# Effect of introducing context on the temporal dynamics of an ECoG based semantic encoding model

Nimra Nadeem

January 11, 2020

**Abstract**

The Hasson Lab in the Princeton Neuroscience Institute has developed a novel pipeline for the collection of high quality intracranial data for understanding neural representations of language. Particular, they have worked on a collecting and preprocessing a large dataset of high quality speech and Electrocorticography (ECoG) recordings of two patients in the NYU medical epilepsy unit. In this paper, I use this dataset to build a semantic encoding model by using contextualised word embeddings derived from a pre-trained Bidirectional Encoder Representations from Transformers (BERT) model. My aim is to (a) explore the effect of incorporating context on the overall performance of the encoding model, (b) find the temporal lag with the highest correlation for production and comprehension after incorporating context and (c) compare the performance of contextualized embeddings extracted from 4 different layers (BERT). The results of this study will improve our understanding of both computational and neurological semantic representations of language. Any improvements introduced to the semantic encoding model using ECoG data will (a) enhance our understanding of sensitivity to context in the human cerebral cortex and (b) yield significant improvements in the performance of brain-computer interfaces (BCIs) for patients with neurological impairment.

## 1 Introduction

Semantic encoding models of the brain help us better understand both computational and neurological representations of language. Previous studies have demonstrated the importance of such encoding models in the development of brain-computer interfaces (BCI) Leuthardt et al. (2006); Merel et al. (2015) and in increasing our understanding of neural representation of semantic meaning Huth et al. (2016). In the development of BCIs, previous work has also shown that the incorporation of Electrocorticographic (ECoG) data is particularly effective in enhancing the performance of the interfaces, including faster user training and communication rates. Leuthardt et al. (2006); Wilson et al. (2006) Improvements in the performance of BCIs could yield vital benefits for patients with neurological impairment. Leuthardt et al. (2006)

In this paper, I contribute to the work being done in the Hasson lab at the Princeton Neuroscience Institute, to build a language encoding model for the brain using large amounts of high quality Electrocorticography (ECoG) data, with the aim to explore the temporal dynamics of meaning formation in the brain. The lab has been building this encoding model using word-level embedding vectors derived from GloVe. My particular contribution is to incorporate context into this encoding model by using contextual word embeddings instead of independent word embedding (GloVe), derived from the hidden layers of Googleâs state of the art bidirectional language model, BERT.

In this paper, my aim is to (a) explore the effect of incorporating context on the overall performance of the encoding model, (b) find the temporal lag with the highest correlation for production and comprehension after incorporating context and (c) compare the performance of contextualized embeddings extracted from 4 different layers of the Bidirectional Encoder Representations from Transformers (BERT). My goal is to improve the current encoding model developed by the Hasson lab by discovering a higher correlation after incorporating contextual word embeddings derived from BERT. My secondary goal is to discover what effect this context has on the temporal lag during encoding of production and comprehension. I hypothesize that incorporating context would increase the negative lag for production and decrease the positive lag for comprehension, given that the non-contextualized highest correlation production lag is negative, i.e. before the word onset and positive, i.e. after the word onset for comprehension.

## 2 Problem background and related work

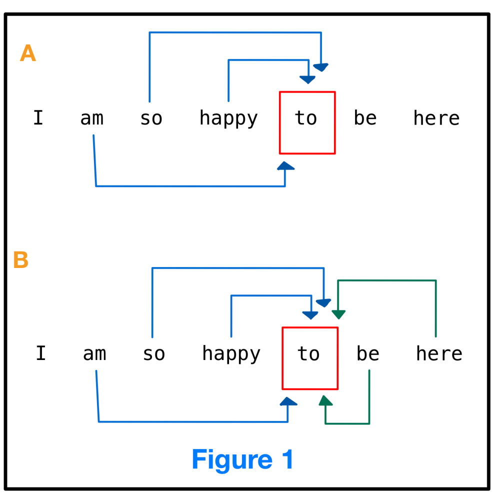
A lot of previous work has been done in developing semantic encoding models for the brain using word-vectors to represent the meaning of individual words. These studies have demonstrated the effectiveness of using word2vec or GloVe generated word vectors to represent semantic meaning as it is mapped in the human brain Huth et al. (2016); Jain & Huth (2018); Pereira et al. (2018) However, there are quite a few areas unexplored by previous work done in this area. Firstly, by using word-level embedding vectors, most of these studies ignore the effect of context on the semantics of a single word. de Heer et al. (2017); Huth et al. (2016); Pereira et al. (2018) Each word has one unique embedding, regardless of the context. While, as we know from everyday life, significant semantic differences occur between the same words in different contexts (example in footnotes) In fact, previous studies have shown that almost all regions of the human cerebral cortex have varying degrees of dependencies on the context of incoming information. Jain & Huth (2018); Wehbe et al. (2014) Secondly, previous language encoding studies do not account for the temporal dynamic of the formation of semantic representation in the brain. Intuitively, we know that we often think of a word before saying it, and sometimes it takes us a while to catch from someone’s spoken words what they actually mean. Thirdly, there is a lack of availability of large amounts of data to build such brain encoding models, so most previous studies have focused on limited amounts of data with only comprehension Jain & Huth (2018) or production of very limited range of vocabulary. The Hasson lab has been working on developing a novel pipeline to get access to large amounts of high quality data of ECoG recordings during natural speech production and comprehension. This big amount of data is invaluable in building better encoding models.

