Unsupervised Methods for the Study of Transformer Embeddings

Mira Ait Saada, François Role, Mohamed Nadif

Unsupervised Methods for the Study of Transformer Embeddings

Mira Ait Saada[1,2], François Role[1], Mohamed Nadif[1]

[1] Université de Paris, CNRS, Centre Borelli, F-75006 Paris
[2] Groupe Caisse des Dépôts, Paris
firstname.lastname@u-paris.fr

Abstract. Over the last decade neural word embeddings have become a cornerstone of many important text mining applications such as text classification, sentiment analysis, named entity recognition, question answering systems, etc. Particularly, Transformer-based contextual word embeddings have gained much attention with several works trying to understanding how such models work, through the use of supervised probing tasks, and usually emphasizing on BERT. In this paper, we propose a fully unsupervised manner to analyze Transformer-based embedding models in their bare state with no fine-tuning. We more precisely focus on characterizing and identifying groups of Transformer layers across 6 different Transformer models.

Keywords: Transformer-based Language Models · Unsupervised Learning · Word Embeddings

1 Introduction

Transformer-based word embeddings provided by neural language models are today increasingly used as the initial input to many text mining applications where they greatly contribute to achieve impressing performance levels. This has motivated a growing number of researchers to investigate the reasons behind this effectiveness as part of the general effort to unlock the black box of AI models. Since a Transformer model produces several embeddings for each word (one for each layer of its deep architecture), it is natural to study the nature of the embeddings learned at the different layers of the model. So far, the common way of doing this is to feed them as input to some supervised probing tasks (text classification, question answering, etc.) and then measure how well they perform on these tasks. From the observed performance, and depending on the probing task used, one may deduce, for example, that a given set of layers seems to be good at capturing some features of language while another set seems to encode another kind of information. While these experiments have allowed to draw some conclusions, the observed results depend both on the tasks and the train and test datasets, and so are not always generalizable. This observation prompted us to explore if it could be possible to gain additional insight into the behavior of the layers without having to rely on supervised probing tasks and external datasets.

In this paper, we propose unsupervised techniques that completely dispense of probing tasks, and demonstrate their interest by applying them to real datasets and several widely used Transformer models. The contributions of the paper are as follows:

1. We propose a set of unsupervised methods that allow to gain insights into the nature of the embeddings available at the different layers of a Transformer model, and how these embedding layers relate to each other. This approach, which directly leverages the intrinsic features of the layers, is in contrast to other studies that rely on probing tasks.
2. The experimental section shows that applying these methods on real datasets allows to acquire new knowledge about the layers of several Transformer models that seem to best perform on the important word clustering task.
3. Also, while most layer interpretation studies have so far focused mainly on BERT we provide a performance comparison for 3 different models namely BERT, RoBERTa and ALBERT, in both their base and large flavors.

2 Related Work

In the supervised learning realm, a growing body of research has been devoted to investigating the linguistic features learned by contextual word embedding models including LSTM-based models as in [9] and Transformer-based models like BERT as in [12]. Both authors agree to say that early layers encode most local syntactic phenomena while more complex semantics appear at the higher layers. In [8], the authors evaluate the performance of contextualized word representations on several supervised tasks and compare layers with each other, including ELMo, BERT (base and large) and OpenAI Transformer models. They especially observe that Transformers' middle layers allow for a better transferability. On the other hand, the authors in [5] observe that the early layers of BERT are more invariant across tasks and hence more transferable. It has also been shown in [1] that, after fine tuning BERT on Question Answering, the model acts in different phases starting from capturing the semantic meaning of tokens in the first layers to separating the answer token from the others in the last layers. It has been concluded that the closer we get to the last layer, the more task specific the representations are. This explains the results obtained in [7] which studies the changes between pre-trained and fine-tuned BERT-base model in terms of attention weights. A significant change in the two last layers in terms of cosine similarity between original and fine-tuned attention weights has been observed on 6 GLUE tasks. The authors deduced that the BERT-base's two last layers learn more task specific features. Several papers focus on identifying the linguistic structure implicitly learned by the models [2, 6]. For example, Goldberg [4] evaluates how well BERT captures syntactic information for subject-verb agreement. Ethayarajh et al. [3] try to assess how context-specific are the representations at the different layers of ELMo, BERT and GPT-2.

In contrast to the above studies we propose to identify coherent groupings of layers, based on the intrinsic characteristics of the layers and not by resorting to external probing tasks.

3 Unsupervised Methods for Layer Analysis

Deep Transformer models may have dozens of layers (see Table 3). In order to better understand their behavior we argue that it is useful to compare them, and try to identify groups with similar characteristics.

