Functional MRI marked a revolution in the field of neuroimaging in the 1990's. For the first time, unambiguous images of human brain changes could be seen with MR. This technology opened new doors for Psychology and Neurology research, and also brought new challenges for the analysis of such images that have lower resolution but higher dimension. How can I deal with the new time dimension in addition to the spatial dimension ? How can I find the relationships between the various areas in the brain across various timescales ?
I will focus here on the context related to ASD that, even as a particular application, reflects the methodologies applied to analyse fMRI data well.

So far, the outcomes of predictive models, mostly based on Machine Learning (ML), built on fMRI data have appeared more performant in the binary classification of ASD / non-ASD compared to approaches on sMRI data [21].
Traditional fMRI analysis with ML involves a lot of preprocessing steps, compressing the information to extracted and derived neuroimaging parameters, like Regional Homogeneity (ReHo) [22] or intrinsic connectivity networks [23-24]. While these features make sense in neuroscience, I may lose important information to build predictive models during this first step in the analytical pipeline.
A key challenge, that is global to ML, is to find what the appropriate amount of data is for a given model and task in order to be able to conclude on true and replicable neuroimaging markers (He et al., 2020; Traut et al., 2021). A bias is induced by preprocessing raw fMRI data that may raise the difficulty to reach such a balance (Traut et al., 2021)(Dadi et al, 2019). For instance, (Churchill et al. 2012) examined how different preprocessing steps like realignment and physiological noise correction impact resting-state fMRI data and connectivity results. They showed that optimising pipelines on an individual-subject basis revealed brain activation patterns either weak or absent under fixed pipelines, which has implications for the overall interpretation of fMRI data, and the relative importance of preprocessing methods.
Unintended replicability issues of fMRI studies or non-reproducible neuroimaging markers have been raised as main concerns in the domain (He et al., 2019)(Churchill et al. 2012)(Dadi et al, 2019)(Traut et al., 2021), and these

events aggravate when studying small samples while getting large datasets remains arduous.

Predictive modelling with Deep Learning (DL) algorithms have been promising in the task of ASD classification, sometimes outperforming ML approaches [21, 25-40]. In its philosophy and its functioning, DL should use much less preprocessed data than ML (LeCun et al., 2015).

Reviewing the literature, I noticed that many DL algorithms were trained on connectivity matrices, often resulting from the brain parcellation of the 4D scan into regions of interest (ROIs) according to relevant and recognised atlases like the Automated Anatomical Labeling (AAL) [41], the extraction of mean-time series inside each ROI, and the computation of a Pearson correlation or a partial correlation coefficient between all the pairs of ROIs, resulting in a connectivity matrix (Biswal et al., 1995)(Dadi et al., 2019).

Connectivity matrices can be seen as graphs or as 3D numerical images when concatenated, that is why many studies have lately involved graphical networks or 3D CNN on connectivity matrices to detect ASD [25, 27, 29, 30, 35, 36]. For instance, [25] used connectivity “fingerprints” as 3D images and constructed a model with 3D CNN. Another example is given by [29], where they built graphs from connectivity matrices as templates of ASD and of non-ASD fMRI information. [29] were hence able to build a discriminative network with spectral convolutions operated on these graphs.

Connectivity matrices based on Pearson's correlation might remove certain patterns when comparing two time series like the phase, whereas this parameter might be important to indicate when certain brain areas are asynchronously responding to the activation of others.

Integrating the time feature in the analysis is tough, but several research teams took up the challenge thanks to DL approaches in the context of ASD prediction. For instance [42] used LSTMs to generate an embedding vector of all the rs-fMRI time series of a scan, and next the vector was used as input to a dense network. [43-45] used high dimensional convolutional neural networks on 3D or 4D data.

While space and time is important to build models on fMRI data that are biologically consistent, it is also important in the field of Natural Language Processing (NLP).