The Hasson lab has been working on using this large amount of high quality data to build language encoding models that explore the temporal dynamic of the formation of semantic representation in the brain. I joined the team with the idea to incorporate context using contextual embedding vectors derived from Googleâs state-of-the-art bidirectional language model, BERT, hoping to improve the current encoding model.

Two recent studies attempted to incorporate context into language encoding models. Jain & Huth (2018); Jat et al. (2019) One of these showed significant improvement in the performance of an fMRI-based language encoding model for comprehension of narrated text after incorporating context using contextual embeddings derived from a small, self-trained Long-Short Term Memory Language Model. Jain & Huth (2018) Another study showed that sentence level representations derived from BERT correlate strongly with MEG brain responses to reading syntactically and semantically simple sentences. Jat et al. (2019) These results indicate that incorporating context will likely improve the performance of the encoding model being developed in the Hasson Lab. The availability of large amount of high quality data for both natural speech production and comprehension, and the exploration of the temporal dynamic in this encoding model make this study unique from previous work on incorporating context into language encoding models.

## 3 Approach

I introduced a new theoretical approach to the encoding model being developed in the Hasson lab, that of incorporating context into the model to study its effects on the temporal lags and the overall model performance. In terms of the design, I incorporated context by extracting contextualized word vectors from the hidden layers of Googleâs pre-trained bidirectional language model, BERT.



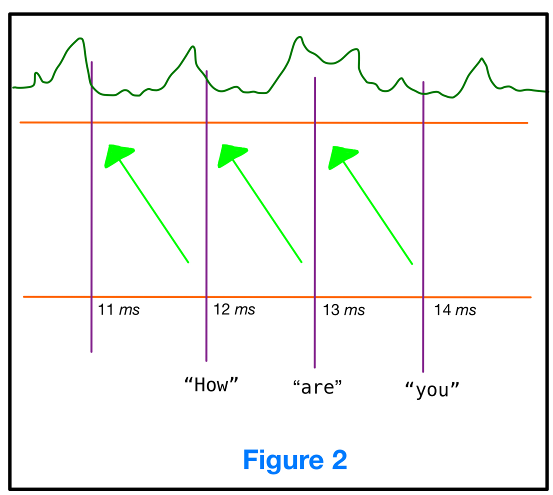




**Figure 1:** Left A: A unidirectional language model, where predicting that the next word in the sequence is 'to' is done only in the context of the words to the left. Left B: A bidirectional language model, where predicting the the current word in the sequence is 'to' is done considering the context of the the words on both sides. Right: Building a linear model with a temporal lag means using the embedding for the word said at $t$ms to predict the brain signal at $(t + \phi)$ms where $\phi$ is the value of some lag in the range $-2000$ms to $+2000$ms

The primary task of a language model is to predict the next word in a sequence of words, and they are subsequently used in a wide range of NLP tasks. Several previous studies have found that the representation of words in the hidden layers of these language models can be used as contextualized word embeddings. Jain & Huth (2018); Jat et al. (2019) This is because the way the language model learns to predict words in a sequence is by taking into account the previous words, and therefore at every step a language model needs to incorporate information about the words that have been seen so far. Earlier language models are unidirectional, which means that the current word is represented only in the context of the words to its left, as seen in Figure 1A. However, bidirectional language models have been shown to be much more powerful, because intuitively the meaning of a word in a sentence does not depend only on the words to its left (Figure 1B). BERT is an example of such a bidirectional language model.

