Deep learning architectures for music audio classification: a personal (re)view

Jordi Pons

jordipons.me – @jordiponsdotme

Music Technology Group
Universitat Pompeu Fabra, Barcelona

Acronyms

MLP: multi layer perceptron ≡ feed-forward neural network
RNN: recurrent neural network
LSTM: long-short term memory
CNN: convolutional neural network
BN: batch normalization

..the following slides assume you know these concepts!

Outline

Chronology: the big picture

Audio classification: state-of-the-art review

Music audio tagging as a study case

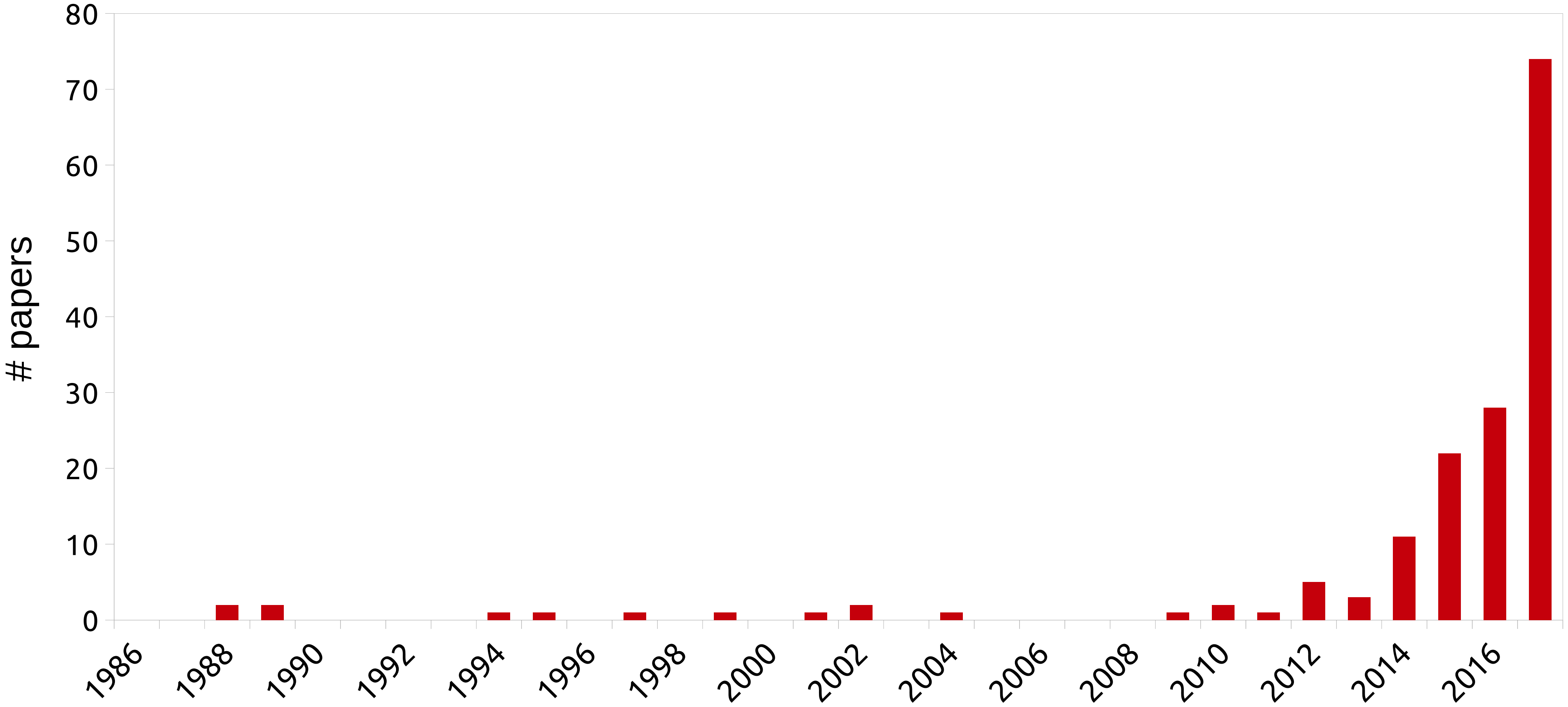
Outline

Chronology: the big picture

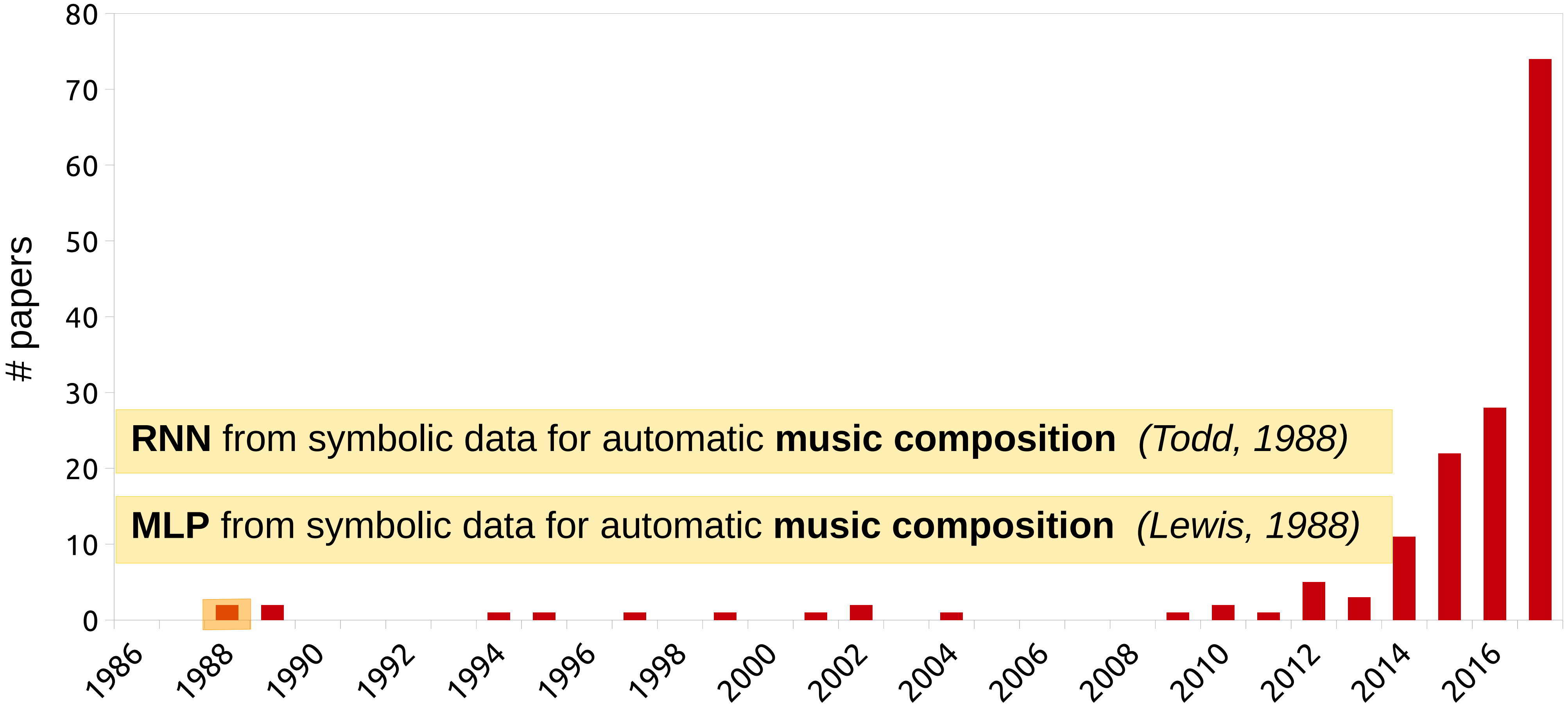
Audio classification: state-of-the-art review

