A SUPERLETS

Superlet transforms are a technique for time-frequency analysis introduced by Moca et al. (2021). The superlet transform is formed from a composite Morlet wavelet transforms. In contrast to traditional time-frequency techniques, such as the Short-time Fourier Transform (STFT) which has a fixed time-frequency resolution trade-off for all frequencies, the Morlet wavelet has lower temporal resolution at lower frequencies. This variable trade-off allows the resulting representation to better capture the dynamics of neural signal, for which high frequency oscillations often occur in short bursts and low frequency oscillations persist for longer durations.

However, the Morlet transform is not Pareto optimal, and the superlet transform makes improvements to both time and frequency resolution simultaneously in order to better capture the self-similar oscillations across different frequencies which are often present in neural signal. In this section, we briefly summarize the content of Moca et al. (2021) relevant to our work.

Consider the following formulation of the Morlet wavelet for a given frequency of interest f:

$$\psi_{c,f}(t) = \frac{1}{B_c\sqrt{2\pi}} \exp\left(-\frac{t^2}{2B_c^2}\right) \exp\left(j2\pi ft\right) \tag{4}$$

$$B_c = \frac{c}{f} \tag{5}$$

Here, c is the number of cycles, which is a parameter that controls the time-frequency resolution trade-off. The expression for the wavelet on the right side of the equation eq. (4) can be understood as being composed of two terms: the first term can be thought of as a complex wave function (Euler's formula), and the second term can be understood as a Gaussian windowing function that modulates the amplitude of the wave. Note that the width σ of the windowing function falls off with frequency.

For a given frequency f, the superlet transform of order o is the geometric mean of Morlet transforms $\psi_{c,f}$ for a range of values $c \in [1, o]$. The superlet representation at a frequency of interest f for a signal x is then:

$$\sqrt[o]{\prod_{i=1}^{o} \sqrt{2} \cdot x * \psi_{c,f}}, \tag{6}$$

where $*$ is the complex convolution operator. Although, compared to the Morlet wavelet transform, the overall resolution is improved, the f in the denominator of eq. (5) means that there continues to be relatively lower time resolution at lower frequencies. In other words, there is more temporal smearing at the bottom of the spectrogram than at the top. To account for this, we use an adaptive masking strategy (see section 2: Pretraining).

B TIME-FREQUENCY REPRESENTATION PARAMETERS

For the STFT, we use a window size of 400 samples ($\approx 200ms$) with an overlap of 350 samples ($\approx 175ms$) with frequency channels evenly spaced from 0 to 200Hz.

For the superlet, we use $c_1 = 1$ with orders $o = 3-30$. The frequencies of interest are evenly spaced from 0.1 to 200Hz. To match the down-sampling rate of the STFT, the superlet representation is decimated by a factor of 50. Finally, for both representations, we remove 5 columns from each array on either side ($\approx 250ms$) to account for edge-effects.

Superlets take the composite of many Morlet wavelets. Thus, a superlet is simply a collection of Morlet wavlets with different c:

$$SL_c = \{\psi_{f,c_i} \mid c_1, ..., c_o\} \tag{7}$$

where o is the order of the superlet.

For this work, we use the multiplicative, adaptive superlet transform. In the multiplicative version of the transform, $c_i = c_1 \cdot i$. And in the adaptive version, the order of the superlet changes according to frequency:

$$o = o_{min} + \left[(o_{max} - o_{min}) \cdot \frac{f - f_{min}}{f_{min} - f_{max}}\right] \tag{8}$$