BERT is composed of a stack of transformer encoders, with each encoder containing a self-attention layer and a feed-forward neural network layer. Devlin et al. (2018); Google (2019) The self-attention layer is most crucial in the incorporation of context, because its task is to learn which parts of the sentence to pay âattentionâ to, i.e. what word dependencies exist. For example, in the sentence, âthe girl was walking when a man bumped into herâ, the self-attention layer will learn that the âherâ at the end of the sentence refers to âthe girlâ at the beginning.

Each encoder layer in the BERT stack outputs a feature vector for each word in the sentence, and the output vectors from each layer serve as the input vectors for the next encoder layer. The feature vectors from any of these layers can be used as a contextualized embedding vector for the words because these feature vectors contain information about the word dependencies in the sentence discovered in the self-attention layer. Devlin et al. (2018); Google (2019)

The open source BERT has been pre-trained in a semi-supervised manner on a massive text corpus. It has been shown to perform extremely well on downstream NLP tasks after being âfine tunedâ on a small dataset for a given task. Though, in the case of our brain encoding model, the actual transcribed corpus was not large enough for me to use it for fine tuning the pre-trained BERT model. However, the original paper introducing BERT (cite) mentions that even without fine tuning, a pre-trained BERT model can be used to extract feature vectors for a given text which can subsequently be used as word embeddings incorporating context. Devlin et al. (2018); Google (2019); Jat et al. (2019)

My purpose is to study the effect of context on the temporal dynamics of semantic encoding in the brain. This idea is displayed more clearly in Fig 2. We try using a linear model to predict the brain signal from the contextualized word embedding at a range of different temporal lags relative to the onset of the word. We then look for the temporal lag with the highest Pearson correlation ($r$) value. This exploration of temporal lag was already being done in the Hasson lab with word-level non-contextual GloVe embeddings. My approach is to do the same analysis but with the BERT-derived contextualized word embeddings instead.

In the ideal case, incorporating context using BERT should increase the lag for production, because the contextualized word embedding should contain information about words well before their onset due to the âcontextâ. In the case of speech production, I would expect the lag to decrease, i.e. get closer to the onset of the word or

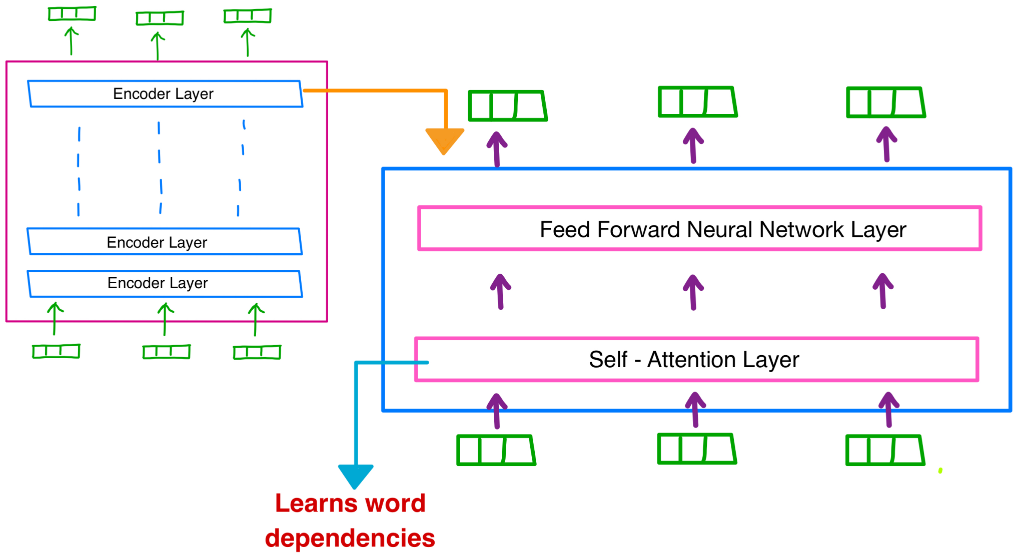



**Figure 2:** Diagram explaining the architecture of Google's Bidirectional Encoder Representations from Transformers (BERT) language model. BERT is composed of a stack of transformer encoders, with each encoder containing a self-attention layer and a feed-forward neural network layer. The self-attention layer learns word dependencies in the given sentence. The encoder layer outputs a feature vector corresponding to each word in the sentence. This feature vector contains information about the inter-word dependencies, i.e. context, of each word.

even shift to before the onset of the word because incorporating context in the linear model for comprehension should mean having greater information before word onset and therefore earlier understanding.