Music audio tagging as a study case

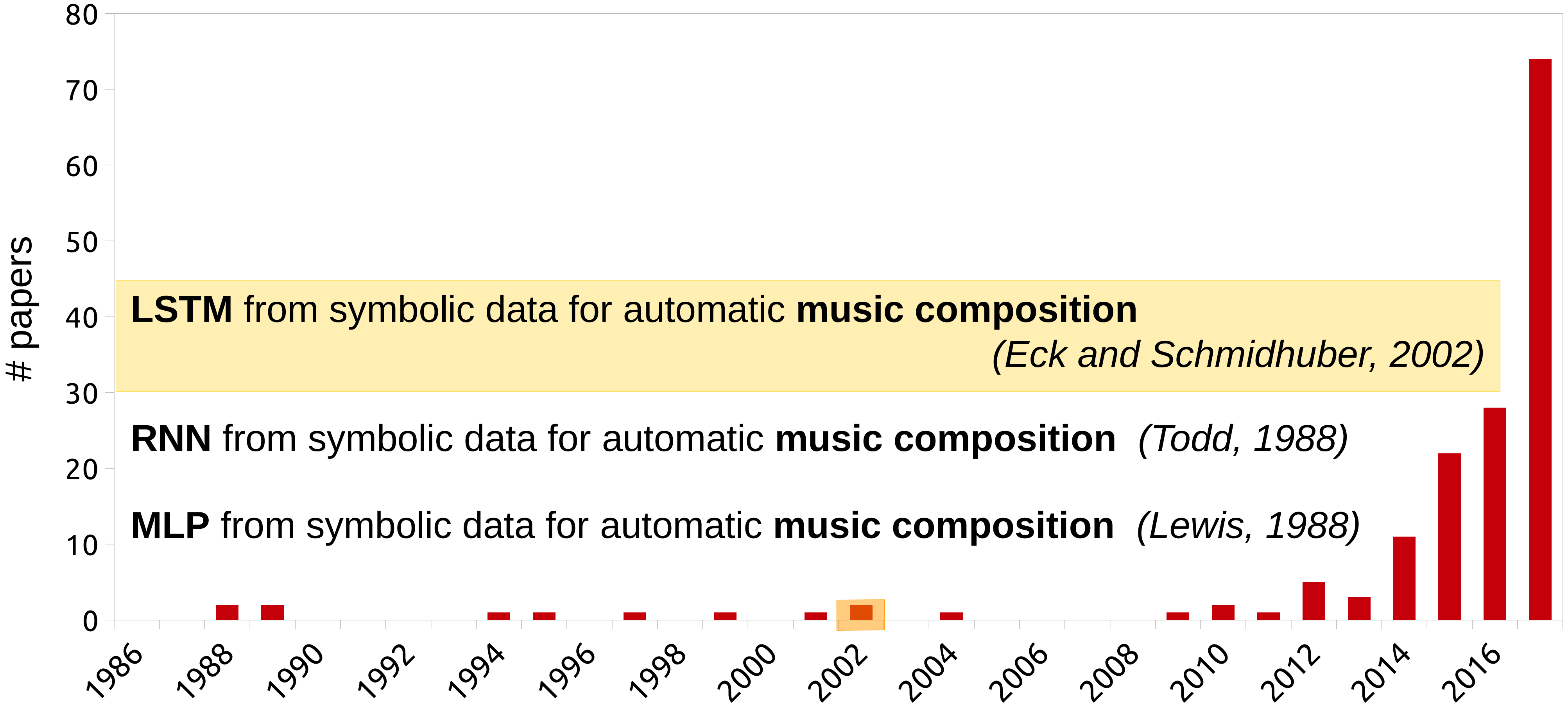
“Deep learning & music” papers: milestones



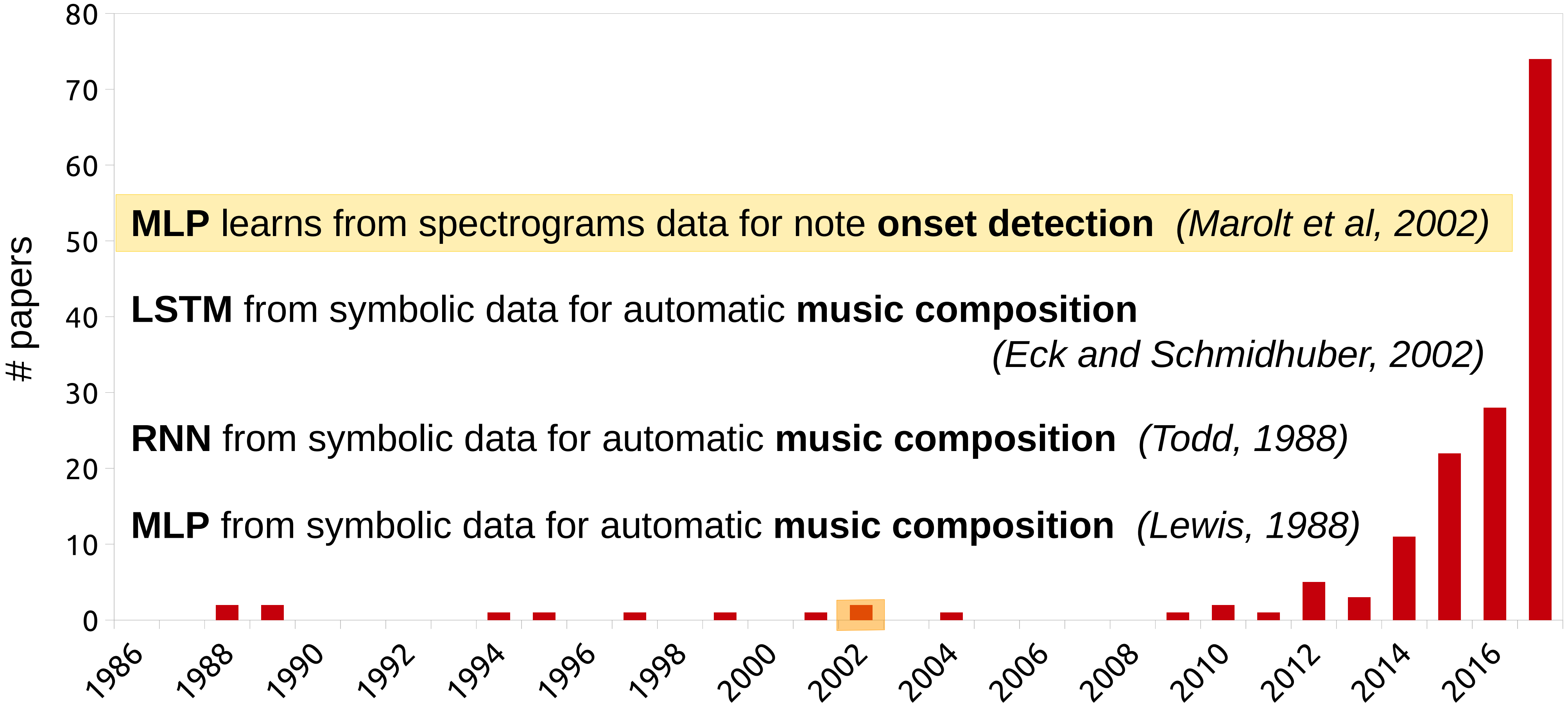
“Deep learning & music” papers: milestones