Recently, Transformers algorithms [46] have been a new top-paradigm in NLP. (add ref to chapter 2) This type of algorithm has inspired new approaches in medical image processing [47-49] that have demonstrated the relevancy of such algorithms in image analysis. In particular, [49] used a model similar to an encoder architecture of Transformers in order to process compressed 4D task-fMRI data, for task recognition. The compression of 4D fMRI was obtained by training a 3D CNN that compressed all the 3D frames into a tensor of 3D embedded frames that was used as input to the encoder. After training, they were able to get what time frames were globally important for every task.

[50] proposed a model to predict brain states from short sequences of task fMRI data based on a Transformer architecture (Vaswani et al., 2017). The model takes as input matrices of voxel time-series of a whole brain and it compresses the spatial information. This is similar to data-driven techniques like ICA (Kiviniemi et al., 2003)(Mckeown et al., 1998) that learn brain regions in an unsupervised way. A second model takes the latent representation as input to predict a brain state. This methodology seems to work well for task-fMRI data. However, it is difficult to understand the learned embedding as well-defined regions replicable across the individuals. In addition, only the spatial information is compressed here, while it is crucial to also process the time information in order to extract spatio-temporal relationships between brain areas when analysing resting-state fMRI data.

[51] proposed a cross window approach that takes both into account spatial and temporal information within a Transformer. They also used a CLS token (Dosovitskiy et al., 2021) that is learned to summarise latent representations of each layer in the Transformer encoder and that can be used in downstream tasks like ASD classification.

[52] built a self-supervised transformer where the input is actually the output of a 3D CNN that compresses the brain volumes in smaller representations. The 3D CNN has an encoder-decoder architecture. The training occurs in two phases: (1) the 3D CNN is trained to encode a brain volume in a latent representation and to reconstruct the same brain volume via the decoder; (2)

the transformer is trained, taking as input the latent representation of the 3D CNN, and the pre-trained 3D CNN is also improved. The issue of the [52] approach is that the spatial information is totally compressed during the 3D CNN encoder-decoder step, which may lose interesting spatial interactions between time-series at different timeframes. This approach may be more appropriate for task fMRI analysis because of the frame by frame decomposition approach.

[54] also proposed a self-supervised approach where two Transformers are used to simultaneously infer common and individual functional networks in both spatial and temporal space.

[55] compared various approaches inspired by NLP applied to fMRI analysis. In particular, they used various types of Transformer architecture and studied various tasks in order to find what NLP framework worked best. They pre-trained the models on broad neuroimaging data and they found that the pre-trained models generalised well in mental state decoding, outperforming models trained from scratch. They found that causal sequence modelling, where a short sequence is predicted from a previous longer sequence as input of a Transformer decoder, outperforms all the models they implemented.

[53] used the connection profile matrix of extracted mean time-series of known ROIs as an input instance of a simple Transformer encoder model. [53] define the connection profile Xi for node i as the corresponding row for each node in the edge weight matrix X. After the encoder, [53] use an orthonormal clustering readout (OCRead) function in order to project the learned latent representation in an orthonormal space, rendering the observations more discriminable. Finally, this latent representation in orthonormal space was used for downstream cases, like gender prediction for instance. The OCRead function made a big difference in the performance of the model.

In our project, I want to understand and model what brain areas have specific interactions that could lead to autistic brain functioning at resting-state. Such a model should find patterns of interaction into time series of different brain areas, which is not possible with [49] since it compresses spatial information via a CNN.

I worked around the topic of Transformers applied to rs-fMRI data. I used a well known functional parcellation (Craddock 200 (Craddock et al., 2012)) in order to extract time-series, each representing meaningful regions of the brain. Hence, I drew a comparison with textual data processing in NLP: a text consists of a set of words that have relationships between each other in order to give a sense to the actual text. Transformers (Vaswani et al., 2017) are designed to find these relationships. In rs-fMRI, a brain in a 4D scan sequence consists of a set of time-series that have relationships between each other and that may explain observations of mental or neurodevelopmental disorder behaviours and specificities. Thus, using Transformers as a way to find implicit patterns of relationships between brain time-series appeared natural.