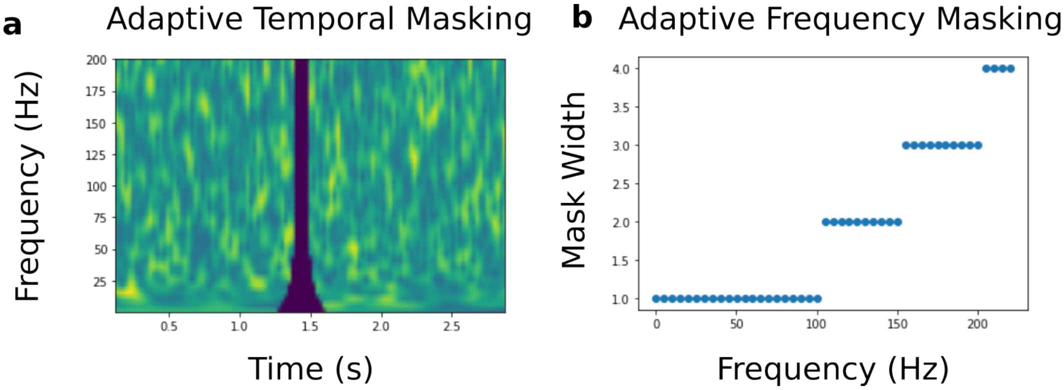

Figure 7: **Adaptive masking** *(Left)* The adaptive temporal mask, for which the width increases with the inverse of frequency. *(Right)* The width of the frequency mask, which specifies the amount of masked frequency channels, increases as a function of frequency.

C Masking time and frequency

During pre-training, BrainBERT receives an augmented version of the spectrogram, from which random time and frequency bands have been removed. In this work, we use two masking strategies: static masking, which is suited for spectrograms of fixed time-frequency resolution, and adaptive masking, which is appropriate for spectrograms with a variable trade-off in time-frequency resolution. **Static masking** For our static masking procedure, we adapt the work of (Liu et al., 2020; 2021). The width of the temporal mask is randomly selected from the range $[1, 5]$, and the width of the frequency mask is randomly selected from the range $[1, 2]$.

Adaptive masking For the superlet transform, we use adaptive temporal and frequency masks. The temporal width of the adaptive mask as a function of frequency f is given by:

$$w_t(f) = 2 \max\left(m, \frac{200}{20+f}\right) \tag{9}$$

where m is the minimum width of the mask, which is randomly selected from $\{1, 2\}$ per example. See fig. 7 for visualization.

The width of the frequency masks is given as a function of frequency f:

$$w_f(f) = \max\left(1, \left\lfloor \frac{4.9f}{250} \right\rfloor\right) \tag{10}$$

The parameters of these equations are selected to roughly match the width of the masks used for the STFT.

D Masking procedure

The masking procedure is described in algorithm 1 (see section 2: Pretraining). Note that intervals are masked without overlap. The procedure is adapted from Liu et al. (2021) and Devlin et al. (2019). During pre-training, we use $p_{\text{mask}} = 0.05$, $p_{ID} = 0.1$, and $p_{\text{replace}} = 0.1$. The purpose of letting some segments go unaugmented is to make the distribution seen at train time and test time more similar. The purpose of replacing some segments with random signal is to prevent the model from simply learning the identity function.

Algorithm 1 Time-masking procedure

```
$\mathbf{Y} \leftarrow n \times m$ spectrogram
$i \leftarrow 0$
while $i \leq m$ do
    $p \sim \text{Unif}(0, 1)$
    if $p < p_{\text{mask}}$ then
        $l \sim \lfloor \text{Unif}(\text{step}_{\text{min}}, \text{step}_{\text{max}} + 1) \rfloor$
        $q \sim \text{Unif}(0, 1)$
        if $q < p_{\text{ID}}$ then
            pass
        else if $p_{\text{ID}} \leq q < p_{\text{ID}} + p_{\text{replace}}$ then
            $j \leftarrow \text{Unif}(0, m - l)$
            $\mathbf{Y}[:, i : i + l] \leftarrow \mathbf{Y}[:, j : j + l]$
        else
            $\mathbf{Y}[:, i : i + l] \leftarrow \mathbf{0}$
        end if
        $i \leftarrow i + l$
    end if
end while
```

E Baselines

Linear baselines (time domain) We train two feed forward networks with layer dimensions $[d_{\text{input}}, 1]$ and sigmoid activations. The first model, Linear (5s time domain), takes 5s of time domain input, sampled at 2048 Hz, so $d_{\text{input}} = 10,240$. The second model, Linear (0.25s time domain), takes 5s of input, so $d_{\text{input}} = 512$

Deep neural network (time domain) We train a deep neural network, Deep NN (5s time domain), which consists of 5 stacked feed forward layers with dimensions $[d_{\text{input}}, 1024, 512, 256, 128]$ and ReLU activations on the hidden layers and a sigmoid activation on the output layer. The network takes 5s of time domain input, sampled at a rate of 2048 Hz, so $d_{\text{input}} = 10,240$.

