PNAS
www.pnas.org

Supplementary Information for

A hierarchy of linguistic predictions during natural language comprehension

Micha Heilbron, Kristijan Armeni, Jan-Mathijs Schoffelen, Peter Hagoort & Floris P. de Lange

Micha Heilbron
E-mail: micha.heilbron@donders.ru.nl

This PDF file includes:

Supporting Information Text

A. Supplementary methods.

A.1. Self-attentional language model. Contextual predictions were quantified using GPT-2 – a large, pre-trained language model (1). Formally, a language model can be cast as a way of assigning a probability to a sequence of words (or other symbols), $(x_1, x_2, ..., x_n)$. Because of the sequential nature of language, the joint probability, $P(X)$ can, via the chain rule, be factorised as the product of conditional probabilities:

$$\begin{aligned} P(X) &= p(x_1) \times p(x_2 \mid x_1) \times \cdots \times p(x_n \mid x_{n-1}, \ldots, x_1) \\ &= \prod_{i=1}^{x} p\left(x_n \mid x_1, \ldots, x_{n-1}\right) \end{aligned} \quad [1]$$

Since the advent of neural language models, as opposed to statistical (Markov) models, methods to compute these conditional probabilities have strongly improved. Improvements have been especially striking in the past few years with the introduction of the *Transformer* (2) architecture, which allows efficient training of very large networks on large, diverse data. This resulted in models that dramatically improved the state-of-the art in language modelling on a range of domains.

GPT-2 (1) is one of these large, transformer-based language models and is currently among the best publcicly released models of English. The architecture of GPT-2 is based on the decoder-only version of the transformer. In a single forward pass, it takes a sequence of tokens $U = (u_1, \ldots, u_k)$ and computes a sequence of conditional probabilities, $(p(u_1), p(u_2|u_1), \ldots, p(u_k \mid u_1, ..., u_{k-1}))$. Roughly, the full model (see Figure S1) consists of three steps: first, an embedding step encodes the sequence of symbolic tokens as a sequence of vectors which can be seen as the first hidden state h_o. Then, a stack of transformer blocks, repeated n times, each apply a series of operations resulting in a new set of hidden states h_l, for each block l. Finally, a (log-)softmax layer is applied to compute (log-)probabilities over target tokens. Formally, then, the model can be summarised in three equations:

$$h_0 = UW_e + W_p \quad [2]$$

$$h_l = \text{transformer\_block}\left(h_{l-1}\right) \forall i \in [1, n] \quad [3]$$

$$P(u) = \text{softmax}\left(h_n W_e^T\right), \quad [4]$$

where W_e is the token embedding and W_p is the position embedding (see below).

The most important component of the transformer-block is the *masked multi-headed self-attention* (Fig S1). The key operation is self-attention, a seq2seq operation turning a sequence of input vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_k)$ into a sequence of output vectors $(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_k)$. Fundamentally, each output vector $\mathbf{y}_i$ is a weighted average of the input vectors: $\mathbf{y}_i = \sum_{j=1}^{k} w_{ij}\mathbf{x}_j$. Critically, the weight $w_{i,j}$ is not a parameter but is *derived* from a function over input vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. The Transformer uses (scaled) *dot product attention*, meaning that the function is simply a dot product between the input vectors $\mathbf{x}_i^T\mathbf{x}_j$, passed through a softmax make sure that the weights sum to one, scaled by a constant determined by the dimensionality, $\frac{1}{\sqrt{d_k}}$ (to avoid the dot-products growing too large in magnitude): $w_{ij} = (\exp \mathbf{x}_i^T\mathbf{x}_j / \sum_{j=1}^{k} \exp \mathbf{x}_i^T\mathbf{x}_j)\frac{1}{\sqrt{d_k}}$.

In self-attention, then, each input $\mathbf{x}_i$ is used in three ways. First, it is multiplied by the other vectors to derive the weights for its own output, $\mathbf{y}_i$ (as the *query*). Second, it is multiplied by the other vectors to determine the weight for any other output $\mathbf{y}_j$ (as the *key*). Finally, to compute the actual outputs it is used in the weighted sum (as the *value*). Different (learned) linear transformations are applied to the vectors in each of these use cases, resulting in the Query, Key and Value matrices (Q, K, V). Putting this together, we arrive at the following:

$$\text{self\_attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad [5]$$

where d_k is dimension of the keys/queries. In other words, self_attention simply computes a weighted sum of the values, where the weight of each value is determined by the dot-product similarity of the query with its key. Because the queries, keys and values are linear transformations of the same vectors, the input *attends itself*.

To be used as a language model, two elements need to be added. First, the basic self-attention operation is not sensitive to the order of the vectors: if the order of the input vectors is permuted, the output vectors will be identical (but permuted). To make it position-sensitive, a position embedding W_p is simply added during the embedding step – see Equation 2. Second, to enforce that the model only uses information from one direction (i.e left), a mask is applied to the attention weights (before the softmax) which sets all elements above the diagonal to $-\infty$. This makes the self-attention *masked*. To give the model more flexibility, each transformer block actually contains multiple instances of the basic self-attention mechanisms from Eq. (5). Each instance (each *head*) applies different linear transformations to turn the same input vectors into a different set of Q, K and V matrices, returning a different set of output vectors. The outputs of all heads are concatenated and then reduced to the initial dimensionality with a linear transformation. This makes the self-attention *multi-headed*.

In total, GPT-2 (XL) contains $n = 48$ blocks, with 12 heads each; a dimensionality of $d = 1600$ and a context window of $k = 1024$, yielding a total 1.5×10^9 parameters. We used the PyTorch implementation of GPT-2 provided by HuggingFace's *Transformers* package (3).

A.2. Deriving word-by-word predictions from arbitrary-length texts. Passing the texts through Equations 2-4 results in a sequence of conditional (log-)probability distributions over tokens $P(U)$, for each token. Since GPT-2 uses Byte-Pair Encoding, a token can be either punctuation or a word or (for less frequent words) a word-part. How many words actually fit into a context window of length k therefore depends on the text. For words spanning multiple tokens, we computed word probabilities simply as the joint probability of the tokens.

Assuming that listeners' expectations would to some extent 'reset' during the break, we passed the text for each run separately through GPT2. In the EEG experiment, the text of each run was shorter than 1024 tokens for each run, implying that word probabilities were conditional on *all* preceding words in the run (i.e. contexts up to 3 minutes long). In the MEG experiments, runs were considerably longer, and often contained more tokens than the context window size of 1024. For these texts, we used a windowed approach. For window-placement, we used the constraint that the windows had an overlap of at least 700 tokens, and that they could not start mid-sentence (ensuring that the first sentence of the window was always well-formed). As such, for each word w_i we computed $p(w_i|\text{context})$, where 'context' consisted either of all preceding words in the run, or of a sequence of prior words constituting a well-formed context that was at least 700 tokens long.

A.3. Syntactic and semantic predictions. Feature-specific predictions were computed from the lexical prediction. To this end, we first truncated the unreliable tail from the distribution using a combination of top-k and nucleus truncation. The nucleus was defined as the "top" k tokens with the highest predicted probability, where k was set dynamically such that the cumulative probability was at least 0.9 – a criterion known to be effective at preserving the core of the prediction, while removing the less reliable tail of the distribution (4). To have enough information also for very low entropy cases (where k becomes small), we forced k to be a least 40.

From this truncated distribution, we derived feature-specific predictions by analysing the predicted (top-k) words. For the part-of-speech predictions, we performed part of speech tagging on every potential sentence (i.e. the context plus the predicted word) with Spacy (5) to derive the probability distribution over parts-of-speech, from which the POS surprisal was calculated as the negative log probability of the POS of the presented word. In other words, for word n in the text, the POS surprisal is:

$$\log(P(\text{POS}_n\,|\text{context})) = \log(\frac{\sum_{q\in Q} P(w_q^{(n)}|\text{context})}{\sum_{i\in K} P(w_i^{(n)}|\text{context})}), \quad [6]$$

where Q is the range of predicted words with the same POS as the actually presented word n, and K is the range of the top-k most probable words in the distribution.

For the semantic prediction, we took a weighted average of the glove embeddings of the predicted words to compute the expected vector: $\mathbb{E}[\text{G}(w_n)] = \sum_{i=1}^{k} P(x_i|\text{context})\,\text{G}(x_i)$, where $\text{G}(x_i)$ is the GloVe embedding for predicted word x_i, where i ranges over the top-k words in the truncated distribution, and $P(x_i|\text{context})$ is renormalized such that this truncated distribution sums to one. From this prediction, we computed the semantic prediction error as the cosine distance between the predicted and observed vector:

$$\text{PE}_{\text{semantic}} = 1 - \frac{\mathbb{E}\left[\text{G}\left(w_n\right)\right]\text{G}\left(w_n\right)}{\left\|\mathbb{E}\left[\text{G}\left(w_n\right)\right]\right\|\left\|\text{G}\left(w_n\right)\right\|}. \quad [7]$$

Technically, this is a *prediction error* because it is defined as a distance, rather than as as a negative log probability. Conceptually however, there is no important difference: surprisal is a probabilistic measure of prediction error.

A.4. Phonemic predictions. Phonemic predictions were formalised in the context of incremental word recognition (6, 7). This process can be cast as probabilistic prediction by assuming that brain is tracking the *cohort* of candidate words consistent with the phonemes so far, each word weighted by its prior probability. We compared two such models that differed only in the prior probability assigned to each word.

The first model was the single-level or frequency-weighted model (Fig 6), in which prior probability of words was fixed and defined by a word's overall probability of occurrence (i.e. lexical frequency). The probability of a specific phoneme (A), given the prior phonemes within a word, was then calculated using the statistical definition:

$$P(\varphi_t = A \mid \varphi_{1:t-1}) = \frac{f(C_{\varphi_t=A})}{f(C_{\varphi_{1:t-1}})}. \quad [8]$$

Here, $f(C_{\varphi_t=A})$ denotes the cumulative frequency of all words in the remaining cohort of candidate words if the next phoneme were A, and $f(C_{\varphi_{(1:t-1)}})$ denotes the cumulative frequency of all words in the prior cohort (equivalent to $f(C)$ of all potential continuations). If a certain continuation did not exist and the cohort was empty, $f(C_{\varphi_t=A})$ was assigned a laplacian pseudocount of 1. To efficiently compute Eq. (8) for every phoneme, we constructed a statistical phonetic dictionary as a digital tree that combined frequency information from SUBTLEX database and pronunciation from the CMU dictionary.