Here, I wondered if bringing variables like gender, age, or comorbidities in a multitask fashion could improve the performance of a classification model of ASD based on a Transformer encoder architecture. Why? I had the hypothetical idea that a multitask model trained on ASD detection and on relevant confounds can help to better predict Autism by better adjusting the weights of the model. I designed various experiments to study this question. I introduce preliminary results that have not shown a clear difference between the models so far. I discuss potential improvements and new experiments to run in the discussion.

Methods

Data preparation:

- fMRI data was preprocessed with the C-PAC pipeline (version 0.4.0 for HBN and version 0.3.9 for ABIDE 1), with global signal correction and band-pass filtering (0.01-0.1Hz). A functional parcellation - Craddock 200 (Craddock et al., 2012) - was accomplished, extraction of mean time series in each region. The preprocessed data is open source for ABIDE 1 and, for HBN, available for researchers authorised to use the database. In total, 1102 time-series files are available in ABIDE 1, and 1096 were available in HBN databases.
- For ABIDE 1, quality control manual annotations were provided. I kept only the scans where at least one rater assessed those to be good quality scans. 1022 scans remained after this step. No quality control was provided with preprocessed HBN data. However, Functional MRI data went through automated quality control using tools like MRIQC to identify issues with coverage, motion, signal loss, etc. Low quality fMRI scans were excluded based on thresholds on these QC metrics.
- For the set of time-series of each participant, I checked that all the time-series had at least one value different from 0, that the time-series lengths were long enough (the minimal length of 100 frames was chosen arbitrarily to keep as many time series possible), and that there were 200 time-series (following the Craddock 200 parcellation). I ruled out participant data that did not match these conditions. Hence, I generated time-series harmonised in length (100 frames) across the whole dataset.
- In ABIDE 1, I chose the full sample 1 of the data-collecting site University of Michigan to be the independent test set. In HBN data, I took 10% of the dataset as the independent test set, where each site was represented and balanced according to their proportion in the full dataset.
- The rest was used to train the model in a 100-folds cross-validation (CV) fashion. The CV was stratified on the label ASD/non-ASD. I used the class StratifiedKFold from the module model_selection of the library scikit-learn in Python to generate our folds, with a random state set to 42.

- Add normalisation chosen
- Models 1,2,3 - ABIDE 1 and ASD only: 773 for training in 100-folds CV, 94 for testing
- Model 4 - ABIDE 1 + HBN, ASD and comorbidities: 1822 for training in 100-folds CV, 213 for testing
- Model 5 - HBN only for training: 975 for training in 100-folds CV, 108 for testing

	Training - Validation sets		**Testing set**	
	ABIDE 1	*HBN*	*ABIDE 1*	*HBN*
Model 1	773 (353 ASD)	/	94 (42 ASD)	/
Model 2	773 (353 ASD, 659 males)	/	94 (42 ASD, 70 males)	/
Model 3	773 (353 ASD, 277 aged between 10-15)	/	94 (42 ASD, 50 aged between 10-15)	/
Model 4	847 (413 ASD)	975 (67 ASD)	105 (51 ASD)	108 (6 ASD)
Model 5	/	975 (67 ASD)	/	108 (6 ASD)

Table X. Description of data used in training-validation sets (100-folds CV) and in testing set for each model

Models:

- I first trained a classification model of ASD that I can depict as a Transformer encoder followed by a fully connected layer block (add fig). In this experiment, I used data with participants with no diagnosis and with no other diagnosis than ASD.