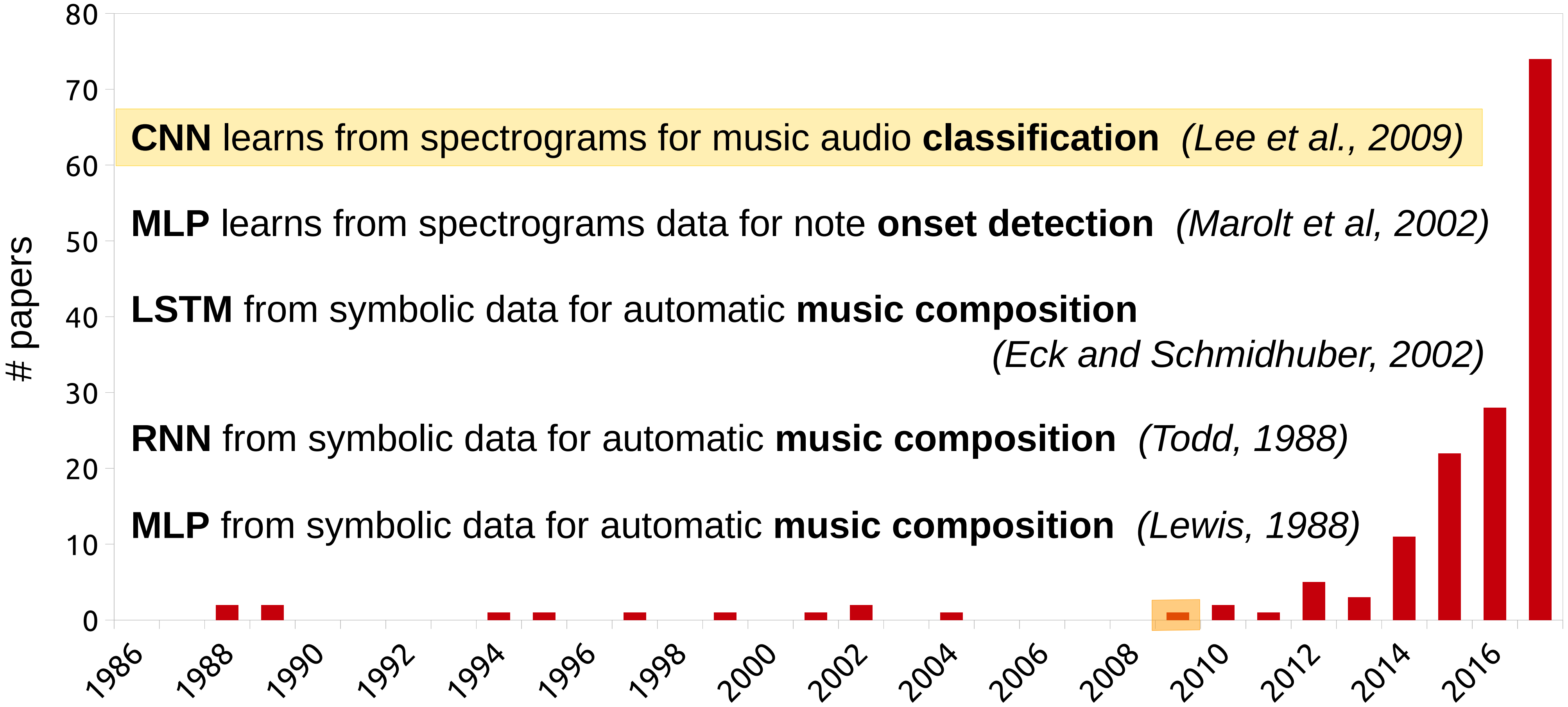
“Deep learning & music” papers: milestones



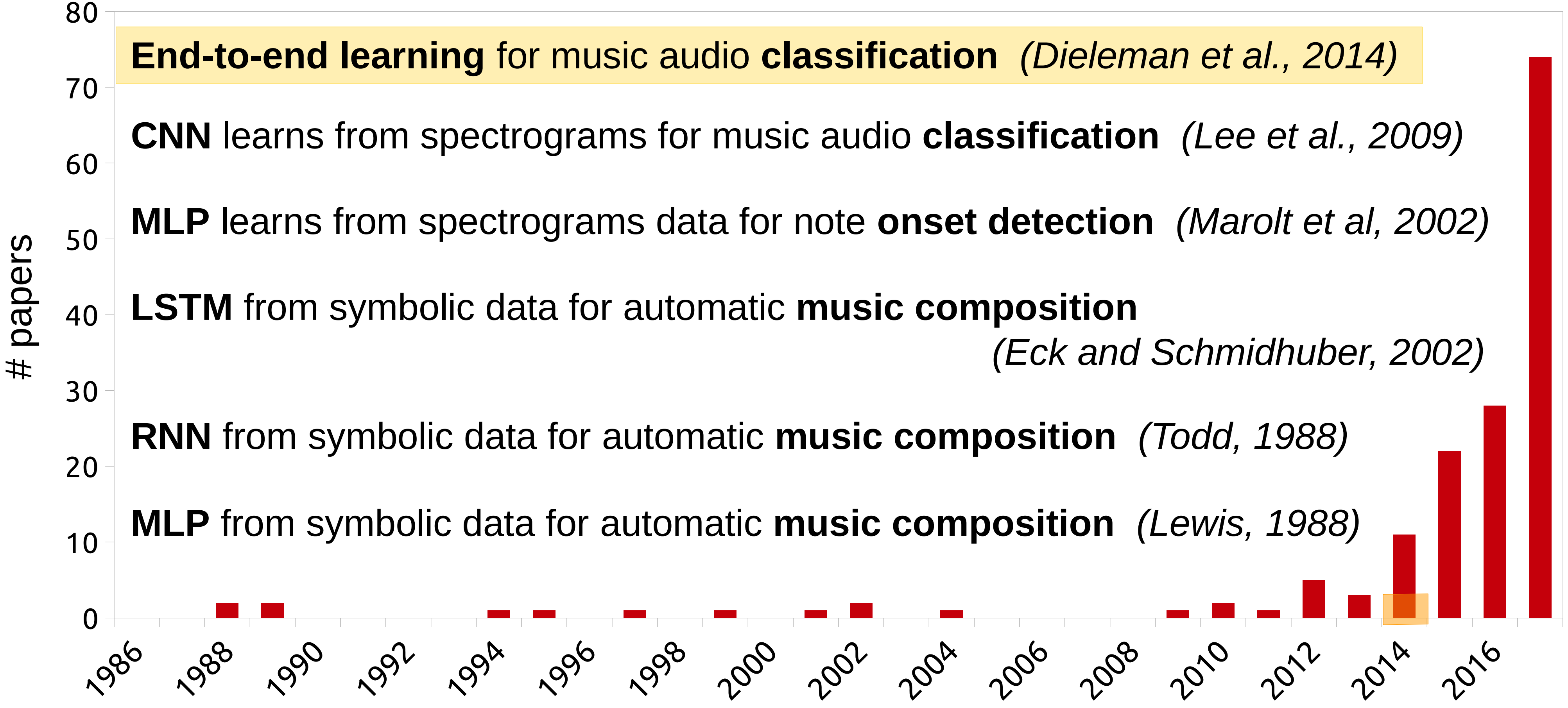
“Deep learning & music” papers: milestones