The second model was equivalent to the first model, except that the prior probability of each word was not defined by its overall probability of occurrence, but by its conditional probability in that context (based on GPT-2). This was implemented by constructing a separate phonetic dictionary for every word, in which lexical frequencies were replaced by implied counts derived from the lexical prediction. We truncated the unreliable tail from the distribution and replaced that by a flat tail that assigned each word a pseudocount of 1. Since all counts in the tail are 1, the cumulative implied counts of the nucleus

is complementary to the the length of the tail, which is simply the difference between the vocabulary size and nucleus size $(V - k)$. As such a little algebra reveals:

$$\text{freqs}_n = P_{tr}(w^{(i)}|\text{context}) \frac{V - k}{1 - \sum_{j=1}^{k} P(w_j^{(i)}|\text{context})}, \quad [9]$$

where $P_{tr}(w^{(i)}|\text{context})$ is the trunctated lexical prediction, and $P(w_j^{(i)}|\text{context})$ is predicted probability that word i in the text is word j in the sorted vocabulary.

Although we computed probabilities using simply the implied relative frequencies, these two ways of assigning relative probabilities are equivalent to two kinds of priors in a Bayesian model. Specifically, in the first model the prior over words is the fixed unconditional word probability, while in the second model the prior is the contextual probability, itself based on a higher level (lexical) prediction. This makes the second computation *hierarchical*. Because phoneme predictions are based on not just (at the first level) on short sequences of within-word phonemes, but also on a contextual prior which itself (at the second level) is based on long sequences of prior words.

B. Reduced regression control analysis. In our analysis of spatial patterns of explained variance (Figures 4, S9) and coefficients (Figures 5, S10) we observed dissociable spatiotemporal signatures of different prediction error regressors. We interpreted these as independent neural effects of different types of linguistic unexpectedness. However, an alternative possibility is that these signatures may reflect an artefact of our analysis, in which we subdivide unexpectedness into multiple, partially correlated regressors (Figure 3). These multiple regressors could then reveal a linear approximation of a single non-linear effect, instead of reflecting multiple, independent underlying effects.

To assess this possibility, we performed reduced regressions. Here, the logic is that if the signatures indeed reflect independent effects, these effects should be preserved in a reduced regression in which only a single covariate, or only a single unexpectednes covariate, is included. Note that due to the inherent correlations between unexpectedness regressors, the results of the reduced regressions are not expected to be identical to the full regression. Specifically, in the reduced regression there may be additional variance assigned to an unexpectedness regressor – variance that in the full model would be assigned to a different regressor. We thus expect the effects found in the reduced regression to be a *superset* of those in the full regression. To test this, we compared the patterns of unique explained variance (Figures 4, S9) and of the coefficients (Figures 5, S10) observed in the *full model*, with those obtained with two reduced models, that either contained (i) only baseline covariates, but no surprisal or prediction error covariates; or (ii) including only onset regressors, and no other covariates aside from the surprisal or prediction error covariate of interest. We focus on the spatiotemporal coefficients in the EEG data and patterns of additional explained variance in the MEG data, as these are the primary targets of statistical evaluation.

The results show that as expected, the overall patterns are preserved in the reduced regression analysis, plus some additional effects (i.e. in line with our expectation of finding a *superset* of the effects in the full regression model). In the coefficient analysis, this is especially clear for the POS surprisal regressor (Figure S17). In the reduced models, the coefficients for POS surprisal preserve the characteristic early frontal positivity (like in the full regression model), but additionally reveal a N400-like posterio-central negativity – an effect which in the full model is not attributed to the POS surprisal, presumably because it is already explained by overall lexical surprisal (Fig S12). A similar 'residual N400 effect' is also observed in the coefficients for the semantic prediction error regressor. For the spatial patterns of explained variance, the additional effects are best visible in terms of extra areas and higher amount of 'unique' variance explained, due to variance that is shared between unexpectedness regressors, is now being attributed solely to a single unexpectedness regressor (see Figure S16). Another notable aspect of patterns of explained variance, is that in the reduced regressions, the differences between participants appear less stark, especially in terms of lateralisation. This may reflect that in the full model, we only focus on *unique* explained variance, and discard 'shared' variance equally well accounted by multiple regressors. This may amplify apparent differences between participants, by only focussing on the tip of the statistical 'iceberg'.

Together, the reduced regressions suggest that the dissociable signatures are not an artefact of our multiple regression approach, and instead reflect independent effects – in line with our interpretation.

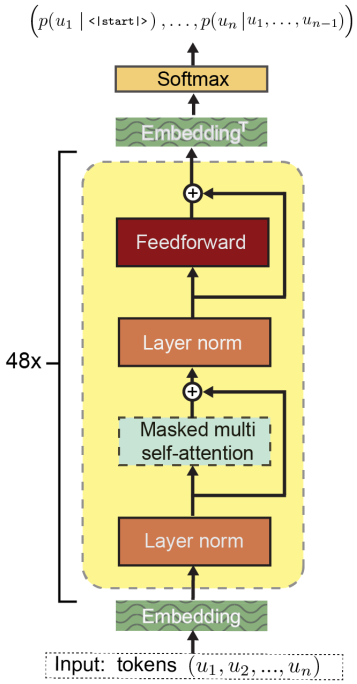

Fig. S1. GPT-2 Architecture. Note that this panel is a re-rendered version of the original GPT schematic, slightly modifyied and re-arranged to match the architecture of GPT-2. For more details on the overall architecture and on the critical operation of self-attention, see *Methods*. In this graphic, Layer Norm refers to layer normalisation. Not visualised here is the initial tokenisation, mapping a sequence of characters into tokens.

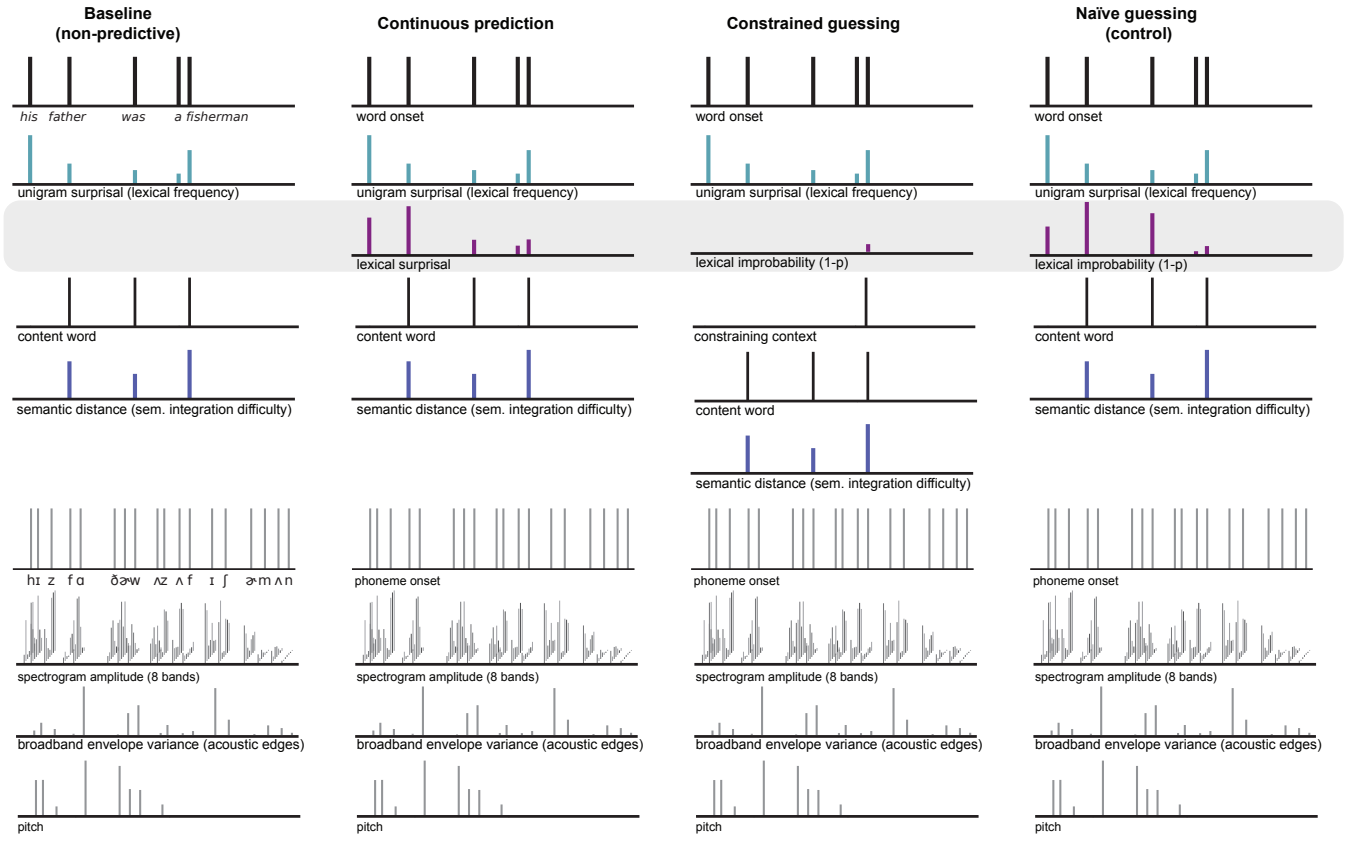

Fig. S2. WORD-LEVEL REGRESSION MODELS. Schematic of the main models plus the control model of the model comparison test for predictive processing at the word level, for an example fragment from the EEG stimulus material. In all models, lexical attributes are modelled on a word-by-word basis; acoustic control variables (grey) are modelled on a phoneme-by-phoneme basis. For voiceless phonemes or other segments where PRAAT could not track the pitch, pitch values were set to zero. Transcription of the words and phonemes are given under the word and phoneme onset impulses in the Baseline model. Because we use a regression ERP/ERF framework (8), aimed at capturing (modulations of) evoked responses to discrete events like words or phonemes, all regressors are modelled as impulses (see *Methods*).

