Representation and Pre-Activation of Lexical-Semantic Knowledge in Neural Language Models

Steven Derby, Paul Miller, Barry Devereux
Institute of Electronics, Communication and Information Technology ECIT
Queen's University Belfast, United Kingdom
{sderby02, p.miller, b.devereux}@qub.ac.uk

Abstract

Neural network language models have the ability to capture the contextualised meanings of words in a sentence by dynamically evolving a representation of the linguistic input in a manner evocative of human language comprehension. While researchers have been able to analyse whether key linguistic regularities are adequately characterised by these evolving representations, determining whether they activate lexico-semantic knowledge similarly to humans remains challenging. In this paper, we perform a systematic analysis of how closely the intermediate layers from LSTM and transformer language models correspond to human semantic knowledge. Furthermore, in order to make more meaningful comparisons with theories of human language comprehension in psycholinguistics, we focus on two key stages where the meaning of a particular target word may arise: immediately before the word's presentation to the model (comparable to *forward inferencing*), and immediately after the word token has been input into the network. Our results indicate that the transformer models are better at capturing semantic knowledge relating to lexical concepts, both during word prediction and when retention is required.

1 Introduction

A wide variety of Natural Language Processing (NLP) tasks have been improved dramatically by the introduction of LSTM (Hochreiter and Schmidhuber, 1997) and transformer-based (Vaswani et al., 2017) neural language models, which can encode the meanings of sentences in such a way that facilitates a range of language tasks (Bengio et al., 2003; Peters et al., 2018; Radford et al., 2018; Dai et al., 2019). Furthermore, both recurrent and transformer networks have been shown to capture a broad range of semantic phenomena and syntactic structure (Dyer et al., 2016; Linzen et al., 2016; Bernardy and Lappin, 2017; Gulordava et al., 2018; Marvin and Linzen, 2018; Lin et al., 2019; Liu et al., 2019; Hewitt and Manning, 2019; Tenney et al., 2019a). Although such models clearly learn aspects of lexical semantics, it remains unclear whether and how these networks capture semantic features associated with conceptual meaning. Some work has demonstrated that word embeddings do reflect conceptual knowledge captured by property norming studies (Rubinstein et al., 2015; Collell and Moens, 2016; Lucy and Gauthier, 2017; Derby et al., 2018), in which human participants produce verbalisable properties for concepts, such as *is green* or *is an amphibian* for concepts such as FROG (McRae et al., 2005; Devereux et al., 2014). Such features correspond to *stereotypic tacit assumptions* (Prince, 1978); common-sense knowledge we have about the real world. There is some evidence that language models implicitly encode such knowledge (Da and Kusai, 2019; Weir et al., 2020); however, coverage of different types of knowledge may be inconsistent, with evidence to suggest that these models fail to capture some types of semantic knowledge such as visual perceptual information (Sommerauer and Fokkens, 2018; Sommerauer, 2020), as well as questions about the completeness of such empirical studies (Fagarasan et al., 2015; Bulat et al., 2016; Silberer, 2017; Derby et al., 2019). In general, there has been only limited work that attempts to investigate whether these neural language models activate lexico-semantic knowledge similarly to humans, further restricted by the fact that such knowledge probing is only performed on latent representations that have received the target concept, ignoring theories of language comprehension and acquisition that emphasise the importance of prediction (Graesser et al., 1994; Dell and Chang, 2014; Kuperberg and Jaeger, 2016).

In this paper, we contribute to the analysis of neural language models by evaluating latent semantic knowledge present in the activation patterns extracted from their intermediate layers. By performing a layer-by-layer analysis, we can uncover

Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, pages 211–221
Online Event, June 10, 2021. ©2021 Association for Computational Linguistics

how the network composes such meaning as the information propagates through the network, eventually emerging as a rich representation of semantic features that facilitates conditional next word prediction, which is directly dependent on the past knowledge. We perform our layer probing analysis at two temporal modalities. That is, we investigate the hidden layer activations of the NNLMs both **before** the concept word occurs (which facilitates next word prediction), and **after** the concept word has been explicitly given to the model. In this way, we determine how richly these latent representations capture real-world perceptual and encyclopaedic knowledge commonly associated with human conceptual meaning.

2 Related Work

The recent popularity of interpretability in NLP has resulted in strong progress on understanding both recurrent (Alishahi et al., 2019) and transformer-based networks (Rogers et al., 2020). A number of these studies rely on probing techniques, where supervised models are trained to predict specific linguistic phenomena from model activations (Adi et al., 2016; Wallace et al., 2019; Tenney et al., 2019b; Hewitt and Liang, 2019).