- Next, I designed a simple multitask model, where the common part is a Transformer encoder and the separate parts are fully connected blocks that predict different targets (add fig).
- In this study, I introduce models implemented with a cross-entropy loss in the simple classification model :

$$H(P^{*} \mid P) \; = \sum_{i} P^{*}(y|x_{i}) log(P(y|x_{i}; \theta)$$

Where:

- H is the cross-entropy between the true class distribution P^{*} and the predicted class distribution P
- y is the class
- x_{i} is an input instance
- θ are the parameters of the model

- A weighted sum of cross-entropy computations was used as the loss of multitask models:

$$H_{1}(P_{1}^{*} \mid P_{1}) \; = \sum_{i} P_{1}^{*}(y|x_{i}) log(P_{1}(y|x_{i}; \theta_{1})$$

$$H_{2}(P_{2}^{*} \mid P_{2}) \; = \sum_{i} P_{2}^{*}(y|x_{i}) log(P_{2}(y|x_{i}; \theta_{2})$$

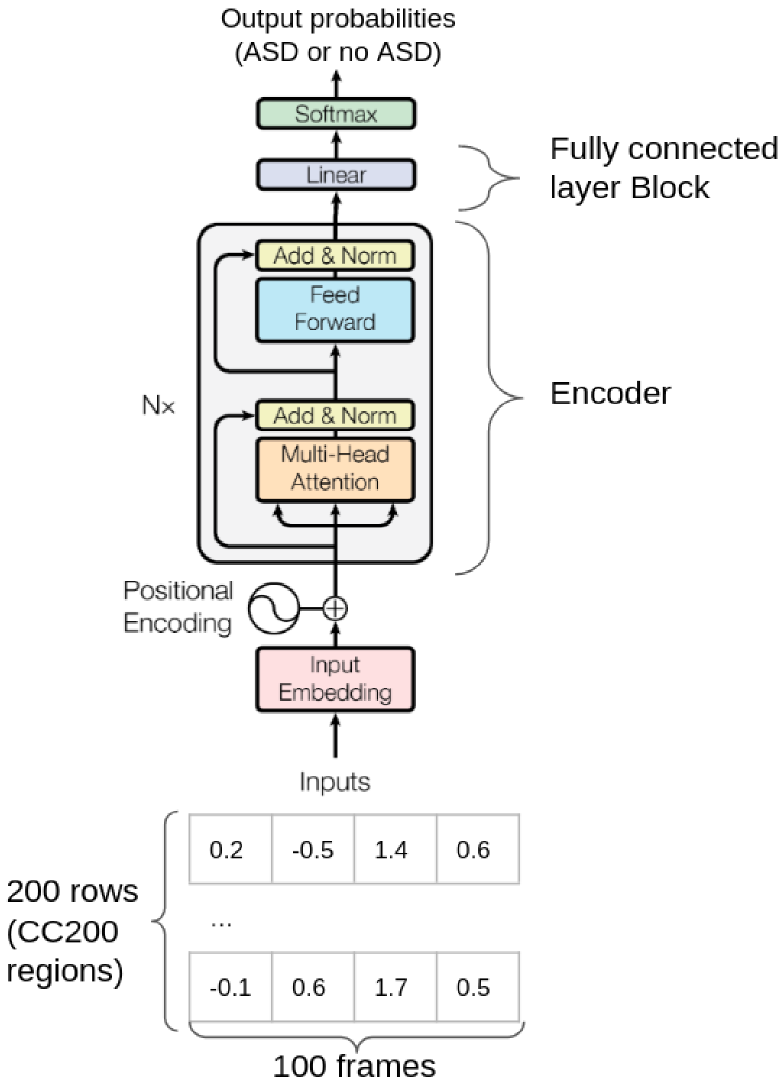
$$H_{sum} = \alpha H_{1}(P_{1}^{*}|P_{1}) \; + \; (1 - \alpha)H_{2}(P_{2}^{*} \mid P_{2})$$

Where:

- H_{1} is the cross-entropy between the true class distribution P_{1}^{*} and the predicted class distribution P_{1}
- H_{2} is the cross-entropy between the true class distribution P_{2}^{*} and the predicted class distribution P_{2}
- H_{sum} is the loss criterion of the model
- θ_{1} are the parameters of the encoder + the FC block 1 (see **Figure X**)