Linear baselines (time-frequency domain) We train two feed forward linear networks with dimension $[d_{\text{input}}, 1]$ and sigmoid activations that take the time-frequency representations as input. Linear (.25s STFT) and Linear (.25s superlet) receive the time-frequency representation, averaged across time in a ≈ 244ms interval, centered on the example. More precisely, given a time-frequency representation $\mathbf{Y} \in \mathbb{R}^{n \times 2l}$, the model receives the vector obtained by averaging $Y_{:,l-5,l+5}$ across the time axis. There are $n = 40$ frequency bands, so $d_{\text{input}} = 40$. This is the same form as the input that our classification network receives from BrainBERT (see section 3).

F Preprocessing

Data was high-pass filtered at 0.1Hz. Line noise at 60Hz and its harmonics were removed. Each electrode was re-referenced by subtracting out the mean signal from the two adjacent electrodes on the same shaft (Laplacian re-referencing). This decreases the cross-correlation between electrodes (Li et al., 2018). Only electrodes which can be Laplacian re-referenced, i.e., have neighbors, are included in the pretraining data. Finally, signals from all electrodes are visually inspected, and recordings which show obvious signs of corruption are removed. In total, data from 1,249 electrodes ($\approx 74\%$ of total electrodes; 4,551 electrode-hours) are used for pretraining.

G Pretraining parameters

All layers ($N = 6$) in the encoder stack are set with the following parameters: $d_h = 768$, $H = 12$, and $p_{\text{dropout}} = 0.1$. We pretrain the BrainBERT model with the LAMB optimizer (You et al., 2019) and $lr = 1e - 4$. We use a batch size of $n_{\text{batch}} = 256$, train for 500k steps, and record the validation performance every 1000 steps. Then, the weights with the best validation performance are retained.

During pretraining, the BrainBERT representations are passed to a spectrogram prediction head which consists of a two feed forward linear layers. The first layer, a hidden layer, has dimension

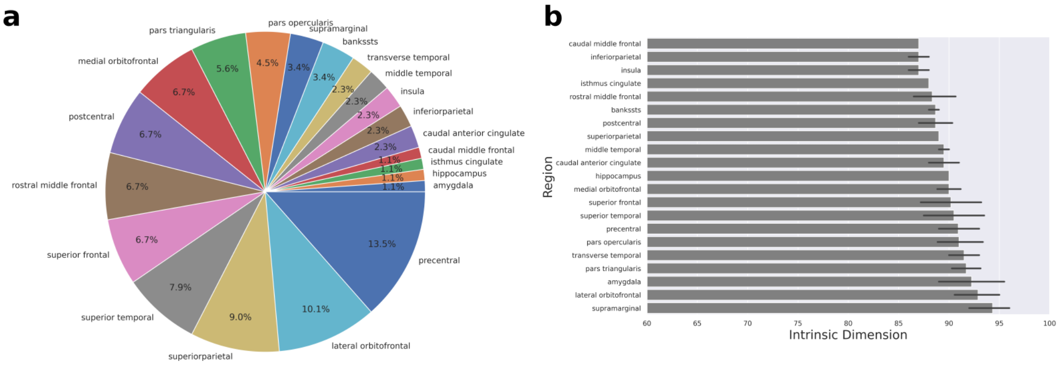

Figure 8: **Intrinsic dimension averaged by region** We find the intrinsic dimension (ID) for all held-out electrodes, and consider the electrodes in the top 10-th percentile. Among these electrodes, the percentage which fall in each region is shown in (a). However, the electrodes are not distributed uniformly across the brain, so to get a normalized view of the regions, we take the mean across electrodes in each region (b). Error bars show a 95% confidence interval

$d = 768$ and a GeLU activation. The second layer, the output layer, has dimension $d = 40$ This matches the height of the spectrogram, which is determined by the number of frequencies of interest; see appendix B.