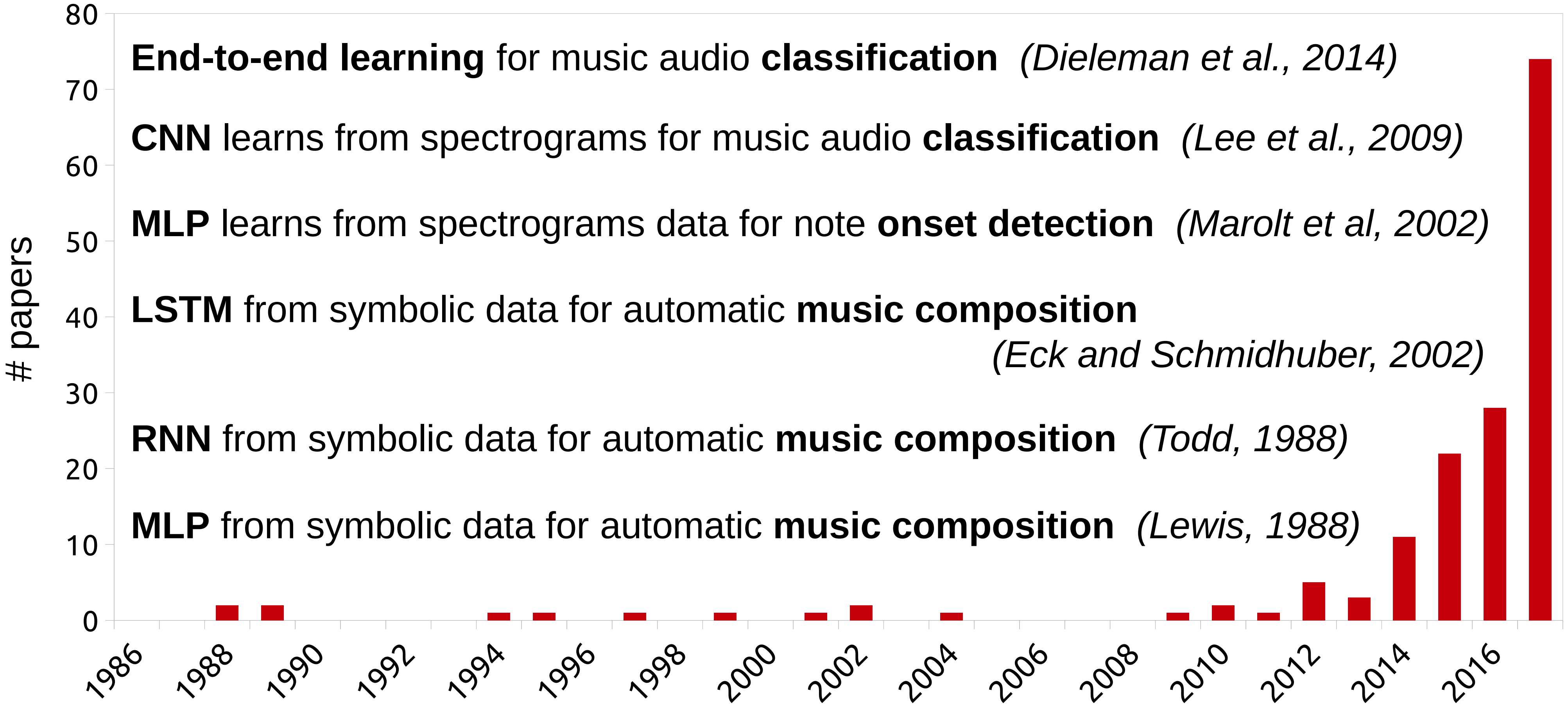
“Deep learning & music” papers: milestones



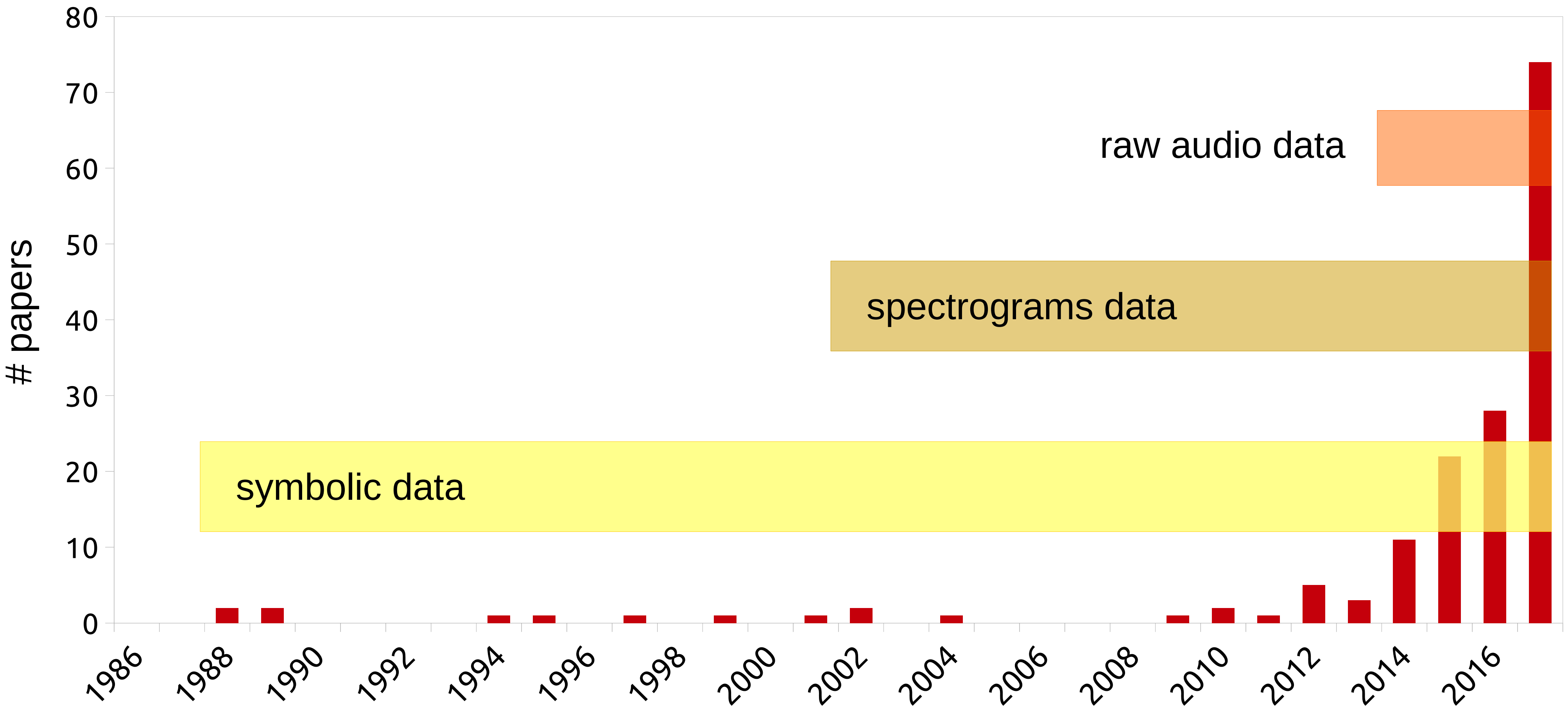
“Deep learning & music” papers: milestones