3.1 Matrix and Vector Representation of Layers

In this section, we propose several alternative (matrix- and vector-based) representations for a Transformer layer, thus allowing to study their correlations from multiple points of view. Given a dataset of n words, and a Transformer model with b layers and embedding dimension d, the dataset can be represented by b different matrices $\mathbf{X}_1, \ldots, \mathbf{X}_b$ of size $n \times d$, where each matrix $\mathbf{X}_\ell$ corresponds to the dataset at the ℓ-th layer. An alternative way of representing a layer ℓ is by averaging the rows of its $\mathbf{X}_\ell$ matrix, leading to a vector representation $\mathbf{v}_\ell$ of the layer. Additional intermediate data structures are then computed from these initial representations (Table 1). The pseudo-code in Algorithm 1 describes in detail how these data structures are created and used during the analysis process.

Table 1: Definitions and notations

Symbol	Description
n	Number of words of the dataset.
d	Number of dimensions: 768 for base models and 1024 for large ones.
b	Number of layers: 12 for base models and 24 for large ones.
$\mathbf{X}_\ell$	Matrix of size $(n \times d)$: data matrix of layer ℓ (cf. Figure 1).
$\mathbf{x}_{\ell i}$	The ith row of $\mathbf{X}_\ell$.
$\mathbf{S}_\ell$	Matrix of size $(n \times n)$: corresponds to the square matrix of $\mathbf{X}_\ell$.
$\mathbf{v}_\ell$	Vector of size d: computed for a layer ℓ as the average of rows of $\mathbf{X}_\ell$.
$\mathbf{r}_\ell$	Vector of size n: similarity ranks of words regarding $\mathbf{v}_\ell$.

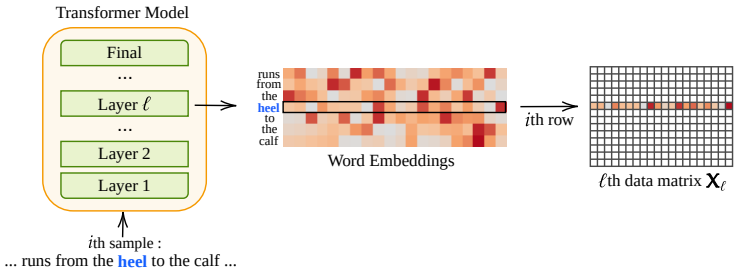

Fig. 1: Construction of the data matrices from contextual word embeddings

Algorithm 1: Unsupervised Process of Layers' Analysis

Input: a dataset D of n words; a Transformer model M with b layers and embedding dimension d; a clustering algorithm $\mathcal{C}$; a ranking function $rank$.

Step 1. Build matrix and vector representations of layers, for each $\ell = 1 \dots b$:

$\mathbf{X}_\ell \leftarrow$ vertical stacking of the n word embeddings provided by the ℓth layer

$\mathbf{S}_\ell \leftarrow \mathbf{X}_\ell \mathbf{X}_\ell^T$

$\mathbf{v}_\ell \leftarrow \sum_i \mathbf{x}_{\ell i}$, where $\mathbf{x}_{\ell i}$ is the ith row of $\mathbf{X}_\ell$

$\mathbf{e}_{\ell i} \leftarrow$ cosine similarity between the word vectors $\mathbf{x}_{\ell i}$ and $\mathbf{v}_\ell$, $i = 1 \dots, n$

$\mathbf{r}_{\ell i} \leftarrow rank(\mathbf{e}_{\ell i})$, $i = 1 \dots, n$

$R_v(\mathbf{X}_\ell, \mathbf{X}_{\ell'}) = \frac{trace(\mathbf{S}_\ell \times \mathbf{S}_{\ell'})}{\sqrt{trace(\mathbf{S}_\ell^2) \times trace(\mathbf{S}_{\ell'}^2)}}$, $\ell, \ell' = 1, \dots, b$

$\rho(\mathbf{r}_\ell, \mathbf{r}_{\ell'}) = \frac{6 \sum_{i=1}^{n} (\mathbf{r}_{\ell i} - \mathbf{r}_{\ell' i})^2}{n(n^2-1)}$, $\ell, \ell' = 1, \dots, b$ where ρ is the Spearman coefficient

Step 2. Identify groups of layers

Visualize the R_v and Spearman coefficients as heatmap matrices.

$\mathbf{V} \leftarrow$ vertical stacking of the b vectors $\mathbf{v}_\ell$, $\ell = 1 \dots b$

$clusters \leftarrow \mathcal{C}(\mathbf{V})$

Visualise $clusters$

Step 3. Interpret the groups identified in step 2.

