Backchannel Prediction for Conversational Speech Using Recurrent Neural Networks

Introduction

What are backchannels (BCs)?

Feedback for the speaker from the listener

- Nodding / head movement
- Eye gaze shift
- Short phrases like “uh-huh”, “yeah”, “right”
- etc.

BCs help build *rapport* (feeling of comfortableness between conversation partners)

- Vary from culture to culture (e.g. Japanese)

BC categories

- non-committal (“uh huh”, “yeah”)
- positive / confirming (“oh how neat”, “great”)
- negative / surprised (“you’re kidding”, “oh my god”)
- questioning (“oh are you”, “is that right”)
- et cetera.

Why backchannel prediction?

- Artificial assistants are becoming ubiquitous (Siri, Google Now, Alexa, Cortana, . . .)
- Conversation with these is still distinctively unhuman
- BCs can help make conversations with an AI agent feel more natural

Goal

- Simplify backchannels to only short phrases
- Predict when to emit backchannels
- Predict what kind of backchannel to emit

Related Work

Ward (2000)

Common approach: manually tuned rules.

> *"[. . .] we formulate the following predictive rule for English:*
> *Upon detection of*
>
> - *a region of pitch less than the 26th-percentile pitch level and*
> - *continuing for at least 110 milliseconds,*
> - *coming after at least 700 milliseconds of speech,*
> - *providing you have not output back-channel feedback within the preceding 800 milliseconds,*
> - *after 700 ms wait,*
>
> *you should produce back-channel feedback."*

Almost always based on

- pitch
- power

Common approach: manually tuned rules.

- error-prone
- a lot of manual work
- hard to generalize

semi-automatic approaches, e.g.

- hand-picked features such as binary pause regions and different speech slopes
- train Hidden Markov Models to predict BCs from that

(Morency 2010)

BC Prediction

Dataset

Switchboard dataset:

- 2400 English telephone conversations
- 260 hours total
- Randomly selected topics
- Transcriptions and word alignments that include BC utterances

BC Utterance Selection (Theory)

- Get a list of all backchannel phrases
- Separate those into categories
- ???

BC Utterance Selection (Practice)

- BC phrase list from the *Switchboard Dialog Act Corpus* (SwDA)

	name	act_tag	example	full_count
1	Statement-non-opinion	sd	Me, I'm in the legal department.	75145
2	**Acknowledge (Backchannel)**	b	Uh-huh.	38298
3	Statement-opinion	sv	I think it's great	26428
4	Agree/Accept	aa	That's exactly it.	11133
5	Abandoned or Turn-Exit	%	So, -	15550
6	Appreciation	ba	I can imagine.	4765
7	Yes-No-Question	qy	Do you have to have any special training?	4727

Figure 1: Most common categories

BC Utterance Selection (Practice)

- SwDA incomplete $\rightarrow$ Identify utterances only from their text

BC Utterance Selection (Practice)

Something like “uh” can be a disfluency or a BC.

$\rightarrow$ only include phrases with silence or BC before them.