# 4 Implementation

The following data collection and pre-processing was done by the Hasson Lab to produce the data that I analysed. I did not contribute to this step of the process but it was essential in producing the high quality data I required for my analysis.

## 4.1 Data collection

Speech and intracranial electroencephalography data was recorded round the clock for around 3-6 days for two patients at the NYU Medical Epilepsy Unit. The brain signals were recorded from 100-200 electrode at a sampling rate of 512Hz. A total of 42 hours of speech were recorded in total across the patients with 120k comprehension words and 120k production words. Ariel Goldstein (2019)

### 4.2 Pre-processing

The speech recording was transcribed and a speaker identity was assigned to each part of the transcription to distinguish production from comprehension. The transcribed text was aligned with the brain signal recording at a precision of milliseconds. The data was split into separate conversations to allow for structured analysis. As a result, a pre-processed datum file was produced for each conversation, which contained the aligned transcript of the entire conversation. This was a text file, formatted as follows: each line had 5 items, the first was a word followed by the onset, offset, accuracy and speaker identity of the word. A few âbad wordâ symbols were used to indicate words in the data that were incomprehensible in the audio recording. The brain signal recordings were similarly split up into the separate conversations and preprocessed to remove noise. Ariel Goldstein (2019)

To build an encoding model, 300-dimensional GloVe embeddings were used to semantically represent the words. 160 different temporal lags for windows of 25s in the range of -2000ms to +2000ms relative to the onset of the word were used. For each lag value, a linear model predicting the brain signal from the word embedding vector was built. The Pearson correlation coefficient (r) was calculated to produce a plot of correlation against temporal lags. This plot revealed the temporal lag corresponding to the maximum correlation between the predicted and real signal. The above analysis had already revealed that semantic information during production could be encoded with maximal correlation up to a few seconds before the actual onset of the word. In the case of production, a general trend of maximal correlation post word onset was shown. Ariel Goldstein (2019)

### 4.3 My work - Introducing Context

My contribution was to run the above analysis of lags versus linear correlation using contextualized word embeddings. My hypothesis was that including information about the context would allow maximal correlation during production to be achieved even earlier than before, i.e. increased negative lag. In the case of comprehension, I hypothesized that incorporating context should bring the point of maximal correlation close to the word onset, i.e. decreased positive lag.

I began by running the existing encoding model on several different non-contextual embeddings available as part of open source projects, to assess whether the model performs differently with different non-contextual single word-level embeddings. I tried 4 different sets of embeddings, GloVe, one-hot, fastText, and another open source embedding trained on the Wikipedia corpus. I did not discover any significant variation in the maximum correlation value or the temporal lag corresponding to the maximal correlation.

To derive contextualized word embeddings, I used a pre-trained uncased BERT-Base Uncased model with 12 encoder layers, 768 hidden layers, 12-heads and 110M parameters. BERT is the current state-of-the-art language model for natural language processing, with significant improvements from past language models such as Transformer and ELMo. (Devlin et al. (2018); Google (2019); Jat et al. (2019)) The first step was to prepare the raw transcribed text which BERT would accept as its input. From the earlier pre-processing in the lab, I had the formatted datum files for each conversation. I wrote a python script to parse this datum file into another file containing only the comprehensible words in the form of sentences. I defined the end of a sentence to be when the speaker was switched.