3.2 Measuring the Correlations between Layers

In this section and the following one, we propose unsupervised methods for comparing the layers against each other the goal being to exhibit layers that share some characteristics.

When using a matrix representation for the layers (the $\mathbf{X}_\ell$ matrices of Table 1), an appropriate correlation measure is the R_v coefficient [10] which can be used in order to visualize the layers' similarities and distinguish any possible bloc structures. The R_v coefficient can be interpreted as a non centered squared coefficient of correlation between two given data matrices $\mathbf{X}_\ell$ and $\mathbf{X}_{\ell'}$ (cf. Algorithm 1). This proportion varies between 0 and 1 and the closer to 1 it is, the better is $\mathbf{X}_{\ell'}$ as a substitute for $\mathbf{X}_\ell$ (and *vice-versa*) to characterize the n samples of the dataset. In order to draw a similarity tendency across layers, we compute for each layer ℓ an *Average-*R_v which corresponds to the mean of R_v values between the layer ℓ and the rest of the layers. The heatmap representation of these values allows to spot groupings of layers.

The vector representation of the layers (the $\mathbf{v}_\ell$ vectors) allow for other possibilities. They can be used as input to a clustering algorithm (this will be described in sections 3.3 and 4.3). But they can also serve as a basis for measuring the correlations between layers. For each layer ℓ, we first compute the cosine similarity of its vector $\mathbf{v}_\ell$ with all the word vector $\mathbf{x}_{\ell i}$. A ranking vector $\mathbf{r}_\ell$ is then computed where $\mathbf{r}_{\ell i}$ is the ranking assigned to word i by layer ℓ. Since for two layers ℓ and ℓ' $\mathbf{r}_\ell$ and $\mathbf{r}_{\ell'}$ contain word ranks, a suitable measure of comparison is the Spearman correlation coefficient ρ that measures the rank correlation between two variables. From the ρ values between each pair of layers we can construct a heatmap matrix of size $b \times b$ which, as with the matrix of R_v values, also allows to identify groupings of layers (cf. Algorithm 1).

3.3 Clustering Layers

The next step of the study is to perform a cluster analysis to confirm the potential groups using the techniques described in the previous section. The data samples are the b layers of a given model where each layer ℓ is represented by its corresponding average vector $\mathbf{v}_\ell$. In theory, any kind of clustering algorithm could be used at this stage. In practice, since the number of layers is relatively low and the number of cluster is unknown (although the techniques of the previous section can give an estimate of it), we often used Agglomerative Hierarchical Clustering (AHC) methods in our experiments. The hierarchical arrangement of the samples provided by the dendrograms indeed allows for a better interpretation of the clustering results as will be shown in the experiments section.

3.4 Interpreting Layers

In order to provide a more qualitative analysis of layers' behavior, we use the vector representation $\mathbf{r}_\ell$ and visualize the ranking of the first m words regarding their similarity to $\mathbf{v}_\ell$. We can also deepen our analysis of layers by using the interpretation abilities offered by dimension reduction techniques such as the Principal Component Analysis (PCA). When applying PCA on the $\mathbf{X}_\ell$ matrices, the samples are the word representations and the features represent the dimensions of the embeddings. The cos^2 measure denotes the correlation between a principal component and a given dimension (feature). It also measures the quality of representation of the feature, which allows us to select only the features that are more influential for interpretation.

4 Experiments

In this section, we first apply the process described in Algorithm 1 to several word datasets, in a step by step manner. Then, in order to validate the above methods and better understand the results they provide, we cluster our word datasets and evaluate each layer in terms of clustering performance. To achieve that, we perform word clustering experiments on the $\mathbf{X}_\ell$ matrices. Each clustering run provides a partition containing the cluster label of each word. To evaluate the partition obtained with each layer, we rely on a standard external measure for assessing clustering quality, namely Normalized Mutual Information (NMI) measure [11].

4.1 Datasets and Models Used

The datasets of size n used in the experiments are described in Table 2. The first dataset, referred to as UFSAC3, is extracted from the UFSAC dataset [13] which consolidates multiple popular datasets annotated with WordNet (such as SemEval and SensEval) into a uniform format. The examples are manually divided into three topics: Body, Botany and Geography. The second dataset, UFSAC4, is

a slightly more difficult dataset since it includes a fourth class (words related to Information Technology) and is augmented with some polysemous examples (such as "lobes" which appears in Body and Botany with different contexts). The third dataset yahoo4, is extracted from the Yahoo! Answers dataset [14] by manually selecting sets of words for each category and some corresponding contexts.