Training Area Selection

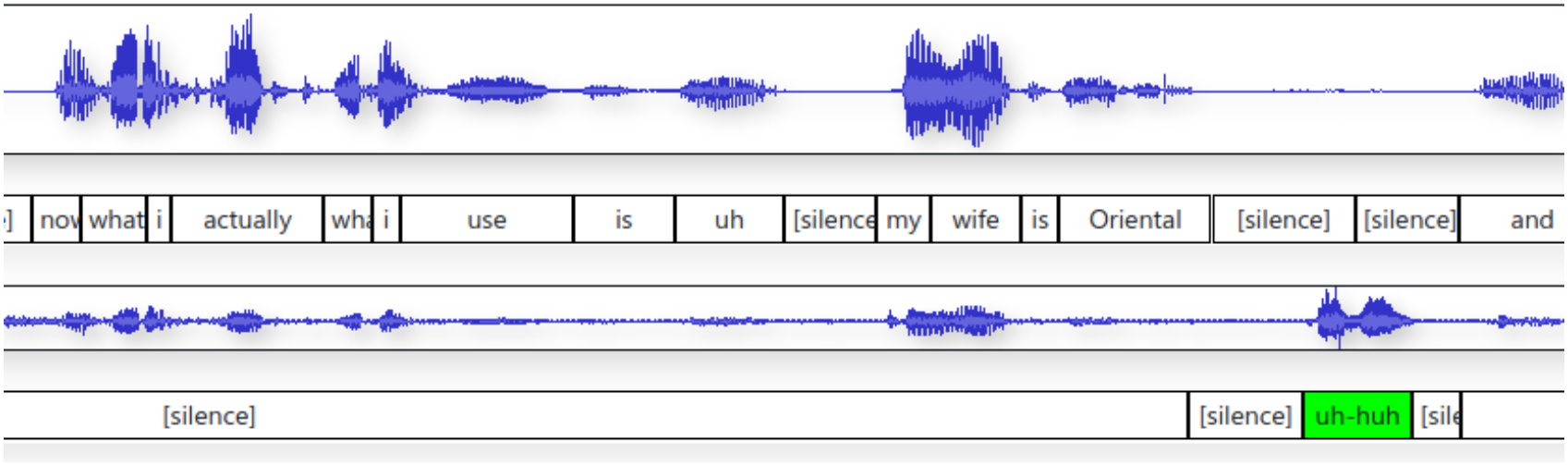


Figure 2: Sample Audio Segment

Training Area Selection

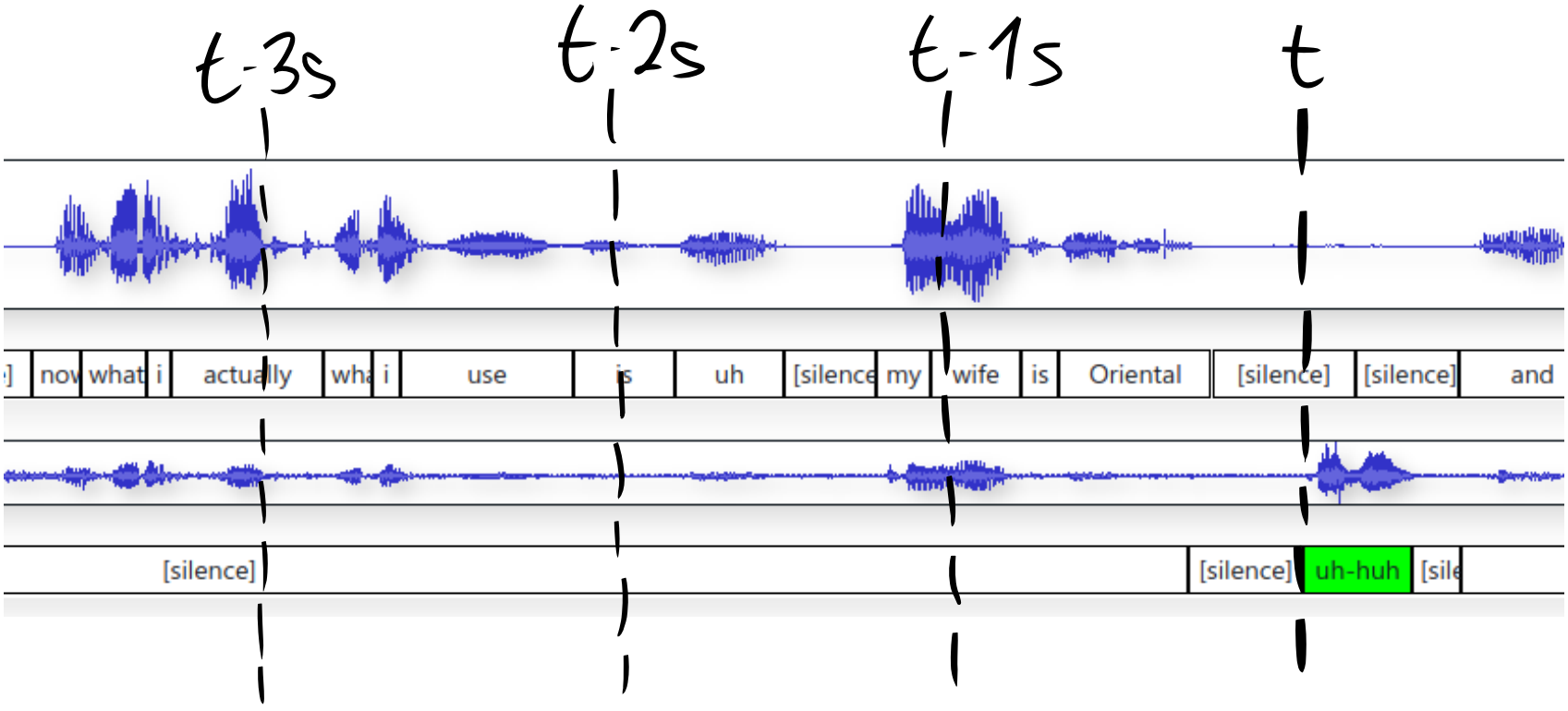


Figure 3: Sample Audio Segment

Training Area Selection

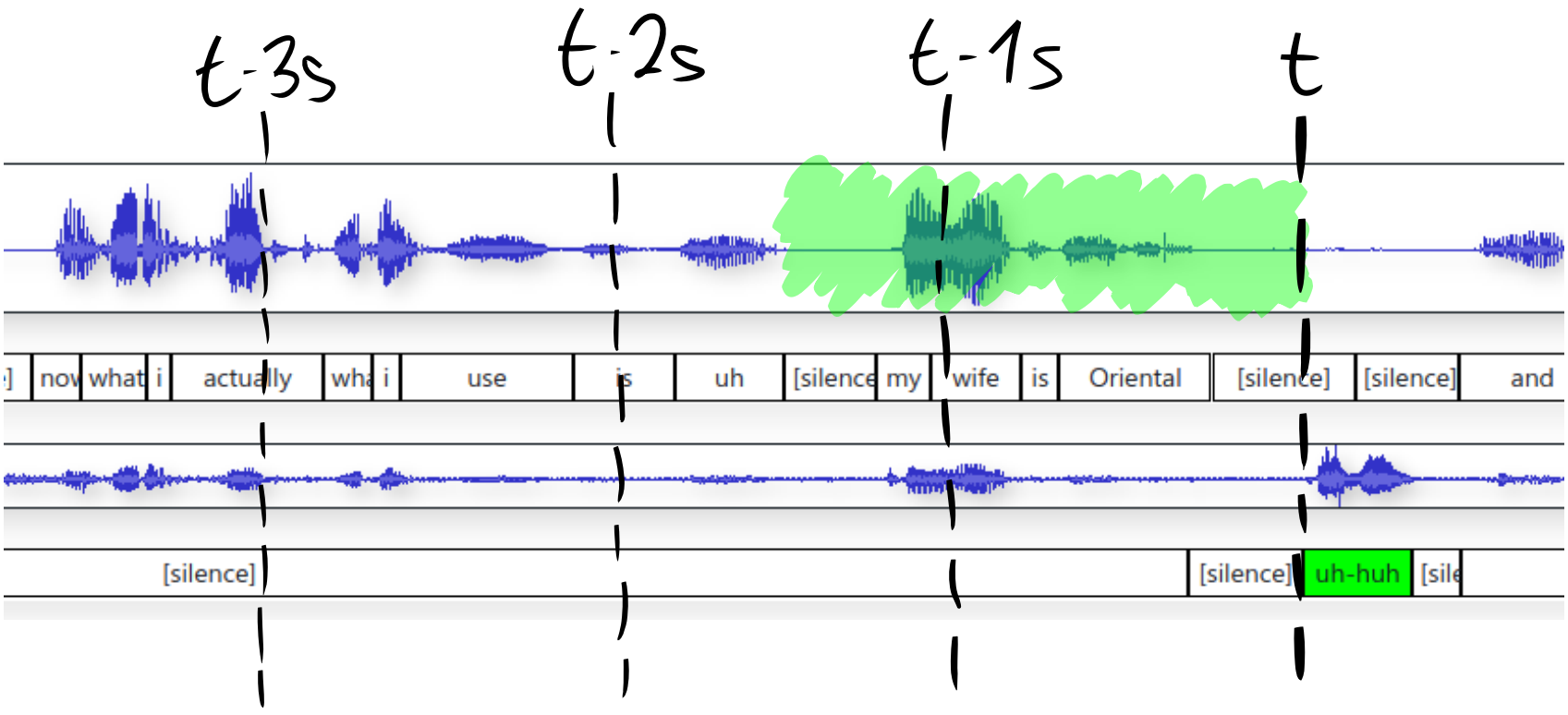


Figure 4: Positive Training Area (width=1.5s)

Training Area Selection

Need area to predict non-BC.

$\rightarrow$ Area of audio where no BC follows

Training Area Selection

Need area to predict non-BC.

$\rightarrow$ Area of audio where no BC follows

Want balanced data set.

$\rightarrow$ Choose area 0.5 seconds before BC area

Training Area Selection

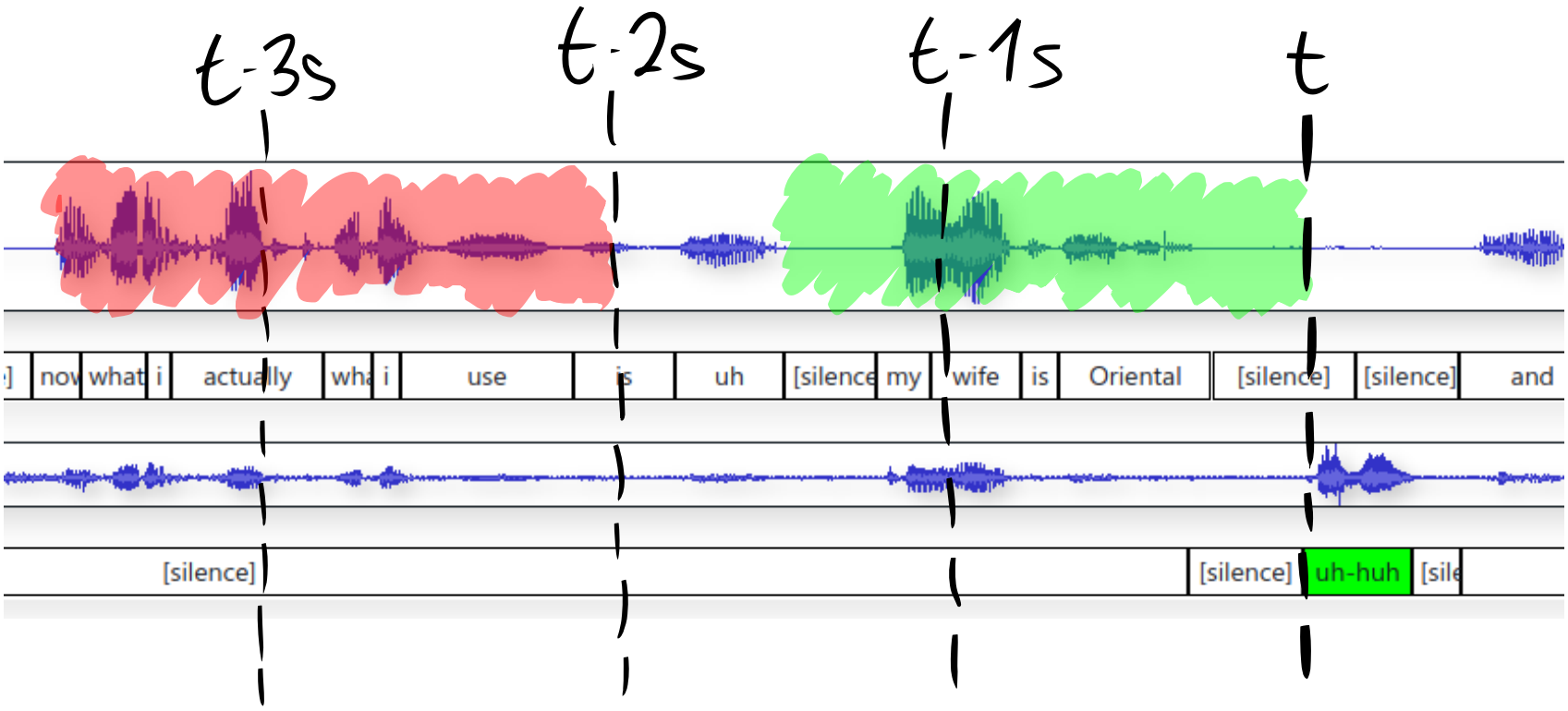


Figure 5: Pos/Neg Training areas

→ Balanced data

Context width?