There exists some work that analyses semantic knowledge in such networks, though to date this has been more limited than investigations of syntax. Koppula et al. (2018) focused on the recurrent layers of LSTM and GRU networks and attempted to interpret their semantic content by using a set of decoders to predict the previous network inputs. Ettinger (2020) devised a set of psycholinguistic diagnostic tasks to evaluate language understanding in BERT, demonstrating that some phenomena such as semantic role labelling and event knowledge are well-inferred, though others such as negation are less so. Similar to our work, Ethayarajh (2019) mined sentences with words in context to demonstrate that context representations are highly anisotropic, while Bommasani et al. (2020) built static word embeddings from contextual representations using pooling methods, analysing their performance on semantic similarity benchmarks.

Language models have also been successfully employed for predicting activation patterns in the brain during human language comprehension (Jain and Huth, 2018; Toneva and Wehbe, 2019). Such work is particularly relevant from the perspective of predictive coding theories of human language comprehension (Kuperberg and Jaeger, 2016), which posits that high-level representations of an unfolding utterance facilitate active prediction of subsequent lexical content in the sentence. Neurolinguistic studies provide evidence that such predictions can be of wordform identity (DeLong et al., 2005), or of the semantic features that are expected for the upcoming word (for example, whether the upcoming word is animate or not; Wang et al., 2020).

3 Neural Language Models

Due to the compatibility issues, we limit our investigation to left-to-right language models that are trained to perform conditional next word prediction, as other SOTA models such as **Bert** (Devlin et al., 2018) fail to capture the desired criterion that facilities similar mechanisms in language comprehension. For the LSTM-based network, we make use of a very large-scale and influential neural language model developed by Jozefowicz et al. (Jozefowicz et al., 2016), which we refer to as **JLM**[1]. The model's architecture consists of character-level embeddings with CNNs, followed by a two-layer LSTM with projection layers to reduce dimensionality and a final linear layer with softmax activation. The vocabulary of the output layer consists of 800000 words, and the model is trained using the One Billion Word corpus (Chelba et al., 2013). For the transformer-based model, we make use of the **GPT-2** (345M) model (Radford et al., 2019), which consists of 24 multi-head attention layers.

3.1 De-Contextualising Representations

There are several problems that emerge when looking to compare concrete conceptual representations of meaning with these neural layer activations. The first is that representations from these latent layers are highly contextualised, which may make it difficult to recover semantic information about a particular concept. The second problem is that recovering a pre-target representation is challenging since it requires contextual information to be supplied to the network before the target word occurs. For our work, we follow a similar approach to Bommasani et al. (2020), and mine a number of sentences from a corpus of text where each target word occurs and then extract representations from each layer of the network **before** and **after** the words are presented. For this, we choose a

[1] https://github.com/tensorflow/models/tree/archive/research/lm_1b

predefined set of target words which are based on the overlap of words in the **JLM** vocabulary and several intrinsic evaluation benchmarks which are employed in the analyses below. We then sample the training corpus for up to 500 sentences for each target, selecting sentences in which the target word occurred in any position except the start of the sentence. By analysing how the representations perform on the semantic benchmarks, we can infer how these language models compose meaning over the layers of the network.

3.2 Feature Pooling

To construct these decontextualized representations, we first compute a hidden state from each of our sentences, and then aggregate them into a single static vector, both at the position of the target word and immediately before. More formally, for each word $w \in W$, where W is our lexicon, we retrieve a set of K sentences $\{S_1, S_2, \dots S_k\}$ from the corpus with corresponding timepoints $T = \{t_1, t_2 \dots t_k\}$ denoting the position of the word w in the sentences, such that $S_i[t_i] = w$ for $1 \leq i \leq K$. Let f_L be the function that maps each sentence fragment to a contextual representation from the model f for each layer L in the network. We construct our word-level representation *before* and *after* the word w occurs at layer L as follows:

$$\text{before}[w]_L = \frac{1}{K}\sum_{i=1}^{K} f_L(S)[t_{i-1}]$$

$$\text{after}[w]_L = \frac{1}{K}\sum_{i=1}^{K} f_L(S)[t_i]$$

This gives us two sets of word embedding vectors for each layer in each network, one set built from activations immediately before the target words and one built from activations immediately after the target words. Since the context differs depending on the sentence, the aggregation performed in the calculations above should preserve only the information associated with the target word. As the model is tasked with predicting the word w, the vectors from the **before** timestep should contain some semantic information relevant to the target word, even if the word has not been explicitly given to the network.

In the case of GPT-2, input tokens are determined using byte pair encodings, and a given word will correspond to several input units in this encoding. For target words that consist of a number of smaller units that combine into the word, we average the representation over all these positions for the **after** representations. For the **before** representations, we take the token immediately before the target word. In the results that follow, we refer to the two sets of embedding vectors for language model M and layer L using the naming convention ***M*[*L*]-before** and ***M*[*L*]-after**. For example, for GPT-2, the word vectors for the fifth multi-head attention layer just before the target word is presented to the network would be **GPT2[5]-before**.

Note that while LSTMs accumulate a representation of the unfolding utterance at each timestep, this is not entirely true for transformers, which directly combine information from all previous words in the sequence at every layer of the network, guided by attention. In our work, we only care about how the semantic information of the network evolves when it must predict the target word and immediately after.