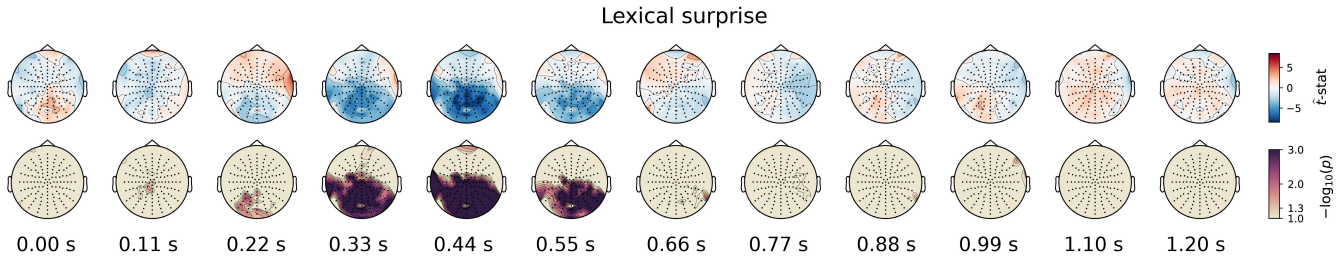

Fig. S3. Full EEG topographies of the effects of lexical surprise These topographies show the average t-statistics of the coefficients (upper row) and respective FWE-corrected significance (lower row) of the lexical surprise regressor from the *continuous prediction* model (Figure S2). As such, while Figure 2b shows the coefficients averaged over channels participating in the cluster (thereby only visualising *the effect*) these topographies visualise the results comprehensively across all channels over time.

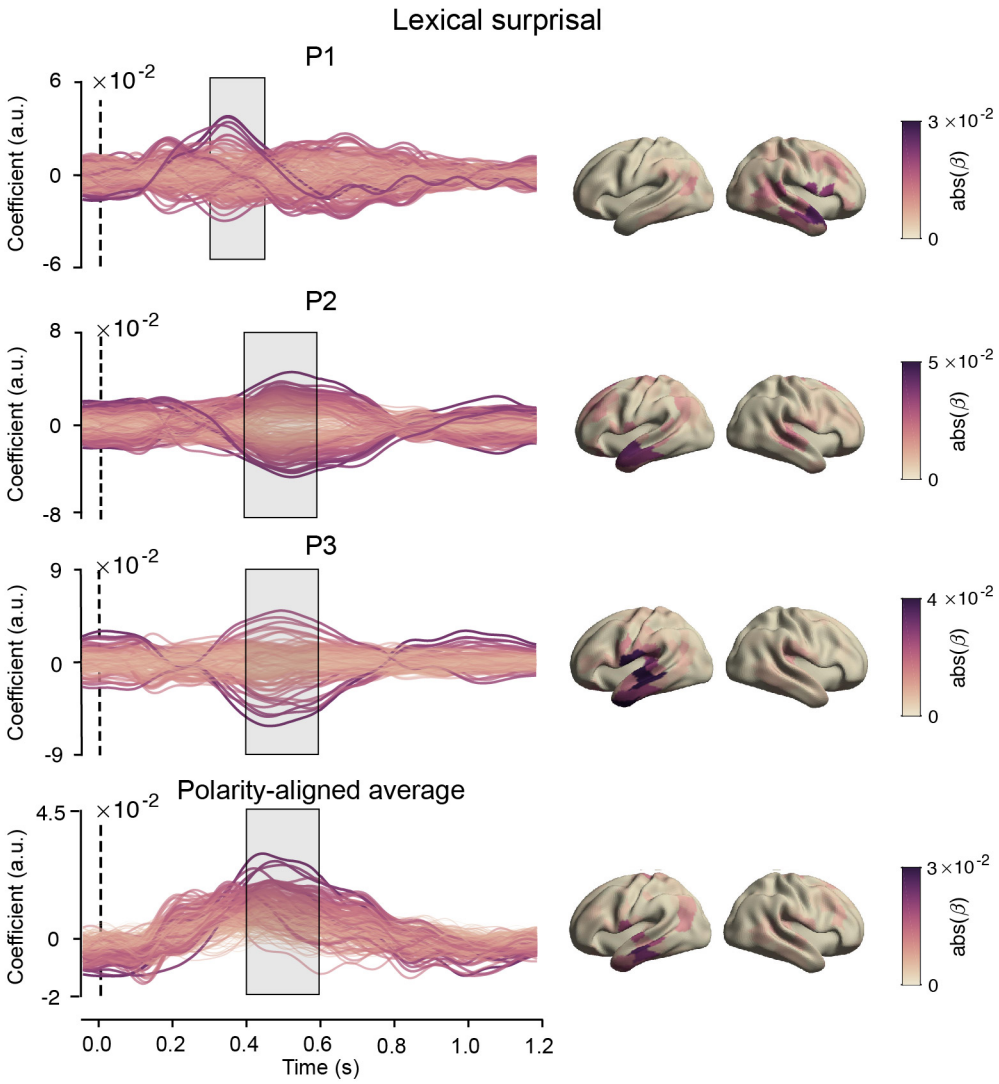

Fig. S4. Coefficients for lexical surprise from the lexical model (Figure S2) Left column: timecourses of the coefficients at each MEG source-localised parcel for lexical surprise for all MEG participants, and the polarity-aligned average across them. Right column: Absolute value of the coefficients averaged across the highlighted period plotted across the brain.

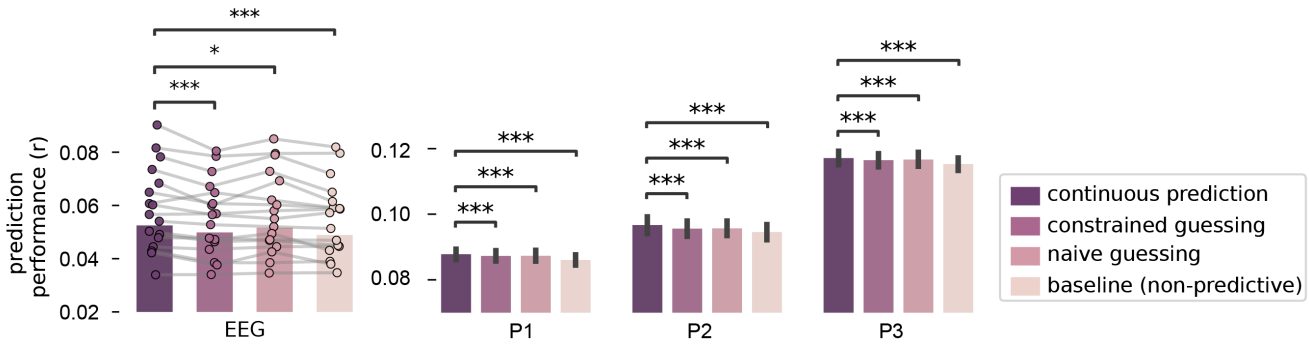

Fig. S5. The effect of word predictability on brain responses is logarithmic and ubiquitous.
Same as in Figure 2a, but now including the 'naive guessing' control model. Like the constrained guessing model, this included a linear (rather than logarithmic) estimate of word improbability, but defined for every word rather than only for constraining contexts. This model was introduced to identify which of the two differences between the *continuous prediction* and *constrained guessing* model – the linearity vs logarithmicity, or the constrainedness vs ubiquity – contributed to the difference in model performance. In both datasets, the *naive guessing* model performed better than the *constrained guessing* model (EEG: $p < 6.24 \times 10^{-4}$; MEG, hierarchical bootstrap t-test across participants: $p = 0.034$) but worse than the *continuous prediction* model (EEG: $p = 0.03$; MEG: all p's $< 4.68 \times 10^{-5}$). This confirms that the effect of unexpectedness is both logarithmic, and not limited to only highly constraining contexts, but found much more generally – in line with the notion of continuous prediction.

Because the naive guessing model includes a *linear* metric of word improbability for *every* word, it strictly speaking formalises the hypothesis that the brain 'naively' makes *all-or-none* guesses about *every* word. Given that this hypothesis appears a-priori implausible, it may seem surprising that the model still performs comparably well. However, note that the predictability regressors of the continuous prediction model (i.e. *surprisal*) and the constrained guessing model (i.e. linear (im)probability) are highly correlated (0.7) because one is a monotonic function of the other. Therefore, we suggest the results are better interpreted the other way around: the fact that – despite being so correlated – the log-probability is consistently a better predictor of neural responses than the linear probability clearly supports that the effect of word predictability on brain responses is logarithmic (see Discussion). This is in line with predictive processing theories, which postulate that the neural response to a stimulus are proportional to negative log-probability of that stimulus.