Feature Selection (Theory)

- Acoustic features like power, pitch
- Linguistic features (from the transcriptions)

Feature Selection – Acoustic

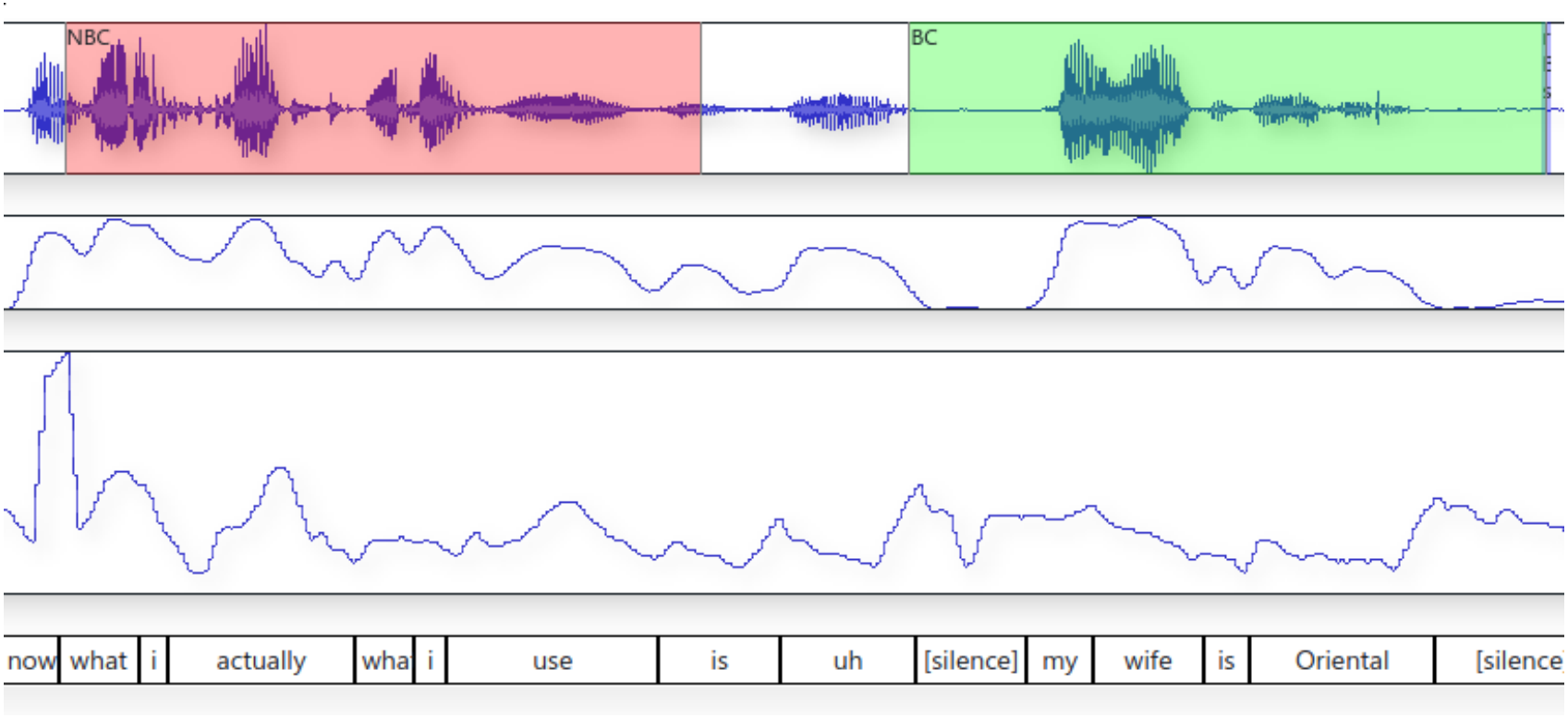


Figure 6: Audio, Power, Pitch

Feature Selection – Linguistic

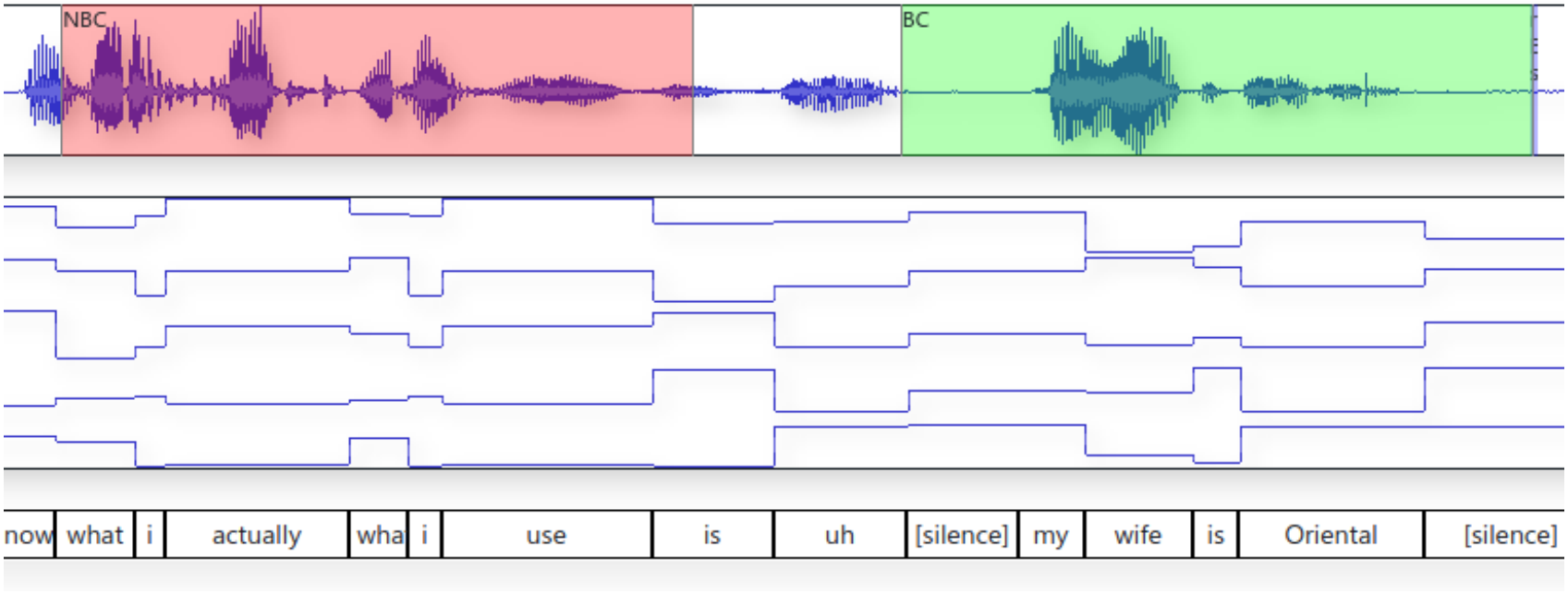


Figure 7: Word2Vec

"what i" has same encoding

Neural network design

Input layer

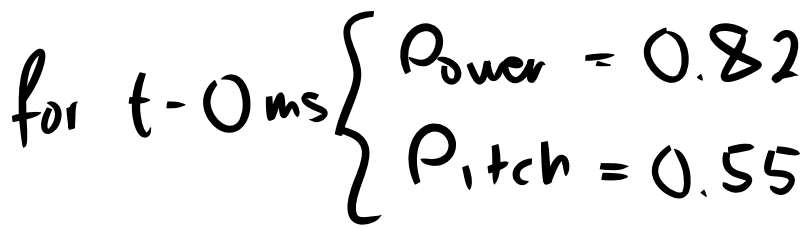


Figure 8:

Input layer

$$\text{for } t\text{-}100\text{ms} \begin{cases} \text{Power} = 000 \\ \text{Pitch} = 000 \end{cases}$$

$$\text{for } t\text{-}0\text{ms} \begin{cases} \text{Power} = 0.82 \\ \text{Pitch} = 0.55 \end{cases}$$

Figure 9:

Input layer

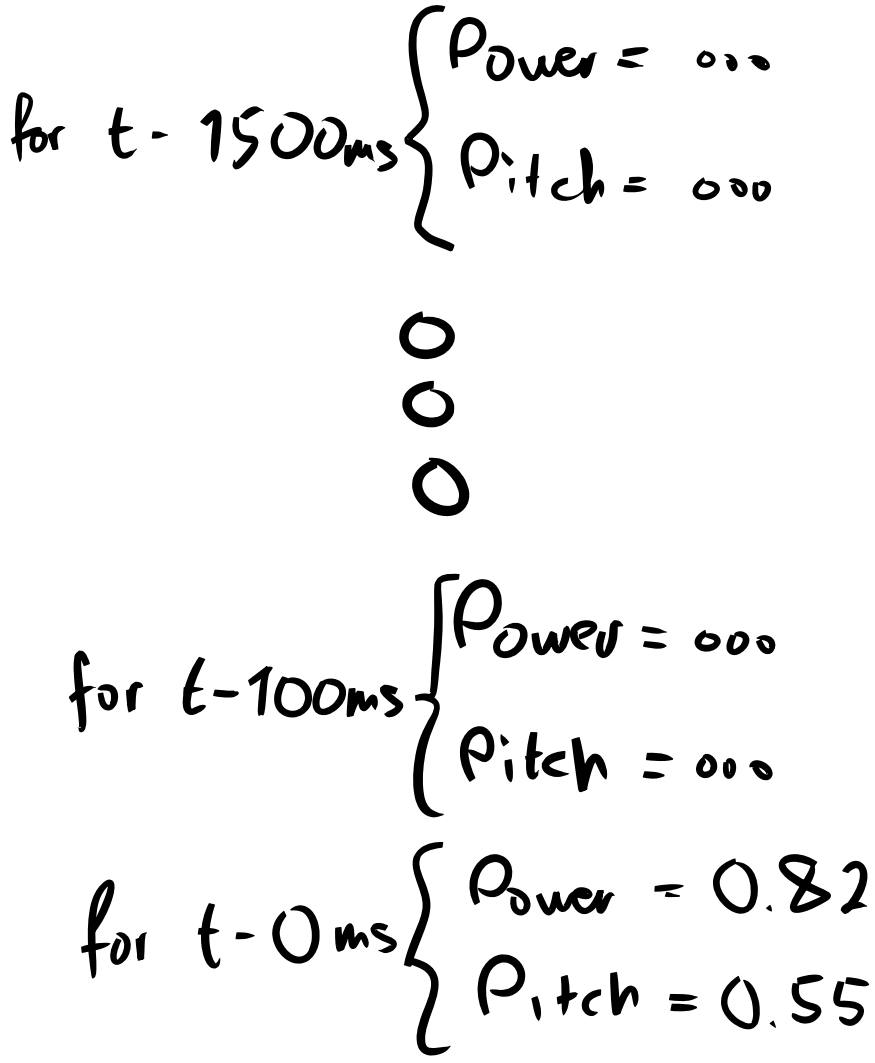


Figure 10:

Input layer

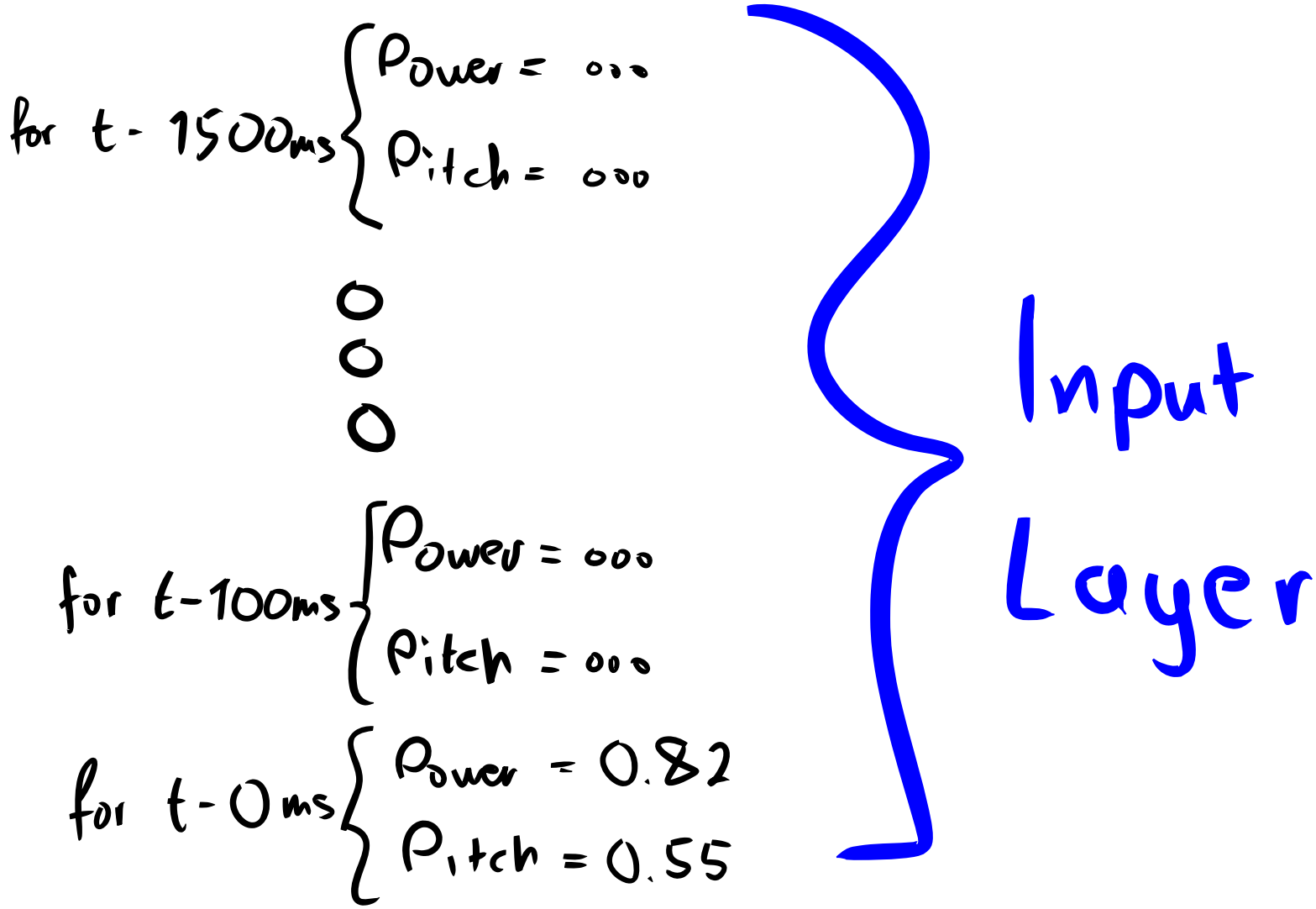


Figure 11:

Hidden layers (Feed forward)

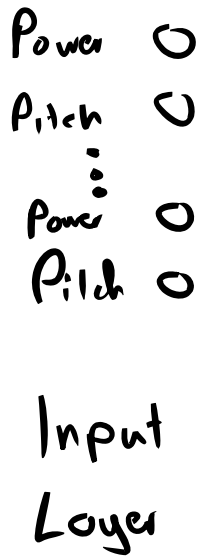


Figure 12:

Hidden layers (Feed forward)

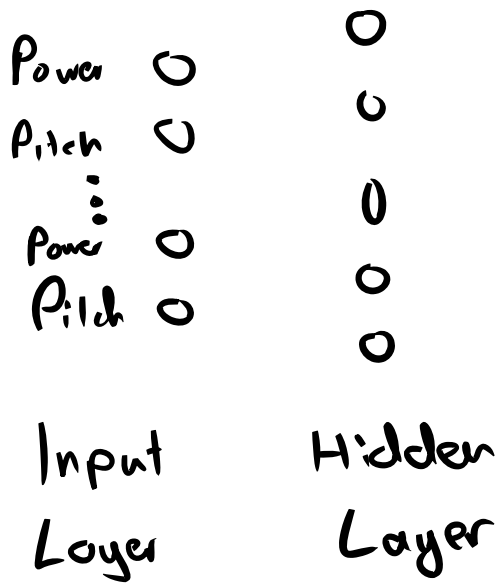


Figure 13:

Hidden layers (Feed forward)

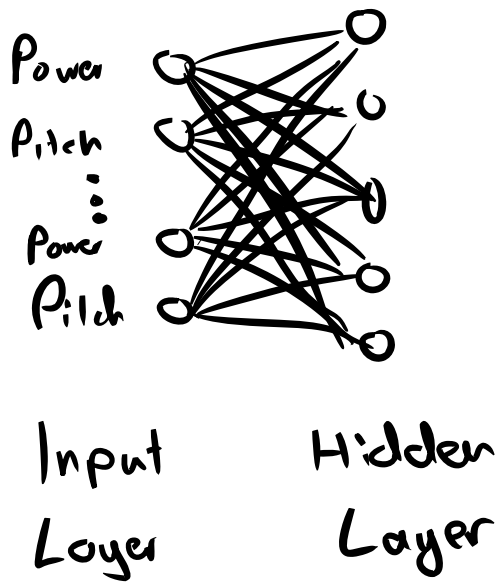


Figure 14:

Hidden layers (Feed forward)

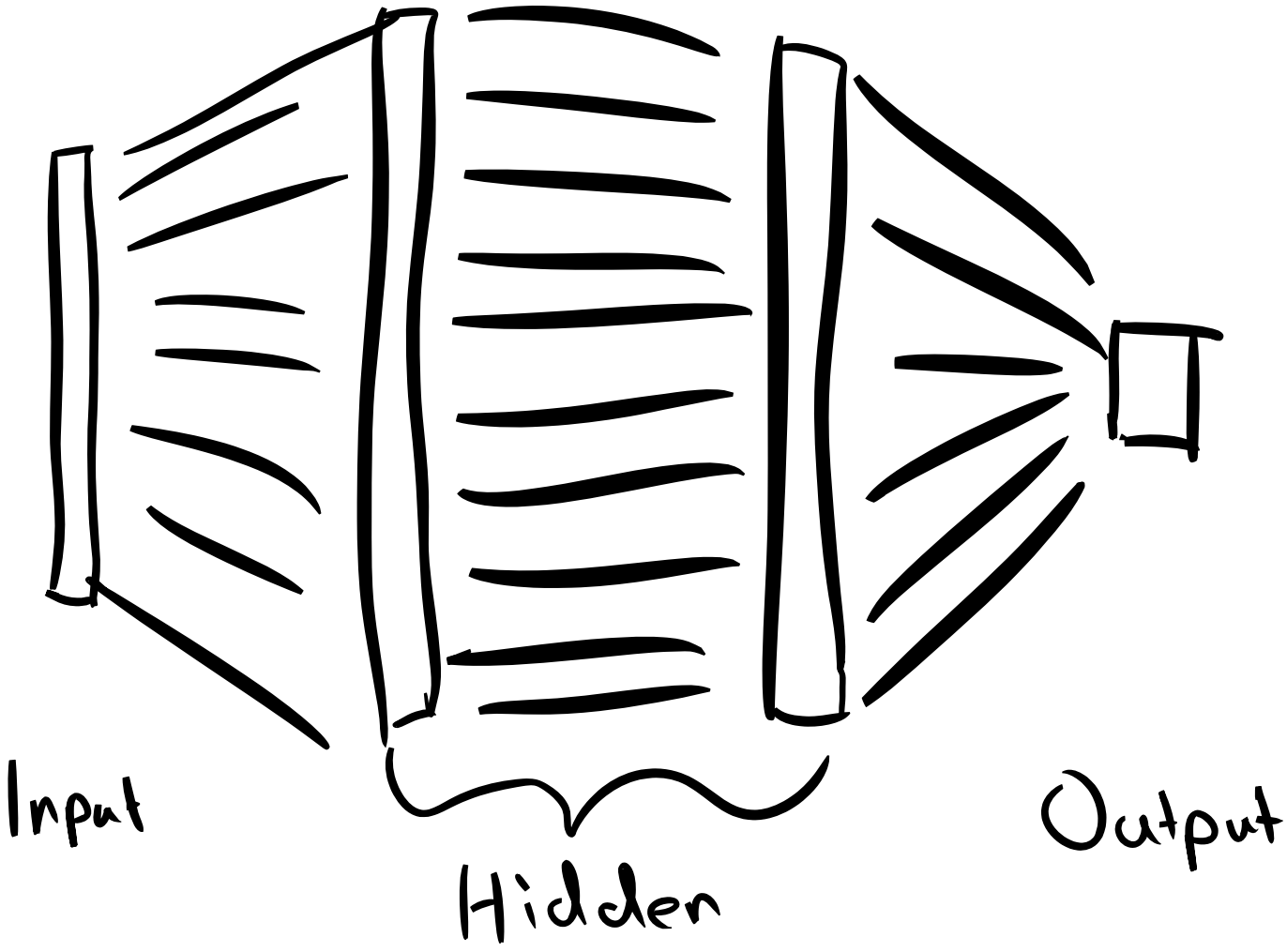


Figure 15:

Feed forward net gets a fixed time context before the BC.

It can not take its previous state into account.

BCs are more probable after a longer period without BCs.

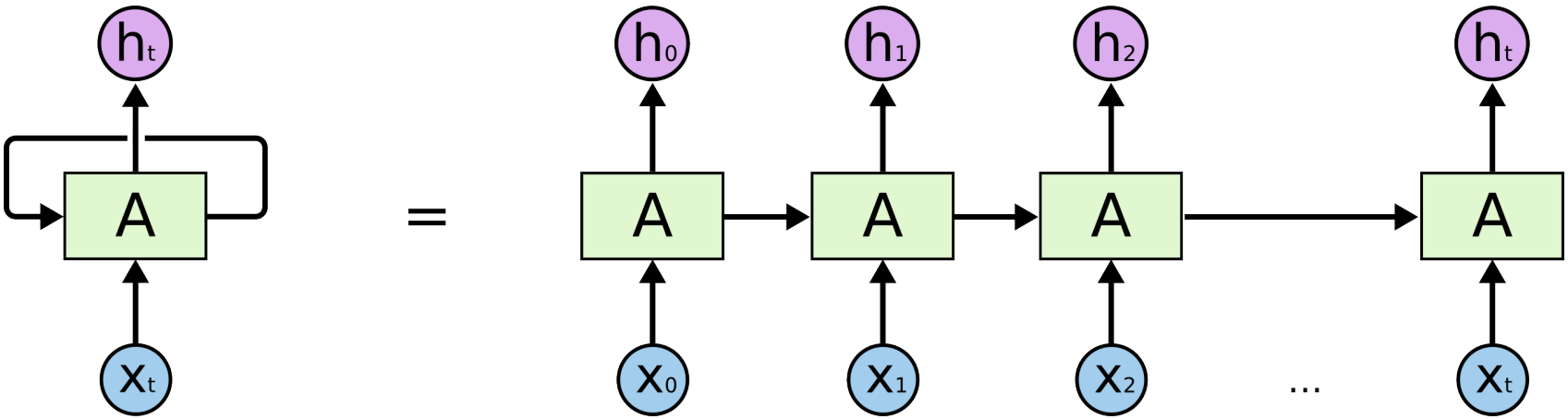


Figure 16: RNN architecture [1]

LSTM is able to take into account it's own past internal state.