“Deep learning & music” papers: milestones



“Deep learning & music” papers: data trends



“Deep learning & music” papers: some references

Dieleman et al., 2014 – **End-to-end learning for music audio**
in International Conference on Acoustics, Speech and Signal Processing (ICASSP)

Lee et al., 2009 – **Unsupervised feature learning for audio classification using convolutional deep belief networks**
in Advances in Neural Information Processing Systems (NIPS)

Marolt et al., 2002 – **Neural networks for note onset detection in piano music**
in Proceedings of the International Computer Music Conference (ICMC)

Eck and Schmidhuber, 2002 – **Finding temporal structure in music: Blues improvisation with LSTM recurrent networks**
in Proceedings of the Workshop on Neural Networks for Signal Processing

Todd, 1988 – **A sequential network design for musical applications**
in Proceedings of the Connectionist Models Summer School

Lewis, 1988 – **Creation by Refinement: A creativity paradigm for gradient descent learning networks**
in International Conference on Neural Networks

Outline

Chronology: the big picture

Audio classification: state-of-the-art review

Music audio tagging as a study case

Which is our goal / task?

input → machine learning → output

waveform

or any audio representation!

deep learning model

phonetic transcription

describe music with tags

event detection

The deep learning pipeline

input → front-end → back-end → output

waveform

or any audio representation!

phonetic transcription

describe music with tags

event detection

The deep learning pipeline: input?

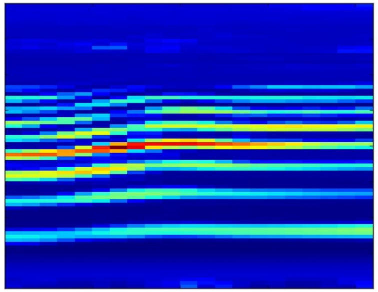
input ➡

?

How to format the input (audio) data?

Waveform
end-to-end learning

Pre-processed waveform
e.g.: spectrogram



The deep learning pipeline: front-end?

input → front-end

waveform

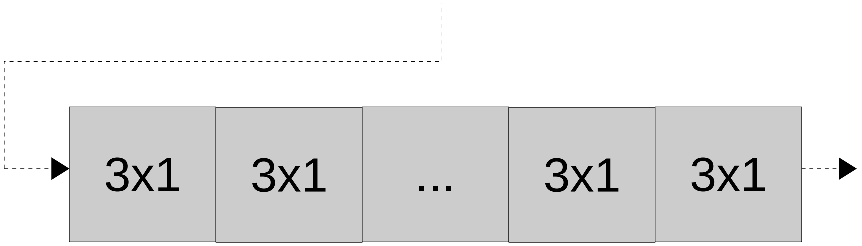
spectrogram

?

based on domain knowledge?	**filters config?**	**input signal?**	
		waveform	*pre-processed waveform*

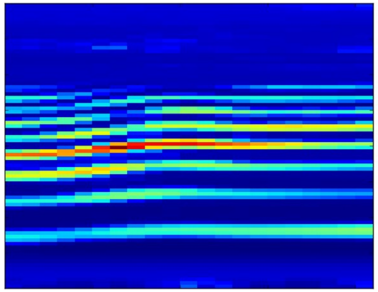
CNN front-ends for audio classification

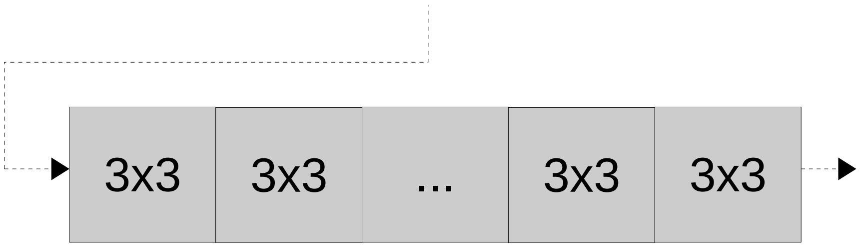
Waveform
end-to-end learning



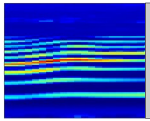
Sample-level

Pre-processed waveform
e.g.: spectrogram





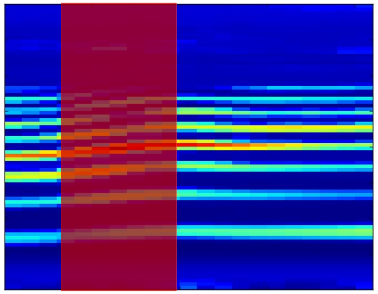
Small-rectangular filters

based on domain knowledge?	**filters config?**	**input signal?**	
		<u>*waveform*</u>	<u>*pre-processed waveform*</u>
no	<u>*minimal*</u> *filter expression*	sample-level  3x1 3x1 ... 3x1 3x1	small-rectangular filters  3x3 3x3 ... 3x3 3x3