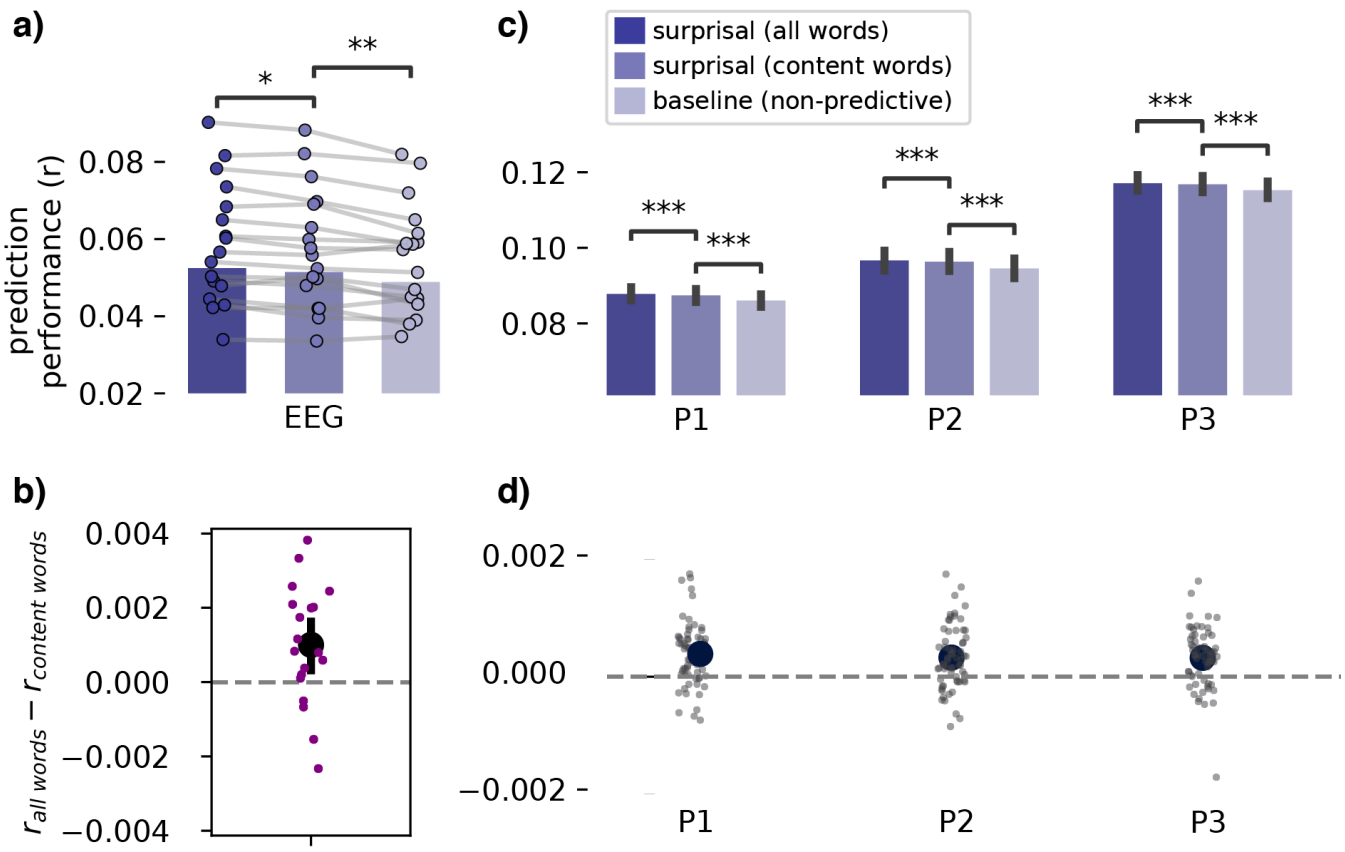

Fig. S6. Surprisal modulations are found for content words and function words.
Expectancy modulations are primarily studied on content words, and studies looking at predictability modulations on function words have obtained mixed results. To test whether the observed surprisal modulation generalises to function words, we compared a model that included surprisal for all content words, to a model that included it additionally for function words (equivalent to the *continuous prediction* model). In both datasets, we found that while the model including surprisal for only content words performed considerably better than the baseline (EEG: $t_{18} = 3.66$, $p = 1.81 \times 10^{-3}$; MEG: all p's $< 3.52 \times 10^{-11}$), the model that also included function words performed better still (EEG (panel **a,b**): $t_{18} = 2.72$, $p = 0.014$; MEG **c,d**: all p's $< 2.13 \times 10^{-5}$). Upper panels show mean cross-validated prediction performance (pearson r), averaged across all channels (EEG, **a**), or all sources in the language network (MEG, **b**). Lower panels show the pairwise differences in model performance between the model including surprisal only for content words, and the model that also included surprisal for function words, for every participant (EEG, **b**) or every run (MEG, **d**). Although the gain in model performance from including function words is small, it is consistent in both datasets.

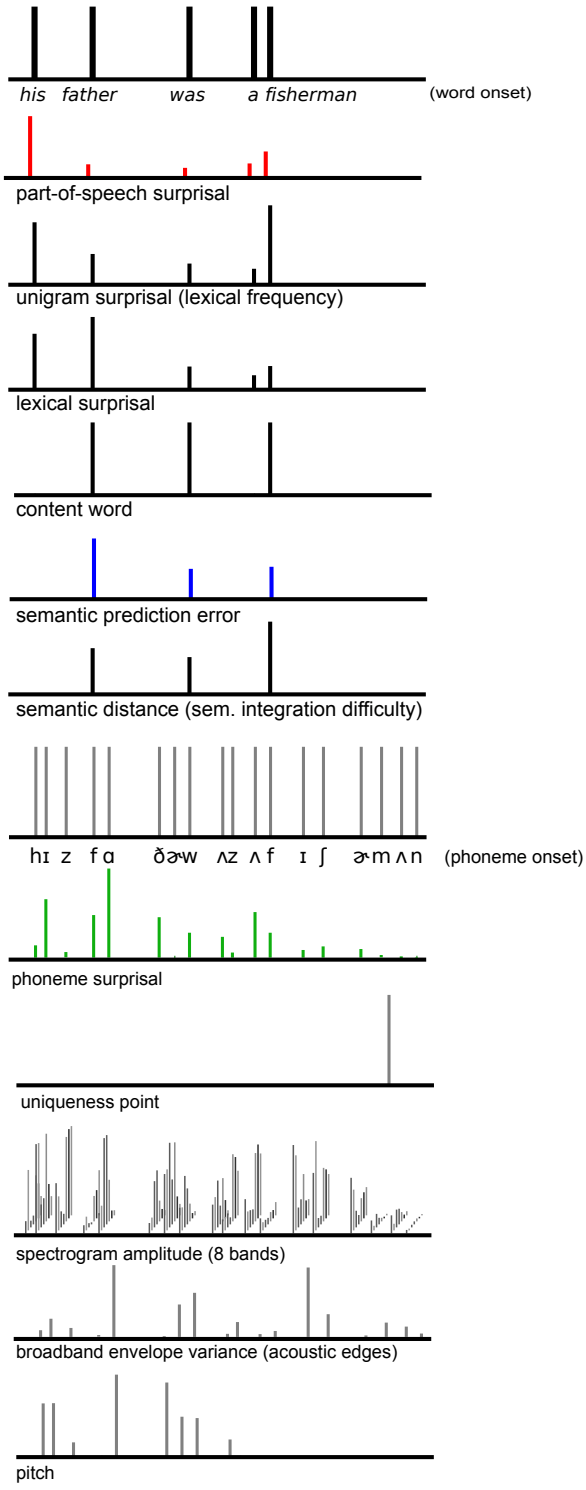

Fig. S7. Regressors of the integrated feature-specific model. Same as Figure S5, but for the integrated feature-specific regression model. The three regressors of interest – POS surprisal, semantic prediction error and phonemic surprisal – are coloured, all control regressors are in black. For voiceless phonemes or other segments where PRAAT could not track the pitch, pitch values were set at zero. Transcription of the words and phonemes from this particular example fragment from the stimulus are given under the word and phoneme onset impulses. Following the regression ERP/ERF scheme (8) all regressors are modelled as impulses (see *Methods*).

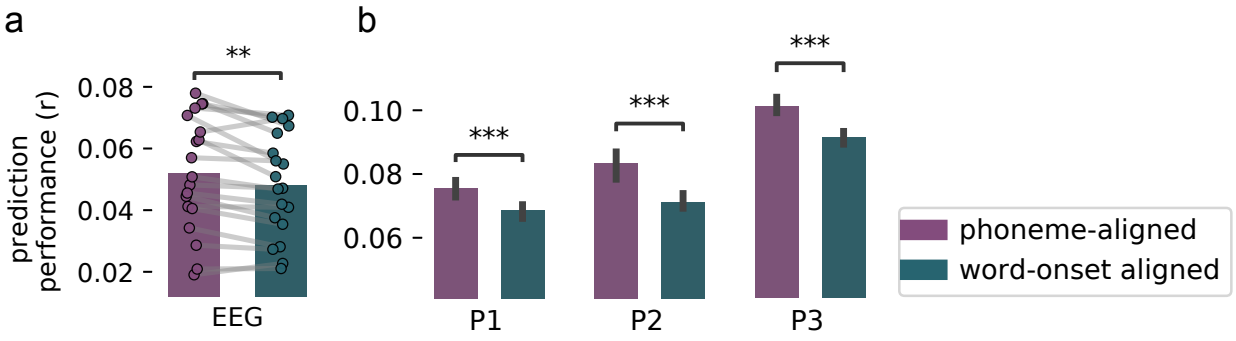

Fig. S8. PHONEME ONSET AND SURPRISAL RESPONSES ARE ALIGNED TO PHONEME ONSETS. Model comparison results of phoneme-alignment test. To test whether the phoneme-level and phoneme surprisal response truly captures a response locked to individual phonemes, rather than being an overparametrized way to capture responses to words, we performed a model comparison. In this comparison we compared a model in which the phoneme-level surprisal was aligned to each phoneme, to a control model in which also contained phoneme-level surprisal, but in which the values for individual phonemes were summed and temporally aligned with the word-initial phoneme. In this analysis, we omitted the acoustics regressors, since for these low-level, sub-linguistic regressors adding more temporal granularity arguably increases prediction performance, irrespective of phoneme alignment. Results show that in both EEG (**a**) and MEG (**b**), the model with phoneme aligned regressors predicts brain responses significantly better (EEG: $t_{18} = 3.8, p = 2.32 \times 10^{-3}$; MEG, Wilcoxon sign-rank: all p's $< 3.75 \times 10^{-37}$. This confirms that the model is not just picking on word-level responses.