- θ_2 are the parameters of the encoder + the FC block 2 (see **Figure X**)
- y is the class
- x_i is an input instance
- α = 0,5 (arbitrary choice)

- I paid attention mainly to the accuracy and AUROC scores as metrics of performance, and to the mean of these two scores to assess the balance of a model.
- I used the Adam optimizer (Kingma et al., 2017) with a learning rate of 0,001 and a weight decay of 10^{-7}.
- I arbitrarily chose a space of dimension 16 to represent each time-series.
- The input embedder consists of one linear layer that projects the input data into an initial space of dimension 16; a positional encoding similar to the original by (Vaswani et al., 2017) is added to the input embeddings.
- The encoder block (see **Figure X**) consists of 3 encoder layers, and each encoder layer includes 4 multi-head attention modules.
- After the Encoder block, the representation of each input (of shape 200 by 16) is flattened and passed through a Fully Connected block of 3 layers. A softmax function is then applied to get the output probabilities.



0.2	-0.5	1.4	0.6
...			
-0.1	0.6	1.7	0.5

Figure X. Architecture of the models 1, 4 and 5. Inputs are the 200 extracted mean time series (CC200 atlas - Craddock et al., 2012) cropped to 100 non-null frames. The Encoder part is similar to a classical Transformer encoder (Vaswani et al., 2017) and returns an intermediate representation of the inputs. The Fully connected layer block processes the representation flattened and returns the probability to be diagnosed with ASD.