As part of our ablation test, we also include a BrainBERT with randomly initialized weights. For this, weights are initialized according to the uniform Xavier scheme (Glorot & Bengio, 2010).

H Fine-tuning training parameters

The classifier is a fully connected linear layer with $d_{\text{in}} = 768$, $d_{\text{out}} = 1$ and a sigmoid activation. The model is trained using a binary cross entropy loss. When fine-tuning, we use the AdamW optimizer, with $lr = 1e - 3$ for the classification head and $lr = 1e - 4$ for the BrainBERT weights. When training with frozen BrainBERT weights, we use the AdamW optimizer with $lr = 1e - 3$.

All models, both this classifier and the baseline models, are trained for 1,000 updates. Validation performance is computed every 100 updates, and test accuracy is reported using the weights with the best validation performance.

I Intrinsic Dimension

We use a standard method for determining intrinsic dimension (Fan et al., 2010). For a given threshold β, the intrinsic dimension is the $d \in \mathbb{N}$ such that the ratio of explained variance for d dimensions of a N dimensional PCA is above β:

$$\frac{\sum_{i=1}^{d} var(y_i)}{\sum_{j=1}^{N} var(y_j)} > \beta \tag{11}$$

We use a PCA of $N = 200$ and a threshold of $\beta = 0.95$. For each session in the held out data, we segment the neural recordings into 5s intervals, for which the spectrogram is a matrix $\mathbf{Y} \in \mathbb{R}^{n \times m}$. We produce the BrainBERT embeddings $\mathbf{E} \in \mathbb{R}^{n \times m}$ for each spectrogram, and average across the time-dimension to obtain a single n-dimensional vector representation of the interval. For each electrode, we find the intrinsic dimension of the manifold on which these BrainBERT embeddings lie. We find the regions with the highest intrinsic dimension, by first taking the electrodes in the top 10-th percentile. Among these electrodes, most of them lie in the precentral gyrus (13.5%), the lateral orbitofrontal cortex (10.1%) and the pars opercularis (9.0%). However, it should be noted that the electrodes are not distributed evenly across the cortex, so to get a sense of which regions have the highest ID, we should normalize across the number of electrodes in a region. We find that the regions with the highest mean ID are the supramarginal gyrus, the lateral orbitofrontal cortex, and the amygdala.

I.1 Tasks

Sentence onset Once the movie transcript has been aligned to the brain activity, the brain activity which corresponds to the sentence onsets can be collected, each embedded in 5s of context. These intervals form the set of positive examples.

Speech vs. non-speech For this task, the positive examples are formed from the 5s of context surrounding any word. For both this task and the sentence onset task, the negative examples are formed by finding 1s intervals which do not overlap with any intervals of speech audio. Each example is embedded in 5s of context.

Volume For each word in the transcript, the volume is measured as the root-mean-square for a 500ms interval starting from the word onset. Those words which lie one standard deviation above the mean volume are labeled as “high-volume”. Those words which lie one standard deviation below the mean volume are labeled as ”low-volume”.

Pitch For each word in the transcript, the pitch is extracted using librosa’s `piptrack` function over a Mel-spectrogram (sampling rate 48,000 Hz, FFT window length of 2048, hop length of 512, and 128 mel filters). Those words which lie one standard deviation above the mean pitch are labeled as “high-pitch”. Those words which lie one standard deviation below the mean pitch are labeled as “low-vpitch”.