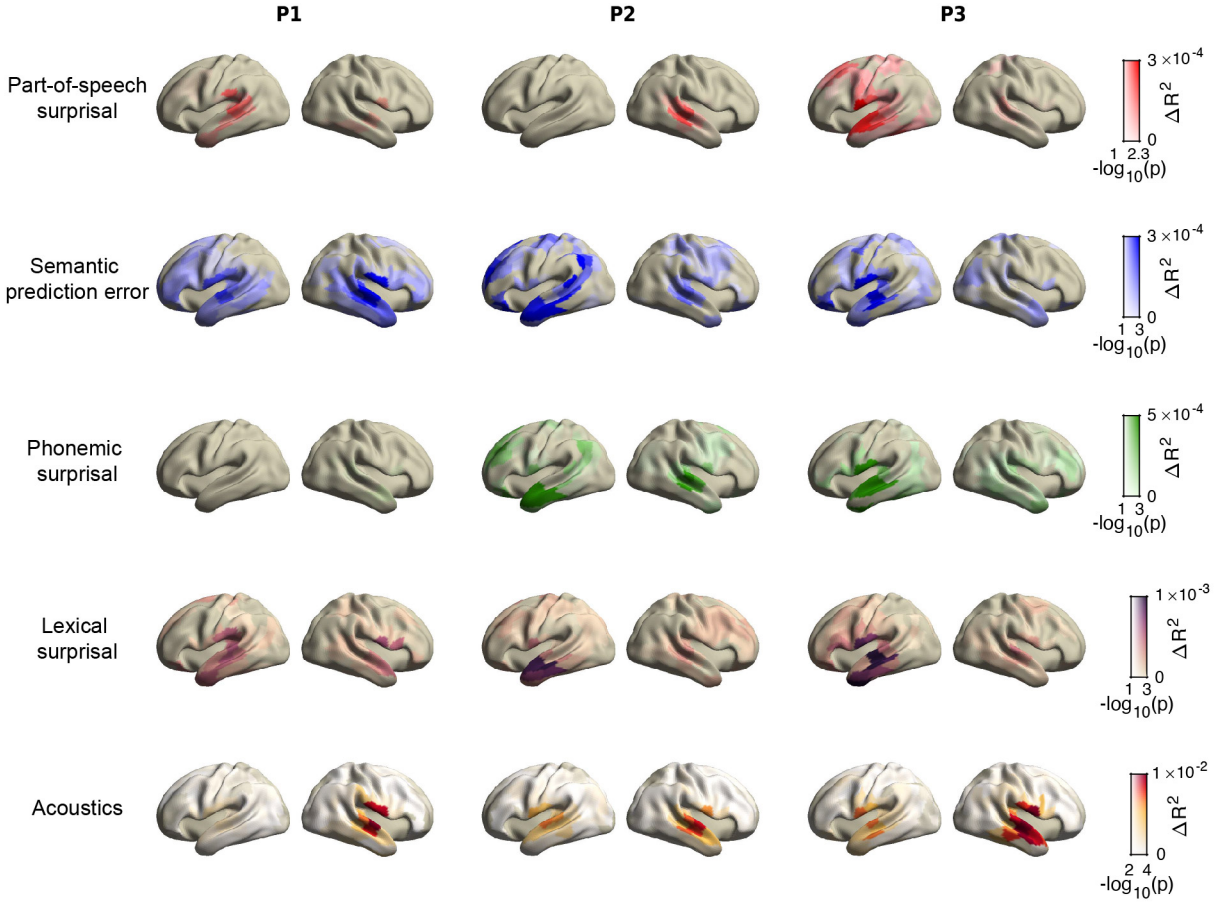

Fig. S9. Unique explained variance for five regressors across the brain.
Same as Figure 4, but including the lexical surprisal control regressor for comparison. Colours indicate amount of additional variance explained by each regressor; opacity indicates the FWE-corrected statistical significance (across cross-validation folds). Note that $p < 0.05$ is equivalent to $-\log_{10}(p) > 1.3$.

	Part-of-speech	Semantic	Phonemic	Lexical
Part-of-speech		0.72, $p = 0.0031$	0.78, $p < 0.0001$	0.78, $p < 0.0001$
Semantic	0.72, $p = 0.0031$		0.71, $p = 0.0014$	0.83, $p < 0.0001$
Phonemic	0.78, $p < 0.0001$	0.71, $p = 0.0014$		0.67, $p = 0.002$
Lexical	0.78, $p < 0.0001$	0.83, $p < 0.0001$	0.67, $p = 0.002$	

Table S1. PAIRWISE DISSOCIABILITY OF SPATIAL PATTERNS OF EXPLAINED VARIANCE.
Pairwise classification performance in the classification-based dissociability analysis (*Methods*) performed to statistically evaluate whether the spatial patterns of additional explained variance explained (see Figures 4, S9) was statistically robust. Tables shows the mean cross-validated classification accuracy, across the 3 participants. P-values indicates a statistical comparison against chance level, using a multi-level nonparametric procedure (hierarchical bootstrapping) to aggregated across all cross-validation folds of all three participants. Classification was based on mean-normalised, non-thresholded spatial maps of explained variance. The fact that all maps of additional explained variance were could be dissociated, indicates that the spatial dissociation was not just present on average across all data, but that it holds up when using cross-validation, indicating that the patterns are robustly different.

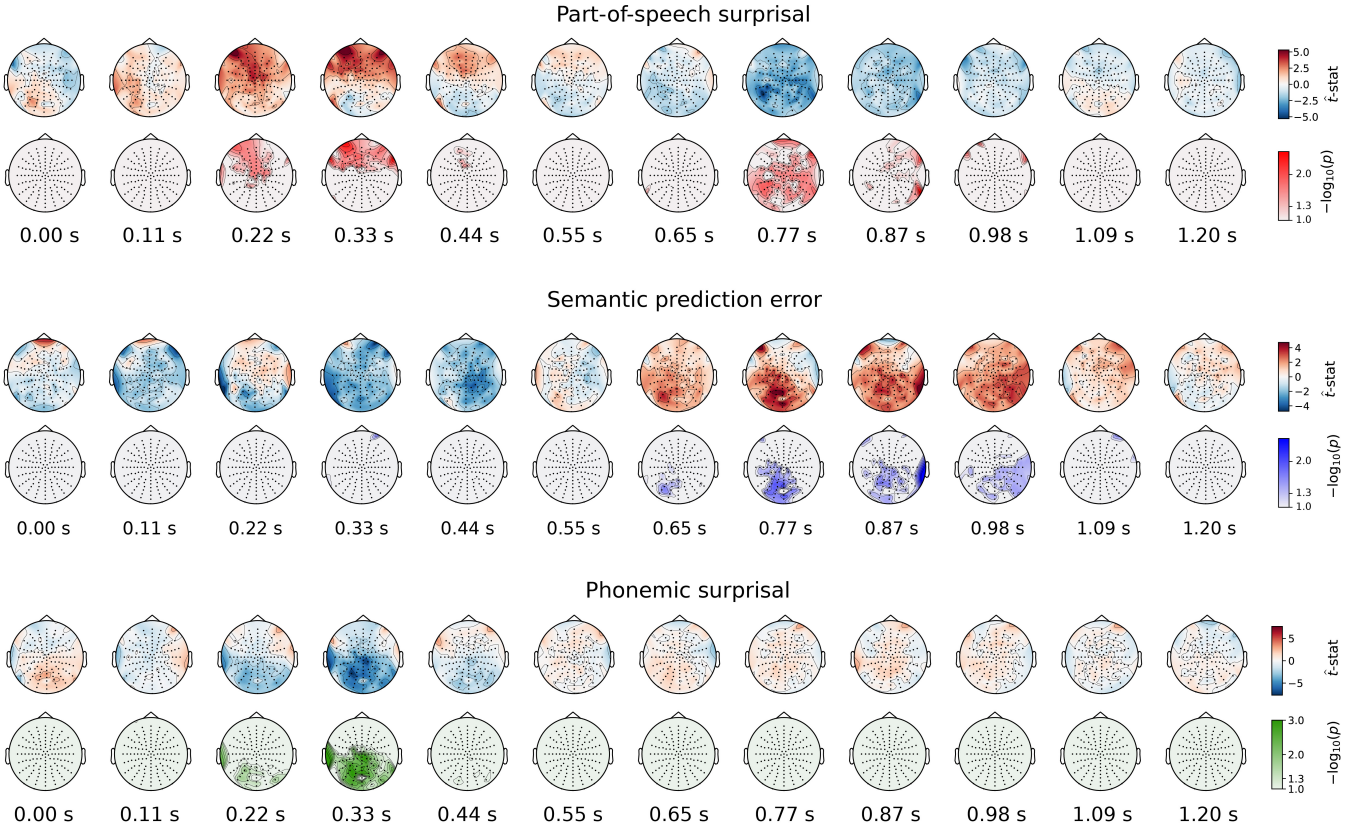

Fig. S10. FULL TOPOGRAPHIES OF THE COEFFICIENTS AND SIGNIFICANCE OF FEATURE-SPECIFIC PREDICTION ERRORS For each feature-specific prediction error regressor, the topographies show the t-statistics of the coefficients (upper row) and the respective TFCE-corrected significance (lower row). So while Figure 5 only shows the coefficients averaged over channels participating in the cluster (i.e. *the effect*) these topographies visualise the results comprehensively across all channels, over time.

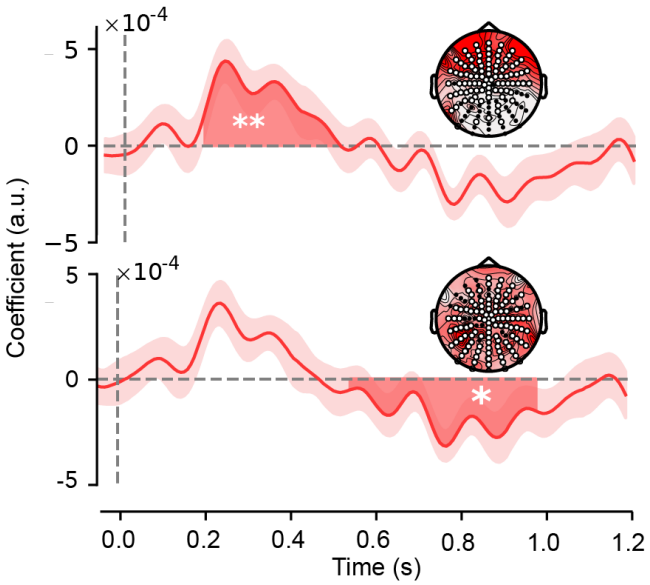

Fig. S11. Significant effects of POS surprisal in the EEG data. Two significant effects were observed in the modulation functions for POS surprisal: an early positive effect with a frontal topography (upper panel) and a later negative effect based on a distributed cluster (lower panel). The early effect tightly replicates recent model-based studies on EEG effects of (morpho)syntactic surprisal, and was also found in the MEG data. By contrast, the late effect of POS surprisal is not in line with any earlier study (note that it is negative unlike the syntactic P600) and importantly was not replicated in the MEG data. Therefore we only consider the early effect a 'main' effect of POS surprisal (visualised in the main Figure 5) and we advice to refrain from interpreting the late effect before it is independently replicated.