Table 3 describes the 6 pre-trained Transformer-based embedding models used for the experiments, without any fine-tuning.

Table 2: datasets description: k denotes the number of clusters.

datasets	n	k	clusters' sizes
UFSAC3	583	3	body: 266, geo: 227, botany: 90
UFSAC4	691	4	body: 275, geo: 227, botany: 99, IT:90
yahoo4	528	4	finance: 150, science-maths.: 152, computers-internet: 144, music: 82

Table 3: Transformer models' description.

	b	d	vocab size
BERT-base-cased	12	768	28,996
RoBERTa-base			50,265
ALBERT-base-v2			30,000
BERT-large-cased	24	1024	28,996
RoBERTa-large			50,265
ALBERT-large-v2			30,000

4.2 Investigating the Correlations between Layers

For comparing the layers with each other, we experimented with the R_v coefficient and the Spearman's rank correlation coefficient (cf. Section 3.2) . Figure 2 shows the similarities computed between the layers of each model in terms of the R_v coefficient which uses the matrix-based representations of the layers. As a result

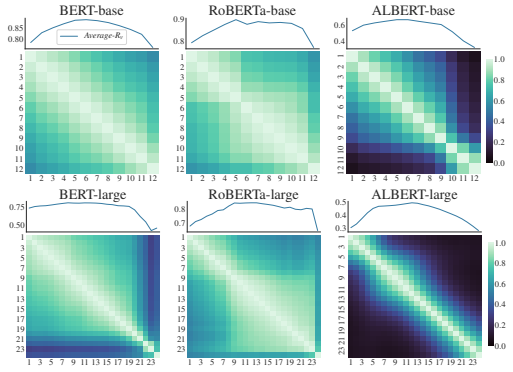

Fig. 2: R_v-coefficient based layer-wise similarity computed between UFSAC4's data matrices $\mathbf{X}_\ell$.

of the way in which Transformer models operate, one would expect that a layer is similar to the one following it. This is indeed what is observed globally in Figure 2. However, when taking a closer look, some interesting remarks can be made:

– Several rectangular blocks can be spotted. This is confirmed by the curve of the average R_v value which is drawn on top of the heatmaps. Clearly there

are breakpoints separating groups of layers that share common characteristics in terms of affinities with other layers.
– One can observe a significant decrease of average similarity on the three last layers with the last layer sometimes having a distinctive behavior.
– ALBERT models and especially ALBERT-large are very different from the other models in terms of layer-wise similarity [3].

Additional insights can be gained from the Spearman correlation coefficients computed on pairs of layers using their ranking vectors $\mathbf{r}_\ell$ (Figure 3).

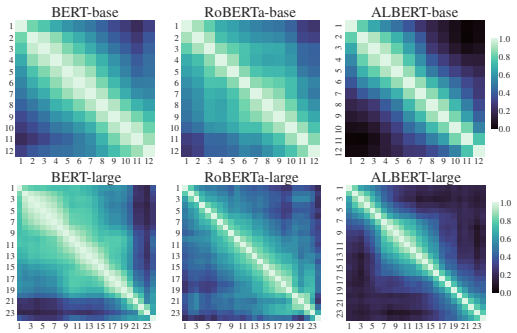

Fig. 3: Layer-wise agreements using Spearman correlation coefficient: the agreement coefficient between two layers ℓ and ℓ' is the Spearman correlation coefficients ρ calculated between $\mathbf{r}_\ell$ and $\mathbf{r}_{\ell'}$.

These coefficients allow to refine the observations made on Figure 2, we can notice an even bigger difference between the 1st layer and the rest of the network in terms of correlation for BERT-base, moving from $\rho = 0.82$ between layers 1 and 2 to $\rho = 0.95$ between 2 and 3. Overall, Figure 3 reveals a certain block structure with groups $\{1\}$, $\{2, 3, 4\}$ and $\{5, 6, 7, 8\}$. Finally, another break can be observed between layers 11 and 12 where $\rho = 0.88$ while it was $\rho = 0.94$ between layers 10 and 11 leading to two new groups $\{9, 10, 11\}$ and $\{12\}$. The same kind of block structure can also be observed when looking at the other models.