[1] Christopher Olah

Postprocessing

NN output is

- a value between 0 and 1
- quickly changing
- noisy

Postprocessing – Low-pass filter

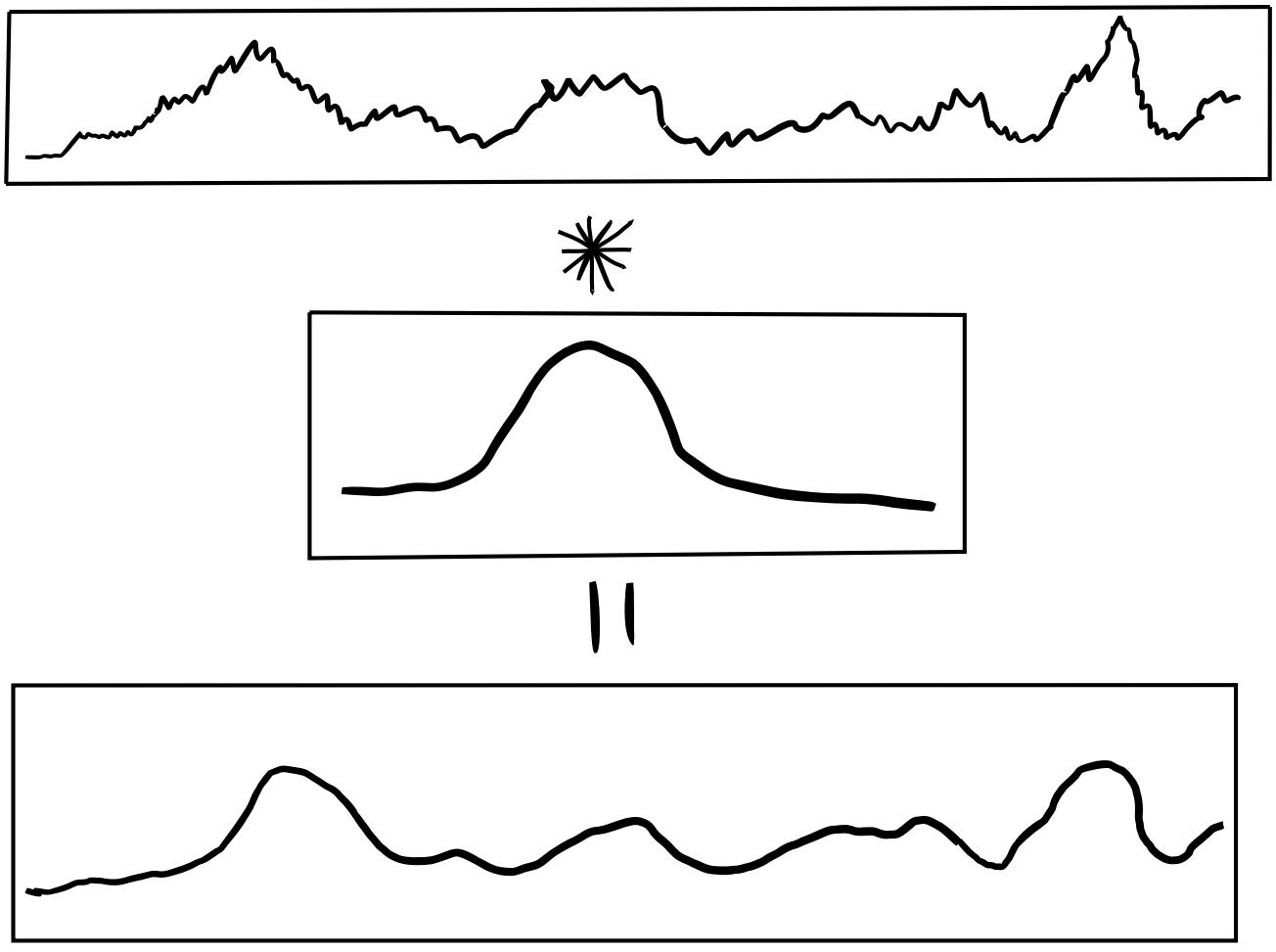


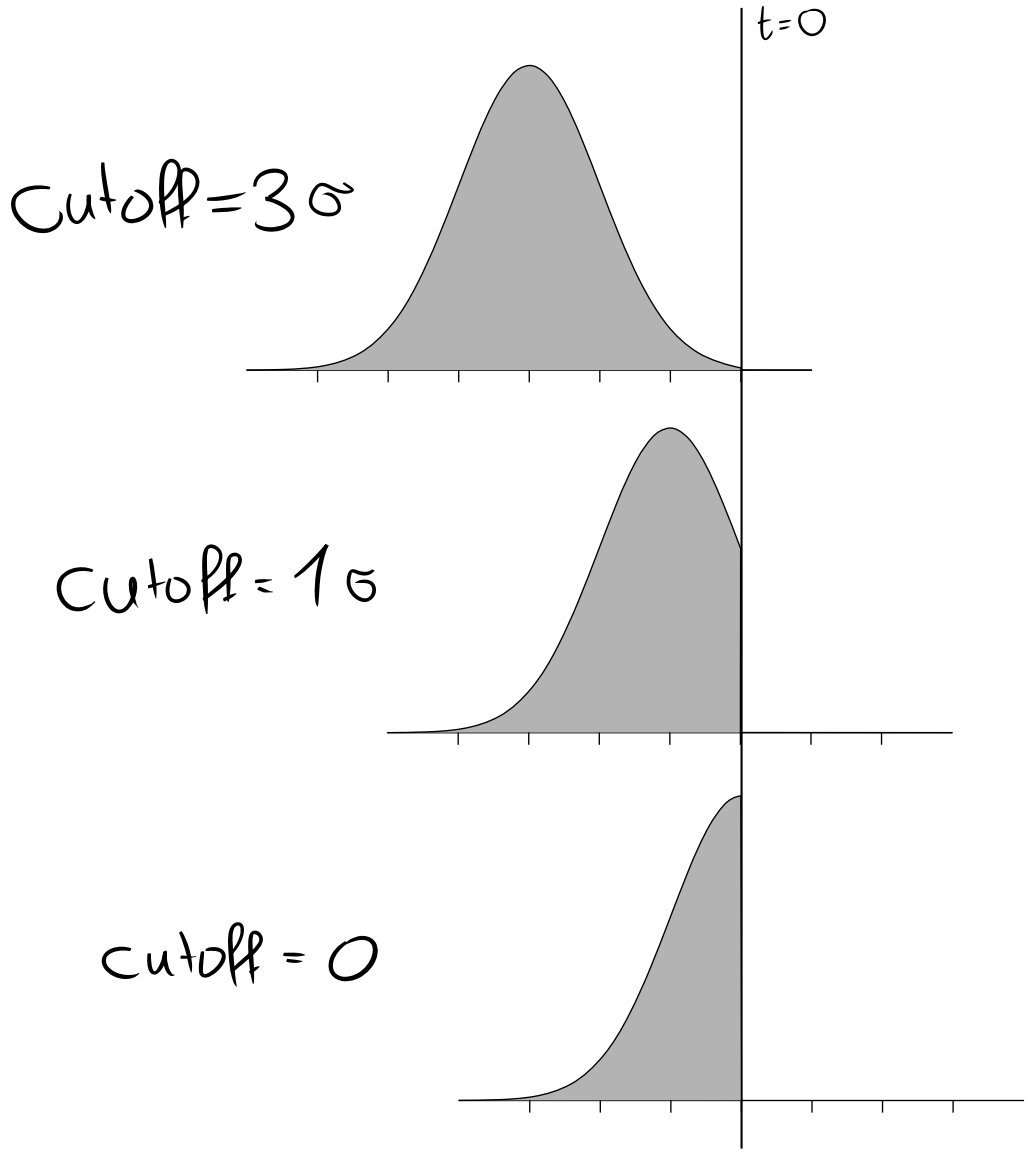
Figure 17:

Gauss filter looks into future

$\rightarrow$ Cut off gaussian filter and shift it

Gauss filter looks into future

$\rightarrow$ Cut off gaussian filter and shift it



Thresholding / Triggering

Use areas of output $>$ threshold ($0 < t < 1$)

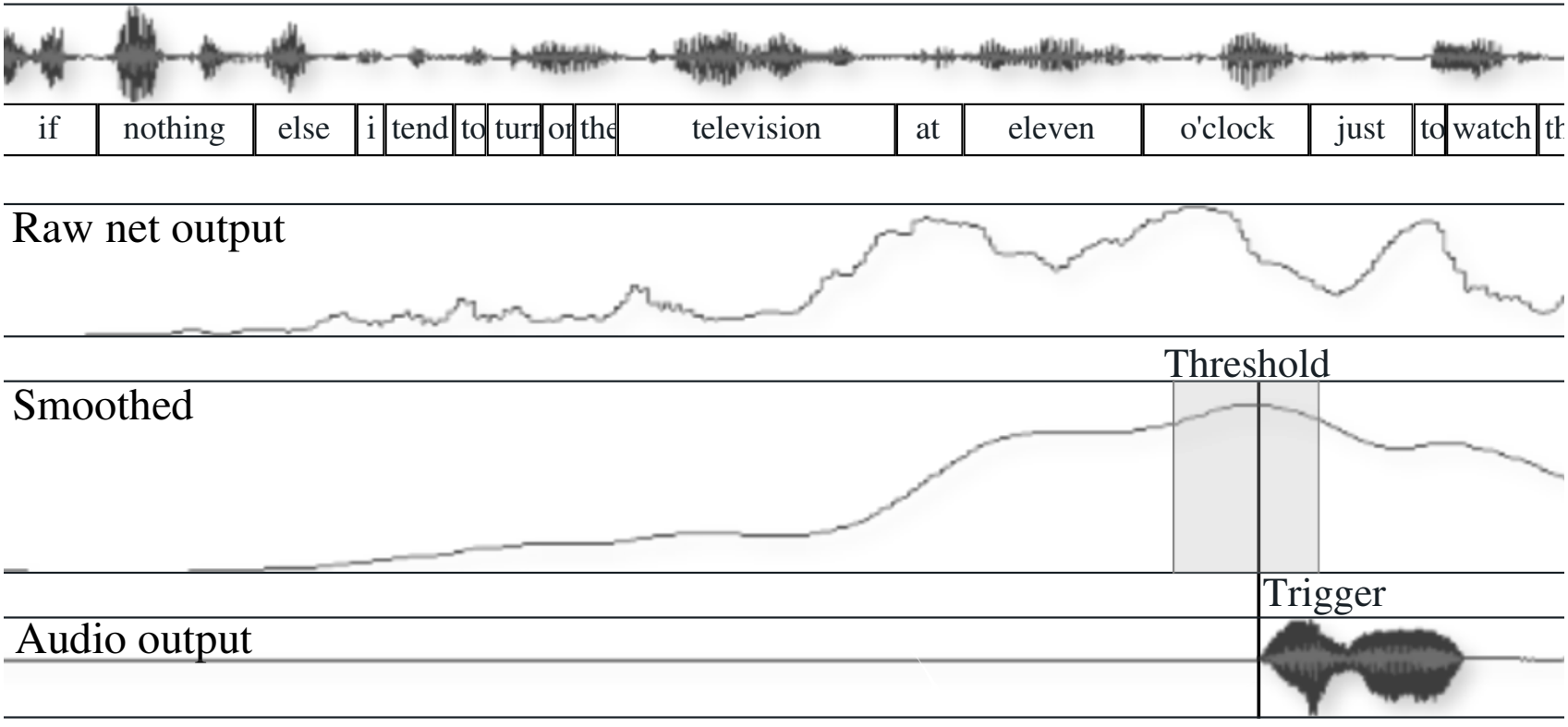


Figure 19: Example of the postprocessing process.

Output audio trigger

place uh-huh sample beginning at trigger time.

Evaluation

Objective Evaluation

- Precision (portion of predictions that were correct)
- Recall (portion of correct BCs that were predicted)
- F1-Score (harmonic mean of Precision and Recall)

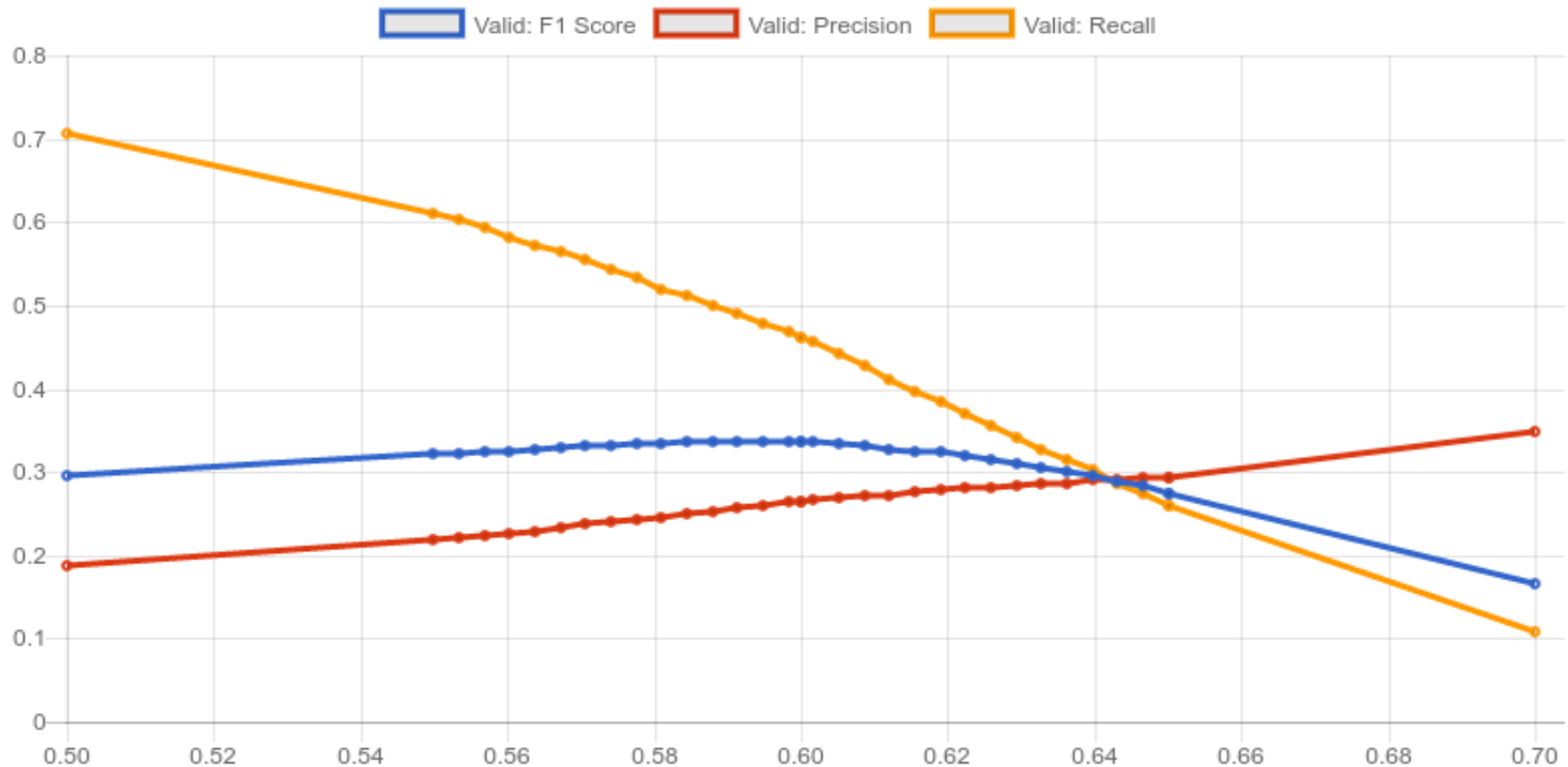


Figure 20: Evaluating the performance of a network while varying the threshold. Note the inverse relationship between *Precision* and *Recall*.