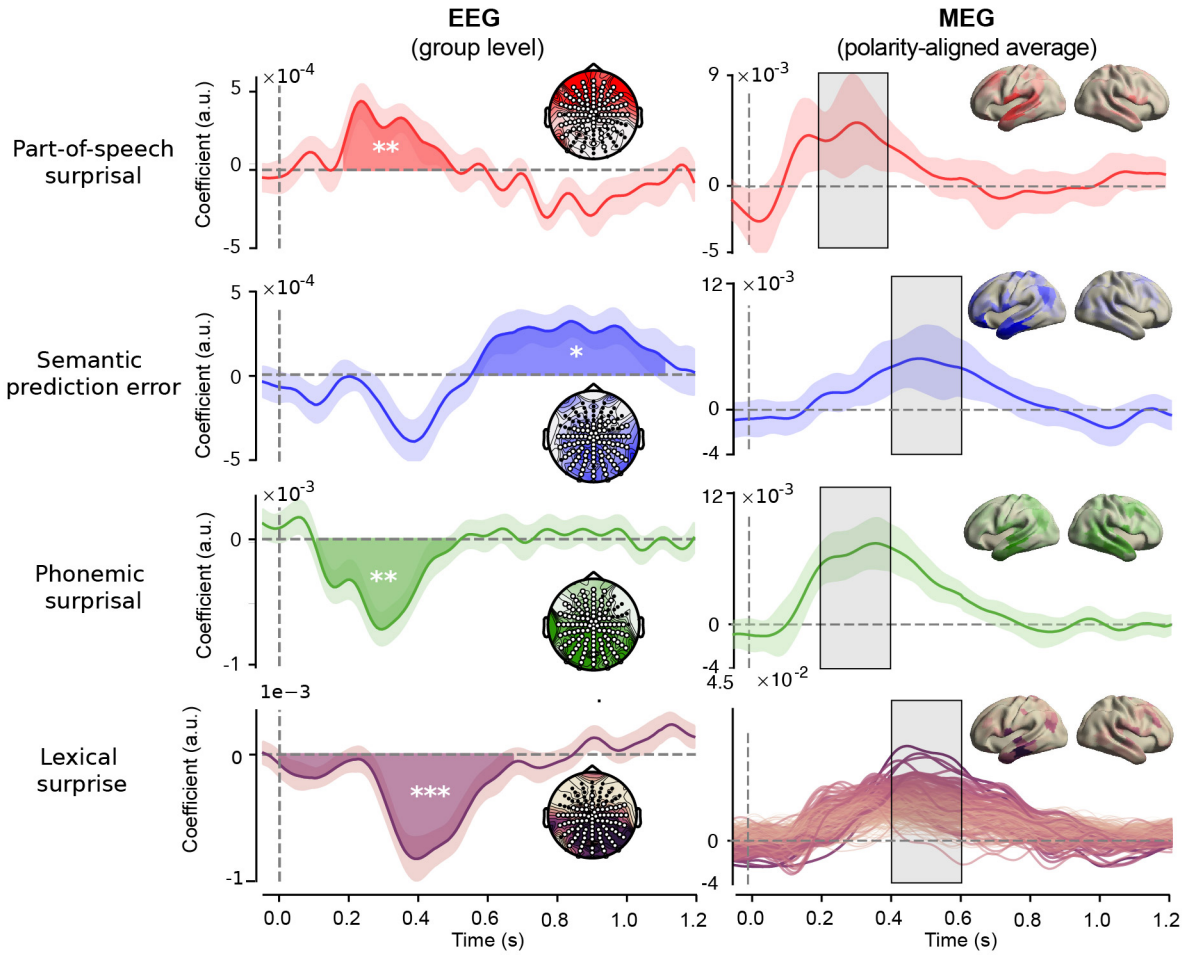

Fig. S12. Coefficients for each feature specific prediction error metric, and lexical surprisal (control variable)
EEG (left column): coefficient modulation function averaged across the channels participating for at least one sample in the significant clusters. Highlighted area indicates temporal extent of the cluster. Shaded area around waveform indicates bootstrapped standard errors. Stars indicate cluster-level significance; $p < 0.05$ (*) , $p < 0.05$ (**), $p < 0.001$ (***). Insets represent channels assigned to the cluster (white dots) and the distribution of absolute values of t-statistics. MEG (right column): polarity aligned responses averaged across participants for all sources (same as in Figure 5 but without averaging over sources, and including two control variables). Insets represent topography of absolute value of coefficients averaged across the highlighted period. Note that due to polarity alignment, sign information is to be ignored for the MEG plots.

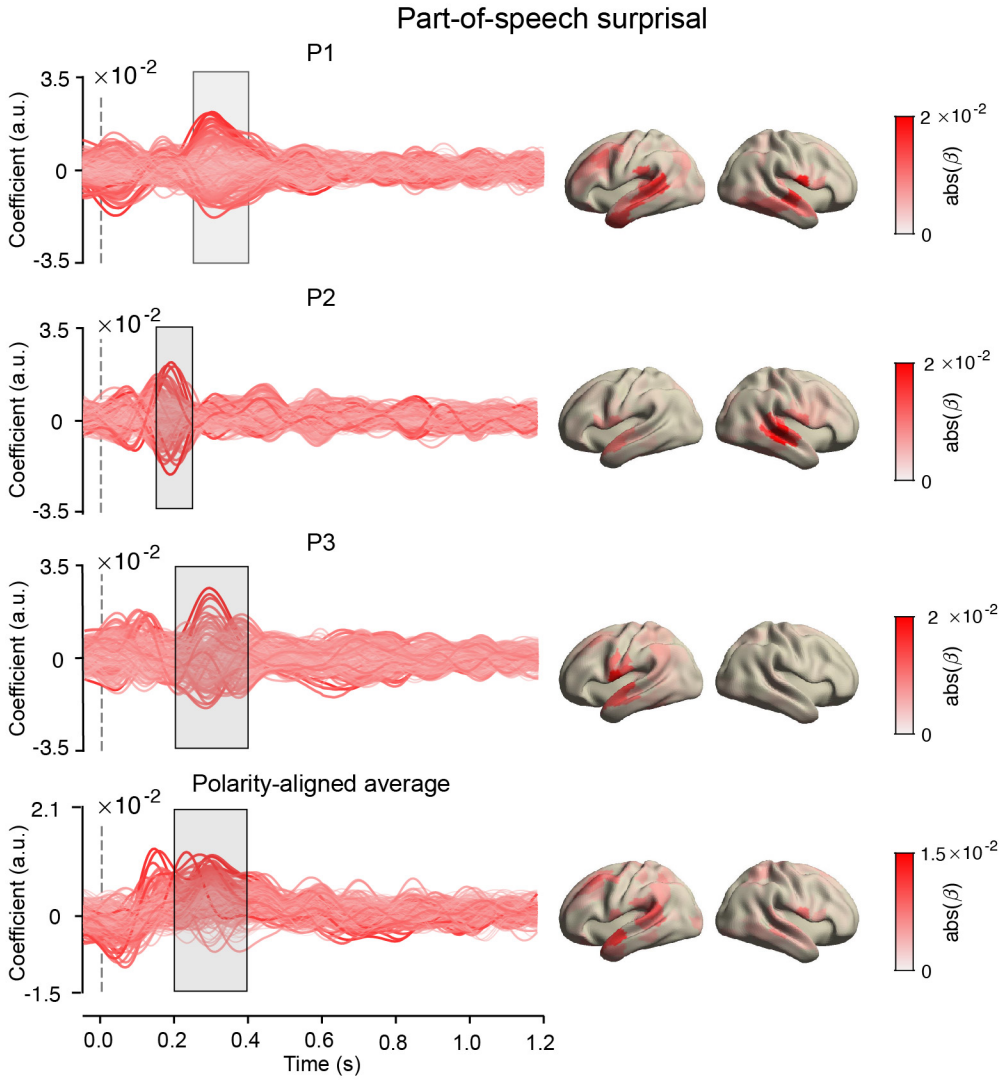

Fig. S13. Coefficients for POS surprisal from the integrated model (Figure S7)
Left column: coefficients for each source for each individual in the MEG experiment, and the polarity-aligned average across participants. Right column: absolute value of the coefficients across the brain, averaged across the highlighted time-period.

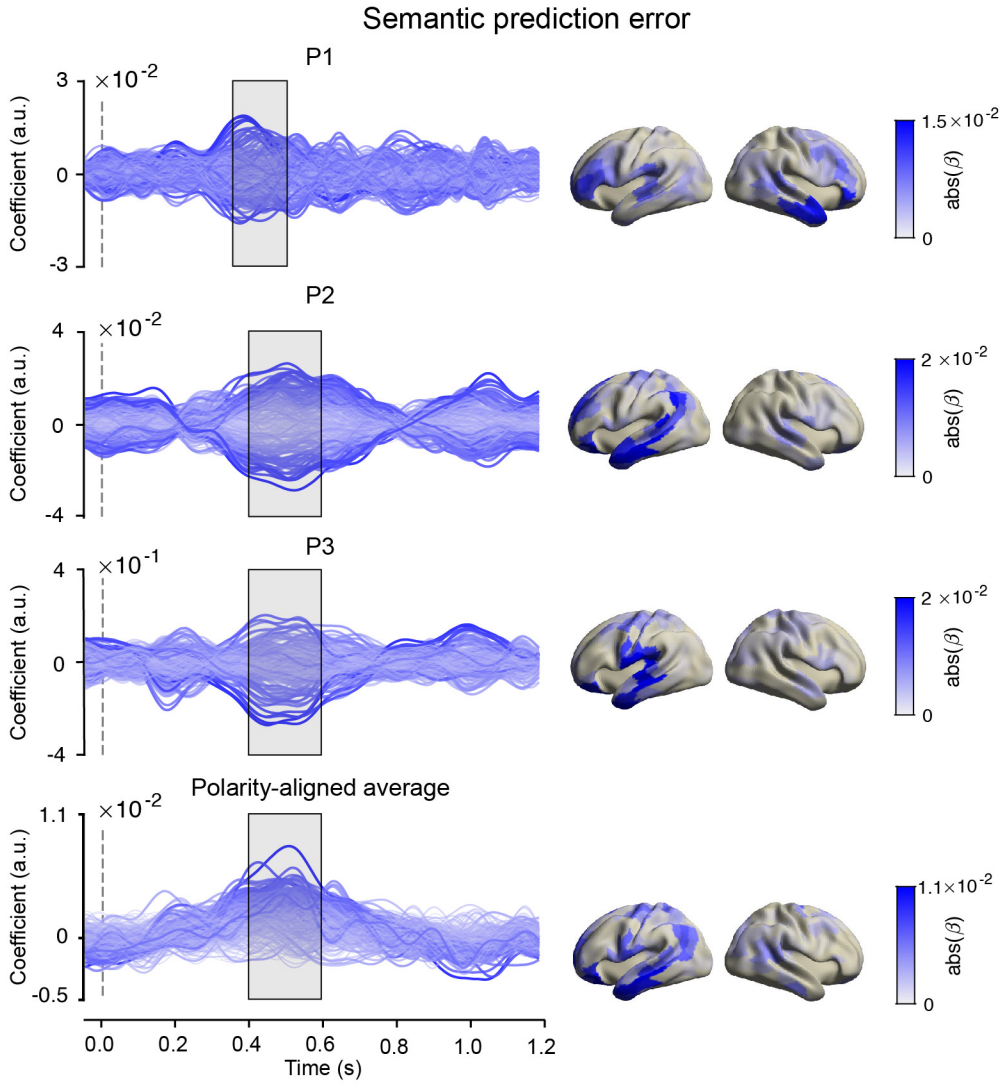

Fig. S14. Coefficients for semantic prediction error from the integrated model (Figure S7)
Left column: coefficients for each source for each individual in the MEG experiment, and the polarity-aligned average across participants. Right column: absolute value of the coefficients across the brain, averaged across the highlighted time-period.