Domain knowledge to design CNN front-ends

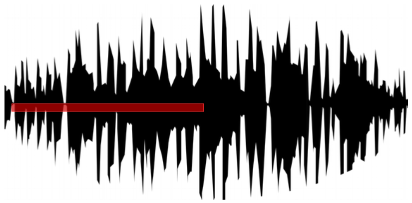
Waveform
end-to-end learning

Pre-processed waveform
e.g.: spectrogram



Domain knowledge to design CNN front-ends

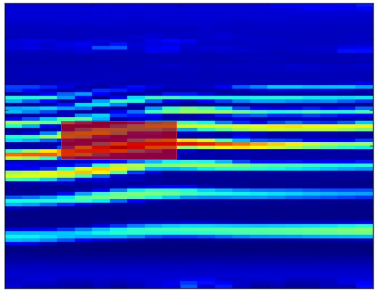
Waveform
end-to-end learning



filter length: 512 — ***window length?***
stride: 256 — ***hop size?***

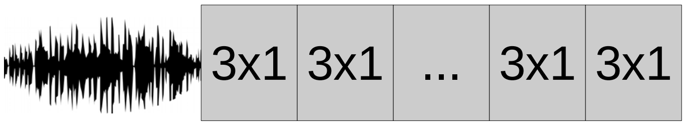
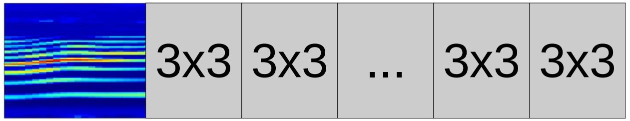
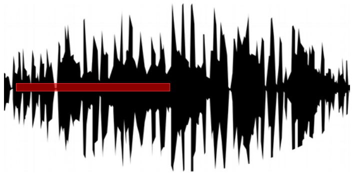
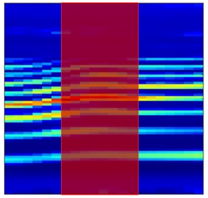
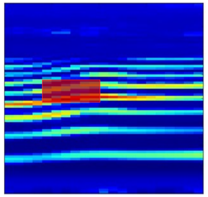
frame-level

Pre-processed waveform
e.g.: spectrogram



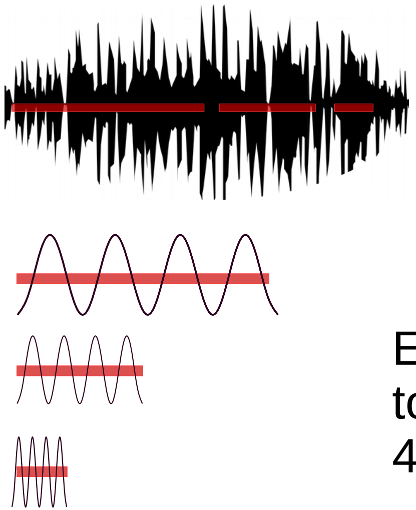
Explicitly tailoring the CNN towards learning temporal ***or*** timbral cues

vertical or horizontal filters

based on domain knowledge?	**filters config?**	**input signal?** *waveform*	*pre-processed waveform*
no	*minimal filter expression*	sample-level 	small-rectangular filters 
yes	*single filter shape in 1st CNN layer*	frame-level 	vertical *OR* horizontal  *or* 

DSP wisdom to design CNN front ends

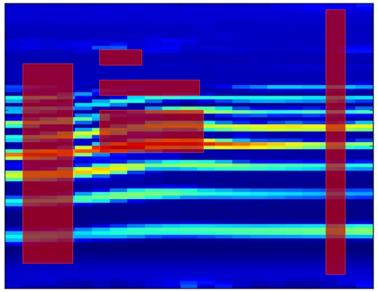
Waveform
end-to-end learning



Efficient way
to represent
4 periods!

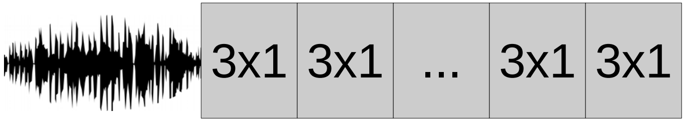
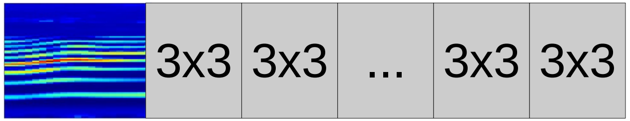
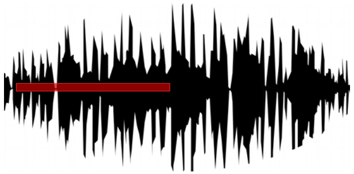
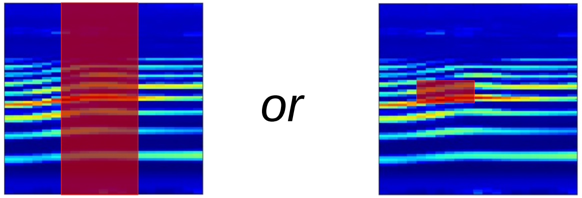
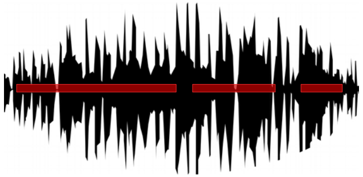
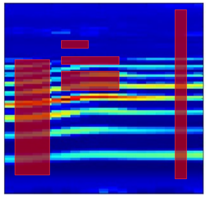
Frame-level (many shapes!)

Pre-processed waveform
e.g.: spectrogram



Explicitly tailoring the CNN towards
learning temporal ***and*** timbral cues

Vertical and/or horizontal

based on domain knowledge?	filters config?	input signal? *waveform*	input signal? *pre-processed waveform*
no	*minimal* filter expression	sample-level 	small-rectangular filters 
yes	*single filter shape in 1st CNN layer*	frame-level 	vertical *OR* horizontal 
yes	*many filter shapes in 1st CNN layer*	frame-level 	vertical *AND/OR* horizontal 

CNN front-ends for audio classification

Sample-level: Lee et al., 2017 – **Sample-level Deep Convolutional Neural Networks for Music Auto-tagging Using Raw Waveforms** *in Sound and Music Computing Conference (SMC)*

Small-rectangular filters: Choi et al., 2016 – **Automatic tagging using deep convolutional neural networks** *in Proceedings of the ISMIR (International Society of Music Information Retrieval) Conference*

Frame-level (single shape): Dieleman et al., 2014 – **End-to-end learning for music audio** *in International Conference on Acoustics, Speech and Signal Processing (ICASSP)*

Vertical: Lee et al., 2009 – **Unsupervised feature learning for audio classification using convolutional deep belief networks** *in Advances in Neural Information Processing Systems (NIPS)*

Horizontal: Schluter & Bock, 2014 – **Improved musical onset detection with convolutional neural networks** *in International Conference on Acoustics, Speech and Signal Processing (ICASSP)*

Frame-level (many shapes): Zhu et al., 2016 – **Learning multiscale features directly from waveforms** *in arXiv:1603.09509*

Vertical and horizontal (many shapes): Pons, et al., 2016 – **Experimenting with musically motivated convolutional neural networks** *in 14th International Workshop on Content-Based Multimedia Indexing*

The deep learning pipeline: back-end?