4.3 Identifying Clusters of Layers

In order to have another look at the possible groupings of layers, we perform an AHC and draw the associated dendrograms (cf. Section 3.3). Figure 4 shows the results obtained using the Ward and Average linkage criteria, used respectively with the euclidean and cosine distances. If we look at the results for BERT-base that are obtained using the Euclidean distance, we can see that the clusters are close to the groupings suggested by the methods of Section 4.2 (compare with the top-left heatmap in Figure 3), with the exception of layer 12 which strangely seems to be close to layer 1. This can be explained by the fact that we use the euclidean distance, which tends to be sensitive to the amplitude of data. If we look at BERT-base's box plots in Figure 5, it can be seen that the variance of the last layer is very close to that of the 1st layer. To confirm that this wrong

[3] This could be explained by the parameter sharing technique used to train the ALBERT model, which consists of duplicating the same parameters for all layers [5]

assignment was due to an amplitude issue, we experimented with the same AHC algorithm using a cosine distance (known to be insensitive to vector magnitude) with the "average" linkage strategy. With this configuration, the 1st layer is even more separated from the following ones and as expected, the last layer is much less close to the 1st and is assigned to a separate cluster, which is coherent with the previous observations (Section 4.2). This intuition is confirmed when looking at RoBERTa-large, for which we don't have the problem of differing variances across layers (Figure 5) and hence presenting almost the same groupings using the two distances. Overall, clustering the layers leads to the following observations:

- As shown in Figure 4 for BERT-base, the 4th layer is merged with the $\{2, 3\}$ cluster before the 1st layer, which confirms the break between the first layer and the following ones. In fact, the first layer is, for most models, isolated in its own cluster. This behavior is visible in Figure 2 and even more in Figure 3 where the 1st layer (1st row) has darker colors than its following neighbors, which indicates lower correlation values compared to the other layers.
- For RoBERTa-large we can also see that the 1st layer joins the 2nd only after layers 3 and 4 (especially in the Cosine version). The last layer is also isolated, joining a cluster only at the 3rd merge of the AHC. The rest of the clusters generally contain successive layers (like $\{5, 6, 7\}$). When cutting at the second merge level we end up with the following partition $\{1\}$, $\{2, 3, 4\}$, $\{5, 6, 7\}$, $\{8, 9, 10\}$, $\{11, 12, 13, 14\}$, $\{15, 16, 17\}$, $\{18, 19, 20, 21\}$, $\{22, 23\}$, $\{24\}$.

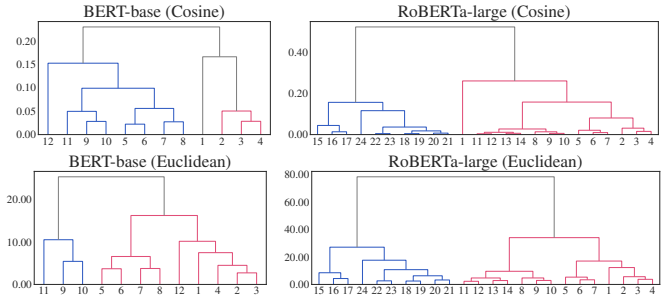

Fig. 4: Dendrograms obtained with AHC from the set of layers where each layer is represented by $\mathbf{v}_\ell$ a $d-$dimensional vector computed on UFSAC4.

In Figure 5, we use the vector representation $\mathbf{v}_\ell$ to draw box plots to analyze the distribution's evolution of layers over the network. We first observe that the three models present different behaviors in terms of variance with from the smallest to the largest: RoBERTa, BERT and ALBERT. Despite that, all distributions are centered around zero with the lower and upper boundaries being quite symmetric. Besides, for BERT (base and large), we can observe a certain break at the last layers (progressive increase followed by a sudden drop) corresponding to the breaks of similarity observed in Figures 2 and 3.

4.4 Qualitative Interpretation

Table 4 shows the first $m = 30$ words that are the closest to the $\mathbf{v}_\ell$ representations of a selection of layers (due to space limitations) of BERT-base for the UFSAC4

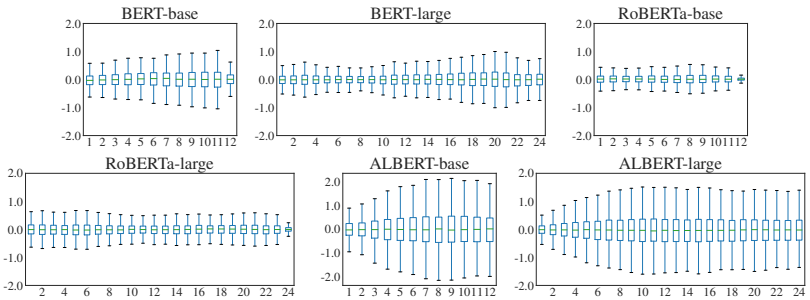

Fig. 5: Evolution of box plots (without outliers) over layers: each layer is represented by its average vector $\mathbf{v}_\ell$ of the UFSAC4 dataset.