4 Evaluation Tasks

For our empirical analysis, we first analyse these layers on classic intrinsic benchmarks that determine their ability to explain human semantic judgments scores on word association, to first determine how well these networks capture the semantic content of the word. We then probe these layers to determine whether they capture a rich set of semantic features related to upcoming concepts and whether such representations are retained by the network for functional use on the prediction task.

4.1 Semantic Similarity Benchmarks

Semantic similarity benchmarks, where a set of word pairs are scored by human annotators based on how similar they are, can be used to determine how correlated word pair distances from a set of embedding vectors are with human judgements of similarity for the same words. For the embedding vectors (from each network and network layer), cosine similarity can be used to determine how similar the word vectors are, and these cosine similarities can then be compared with the human judgements using Spearman correlation. Of course, the notion of similarity that informs human judgements is highly dependent on a number of factors such as context, the stimulus set of word pairs, and the instructions given to the human raters (Batchkarov et al., 2016). For this reason, we make use of a number of benchmarks which can be partitioned into two types of relationships, known as *semantic similarity* and *semantic relatedness*. For semantic relatedness, we use **WordSim353-rel** (Agirre et al., 2009) and **MEN** (Bruni et al., 2012), where a high score between word pairs indicates a greater chance of occurring in the same sentence with some syntactic relation (for example “coffee" and “cup"). For semantic similarity, we use **WordSim353-sim** (Agirre et al., 2009) and **SimLex999** (Hill et al., 2015), where a high score between word pairs indicates a high overlap in semantic attributes or replaceability in a sentence (for example “coffee" and “tea"). Though it does not clearly fall into either the similarity or relatedness categories, we also include the original version of the WordSim judgements, **WordSim353** (Finkelstein et al., 2001). Evaluations were performed using the Vecto-ai python package (Rogers et al., 2018).

4.2 Neural Activation Similarity

As an extension to these results, we also evaluate how reliable the vector representations from each layer of the networks are in terms of their ability to predict brain imaging data gathered from participants viewing a set of concept words. In this analysis, we use BrainBench (Xu et al., 2016)[2], a semantic evaluation platform that includes fMRI and MEG neuroimaging data from humans for 60 concept words. This benchmark evaluates how well the semantic models can make predictions about the patterns of neural activations observed in the human participants. For a set of words V, we calculate two pairwise word correlation matrices $M_D, M_B \in R^{|V|\times|V|}$ for a distributional semantic model (D) and the brain imaging data (B). We then perform a 2 vs. 2 test between M_D and M_B, where, for all pairs of words $w_1, w_2 \in V$, we count how often the similarity structure observed for D agrees with B, i.e. how often

$$\begin{aligned} &r(M_D(w_1), M_B(w_1)) + r(M_D(w_2), M_B(w_2)) \\ &> r(M_D(w_1), M_B(w_2)) + r(M_D(w_2), M_B(w_1)) \end{aligned}$$

where r is Pearson’s correlation and $M(w_1)$ and $M(w_2)$ denote the rows of values corresponding to the concepts w_1 and w_2, omitting the columns that correspond to the correlation between w_1 and w_2. The final score is the proportion of positive cases across all word pairs, with 0.5 indicating chance. Intuitively, this is a measure of how well the similarity profile of the semantic model matches the similarity profile of the brain data.

4.3 Human Property Knowledge

Next, we determine how well the embedding vectors for each network and layer capture commonsense aspects of meaning reflected in conceptual models from cognitive psychology. We achieve this by using probes to determine whether explicit lexico-semantic knowledge from human-derived property norms can be reliably decoded from these embeddings. For example, for the concept APPLE, can we predict from the embedding vector whether human-elicited properties of that concept such as *is-round* or *grows-on-trees* are true? For this analysis, we make use of a dataset of human-elicited property knowledge (the CSLB norms; Devereux et al., 2014)[3], which lists semantic properties for 638 concept words. These semantic properties are partitioned into five distinct categories, which characterise the different types of information they represent: **visual** (e.g. *is-green*; *is-round*), **functional** (e.g. *is-eaten*; *used-for-cutting*), **taxonomic** (e.g. *is-a-fruit*; *is-a-tool*), **encyclopedic** (e.g. *has-vitamins*; *uses-fuel*), and **other-perceptual** (e.g. *is-tasty*; *is-loud*). While property norming studies provide an insight into the types of information characterised by human conceptual representations, supported by human agreement on feature attributes, it should be noted that they are not a literal description of human lexical-semantic representation (Barsalou, 2003).

4.3.1 Probing methodology

For the probing analysis, we fit a number of $L2$-regularised logistic regression models, in order to predict whether or not a semantic feature is decodable from our embedding vectors, largely following previous work (Collell and Moens, 2016; Lucy and Gauthier, 2017; Derby et al., 2018). Due to the small sample size, each model uses class weight balancing and decodability is scored using the F1 score over 5 cross-validation folds. More specifically, we preprocess the CSLB dataset to exclude features occurring for fewer than five words. For each feature, we then partition the concepts into five folds using stratified sampling and perform 5-fold cross-validation on each feature.