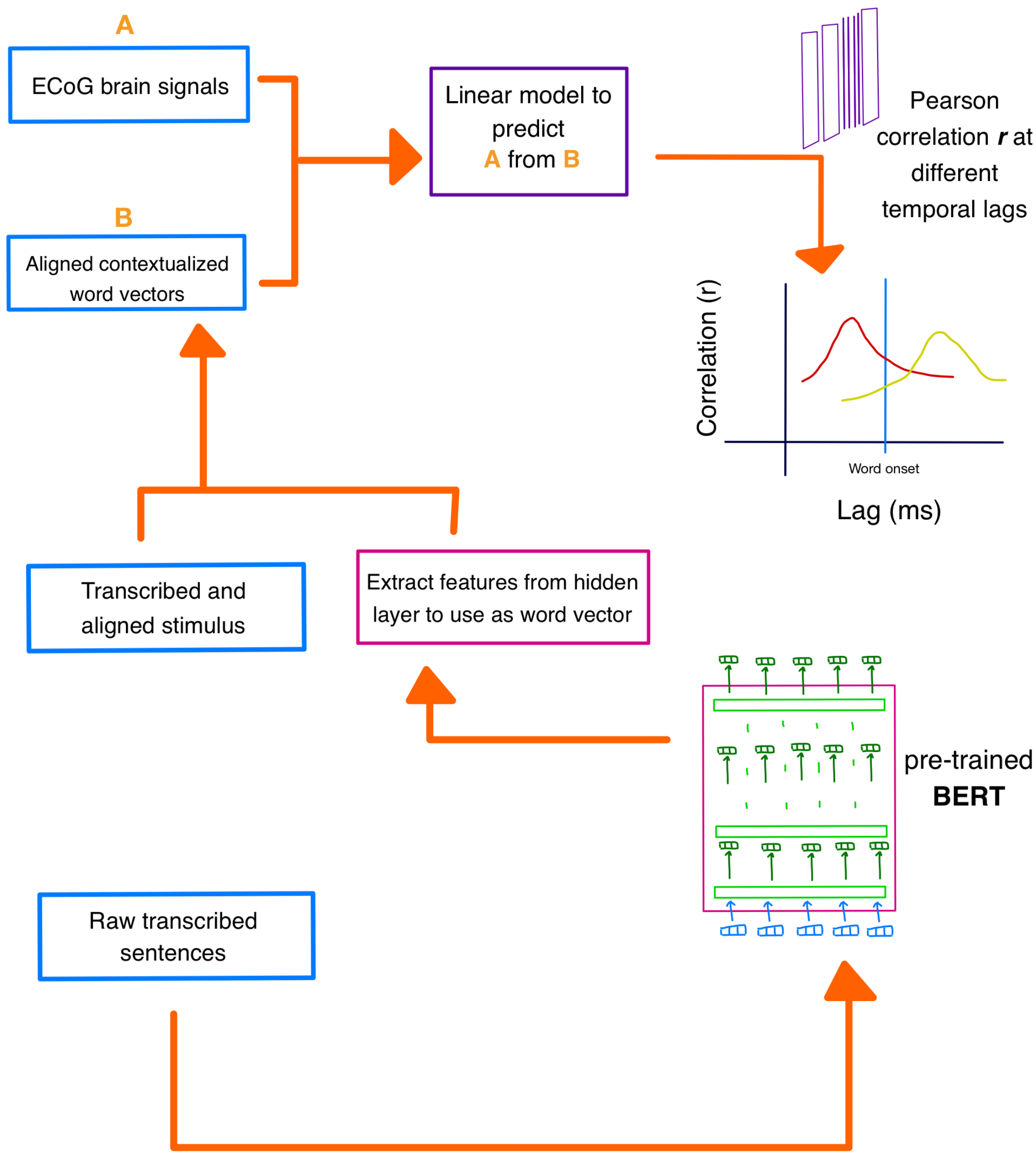



**Figure 3:** Flowchart to describe the work flow of my implementation

167 I used a Keras environment on a remote GPU to feed this sentence file as input to the pre-trained BERT model, and ran the extract features python script available on the open source BERT GitHub repository (Google (2019)) for the last and third last layer. The script produced a json file as its output which contained information about the feature vectors from each layer. I wrote a python script to parse this json file and create a word embedding file from one layer at a time.

The first problem in using these embeddings for encoding the datum was that BERT creates token-based embeddings. This means that it splits up the input sentence into bite-size chunks it can recognize. Each of these chunks is called a token. For example, every contraction like "don't" is split up into three tokens, "*don*", "'", and "*t*". BERT outputs an embedding for each token, which means that we have 3 embeddings for one word in the case of "*don't*".

To deal with this issue, I wrote a python script that took the tokenized embeddings as input and gave single word embeddings as output. This script used an array of indices that mapped each original word in the input sentence to an index in the sequence of tokenized embeddings, indicating where the tokens for the original word start from. (Google (2019)) This meant that for each word that had multiple embeddings due to BERT's tokenization, I used the embedding for the first token to represent the entire word, and discarded the embeddings for the remaining tokens. So in the example of "don't" given earlier, I used the embedding for "don" and discarded the embeddings for "'" and "t". This is one possible way of dealing with tokenization. An equally valid way would have been to take the average of the tokenized embeddings and use the result as the embedding for the entire word.