dataset. Confirming the insights provided by the previous rank-based comparisons between layers as well as the clustering experiments, one can note a significant break between layer 1 and its immediate neighbors. Layers 2, 3 and 4 resemble each other more than they resemble layer 1, and share more words such as axons, sclera and scrotum. The correlation scores displayed on top of each pair of layers in Table 4 confirm this visual inspection. Between layers 5 and 8 (not shown here), we observed a continuous shift of words in the sense that a limited number of words appeared and disappeared from a layer to another, with the vanishing of Botany words from the 5th layer. More new words start to appear on the 9th layer

Table 4: Ranking of the words (colored according to their topic) that are closest to $\mathbf{v}_\ell$ representations the BERT-base layers for the UFSAC4 dataset. The first row contains the pairwise Spearman correlation coefficient.

	$\rho = 0.82$	$\rho = 0.95$	$\rho = 0.94$		$\rho = 0.93$	$\rho = 0.94$	$\rho = 0.88$	
1	2	3	4	9	10	11	12
cerebellar	cerebellar	cerebellar	bronchial	adenoids	adenoids	anus	penis
bulbs	kernels	bronchial	cerebellar	atrium	cerebellum	eardrum	eardrum
perennials	maxillae	sclera	molars	hipbones	anus	penis	anus
orchids	bronchial	bronchioles	cranial	anus	Bermuda	armpit	ribs
bronchioles	sclera	clavicles	axons	cerebellum	atrium	cortical	cortical
lymphocyte	cerebellum	cerebellum	arterioles	armpit	eardrum	Bermuda	sphincter
mosses	deserts	cranial	sclera	gyral	armpit	cerebellum	clitoris
rootstocks	clavicles	molars	bronchioles	Bermuda	penis	atrium	pelvic
bronchial	axons	axons	clavicles	sinus	gyral	skull	bulbar
clavicle	bronchioles	arterioles	cerebellum	sphincter	sphincter	Egyptian	calf
leaflets	arterioles	brachial	follicle	leg	Armenia	breastbone	skull
follicle	molars	maxillae	epidermis	clitoris	clitoris	adenoids	armpit
arteriovenous	hindbrain	axon	axon	brachial	pelvic	pelvic	peritoneum
occipital	interface	cervical	brachial	breastbone	hipbones	Bavaria	palmar
maxillae	bulbs	rootstocks	scrotum	incisors	breastbone	clitoris	cerebellum
cheekbone	scrotum	kernels	cervical	skull	calf	Armenia	Carpal
cerebellum	cranial	scrotum	cheekbone	eardrum	skull	gyral	gallbladder
cranial	pods	interface	hindbrain	hepatic	sinus	sphincter	distal
epidermis	sphincter	cerebrum	maxillae	brain	arteriovenous	liver	leg
mucosa	pylorus	deserts	peritoneum	arcuate	Egyptian	calf	wrist
clavicles	rootstocks	epidermis	saliva	cheekbone	leg	peritoneum	hips
pods	epidermis	follicle	incisors	eye	Bavaria	sinus	gut
herbaceous	areolae	sphincter	triceps	muscles	cortical	membrane	bronchial
brachial	follicle	peritoneum	aorta	bones	arcuate	Syria	anal
metatarsal	axon	hindbrain	rootstocks	rump	body	bulbar	scrotum
kernels	glomeruli	saliva	atrium	liver	liver	arcuate	brachial
arterioles	peritoneum	cheekbone	lumbar	cilia	CNS	leg	fibula
pylorus	lymphocyte	triceps	cerebrovascular	Armenia	hepatic	hipbones	Bermuda
metatarsus	arteriovenous	cerebrovascular	gyral	ribs	rump	palmar	liver
molars	occipital	arteriovenous	kernels	dental	cardiovascular	hips	basal

like Bermuda, sinus and hepatic with an increasing number of Geography words, until layer 11 inclusive. Layer 12 includes new words that do not appear before like ribs, Carpal and gallbladder and fewer number of Geography words. Overall, the qualitative analysis seems to be in good agreement with what was observed with the previous methods. For example, it seems to attest to the existence of

five groups of layers in the standard BERT-base model, namely $\{1\}$, $\{2,3,4\}$, $\{5,6,7,8\}$, $\{9,10,11\}$ and $\{12\}$.