input → front-end → back-end

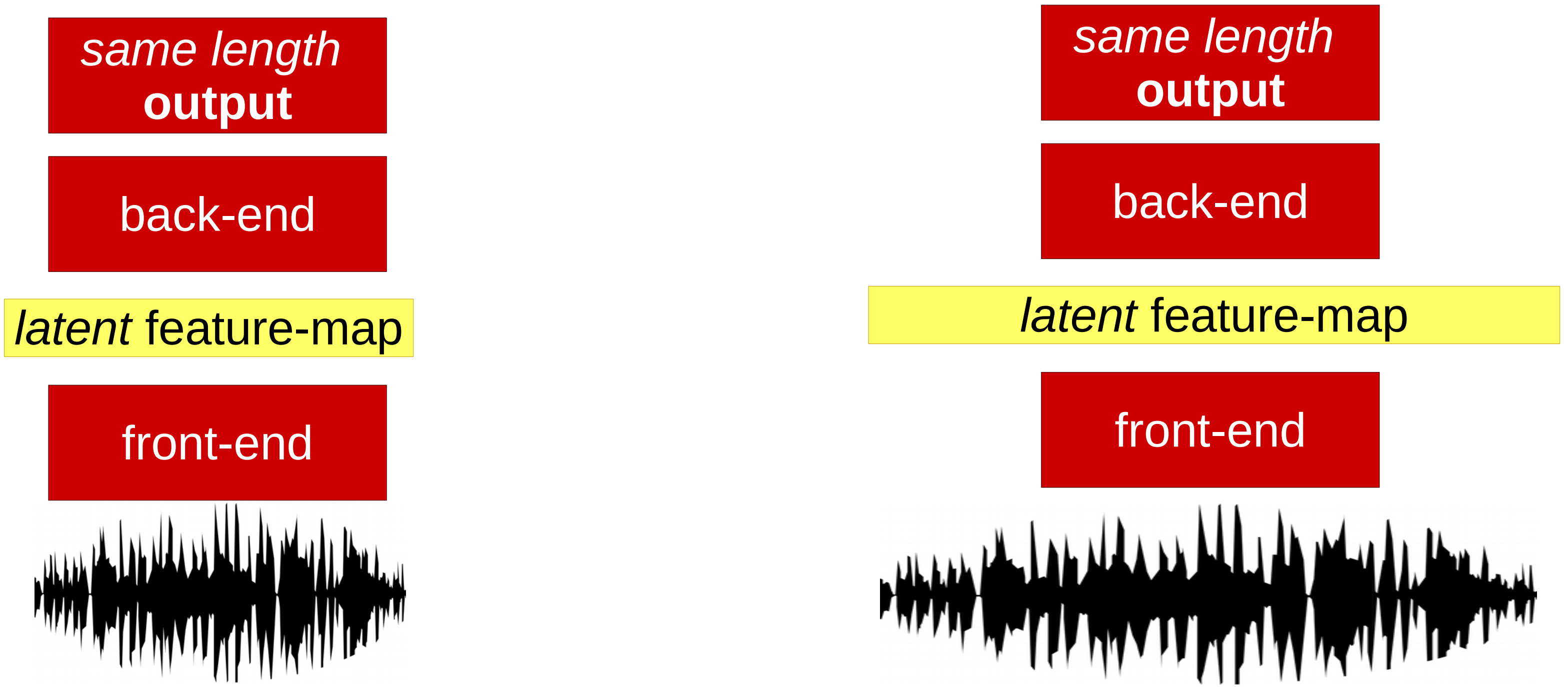
waveform

spectrogram

several CNN architectures

?

What is the back-end doing?



*Back-end **adapts** a variable-length feature map to a fixed output-size*

Back-ends for variable-length inputs

- **Temporal pooling:** max-pool or average-pool the temporal axis

  Pons et al., 2017 – **End-to-end learning for music audio tagging at scale**, in proceedings of the ML4Audio Workshop at NIPS.

- **Attention:** weighting latent representations to what is important

  C. Raffel, 2016 – **Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching.** PhD thesis.

- **RNN:** summarization through a deep temporal model

  Vogl et al., 2018 – **Drum transcription via joint beat and drum modeling using convolutional recurrent neural networks,** In proceedings of the ISMIR conference.

..music is generally of variable length!

Back-ends for fixed-length inputs

Common trick: let's assume a fixed-length input

- **Fully convolutional stacks:** adapting the input to the output with a stack of CNNs & pooling layers.

Choi et al., 2016 – **Automatic tagging using deep convolutional neural networks** in proceedings of the ISMIR conference.

- **MLP:** map a *fixed-length* feature map to a *fixed-length* output

Schluter & Bock, 2014 – **Improved musical onset detection with convolutional neural networks** in proceedings of the ICASSP.

..such trick works very well!

The deep learning pipeline: output

input → front-end → back-end → output

input	front-end	back-end	output
waveform spectrogram	*several CNN architectures*	*MLP* *RNN* *attention*	phonetic transcription describe music with tags event detection

The deep learning pipeline: output

input → front-end → back-end → output

input	front-end	back-end	output
waveform	*several CNN architectures*	*MLP*	phonetic transcription
spectrogram		*RNN*	**describe music with tags**
		attention	event detection

Outline

Chronology: the big picture

Audio classification: state-of-the-art review

Music audio tagging as a study case

Pons et al., 2017. **End-to-end learning for music audio tagging at scale**,
in ML4Audio Workshop at NIPS *Summer internship @ Pandora*

The deep learning pipeline: input?

input → front-end → back-end → output

?

describe music
with tags

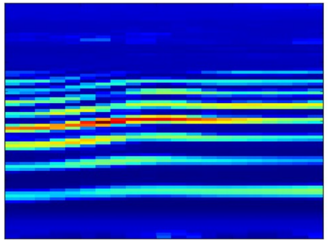
How to format the input (audio) data?

waveform

already: zero-mean
& one-variance

NO pre-procesing!

log-mel spectrogram



– **STFT & mel mapping**
reduces size of the input by removing perceptually irrelevant information

– **logarithmic compression**
reduces dynamic range of the input

– **zero-mean & one-variance**

The deep learning pipeline: input?

input → front-end → back-end → output

waveform

log-mel
spectrogram

describe music
with tags

The deep learning pipeline: front-end?

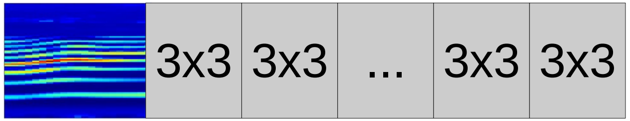
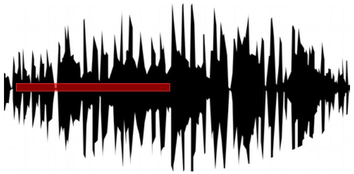
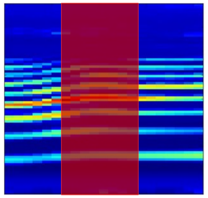
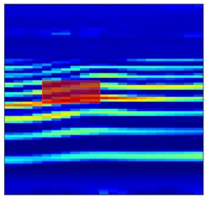
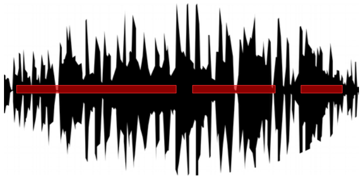
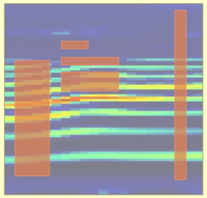
input → front-end → back-end → output

waveform

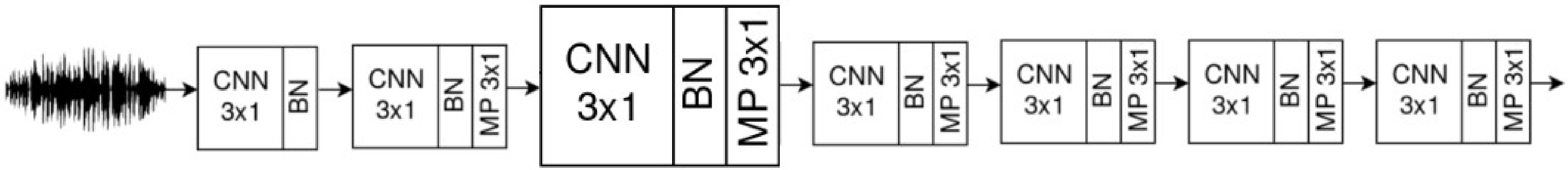
log-mel
spectrogram

?