J Dataset statistics

	Sentence onset		Speech/Non-speech		Volume		Pitch	
Subject	Onset	Non-speech	Speech	Non-speech	High	Low	High	Low
subject-1	1,587	1,587	6,333	6,333	1,135	542	1,723	1,724
subject-2	1,071	1,071	6,413	6,413	662	137	1,042	1,066
subject-3	2,057	2,057	2,581	2,581	1,350	980	1,591	1,592
subject-4	542	542	2,500	2,500	295	229	378	364
subject-5	1,059	1,059	3,183	3,183	769	521	1,009	980
subject-6	1,668	1,668	2,367	2,367	1,092	815	1,311	1,309
subject-10	1,971	1,971	1,971	1,971	1,350	980	1,591	1,592

Table 3: **Annotated data statistics** The number of examples per task and per subject for the held-out sessions are shown here. For the sentence onset task and speech vs. non-speech task, the number of examples is explicitly balanced between classes. Non-speech examples correspond with 1s intervals which do not overlap with any word-audio in the movie.

Subj.	Age (yrs.)	# Electrodes	Movie	Recording time (hrs)	Held-out
1	19	91	Thor: Ragnarok	1.83	
			Fantastic Mr. Fox	1.75	
			The Martian	0.5	x
2	12	100	Venom	2.42	
			Spider-Man: Homecoming	2.42	
			Guardians of the Galaxy	2.5	
			Guardians of the Galaxy 2	3	
			Avengers: Infinity War	4.33	
			Black Panther	1.75	
			Aquaman	3.42	x
3	18	91	Cars 2	1.92	x
			Lord of the Rings 1	2.67	
			Lord of the Rings 2 (extended edition)	3.92	
4	9	135	Megamind	2.58	
			Toy Story	1.33	
			Coraline	1.83	x
5	11	205	Cars 2	1.75	x
			Megamind	1.77	
6	12	152	Incredibles	1.15	
			Shrek 3	1.68	x
			Megamind	2.43	
7	6	109	Fantastic Mr. Fox	1.5	
8	4.5	102	Ant Man	2.28	
9	16	72	Sesame Street Episode	1.28	
10	12	173	Cars 2	1.58	x
			Spider-Man: Far from Home	2.17	

Table 4: **Subject statistics** Subjects used in BrainBERT training, and held-out downstream evaluation. The number of uncorrupted, electrodes that can be Laplacian re-referenced are shown in the second column The average amount of recording data per subject is 4.3 (hrs).

K Electrode visualization

For each subject, pre-operative T1 MRI scans without contrast were processed with FreeSurfer Fischl et al. (2004), which performed skull stripping, white matter segmentation, and surface generation. iELVis Groppe et al. (2017) was used to co-register a post-operative fluoroscopy scan to the preoperative MRI. Electrodes were manually identified using BioImageSuite Joshi et al. (2011), and then assigned to one of 94 regions (according to the Desikan-Killiany atlas (Desikan et al., 2006)) using FreeSurfer's automatic parcellation. The alignment to the atlas was manually verified for each subject.

Depth electrodes in the white matter, if they were within 1.5 mm of the gray-white matter boundary, were projected to the nearest point on that boundary, and were labeled as coming from that region (for the purposes of region analyses). This procedure is very similar to the post brain-shift correction methods used for electrocorticography electrodes Yang et al. (2012). For solely visualization purposes, all electrodes identified to lie in the gray matter or on the gray-white matter boundary were first projected to the pial surface (using nearest neighbors), and then mapped to an average brain (using Freesurfer's fsaverage atlas) for the visualizations shown in fig. 3 and fig. 6.