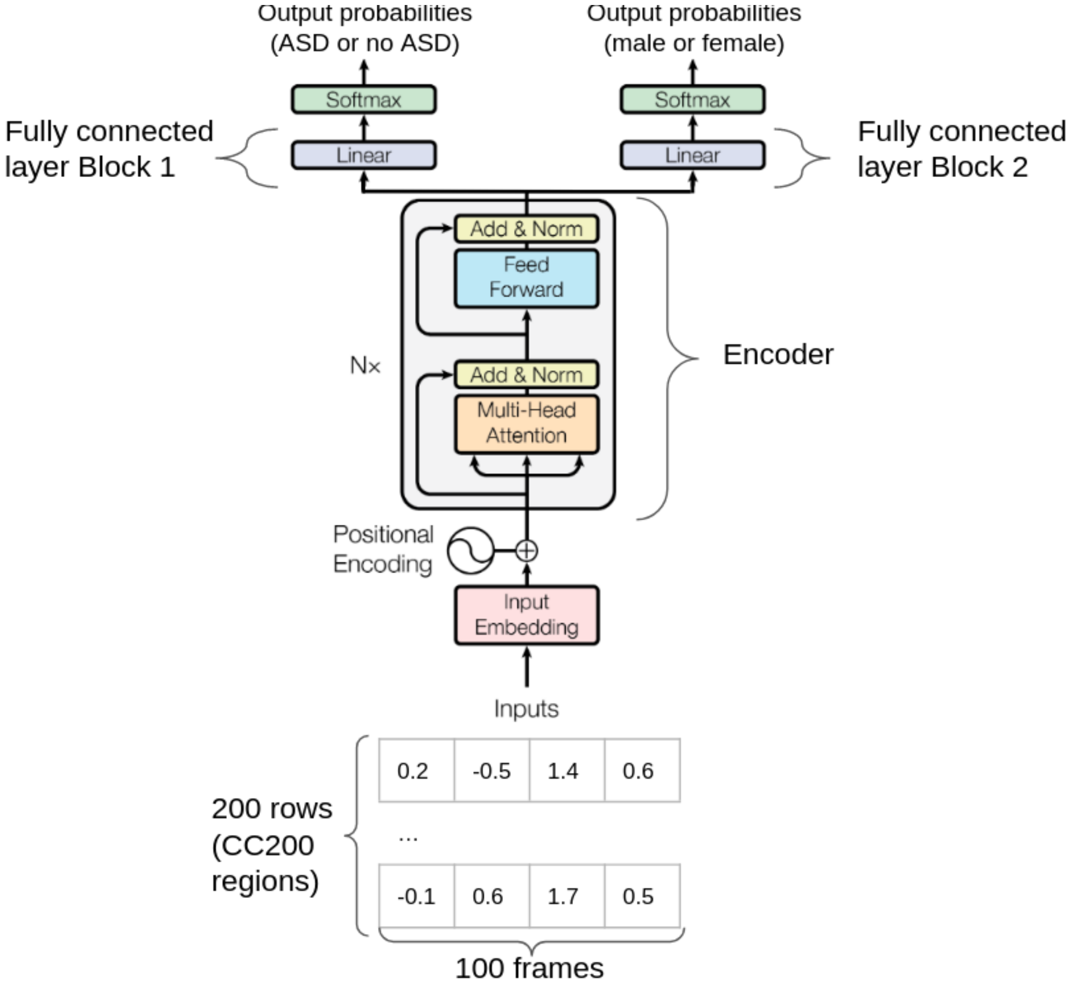


Figure X. Architecture of the multitask models 2 and 3. Inputs are the 200 extracted mean time series (CC200 atlas - Craddock et al., 2012) cropped to 100 non-null frames. The Encoder part is similar to a classical Transformer encoder (Vaswani et al., 2017) and returns an intermediate representation of the inputs. The Fully connected layer block 1 processes the representation flattened and returns the probability to be diagnosed with ASD. The Fully connected layer block 2 processes the same representation flattened and returns the probability to be a male (for model 2, or aged between 10-15 for model 3). The two FC blocks are optimised separately while the Encoder is optimised taking into account the two tasks.

Interpretations:

- I evaluated how the model fitted - underfitting or overfitting - the data by plotting train versus validation accuracy and AUROC score curves over the epochs.
- I defined a metric as being the mean between the accuracy score and the AUROC score, computed on the validation set at each epoch for each fold training. For each fold, I obtained one best metric over all the epochs that corresponded to the best model for this fold too. I compared the best metrics across all the folds between the models to assess if there was an improvement with the multitask framework or not. I could perform paired T-tests after verifying the related assumptions, Mann-Whitney U-tests otherwise .
- I ran the best models (i.e. for each model, the best among all the folds) on the independent test sets to see how each model replicates on new data.
- To visualise the representation made by each model, I computed the mean representation across all the subjects after the Transformer encoder part (see model on **Figure X**). Then, I performed a Pearson's correlation between all the 200 regions of the mean representation of each model and plotted the results of the highest correlations (absolute values greater than 0,6) as a chord diagram.
- In addition to the visualisations, I implemented the algorithm LIME (add ref). LIME is a model-agnostic method that interprets the decision locally of any model that predicts probabilities. For a given observation passed into the algorithm, it provides an approximation of the decision taken on each feature of the observations (e.g. greater or lower than a threshold for continuous features, one modality rather than another for categorical features) and classifies the importance of each feature for this particular observation. The classment and the approximated decision function may help to explain what the algorithm does.

Results

Add figure summarising models & data used to train

In this section, I introduce the outcomes of five models designed differently, that are represented on **Figure X**.

Model 1 consisted of a Transformer encoder followed by a Fully Connected block that returned a prediction of the binary target ASD / non-ASD. Model 1 was trained only on a subset of ABIDE 1, where all the participants having another diagnosis than ASD were excluded.

Model 4 and 5 had the same architecture as Model 1, but were trained on ABIDE 1 and HBN, and on HBN only respectively. It means that other diagnoses than ASD were included in the datasets used for Model 4 and 5.

Model 2 and 3 were intended to be multitask models. Their architecture consisted of a common part that is a Transformer encoder, followed by two separate parts being Fully Connected layer blocks that predicted the binary target ASD / non-ASD and another binary target (Male/Female for Model 2 and Age between 10-15/or not for Model 3) respectively. Data used to train Model 2 and Model 3 was the same as data used for Model 1 (ASD only, excluding comorbidities). (give dataset sizes)

For all the models, I used the Adam optimizer (add ref) with a learning rate of 10^{-3} and a weight decay of 10^{-7}.