describe music
with tags

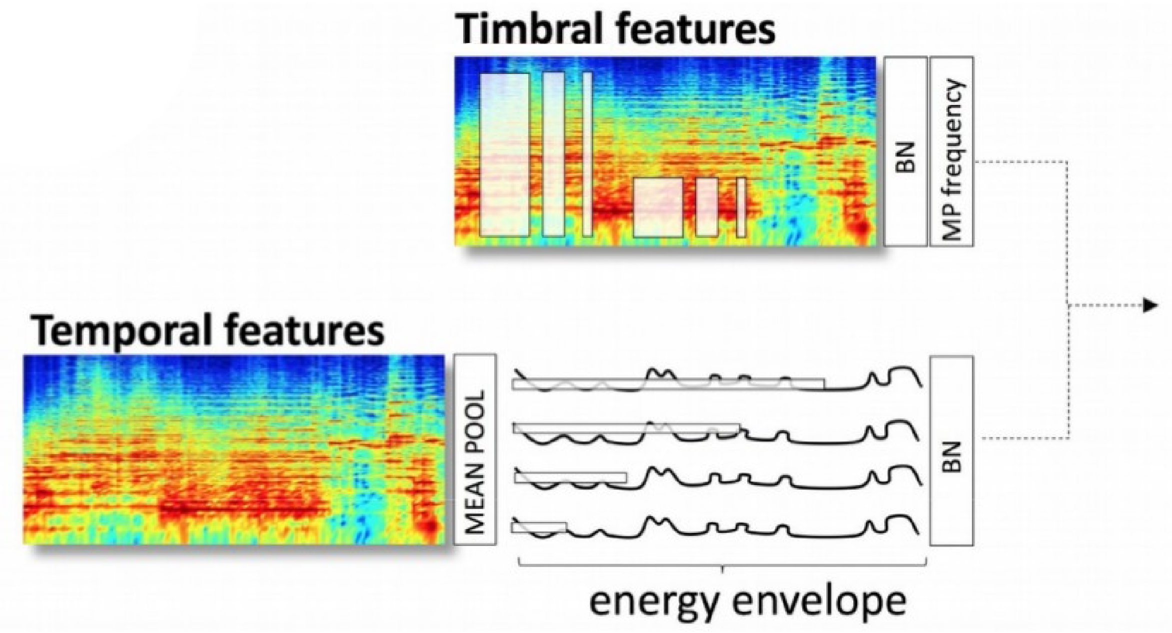
based on domain knowledge?	**filters config?**	**input signal?** *waveform*	**input signal?** *pre-processed waveform*
no	*minimal filter expression*	sample-level 	small-rectangular filters 
yes	*single filter shape in 1st CNN layer*	frame-level 	vertical *OR* horizontal  *or* 
yes	*many filter shapes in 1st CNN layer*	frame-level 	vertical *AND/OR* horizontal 

Studied front-ends: waveform model



sample-level *(Lee et al., 2017)*

Studied front-ends: spectrogram model



vertical and horizontal
musically motivated CNNs

(Pons et al., 2016 – 2017)

The deep learning pipeline: front-end?

input → front-end → back-end → output

waveform

log-mel spectrogram

sample-level

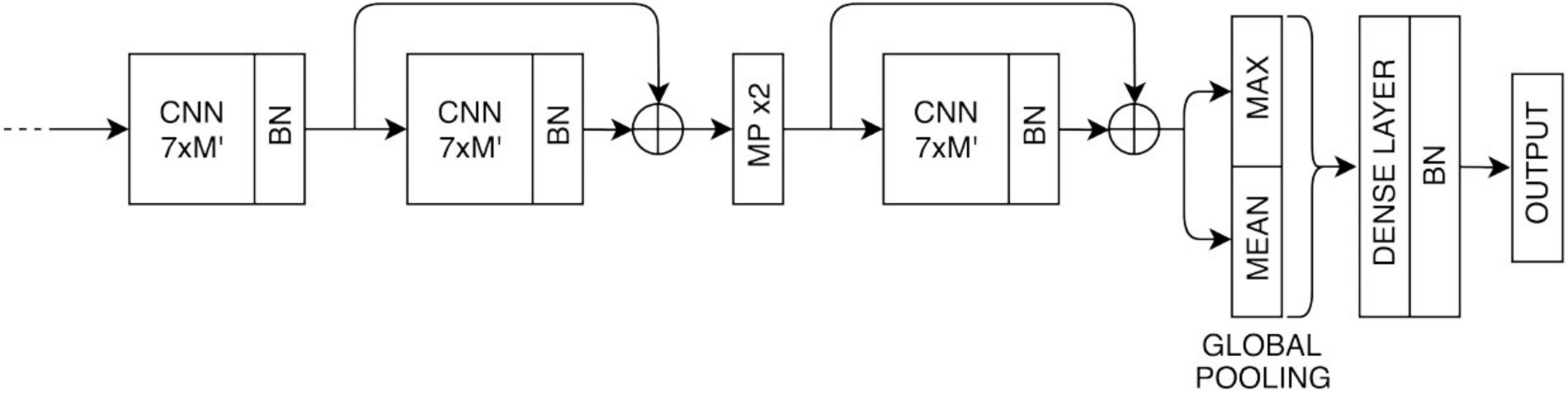
vertical and horizontal

describe music with tags

The deep learning pipeline: back-end?

input		front-end		back-end		output
	→		→		→	
waveform		sample-level		?		describe music with tags
log-mel spectrogram		vertical and horizontal				

Studied back-end: music is of variable length!



Temporal pooling

(Dieleman et al., 2014)

The deep learning pipeline: back-end?

input → front-end → back-end → output

waveform

log-mel spectrogram

sample-level

vertical and horizontal

temporal pooling

describe music with tags

MagnaTT
25k
songs

Million song dataset
250k
songs

1M
songs

spectrograms > waveforms

MagnaTT
25k
songs

Million song dataset
250k
songs

1M
songs

waveforms > spectrograms

spectrograms > waveforms

MagnaTT

25k

songs

Million song dataset

250k

songs

1M

songs

Let’s listen to some music: **our model** in action

acoustic

string ensemble

classical music

period baroque

compositional dominance of lead vocals

major

Deep learning architectures for music audio classification: a personal (re)view

Jordi Pons

jordipons.me – @jordiponsdotme

Music Technology Group
Universitat Pompeu Fabra, Barcelona