Due to the high likelihood of overfitting, we also regularise each logistic regression by adding λ

[2] http://www.langlearnlab.cs.uvic.ca/brainbench/

[3] https://cslb.psychol.cam.ac.uk/propnorms

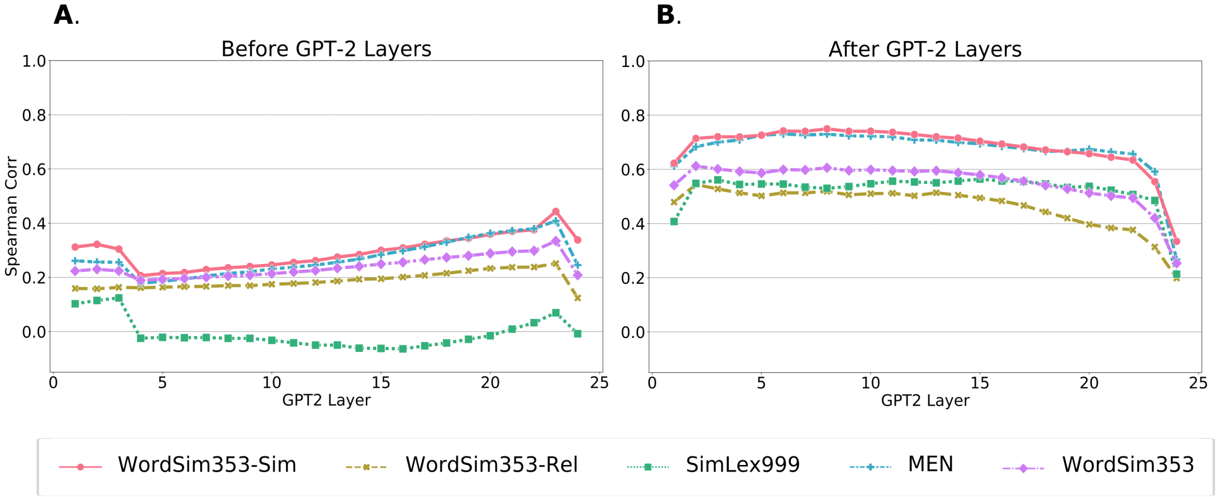

Figure 1: Results (Spearman correlations) for the **before** (on the left) and **after** word embedding vectors across all 24 **GPT-2** layers.

Model	WS353-Rel	WS353-Sim	WS353	SimLex999	MEN	fMRI	MEG
LSTM-based Representations							
JLM[1]-before	0.198	0.496	0.338	0.151	0.353	0.636	0.625
JLM[2]-before	0.314	0.549	0.428	0.115	0.423	0.650	0.638
JLM[1]-after	0.444	0.709	0.557	0.409	0.644	**0.681**	**0.701**
JLM[2]-after	0.280	0.580	0.414	0.423	0.544	0.669	0.692
Transformer-based Representations							
GPT2[Best]-before	0.251 [23]	0.439 [23]	0.334 [23]	0.124 [3]	0.409 [23]	0.627 [23]	0.648 [23]
GPT2[Best]-after	**0.544 [2]**	**0.749 [8]**	**0.612 [2]**	**0.561 [3]**	**0.730 [6]**	0.673 [14]	0.696 [16]

Table 1: Results (Spearman correlations) for each embedding model on the word similarity benchmarks, along with BrainBench results (accuracy) for the fMRI and MEG data. For GPT-2, we include the best performance across all 24 layers from the **before** and **after** representations (best layer number given in [brackets]).

times the $L2$ norm of the coefficient weights to the loss, where λ is a scaling parameter. Since we want to predict each individual property, we determine what value of λ to use by first performing 5-fold cross-validation for each property over a range of potential values, and choosing the best for each feature.

To calculate a decodability score for each feature, we run 5-fold cross-validation using the best λ value for each feature, for which we obtain the final F1 score on the predictions from the test folds. Furthermore, we repeat this cross-validation process three times and take the average score over each run. We note that just because a linear model does not predict the presence of a property does not mean that it is not encoded in the representation (Collell and Moens, 2016). Nevertheless, linear read-out from model activation patterns (and brain activation patterns) remains a useful tool for determining the presence of high-level information such as linguistic structure in those representations (Hewitt and Liang, 2019).