L Pretraining performance

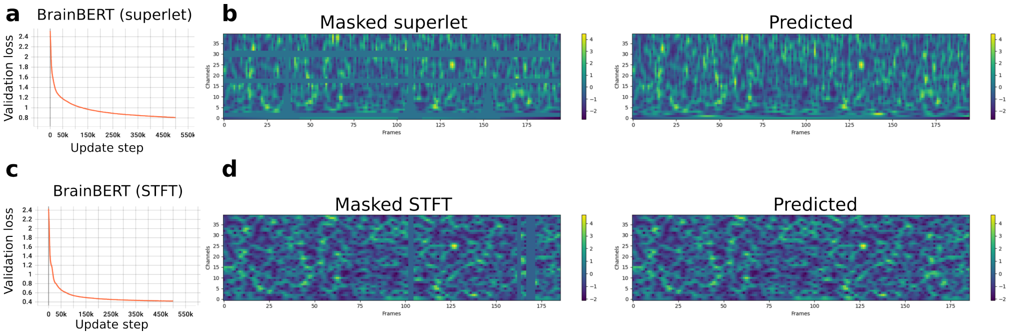

Figure 9: **Pretraining performance** (a) During pretraining, BrainBERT with superlet inputs achieves 0.81 content-aware reconstruction loss. The L1 reconstruction component (not pictured) of this loss is 0.41. (c) BrainBERT with STFT inputs achieves 0.42 content aware reconstruction loss, of which the L1 component accounts for 0.20. (b) and (d) show sample reconstructions produced by BrainBERT after 500k updates.

M Supplementary Figures

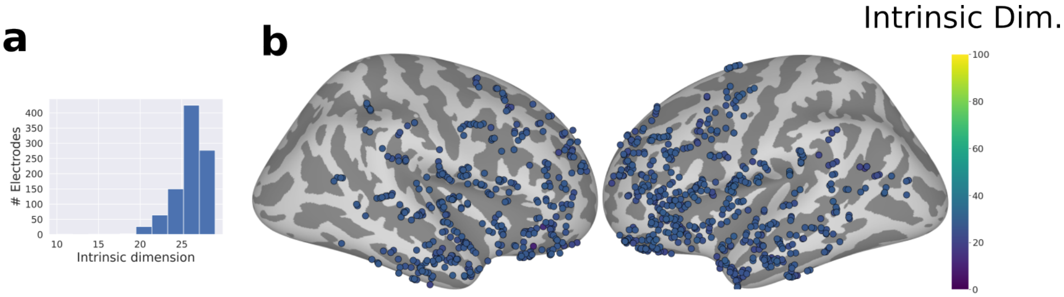

Figure 10: **Intrinsic dimension of STFT features** In fig. 6, we show plot the intrinsic dimension for each electrode based on the BrainBERT representation of that electrode's activity. For comparison, a similar plot (b) can be made using the raw short-time Fourier transform (STFT) features. Unlike for the BrainBERT representations, the distribution of dimensions is tightly grouped around a few values (a).

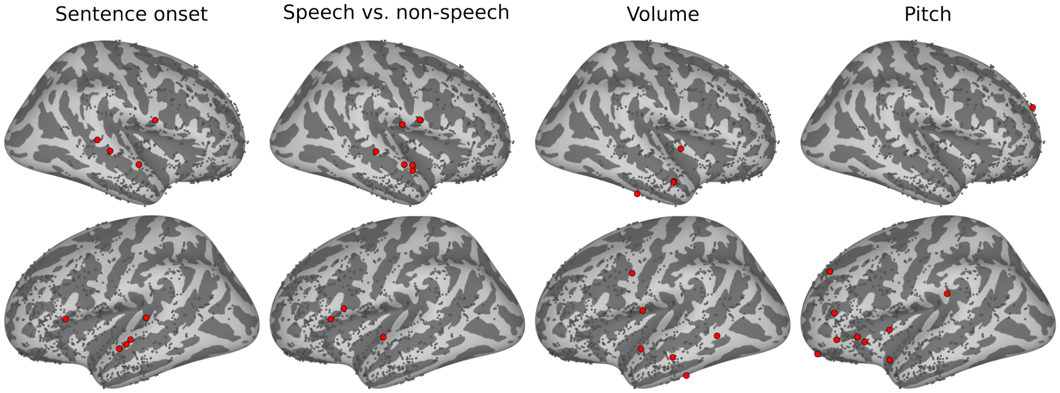

Figure 11: **Location of test electrodes** The results in section 4 are given according to a fixed subset of electrodes per task. For each task, this subset is determined by training a linear classifier over all electrodes and taking the top 10 electrodes with the best performance. The location of these electrodes is shown here.