Also as part of the tokenization, BERT appends a $[CLS]$ token to the start and a $[SEP]$ token to the end of every sentence. In the same script that I dealt with extra tokenized embeddings, I also removed the embeddings for the $[CLS]$ and $[SEP]$ tokens.

Because I had removed the 'bad words' from the datum when generating the sentence file which I used as input to BERT, I had to create a copy of the original datum file with the lines for bad words removed.

At this point, theoretically the datum and embedding files should have been aligned. However, I discovered that during the pre-processing of the raw data in the lab, certain words had been concatenated in the datum file into one line, due to imprecision in the alignment. Certain lines had two or three words all together, followed by the onset, offset, accuracy and speaker fields. This concatenation meant that the number of embeddings was greater than the number of data points, because the embeddings were generated for each word in the raw transcribed text separately. There were several different approaches that I considered to fixing this. Considering the example of a datum line containing the concatenated set of words 'you want to' on a single line, I could:

1. Similar to the way I resolved the tokenization problem, keep the embedding for the first word in the concatenated words, and discard the embeddings corresponding to the remaining words in the set. In the above example, I would keep the embedding for 'you' and use it for that single data point, while discarding the embeddings for 'want' and 'to'.

2. I could average the embeddings of 'you', 'want' and 'to' and use the resultant vector as the embedding for 'you want to'.

3. I could regenerate a copy of the datum conversation file with all the concatenated words on separate lines with the same onset/offset/accuracy/speaker values repeated for each of the separated words.

4. Or I could simply discard the lines with the concatenated words and their corresponding embeddings, i.e. not use those data points at all.

Since the proportion of concatenated words was really small, I chose to go with the last option. I wrote a script to discard the datum lines which contained concatenated words and also remove the corresponding embeddings in the embeddings file.

At this point most of the datum files were aligned with the embedding files. However, for seven of the conversations, the alignment between the datum words and the embeddings was still skewed. Upon inspection of the pre-processed files, I observed that during the extraction of features using BERT, there were some parts of the original datum that did not have an output in the embedding file. There were random snippets in the middle of these seven conversations which were absent from the embedding file. The reason behind this remains unclear to me, and due to time constraints I did not manage to fix the alignment for these conversations. I discarded these 'bad' conversations and ran my encoding analysis with lags for the remaining conversations that now had perfectly aligned datum and embedding files.

I ran the encoding analysis for 14 electrodes that had been identified to have clean data by the previous Hasson lab members. The following technique for building and evaluating the performance of an encoding model at varying lags had already been developed in the lab; I adapted the same technique for my analysis. For each electrode, I used a linear model to predict the brain signal from the word embedding vector, at different temporal lags relative to the onset of the word. The temporal lag values ranged from negative 2000ms to positive 2000ms, and occurred at intervals of every 25ms. For the linear model at each temporal lag, I computed the Pearson correlation value $r$. Finally I generated a plot of the correlation values against each temporal lag. I ran the same analysis on these âgoodâ conversation using the GloVe embedding that had been used by the Hasson Lab before.

# 5 Results

Figure 4A shows the correlation vs lag plot for one of the electrodes that gave a significant correlation for GloVe embeddings. Figure 4B shows the correlation vs lag plot for the same electrode, but with using the contextualized word embeddings extracted from BERT. It appears that contrary to my hypothesis, using contextualized word embeddings worsened the performance of the encoding model. The signal present using GloVe was lost when shifting to contextual embeddings. Because of the lack of high correlation values in the results of the contextual embeddings, the temporal lag values corresponding to the maximum correlation does not reveal any significant information about the actual temporal dynamics of neural representation of semantic context. Furthermore, the results from the two different layers of BERT used to extract contextual embeddings were equally inconclusive.