5 Results

5.1 Semantic Similarity Benchmarks

The similarity benchmark results are displayed in Table 1 and Figure 1. For both JLM and GPT-2, the word vector representations computed **after** the target word has been presented as an input token to the model perform better in comparison to when the network must predict the target word (the **before** representations). This result is not surprising, since in the **after** scenario the models have access to the target word itself. Nevertheless, we still see high correlations for the **before** representations for most models and layers, indicating that the representational state of the language models immediately before the target word reflect semantic content of the to-be-predicted word. GPT-2 produces the strongest correlations with human similarity judge-

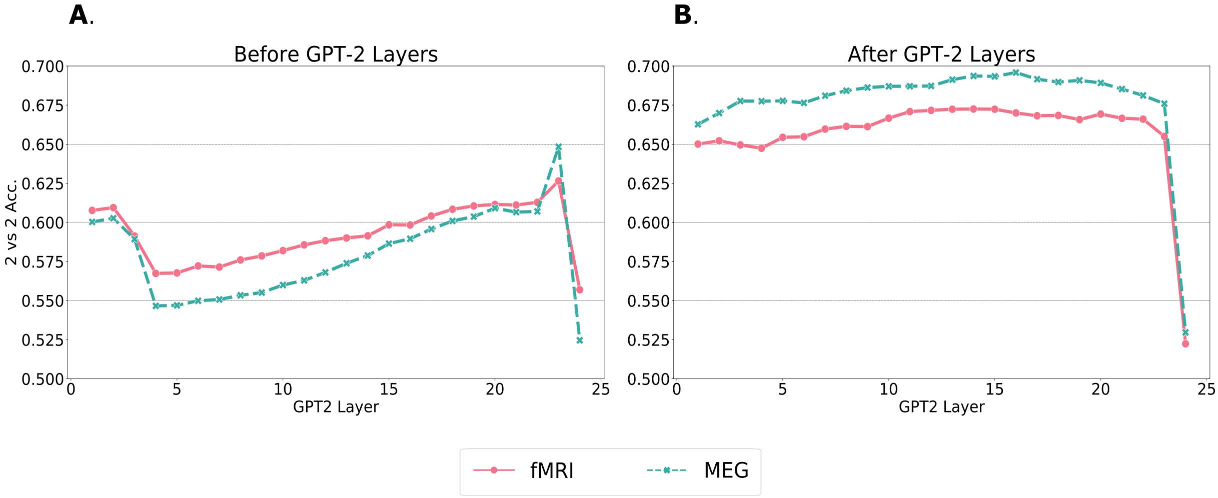

Figure 2: Results (accuracy %) on BrainBench for the MEG and fMRI data for each **before** (on the left) and **after** (on the right) word embedding models for each of the 24 layer of **GPT-2**. Scores are measured using accuracy from a $2 vs.$ 2 test, with a score of 0.5 indicating random chance (see text).

ments overall (particularly in earlier layers of the **after** representations; Fig. 1B). Interestingly, JLM outperforms GPT-2 in how accurately it predicts the brain data, perhaps due to a more cognitively plausible neural architecture that incrementally integrates information over the course of a sentence.

Focusing on the **before** representations, we see that the **JLM-before** semantic representations tend to perform better than the **GPT2-before** representations. This is likely because the LSTM is directly trained on the sampled sentences, which produces a lower perplexity measure than the transformer network, and thus it yields more accurate predictions about the target word. Comparing the **before** representations from different layers in each model, we see that JLM better represents semantic information in the second of its two layers, while for GPT-2 the results are more complex, though later layers are generally better, with the second last layer (23) being best for most evaluations. For both models, then, the upper layers tend to have the best overall semantic representations of the upcoming target word, which follows from the fact that the upper layers directly feed into predictions about the upcoming word in the language modelling task, with the models reflecting the predicted semantic content of that word.

When the target word is available to the model (the **after** representations), we would expect the network to represent meaningful information about the concept, which is why this approach is the most common method for building contextual representations. Our results support this notion, since the **after** representations consistently outperform the **before** representations, on both the word similarity and brain imaging data (see Fig. 2 for the GPT-2 BrainBench results). Notably, **JLM[1]-after** outperforms **JLM[2]-after**, since the activation patterns from the second layer should aim to predict the next word in the sequence (i.e. the word following the target word). Similarly, the **GPT2-after** representations retain semantic information of the word quite well for all but the final layer, with early layers performing well in the semantic similarity evaluations (Fig. 1B and Fig. 2B). **GPT2[24]-after** experiences a dramatic loss in performance, similar to what is observed for the **JLM[2]-after** representations.

Overall, this pattern or results supports the hypothesis that later layers of the language models best reflect semantic information about the to-be-predicted word, whilst earlier layers best reflect semantic information about the just-presented word, though all layers in both models reflect this information to some extent. In the next section, we investigate in more detail the specific kinds of semantic knowledge that is available in different layers of the models.

5.2 Semantic Feature Decoding

The results on the property decoding task are presented in Table 2 and Figure 3. Overall, we see that the GPT-2 layers encode more information about common sense property knowledge than the JLM layers, particularly in the **after** representations.