4.5 Quantitative Interpretation Using Dimension Reduction

In this section, relying on PCA, the objective is to go further in the unsupervised analysis of embeddings at different layers. Figure 6 presents visual representations provided by PCA applied to the $\mathbf{X}_\ell$ matrices of each layer. First, on the projections of samples, one can observe a significant enhancement of the separability of samples between the layers 1 and 2 whereas it is almost the same between 2 and 3. We noticed another difference between layers 4 and 5 with a sort of rotation of samples along with a higher increase of variance explained by the two first components. The separability remains more or less stable until layer 11 inclusive and deteriorates in the 12th layer, which also knows a significant increase of explained variance. These differences in separability indicate that the extremities of the network are not only different, but may be much less efficient. Concerning the correlation circles, we notice more differences between layers 1 and 2 than between 2 and 3, this confirms that the 1st layer constitutes a singleton. We also observe a shift across layers with many dimensions that appear in few consecutive layers and then disappear (like 643 appearing in the 2nd layer and disappearing in the 5th layer). Another significant break is observed at the last layer, where dimensions like 223 and 636 disappear while being important for layers $9, 10$ and 11. These observations reinforce our previous groupings for BERT-base.

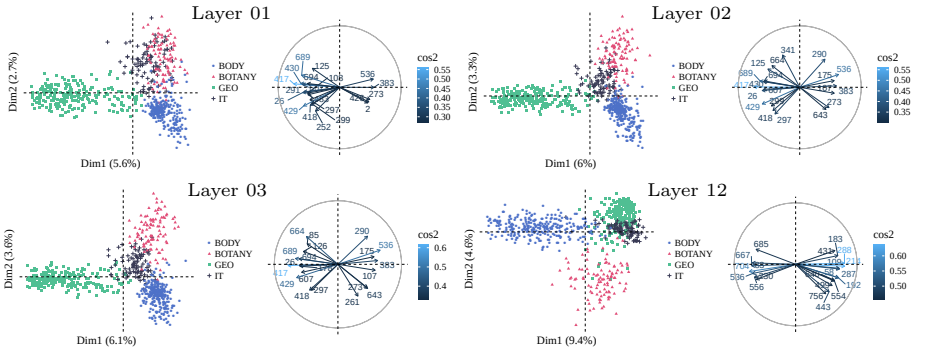

Fig. 6: PCA on BERT-base's data matrices $\mathbf{X}_\ell$, $\ell = 1, \ldots, b$ - Projections (left): coordinates of words on the two first principal components colored w.r.t. their topic. - Correlation circle (right): only the 20 dimensions that are most correlated with the two first components are displayed.

4.6 Results Validation Using a Clustering Performance Metric

In this section, we provide numerical results assessing the layer-wise performance on word clustering using NMI scores (Figure 7) on clustering partitions obtained with K-means applied to $\mathbf{X}_\ell$ matrices. In doing so, we aim to validate the layer groups that have been identified in the previous sections. The main question we try to answer is: Do the previously identified groups share characteristics in terms of clustering performance? This study also gives us an idea of the transferability of each layer and each model for the unsupervised task of word clustering. By

separating layers into groups based on the NMI scores they achieve, one can find clusters of layers that quite resemble the breakdown suggested by the dendograms in Section 4.3 (compare the values in Table 5 with the corresponding dendograms depicted in Figure 4). For BERT-base, in the same way as layer 1 is isolated in its own singleton cluster, its NMI score is also the worst. The group formed by 1, 2 and 3 achieves values between 0.78 and 0.88, while the best performers are the layers from 5 to 8. Performance then decreases, with a marked drop at the last layer, again in agreement with the grouping patterns observed in Figures 3 and 4. The same observations extend to RoBERTa-large where the cluster $\{5, 6, 7\}$ contains the best performing layers. We also clearly see a breaking point of performance between the 1st layer and the following, and another one (more acute) at the last layer. These breaking points are visible in Figures 2 and 3. In

Table 5: NMI scores on blocks of layers with UFSAC4 - The first table corresponds to BERT-base and the second to RoBERTa-large. The groups obtained based on word clustering performance fairly closely correspond to the groups that had been spotted using correlation and cluster analyses.