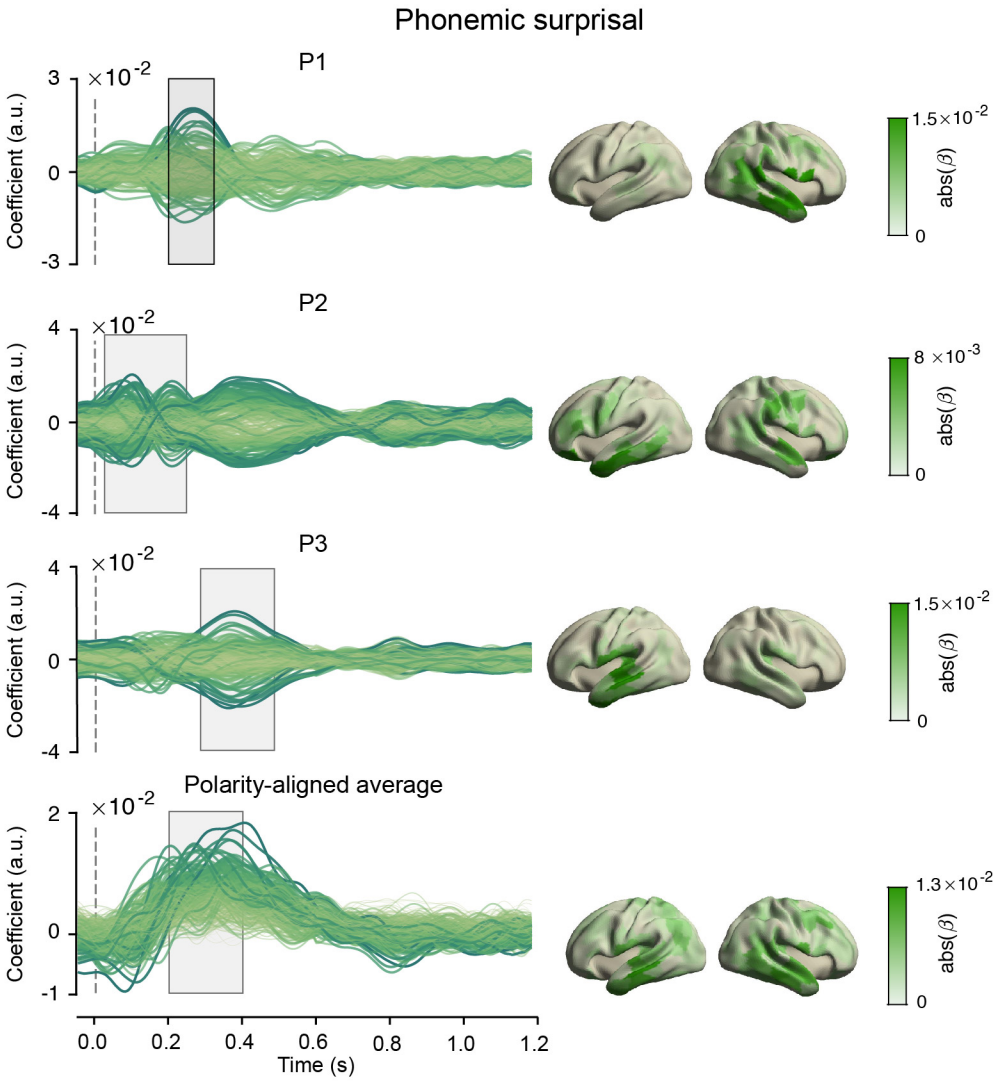

Fig. S15. Coefficients for phoneme surprisal from the integrated model (Figure S7)
Left column: coefficients for each source for each individual in the MEG experiment, and the polarity-aligned average across participants. Right column: absolute value of the coefficients across the brain, averaged across the highlighted time-period.

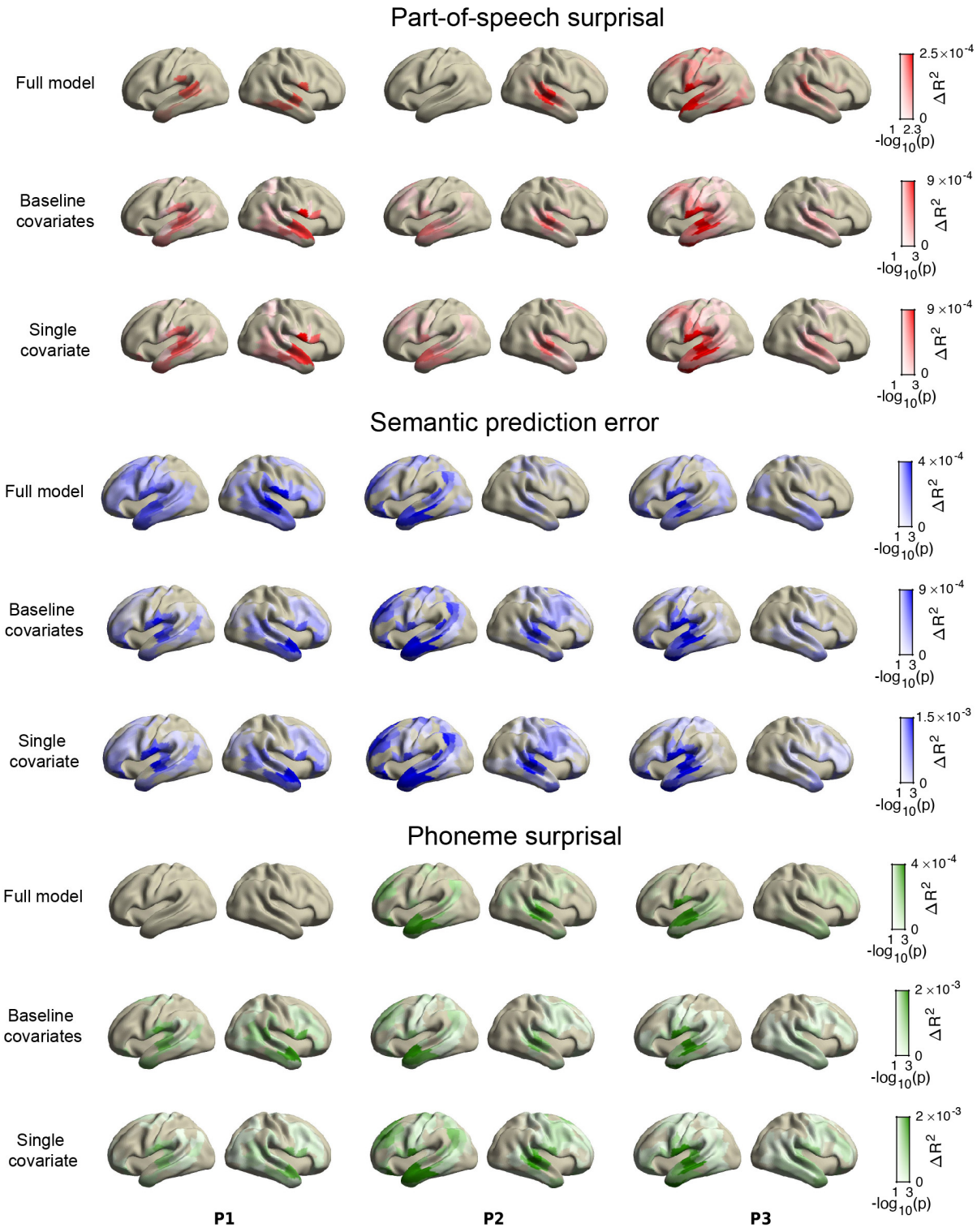

Fig. S16. DISSOCIABLE SPAITAL SIGNATURES ARE PRESERVED IN REDUCED REGRESSIONS.
Same as Figure S9, but comparing the full regression model with two reduced regression models (see Supplementary Note B for details). Colours indicate amount of additional variance explained by each regressor; opacity indicates the FWE-corrected statistical significance (across cross-validation folds). Overall, the spatial patterns are preserved in the reduced regressions, indicating they are not an artefact of our analyses (see Supplementary Note B).

