Features consist of a constant-Q spectrogram, an autocorrelation vector, MFCC and its first and second derivatives (delta-MFFC and delta-delta-MFCC). The size of an example (features from one segment) is 466.

2.2 Euclidean distance

We consider the distance between 2 song segments to be the euclidean distance between the features of those 2 songs. The affinity between a segment and a tag is the ratio:

$$ratio = \frac{distance\ to\ closest\ negative\ example}{distance\ to\ closest\ positive\ example} \quad (1)$$

Positive and negative examples always refer to a particular tag. We looked at the KNN code in the Monte Python package [1] in order to have a time-efficient algorithm.

2.3 Output

We output a continous value for each $(song, tag)$ pair that represents *affinity* between that song and that tag, see equation 1. We also output a binary decision for every $(song, tag)$ pair. We find a threshold for each tag on values output on a validation set (approximately 15% of the training examples) by minimising $F1 - score$ [2].

[1] `http://montepython.sourceforge.net/`

[2] `http://en.wikipedia.org/wiki/F-score`

3 DISCUSSION

This method is not intended to become the state-of-the-art, previously pusblished algorithms [2, 5] have probably more potential. But as it is the first year of this contest at MIREX, we felt it was important to compare a wide variety of algorithms.

Furthermore, on our home-made testing framework inspired by the one in the contest, this NN-like approach performs comparatively as well as other methods. We believe that this is partly due to the small size of the dataset (approximately 2500 songs by 500 artists) and the inherent producer effect. In a similar experiment, Mandel and Ellis [3] could not see any significant learning on such a small dataset. Thus, we feel confident that this algorithm will provide a reasonable benchmark for comparison.

4 ACKNOWLEDGEMENTS

The authors would like to thank everyone that pulled this contest together, in particular the IMIRSEL team, Kris West, Michael Mandel and Edith Law.

5 REFERENCES

[1] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl. Aggregate features and AdaBoost for music classification. *Machine Learning*, 65(2-3):473–484, 2006.

[2] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green. Automatic generation of social tags for music recommendation. In *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, MA, 2008.

[3] M. Mandel and D. Ellis. Multiple-instance learning for music information retrieval. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR 2008)*, 2008.

[4] MIREX: Music Information Retrieval Evaluation eXchange, information at `www.music-ir.org/mirex/2008/index.php`.

[5] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech & Language Processing*, 16(2), 2008.