ℓ	01	02	03	04	05	06	07	08	09	10	11	12
NMI	0.64	0.78	0.81	0.88	0.9	0.94	0.9	0.92	0.91	0.91	0.88	0.83

ℓ	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
NMI	0.51	0.57	0.55	0.61	0.81	0.87	0.9	0.7	0.64	0.62	0.63	0.62	0.61	0.62	0.61	0.55	0.53	0.51	0.48	0.41	0.41	0.58	0.57	0.38

addition, these observations allow us to confirm some findings presented in the supervised study [8] showing that BERT models achieve their best performance on the intermediate layers. We also extend this observation to RoBERTa with fewer well performing layers, situated more earlier in the network. More generally,

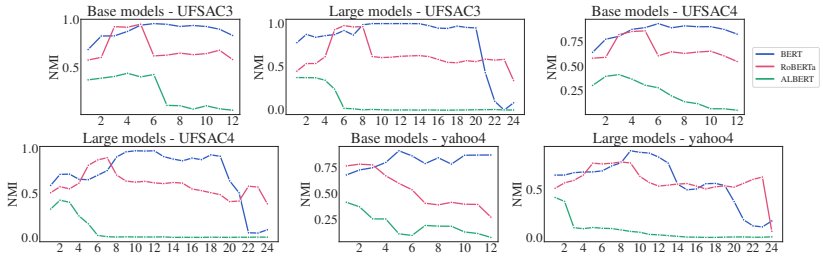

Fig. 7: NMI scores obtained by the word clustering on the $\mathbf{X}_\ell$ data matrices for each layer ℓ.

on both base and large versions, BERT outperforms the three other models, followed by RoBERTa and far away by ALBERT. This is surprising considering that ALBERT is supposed to outperform BERT and RoBERTa when fine-tuned on supervised tasks. We then show that in a no fine-tuning configuration, BERT word embeddings are of higher quality (BERT-large is the only model to achieved the perfect score on UFSAC3). Finally, ALBERT is the only model for which the base version is better than the large one. Moreover, both versions present very poor results on word clustering (especially the large version) and we can notice a better (but still poor) performance with the first layers. One possible explanation is that ALBERT's layers start to be task specific from the beginning of the network, particularly in view of the architecture of ALBERT where all parameters (including attention parameters) are shared across layers.

5 Conclusion

Knowing more about contextualized word embeddings and what can really be expected from them is an important topic. This paper provides a novel way of analysing Transformer embeddings, based on unsupervised methods, more specifically a correlation and cluster analysis of the layers. Applying these methods to real datasets made it possible to spot precise groups of layers (e.g. 5 groups of layers in BERT-base and 9 in RoBERTa-large) which subsequently proved to fairly closely match the groups obtained when grouping layers based on their clustering performance. This suggests that the proposed method, when applied to a dataset is capable of identifying in advance groups of layers that are likely to best or worst perform on the clustering task. This study also allowed to bring out major differences between Transformer models on the important text clustering task, for example the specificity of ALBERT, which is most likely due to its different network architecture, or the fact that BERT seems to outperform RoBERTa on the clustering task. Future path for research is to further investigate these differences as well as the potential of dimension reduction techniques on contextual word embeddings, an issue that deserves to be the subject of further study, allowing in particular to highlight the potential redundancies present in the Transformer networks.

References

1. van Aken, B., Winter, B., Löser, A., Gers, F.A.: How does bert answer questions? a layer-wise analysis of transformer representations. In: CIKM. pp. 1823–1832 (2019)
2. Clark, K., Khandelwal, U., Levy, O., Manning, C.D.: What does bert look at? an analysis of bert's attention. arXiv (2019)
3. Ethayarajh, K., Duvenaud, D., Hirst, G.: Understanding undesirable word embedding associations. arXiv (2019)
4. Goldberg, Y.: Assessing bert's syntactic abilities. arXiv (2019)
5. Hao, Y., Dong, L., Wei, F., Xu, K.: Visualizing and understanding the effectiveness of bert. arXiv (2019)
6. Jawahar, G., Sagot, B., Seddah, D.: What does bert learn about the structure of language? (2019)
7. Kovaleva, O., Romanov, A., Rogers, A., Rumshisky, A.: Revealing the dark secrets of bert. arXiv (2019)
8. Liu, N.F., Gardner, M., Belinkov, Y., Peters, M.E., Smith, N.A.: Linguistic knowledge and transferability of contextual representations. arXiv (2019)
9. Peters, M.E., Neumann, M., Zettlemoyer, L., Yih, W.t.: Dissecting contextual word embeddings: Architecture and representation. arXiv (2018)
10. Robert, P., Escoufier, Y.: A unifying tool for linear multivariate statistical methods: the rv-coefficient. Journal of the Royal Statistical Society **25**(3), 257–265 (1976)
11. Strehl, A., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. Journal of machine learning research **3**, 583–617 (2002)
12. Tenney, I., Das, D., Pavlick, E.: Bert rediscovers the classical nlp pipeline (2019)
13. Vial, L., Lecouteux, B., Schwab, D.: Ufsac: Unification of sense annotated corpora and tools (2018)
14. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: Advances in neural information processing systems (2015)