There could be several reasons behind the inconclusive results. The original datum files for each conversation from the pre-processing step in the lab were in fact âtrimmedâ and did not contain the full content of the original conversation. I believe this was because certain parts of the brain signal recording and audio recording were not clean enough to be used as part of the data. While this trimming of the content at several different points in each of the conversations did not have any significant effect on the linear models built using independent word embeddings (GloVe), they did affect my approach. Discarding part of the speech content meant losing part of the context, and therefore introducing error into the contextualized word embeddings that were extracted using BERT. In addition to this problem, there is another plausible source of error. I extracted contextualized word

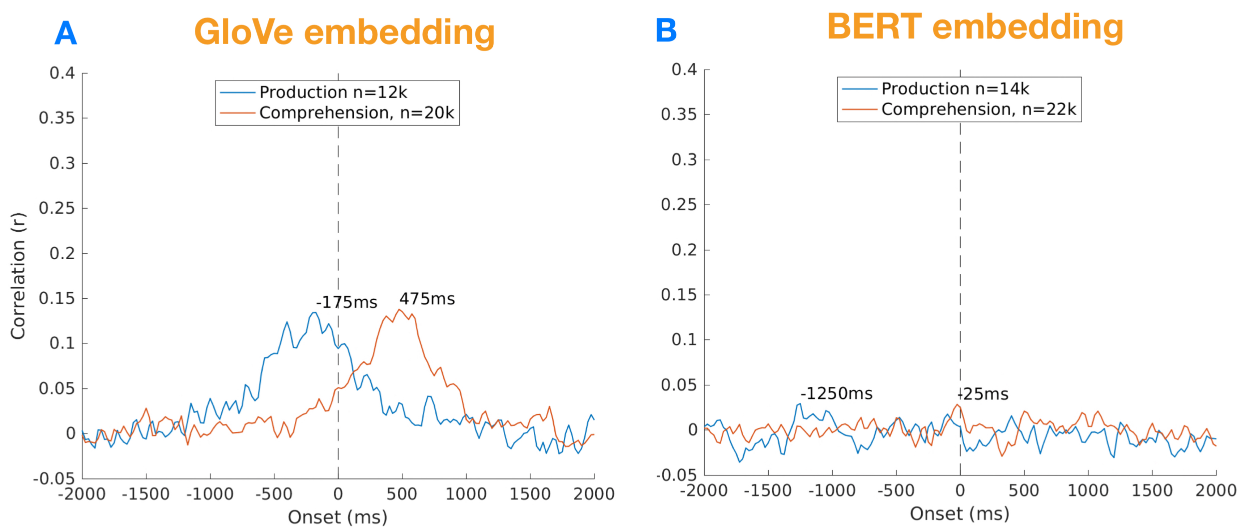



**Figure 4:** Left: Pearson Correlation *r* v. Temporal Lags *ms* plot of linear encoding models built using GloVe based independent word embeddings. This result had already been achieved by the Hasson Lab before I joined the team. A maximum correlation value of 0.15 is achieved at 175ms before the word onset during production and 475ms after the word onset during comprehension. Right: Pearson Correlation *r* v. Temporal Lags *ms* plot of linear encoding models built using BERT based contextual word embeddings. The plot shows an overall drop in the performance of linear encoding models that use contextual embeddings. Due to the low correlation values, the 'maximal correlation temporal lag' does not appear to be very distinct from correlation values at other lags. Therefore the results remain inconclusive on the question of contextual effect on temporal dynamics of semantic representation

embeddings using the pre-trained BERT model. The original paper introducing BERT recommends finetuning the pre-trained model for downstream NLP tasks. (BERT paper) As I mentioned earlier, the data corpus was not large enough to be used for fine-tuning. However, for the purposes of such encoding models, BERT could be fine-tuned on some other large corpus of conversational speech. Since BERT has been pre-trained on regular English text and standard grammar, which can differ significantly from conversational English, fine-tuning might in fact be an essential step in using BERT for this particular task. A data set consisting of dialogues from screenplays, or transcriptions of YouTube interview could be useful for this purpose.

# 6 Conclusion

My results so far show that contrary to my initial hypothesis, introducing context into word embeddings decreases rather than increases the maximum correlation achieved in the linear encoding models. Due to the extremely low correlation values, my results do not reveal anything conclusive about my hypothesis that introducing context would increase negative lag during production and decrease positive lag during comprehension. I was also unable to compare the performance of the encoding model based on word vectors extracted from several different layers of BERT. A previous study showed a variation in performance of the encoding model depending on which layer the contextualized word embeddings were extracted from. Jain & Huth (2018) The study also mapped which areas of the brain were more sensitive to context than other areas. Jain & Huth (2018) I had hoped to compare my results with these previous findings to see whether the sensitivity of specific brain areas to context could be confirmed in my results.