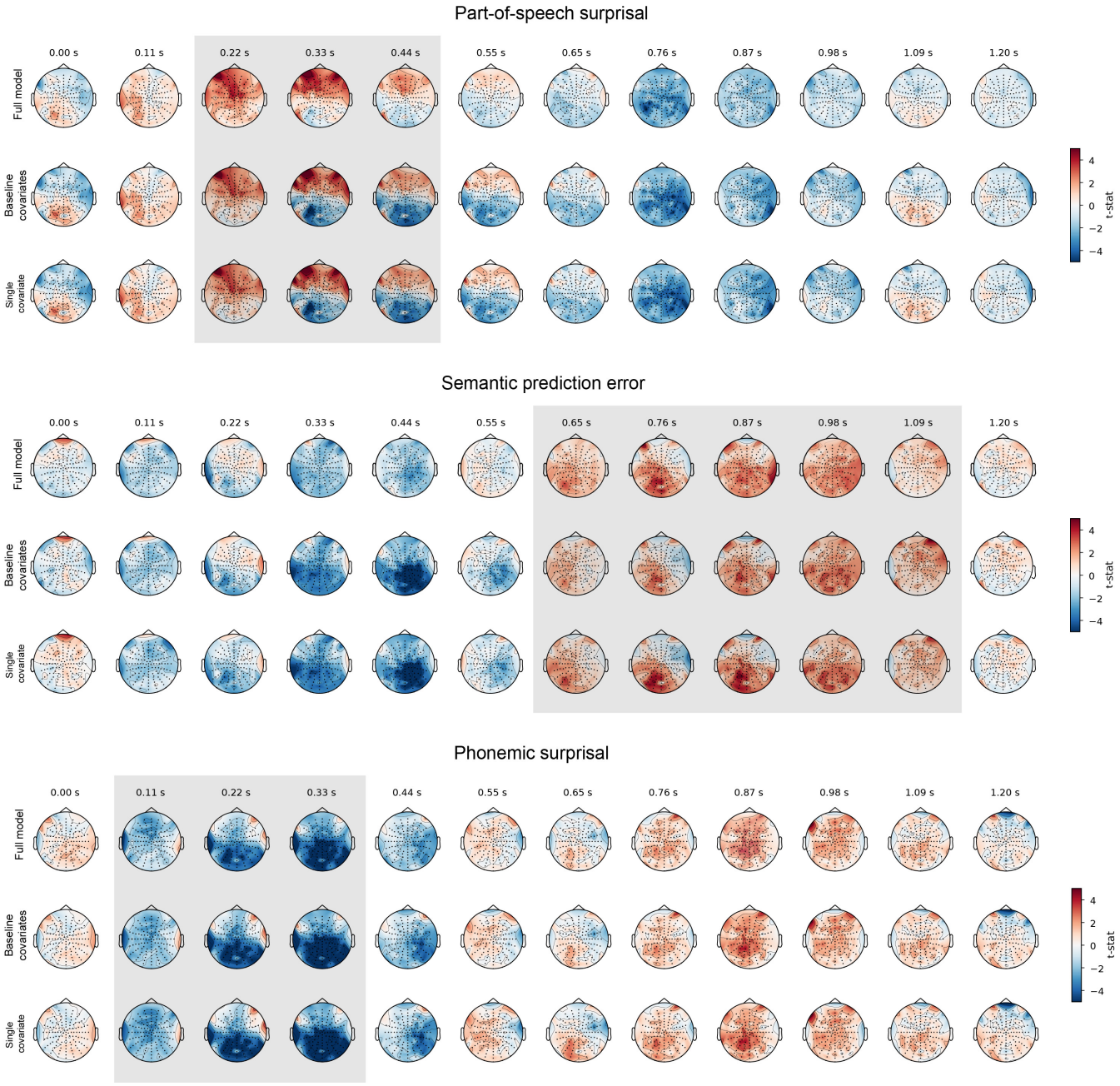

Fig. S17. DISSOCIABLE (SPATIO)TEMPORAL SIGNATURES ARE PRESERVED IN REDUCED REGRESSIONS.
Like Figure S10 but comparing the full regression model with two reduced regression models (see Supplementary Note B for details). Topographies show the the t-statistics of a one sample t-test on the coefficients across participants, over time. Shaded area indicates the time period where the main effect of interest occurs. Overall, the key signatures are preserved in the reduced regressions, indicating these are not an artefact of our analyses but reflect independent, dissociable effects (see Supplementary Note B for details).

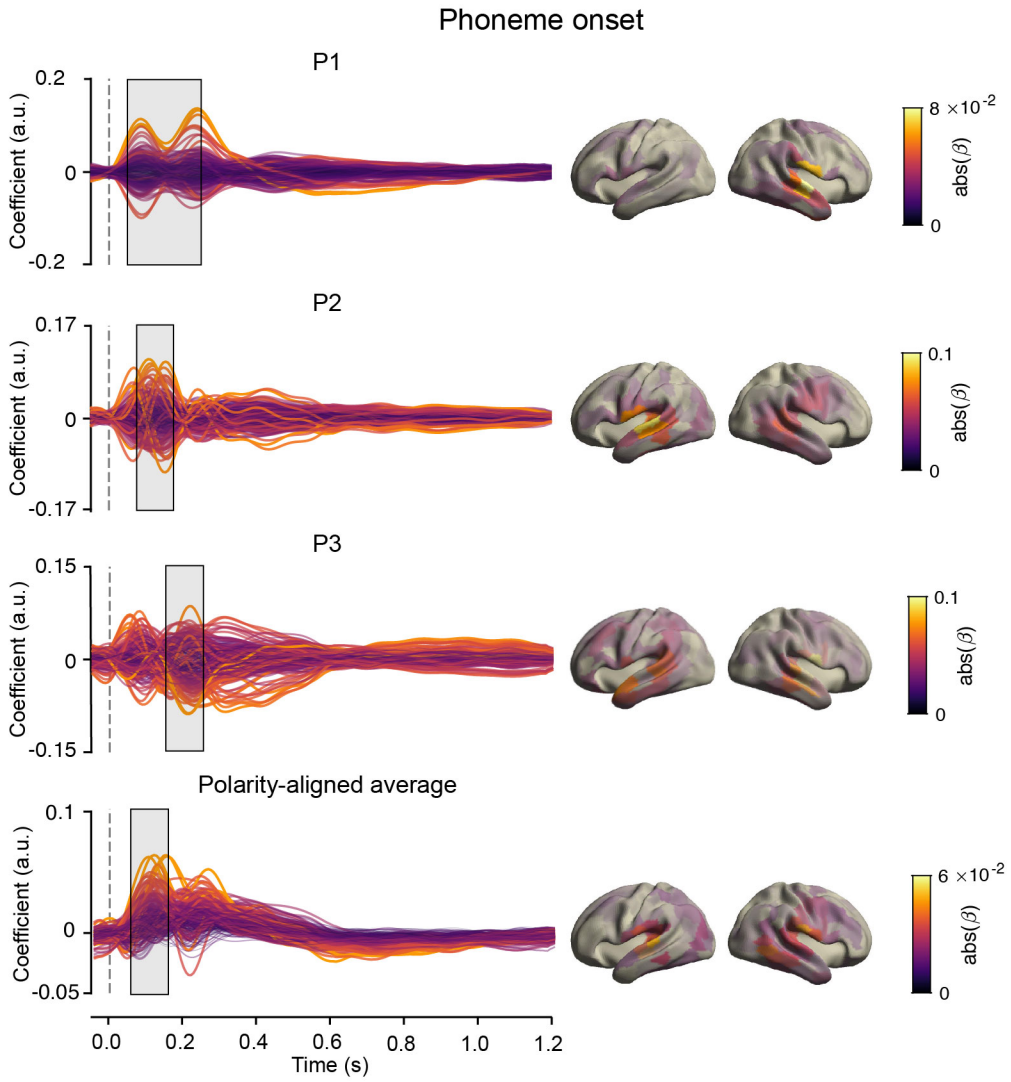

Fig. S18. Coefficients for phoneme onsets from integrated regression model (Figure S7)
Left column: coefficients for each source for each individual in the MEG experiment, and the polarity-aligned average across participants. Right column: absolute value of the coefficients across the brain, averaged across the highlighted time-period.

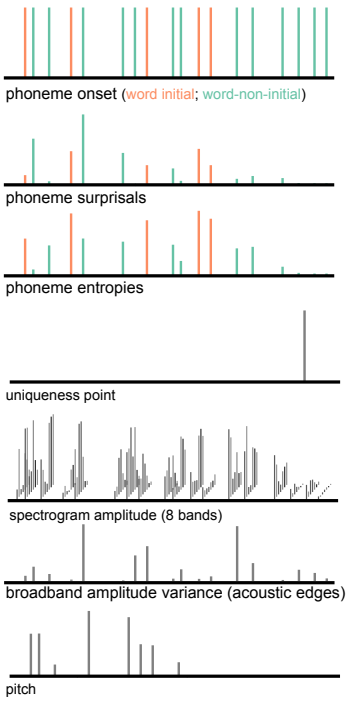

Fig. S19. REGRESSORS OF THE PHONEME MODEL. As indicated by the different colours, both the constants and covariates were modelled separately for word-initial and word-non-initial phonemes.

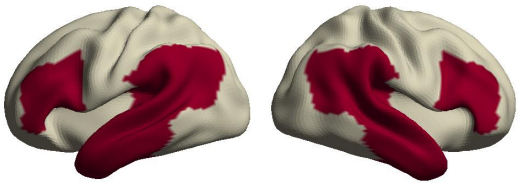

Fig. S20. LANGUAGE NETWORK DEFINITION The language network was defined as temporal cortex plus temporo-parietal junction, and IFG and dorsolateral prefrontal cortex; all bilaterally. In terms of Brodmann areas this corresponded to 20, 21, 22, 38, 39, 40, 41, 42, 44, 45, 46 and 47, amounting to a total of 100 out of 370 cortical parcels.

References

1. A Radford, et al., Language models are unsupervised multitask learners. *OpenAI Blog* **1**, 8 (2019).
2. A Vaswani, et al., Attention is all you need in *Advances in neural information processing systems*. pp. 5998–6008 (2017).
3. T Wolf, et al., HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771 [cs]* (2020) arXiv: 1910.03771.
4. A Holtzman, J Buys, L Du, M Forbes, Y Choi, The Curious Case of Neural Text Degeneration in *International Conference on Learning Representations (ICLR)*. (2019).
5. M Honnibal, I Montani, spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear* **7** (2017).
6. L Gwilliams, D Poeppel, A Marantz, T Linzen, Phonological (un)certainty weights lexical activation in *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*. (Association for Computational Linguistics, Salt Lake City, Utah), pp. 29–34 (2018).
7. C Brodbeck, LE Hong, JZ Simon, Rapid Transformation from Auditory to Linguistic Representations of Continuous Speech. *Curr. biology: CB* **28**, 3976–3983.e5 (2018).
8. NJ Smith, M Kutas, Regression-based estimation of ERP waveforms: I. The rERP framework. *Psychophysiology* **52**, 157–168 (2015).