N Supplementary Results

		Sent. onset	Speech/Non-speech	Pitch	Volume
Top 10	Linear (5s, time domain)	.63±.04	.58 ± .06	.58 ± .07	.56 ± .19
	BrainBERT (STFT)	.82±.07	.93 ± .03	.75 ± .03	.83 ± .09
	BrainBERT (superlet)	.78±.08	.86 ± .06	.62 ± .05	.70 ± .10
Top 20	Linear (5s, time domain)	.63±.04	.57 ± .04	.58 ± .06	.61 ± .10
	CortexBERT (STFT)	.82±.08	.91 ± .10	.74 ± .06	.85 ± .07
	CortexBERT (superlet)	.77±.10	.81 ± .12	.68 ± .06	.76 ± .09

Table 5: Performance is evaluated per task and per electrode. In order to keep computational costs reasonable, since there are many baselines and ablations to compare against, we only evaluate the performance for a subset of all electrodes. For a given task, this subset is selected by first training a linear classifier over all electrodes. In section 4, we find the top 10 electrodes with the highest linear classifier performance. This subset is then held fixed over all comparisons. The variance reported is calculated with respect to this set of electrodes. In this table, we report the results for the top 20 electrodes as well for comparison.

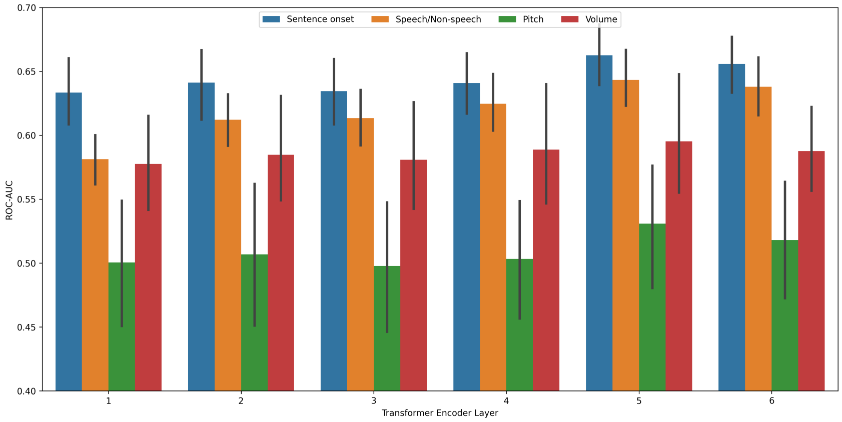

Figure 12: **Layerwise analysis** We compute the decoding performance of the frozen BrainBERT features per layer of the transformer encoder stack. We see that for all tasks, performance increases with depth in the stack, peaking at the second to last layer. Note that in this work, we used the features taken from the last layer. And for the purposes of showing the advantages of BrainBERT's pre-training, the performance we obtain is sufficient.

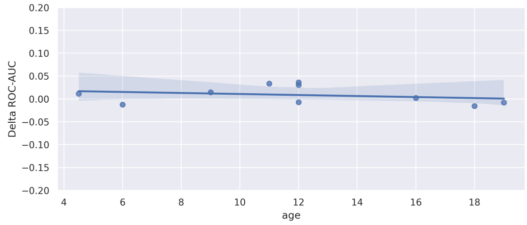

Figure 13: **Generalization vs. age.** The ability of BrainBERT to transfer to new subjects does not depend on age (Pearson's $r = -0.27$, with p-value 0.451). The y-axis shows the delta between the AUC-ROC, averaged across all tasks, between BrainBERT pretrained with and without a particular subject. The x-axis shows the age of the subject. See table 6 for complete breakdown per task.