# 7 Future Work

The first step in taking this study forward would be to get the full transcribed text of the conversations to extract contextual embeddings from and to fine-tune BERT on conversational English data. Another way this study could be enhanced by trying one of the alternative options when dealing with BERT's tokenization, for example, by using the average of the token vectors instead of just the first tokenâs vector for a single word. On a higher level, this paper attempts to incorporate only single-sentence level context. This means that a word's embedding is informed only by the content of the current speaker's speech. A step forward from this would be to incorporate context of the previous speaker as well, i.e. incorporate context from more than one sentence. This analysis could result in fascinating findings, not just in terms of improving the encoding model but also revealing the neurological dependency of our speech on that of others communicating with us. Finally, this paper aimed to show the effect of incorporating context in language encoding models. A similar approach to incorporating context in decoding models would be a useful extension of this project. Such work could yield significant improvements in decoding of semantic meaning from neurological data, which would allow for vital improvements in brain computer interface (BCI) development.

# 8 Acknowledgements

My sincerest gratitude to Professor Uri Hasson for introducing me to his incredibly ground-breaking work, allowing me to contribute to it and advising me through the process. Thank you to Ariel Goldstein for his mentorship, for his efforts in familiarizing me with this completely new domain, for helping me find my topic, for always being available for feedback and for exposing me to enormous scope for intellectual exploration in the work being done in the Hasson Lab. Thank you to Eric Ham for helping me get familiar with the Keras environment. Thank you to Bobbi Aubrey for her tireless efforts in helping me adapt the existing encoding scripts for my use, familiarizing me with the dataset and answering my countless questions. Thank you so very much to Zaid Zada for getting me started with BERT, for walking me through every step, and for helping debug every time I got stuck. Thank you so much to all other members of the Hasson lab for their tireless efforts in collecting and preprocessing the data and building the existing encoding model. And finally, thank you to Dean Anne Caswell-Klein and the COS Independent Work Coordinators for their patience and support in my time of crisis.

# 9 References


Ariel Goldstein, U. H., 2019. Temporal dynamics of meaning - unpublished, *Unpublished*, **0**(0), 0.

de Heer, W. A., Huth, A. G., Griffiths, T. L., Gallant, J. L., & Theunissen, F. E., 2017. The hierarchical cortical organization of human speech processing, *Journal of Neuroscience*, **37**(27), 6539–6557.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805*.

Google, R., 2019. Tensorflow code and pre-trained models for bert - github, *GitHub*, **0**(0), 0.

Huth, A. G., De Heer, W. A., Griffiths, T. L., Theunissen, F. E., & Gallant, J. L., 2016. Natural speech reveals the semantic maps that tile human cerebral cortex, *Nature*, **532**(7600), 453.

Jain, S. & Huth, A., 2018. Incorporating context into language encoding models for fmri, in *Advances in Neural Information Processing Systems*, pp. 6628–6637.

Jat, S., Tang, H., Talukdar, P., & Mitchell, T., 2019. Relating simple sentence representations in deep neural networks and the brain, *arXiv preprint arXiv:1906.11861*.

Leuthardt, E. C., Miller, K. J., Schalk, G., Rao, R. P., & Ojemann, J. G., 2006. Electrocorticography-based brain computer interface-the seattle experience, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **14**(2), 194–198.

Merel, J., Pianto, D. M., Cunningham, J. P., & Paninski, L., 2015. Encoder-decoder optimization for brain-computer interfaces, *PLoS computational biology*, **11**(6), e1004288.

Pereira, F., Lou, B., Pritchett, B., Ritter, S., Gershman, S. J., Kanwisher, N., Botvinick, M., & Fedorenko, E., 2018. Toward a universal decoder of linguistic meaning from brain activation, *Nature communications*, **9**(1), 963.

Wehbe, L., Murphy, B., Talukdar, P., Fyshe, A., Ramdas, A., & Mitchell, T., 2014. Simultaneously uncovering the patterns of brain regions involved in different story reading subprocesses, *PloS one*, **9**(11), e112575.

Wilson, J. A., Felton, E. A., Garell, P. C., Schalk, G., & Williams, J. C., 2006. Ecog factors underlying multimodal control of a brain-computer interface, *IEEE transactions on neural systems and rehabilitation engineering*, **14**(2), 246–250.