Lots of parameters to tune

- Context width
- Context stride
- Which features
- NN depth
- NN layer sizes
- LSTM vs Feed forward
- Gaussian filter sigma
- Gaussian filter cutoff
- Prediction delay

Results

Context width

Context	Precision	Recall	F1-Score
500 ms	0.219	0.466	0.298
1000 ms	0.280	0.497	0.358
1500 ms	0.305	0.488	**0.375**
2000 ms	0.275	0.577	0.373

Table 1: Results with various context lengths. Performance peaks at 1500 ms.

Context stride

Stride	Precision	Recall	F1-Score
1	0.290	0.490	0.364
2	0.305	0.488	**0.375**
4	0.285	0.498	0.363

Table 2: Results with various context frame strides.

Features

Features	Precision	Recall	F1-Score
power	0.244	0.516	0.331
power, pitch	0.307	0.435	0.360
power, pitch, mfcc	0.278	0.514	0.360
power, ffv	0.259	0.513	0.344
power, ffv, mfcc	0.279	0.515	0.362
power, pitch, ffv	0.305	0.488	**0.375**
word2vec$_{dim=30}$	0.244	0.478	0.323
power, pitch, word2vec$_{dim=30}$	0.318	0.486	0.385
power, pitch, ffv, word2vec$_{dim=15}$	0.321	0.475	0.383
power, pitch, ffv, word2vec$_{dim=30}$	0.322	0.497	**0.390**
power, pitch, ffv, word2vec$_{dim=50}$	0.304	0.527	0.385

Table 3: Results with various input features, separated into only acoustic features and acoustic plus linguistic features.

LSTM vs FF

Layers	Parameter count	Precision	Recall	F1-Score
FF (56 : 28)	40k	0.230	0.549	0.325
FF (70 : 35)	50k	0.251	0.468	0.327
FF (100 : 50)	72k	0.242	0.490	0.324
LSTM (70 : 35)	38k	0.305	0.488	**0.375**

Table 4: Feed forward vs LSTM. LSTM outperforms feed forward architectures.

Layer sizes

Layer sizes	Precision	Recall	F1-Score
100	0.280	0.542	0.369
50 : 20	0.291	0.506	0.370
70 : 35	0.305	0.488	**0.375**
100 : 50	0.303	0.473	0.369
70 : 50 : 35	0.278	0.541	0.367

Table 5: Comparison of different network configurations. Two LSTM layers give the best results.

Final results / Comparison

Predictor	Precision	Recall	F1-Score
Baseline (random)	0.042	0.042	0.042
Müller et al. (offline)	–	–	0.109
Our results (offline, context of −750 ms to 750 ms)	0.114	0.300	**0.165**
Our results (online, context of −1500 ms to 0 ms)	0.100	0.318	0.153

Table 6: Comparison with previous research.

Margin of Error	Constraint	Precision	Recall	F1-Score
−200 ms to 200 ms		0.172	0.377	0.237
−100 ms to 500 ms		0.239	0.406	0.301
−500 ms to 500 ms		0.247	0.536	0.339
0 ms to 1000 ms	Baseline (random, correct BC count)	0.111	0.052	0.071
	Baseline (random, 8x correct BC count)	0.079	0.323	0.127
	Balanced Precision and Recall	0.342	0.339	0.341
	Best F1-Score (only acoustic features)	0.294	0.488	0.367
	Best F1-Score (acoustic and linguistic features)	0.312	0.511	0.388

Table 7: Results with our evaluation method with various margins of error used in other research. Performance improves with a wider margin width and with a later margin center.

Survey

Randomly show participants 6 samples of the following categories

1. Random predictor
2. NN predictor
3. Ground truth

Survey

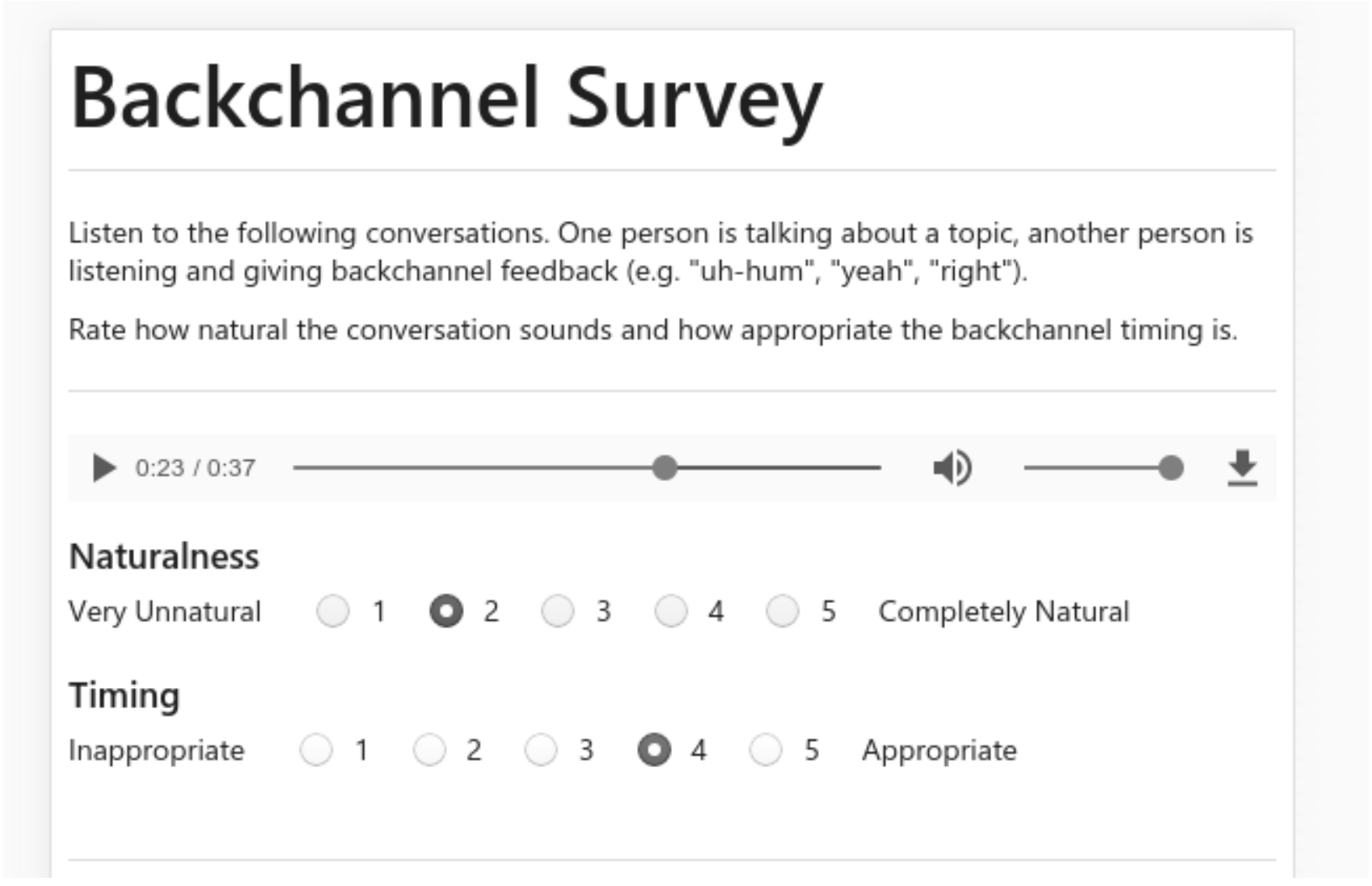


Figure 21: Screenshot of the survey interface.

Survey Results

Predictor	Sample	Timing	Naturalness	Sample Size
random	average	2.33 points	2.63 points	40
nn	average	3.48 points	3.08 points	40
truth	average	4.20 points	4.08 points	40

Conclusion and Future Work