	Sentence onset	Speech/Non-speech	Pitch	Volume	Average
pretrain w/ subject 1	.74±.09	.85 ± .06	.79 ± .05	.92 ± .03	.82 ± .09
pretrain w/o subject 1	.73±.09	.85 ± .05	.77 ± .10	.92 ± .03	.82 ± .10
Linear (5s)	.62±.04	.56 ± .04	.52 ± .02	.62 ± .06	.58 ± .06
pretrain w/ subject 2	.61±.05	.83 ± .09	.74 ± .03	.73 ± .10	.73 ± .11
pretrain w/o subject 2	.62±.08	.85 ± .07	.77 ± .03	.82 ± .13	.77 ± .12
Linear (5s)	.59±.06	.53 ± .03	.52 ± .05	.61 ± .11	.56 ± .08
pretrain w/ subject 3	.85±.07	.81 ± .10	.65 ± .03	.86 ± .10	.79 ± .12
pretrain w/o subject 3	.84±.05	.78 ± .14	.61 ± .07	.88 ± .03	.78 ± .13
Linear (5s)	.64±.02	.57 ± .03	.54 ± .04	.56 ± .03	.58 ± .05
pretrain w/ subject 4	.57±.07	.94 ± .03	.66 ± .08	.69 ± .06	.72 ± .15
pretrain w/o subject 4	.60±.08	.91 ± .12	.66 ± .06	.75 ± .09	.73 ± .15
Linear (5s)	.58±.05	.54 ± .02	.61 ± .04	.64 ± .05	.59 ± .05
pretrain w/ subject 6	.63±.11	.76 ± .12	.69 ± .03	.70 ± .06	.70 ± .10
pretrain w/o subject 6	.64±.11	.73 ± .16	.76 ± .03	.78 ± .04	.73 ± .11
Linear (5s)	.56±.04	.53 ± .04	.53 ± .06	.58 ± .06	.55 ± .06
pretrain w/ subject 7	.74±.06	.91 ± .04	.68 ± .03	.83 ± .08	.79 ± .10
pretrain w/o subject 7	.75±.06	.86 ± .13	.66 ± .06	.85 ± .02	.78 ± .11
Linear (5s)	.63±.04	.57 ± .05	.58 ± .03	.53 ± .05	.58 ± .05
pretrain w/ subject 5	.71±.09	.75 ± .02	.65 ± .03	.84 ± .04	.74 ± .09
pretrain w/o subject 5	.74±.05	.81 ± .04	.65 ± .07	.89 ± .03	.77 ± .10
Linear (5s)	.56±.05	.53 ± .04	.52 ± .04	.55 ± .03	.54 ± .04
pretrain w/ subject 8	.68±.10	.89 ± .13	.66 ± .05	.76 ± .04	.74 ± .12
pretrain w/o subject 8	.70±.06	.85 ± .12	.72 ± .06	.75 ± .08	.76 ± .10
Linear (5s)	.61±.04	.54 ± .04	.59 ± .04	.57 ± .05	.58 ± .05
pretrain w/ subject 9	.75±.11	.80 ± .09	.75 ± .04	.59 ± .21	.72 ± .15
pretrain w/o subject 9	.78±.06	.80 ± .10	.76 ± .03	.56 ± .16	.72 ± .14
Linear (5s)	.55±.04	.53 ± .02	.53 ± .02	.80 ± .08	.60 ± .12
pretrain w/ subject 10	.79±.05	.81 ± .12	.68 ± .07	.90 ± .05	.80 ± .11
pretrain w/o subject 10	.76±.06	.84 ± .10	.70 ± .03	.86 ± .08	.79 ± .09
Linear (5s)	.62±.04	.53 ± .03	.53 ± .04	.56 ± .05	.56 ± .06

Table 6: **Held-one-out analysis**. For each subject, we report the performance of two versions of BrainBERT with superlet input. One version is trained without that particular subject, and one version is trained using all subjects. Fine-tuning performance is reported on a subset of electrodes belonging to the subject in question. As before, this subset is determined by running a linear classifier over all electrodes and taking the top 10 electrodes with the highest performance. The performance of that linear classifier is also shown for comparison. In general, we see a close match between the two versions of BrainBERT. In some instances, the version without a particular subject even performs better. This can be attributed to the fact that one subject's data may have a negative contribution when it comes to learning useful representations for a particular task.