Focusing on the **before** representations we see

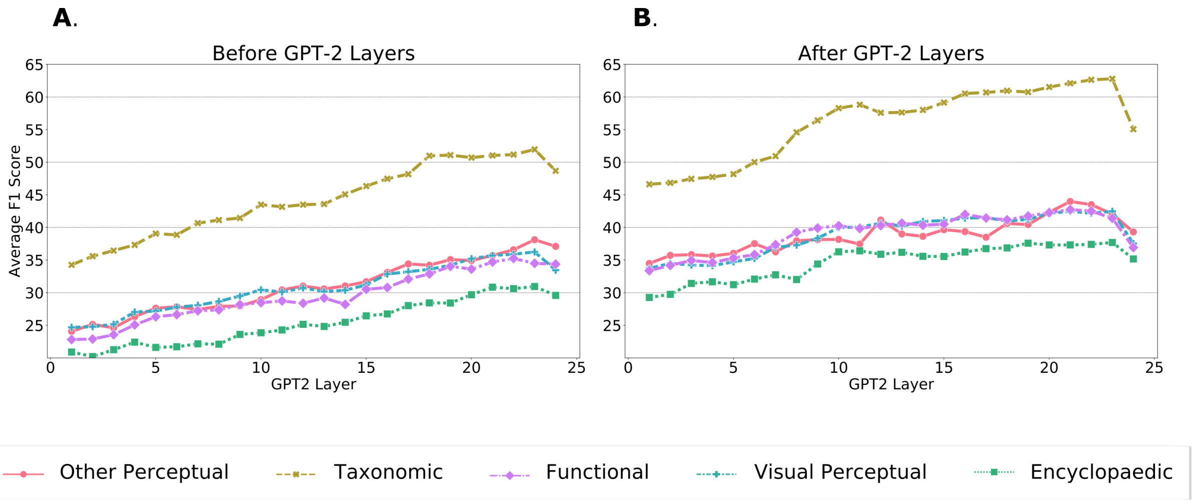

Figure 3: Graph which displays the average cross-validation F1 scores ×100 for each **before** (on the left) and **after** (on the right) transformer-based representations from each layer of **GPT-2**.

Model	**Encyclo.**	**Functional**	**Taxonomic**	**Visual**	**Other Perceptual**	**Overall**
LSTM-based Representations						
JLM[1]-before	21.74	24.92	42.31	29.48	28.91	28.56
JLM[2]-before	26.41	30.22	47.75	32.66	33.39	32.85
JLM[1]-after	33.21	38.29	60.71	38.44	36.83	40.01
JLM[2]-after	33.86	39.06	**62.82**	39.32	40.01	41.14
Transformer-based Representations						
GPT2[Best]-before	30.94 [23]	35.25 [23]	51.95 [23]	36.25 [23]	38.12 [23]	38.35 [23]
GPT2[Best]-after	**37.69 [23]**	**42.74 [21]**	62.79 [23]	**42.47 [23]**	**43.98 [23]**	**45.72 [21]**

Table 2: Average cross-validation F1 scores ×100 for each model and for each of the five property classes. For GPT-2, we include the best performance for each property type across all layers.

that GPT-2 tends to capture more knowledge about conceptual properties than JLM. Most notably, compared to JLM, the GPT-2 model does better at encoding knowledge related to attributive properties (i.e. non-taxonomic properties), which tend to be much more difficult to capture (Rubinstein et al., 2015). Both models show better property decoding performance in the later **before** layers. As these properties are related to conceptual knowledge plausibly associated with the upcoming word, it makes sense that the embedding vectors converge on some particular space related to the semantic restrictions on the upcoming word, which is particularly reflected in the case of taxonomic properties.

Turning to the **after** representations, we see that property knowledge seems to be best reflected in the upper layers of both language models. This is a particularly interesting result, as previous work has demonstrated that the lower layers contain more explicit information relating to the target word such as part-of-speech (Peters et al., 2018) and word association (see Section 5.1). Furthermore, while the **JLM-after** and **GPT2-after** representations perform similarly when predicting taxonomic features, GPT-2 does much better at capturing perceptual, functional, and encyclopedic knowledge. The results indicate that the GPT-2 representation appear to narrow the gap between taxonomic and attributive properties, which distributional models have historically struggled to accomplish. Finally, the network seems to retain and improve performance as we move through the layers.

6 Discussion

6.1 Last Layer Performance

First, we wish to discuss why there is a consistent loss in performance from the representations constructed from the final layer of the network, which is notable given the widespread use of the final layer for transfer learning. To better understand the results for the **GPT2-before** embedding vectors on our evaluation tasks, consider the work of Ethayarajh (2019), who demonstrated that the layers of GPT-2 become more context-specific as we move through the network, more so than LSTM-based networks such as *Elmo*. In particular, Ethayarajh (2019) investigated *intra-sentence similarity*, which measures the average cosine distance between the individual word representations and the sentence representation. In their work, sentence representations were constructed by averaging over the hidden states from all time steps in the sentence, which is similar to the **before** representations (averaging the vectors across sentences given the target word's position). They showed that, when adjusted for anisotropy, the intra-sentence similarity of GPT-2 tends to decrease until layer 4, before uniformly increasing again through the rest of layers. Hence, word representations from different time steps tend to be highly dissimilar from one another by the nature of the network, which demonstrates one limitation of feature pooling. While a limitation, we also note that this approach works well in general for building static word embeddings, supported by previous work (Bommasani et al., 2020)

6.2 Semantic Knowledge

From our initial results on the human judgement benchmarks, we can infer at what layers of the network semantic information about the concept is most representative. When the network must perform next word prediction on the concept, we see that the final layer is most representative, whilst after the word has been given to the network, we see that the semantic information about the concept decreases through as we move through the network. Such a result is not surprising as the network must gradually accumulate information that may be related to the next possible word, focusing less on the previous concept. Generally, the transformer outperforms the LSTM model after the network has received the concept in the lower layers, though the LSTM contained more representative information about the concept during next word prediction.

When probing for human conceptual knowledge, we see that the transformers perform better than the LSTMs, with the transformers performing quite well at predicting attributive features in comparison to taxonomic properties, for which there has historically been a large gap in performance (Rubinstein et al., 2015). These results may indicate that context, for which transformers produce highly contextualised representations (Ethayarajh, 2019), plays an important role in representing conceptual knowledge such as that reflected in semantic property norms. The most interesting result from our investigation is that the semantic knowledge is not forgotten in the later layers of both LSTM and transformer-based networks after receiving the concept, unlike the previous results. These findings may indicate that these networks gradually accumulate such knowledge as the sentence is processed in order to facilitate anticipation of the future. Such ideas have recently been proposed by Ferreira and Chantavarin (2018) who suggested that, in order to reconcile the differences between earlier models of integration (building associations between new concepts and previous information (Kintsch and Van Dijk, 1978; Gernsbacher, 1991)) with more recent theories of prediction, we should replace the notion of *Prediction* with *Preparedness*. Instead of considering direct prediction of future lexical items, which is usually rare (Luke and Christianson, 2016), the authors suggest that given some new information which is processed along with the past information with appropriate background knowledge, a new rich semantic representation is produced containing informative semantic features that facilitate anticipation. Our results indicate that these language models may similarly build and retain rich semantic representations that aid the network in its learning objective (conditional next word prediction).

7 Conclusion

In this paper, we present a novel approach to gaining a better understanding of the kinds of semantic information encoded within the layers of large-scale language models. Our analysis allows us to peer inside the hidden state representations of neural language models, and examine how semantically relevant information is encoded in each layer of the networks. We examine the language models on their ability to capture semantic meaning from two perspectives, when the network is predicting the target word, and when the target word is the most recent input. The results demonstrate that the transformer model is much better at capturing attributive features than the LSTM model, whilst both models are able to retain rich semantic representations of the concept after the concept has been given to the network.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. Association for Computational Linguistics.

Afra Alishahi, Grzegorz Chrupała, and Tal Linzen. 2019. Analyzing and interpreting neural networks for NLP: A report on the first BlackboxNLP workshop. *Natural Language Engineering*, 25(4):543–557. ZSCC: 0000012 Publisher: Cambridge University Press.

Lawrence W Barsalou. 2003. Abstraction in perceptual symbol systems. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 358(1435):1177–1187.

Miroslav Batchkarov, Thomas Kober, Jeremy Reffin, Julie Weeds, and David Weir. 2016. A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 7–12. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Jean-Philippe Bernardy and Shalom Lappin. 2017. Using deep neural networks to learn syntactic agreement. *LiLT (Linguistic Issues in Language Technology)*, 15.

Rishi Bommasani, Kelly Davis, and Claire Cardie. 2020. Interpreting pretrained contextualized representations via reductions to static embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Luana Bulat, Douwe Kiela, and Stephen Clark. 2016. Vision and feature norms: Improving automatic feature norm learning through cross-modal maps. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 579–588.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Guillem Collell and Marie-Francine Moens. 2016. Is an image worth more than a thousand words? on the fine-grain semantic differences between visual and linguistic representations. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2807–2817. The COLING 2016 Organizing Committee.

Jeff Da and Jungo Kusai. 2019. Cracking the contextual commonsense code: Understanding commonsense reasoning aptitude of deep contextual representations. *arXiv preprint arXiv:1910.01157*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Gary S Dell and Franklin Chang. 2014. The p-chain: Relating sentence production and its disorders to comprehension and acquisition. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1634):20120394.

Katherine A. DeLong, Thomas P. Urbach, and Marta Kutas. 2005. Probabilistic word pre-activation during language comprehension inferred from electrical brain activity. *Nature Neuroscience*, 8(8):1117–1121. 00359.

Steven Derby, Paul Miller, and Barry Devereux. 2019. Feature2vec: Distributional semantic modelling of human property knowledge. *arXiv preprint arXiv:1908.11439*.

Steven Derby, Paul Miller, Brian Murphy, and Barry Devereux. 2018. Using sparse semantic embeddings learned from multimodal text and image data to model human conceptual knowledge. In *Unpublished Manuscript*.

Barry J Devereux, Lorraine K Tyler, Jeroen Geertzen, and Billi Randall. 2014. The centre for speech, language and the brain (cslb) concept property norms. *Behavior research methods*, 46(4):1119–1127.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A Smith. 2016. Recurrent neural network grammars. *arXiv preprint arXiv:1602.07776*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*.

Allyson Ettinger. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Luana Fagarasan, Eva Maria Vecchi, and Stephen Clark. 2015. From distributional semantics to feature norms: grounding semantic models in human perceptual data. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 52–57.

Fernanda Ferreira and Suphasiree Chantavarin. 2018. Integration and Prediction in Language Processing: A Synthesis of Old and New. *Current Directions in Psychological Science*, 27(6):443–448. ZSCC: 0000022 Publisher: SAGE Publications Inc.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Morton Ann Gernsbacher. 1991. Cognitive processes and mechanisms in language comprehension: The structure building framework. In *Psychology of Learning and Motivation*, volume 27, pages 217–263. Elsevier.

Arthur C Graesser, Murray Singer, and Tom Trabasso. 1994. Constructing inferences during narrative text comprehension. *Psychological review*, 101(3):371.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. *arXiv preprint arXiv:1909.03368*.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Shailee Jain and Alexander Huth. 2018. Incorporating context into language encoding models for fmri. In *Advances in Neural Information Processing Systems*, pages 6628–6637.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Walter Kintsch and Teun A Van Dijk. 1978. Toward a model of text comprehension and production. *Psychological review*, 85(5):363.

Skanda Koppula, Khe Chai Sim, and Kean Chin. 2018. Understanding recurrent neural state using memory signatures. *arXiv preprint arXiv:1802.03816*.

Gina R. Kuperberg and T. Florian Jaeger. 2016. What do we mean by prediction in language comprehension? *Language, Cognition and Neuroscience*, 31(1):32–59. ZSCC: 0000414 Publisher: Routledge _eprint: https://doi.org/10.1080/23273798.2015.1102299.

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside bert's linguistic knowledge. *arXiv preprint arXiv:1906.01698*.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. *arXiv preprint arXiv:1611.01368*.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019. Linguistic knowledge and transferability of contextual representations. *arXiv preprint arXiv:1903.08855*.

Li Lucy and Jon Gauthier. 2017. Are distributional representations ready for the real world? evaluating word vectors for grounded perceptual meaning. *arXiv preprint arXiv:1705.11168*.

Steven G Luke and Kiel Christianson. 2016. Limits on lexical prediction during reading. *Cognitive Psychology*, 88:22–60.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. *arXiv preprint arXiv:1808.09031*.

Ken McRae, George S Cree, Mark S Seidenberg, and Chris McNorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37(4):547–559.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Ellen F Prince. 1978. On the function of existential presupposition in discourse. In *Papers from the... Regional Meeting. Chicago Ling. Soc. Chicago, Ill*, volume 14, pages 362–376.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf.*

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Anna Rogers, Shashwath Hosur Ananthakrishna, and Anna Rumshisky. 2018. What's in your embedding, and how it predicts task performance. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2690–2703.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works. *arXiv preprint arXiv:2002.12327*.

Dana Rubinstein, Effi Levi, Roy Schwartz, and Ari Rappoport. 2015. How well do distributional models capture different types of semantic knowledge? In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 726–730.

Carina Silberer. 2017. Grounding the meaning of words with visual attributes. In *Visual Attributes*, pages 331–362. Springer.

Pia Sommerauer. 2020. Why is penguin more similar to polar bear than to sea gull? analyzing conceptual knowledge in distributional models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 134–142.

Pia Sommerauer and Antske Fokkens. 2018. Firearms and tigers are dangerous, kitchen knives and zebras are not: Testing whether word embeddings can tell. *arXiv preprint arXiv:1809.01375*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*.

Mariya Toneva and Leila Wehbe. 2019. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). *arXiv preprint arXiv:1905.11833*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. ZSCC: 0008106 arXiv: 1706.03762.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do nlp models know numbers? probing numeracy in embeddings. *arXiv preprint arXiv:1909.07940*.

Lin Wang, Edward Wlotko, Edward Alexander, Lotte Schoot, Minjae Kim, Lena Warnke, and Gina R. Kuperberg. 2020. Neural Evidence for the Prediction of Animacy Features during Language Comprehension: Evidence from MEG and EEG Representational Similarity Analysis. *Journal of Neuroscience*, 40(16):3278–3291. ZSCC: NoCitationData[s1] Publisher: Society for Neuroscience Section: Research Articles.

Nathaniel Weir, Adam Poliak, and Benjamin Van Durme. 2020. On the existence of tacit assumptions in contextualized language models. *arXiv preprint arXiv:2004.04877*.

Haoyan Xu, Brian Murphy, and Alona Fyshe. 2016. Brainbench: A brain-image test suite for distributional semantic models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2017–2021.