I trained the models on 50 epochs and performed a validation at every epoch to get the accuracy and AUROC scores on the validation set of each fold. As a reminder, I performed a 100-folds stratified cross-validation. Hence, each model was trained 100 times.

Fig. X. shows the evolution of accuracy and AUROC scores over the epochs for every model. It aggregates over repeated values (each fold) to show the mean and 95% confidence interval at each epoch on training sets and validation sets respectively. From Fig. X., it is noticeable that the models overfitted: they converged too fast on training sets while stagnating to scores close to randomness on validation sets. Model 4 and Model 5 seem to have higher scores on the validation sets. However, these results may be explained by the fact that HBN data is not balanced between ASD and non-ASD participants. (give ratio)

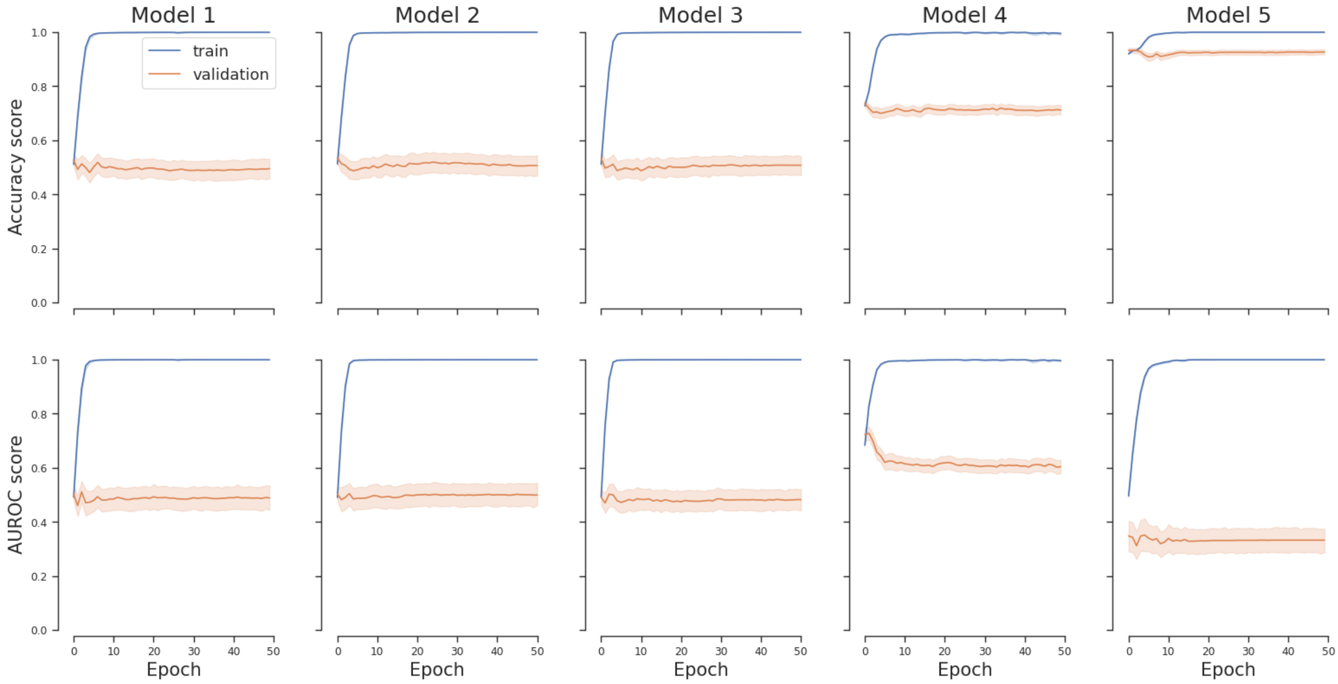
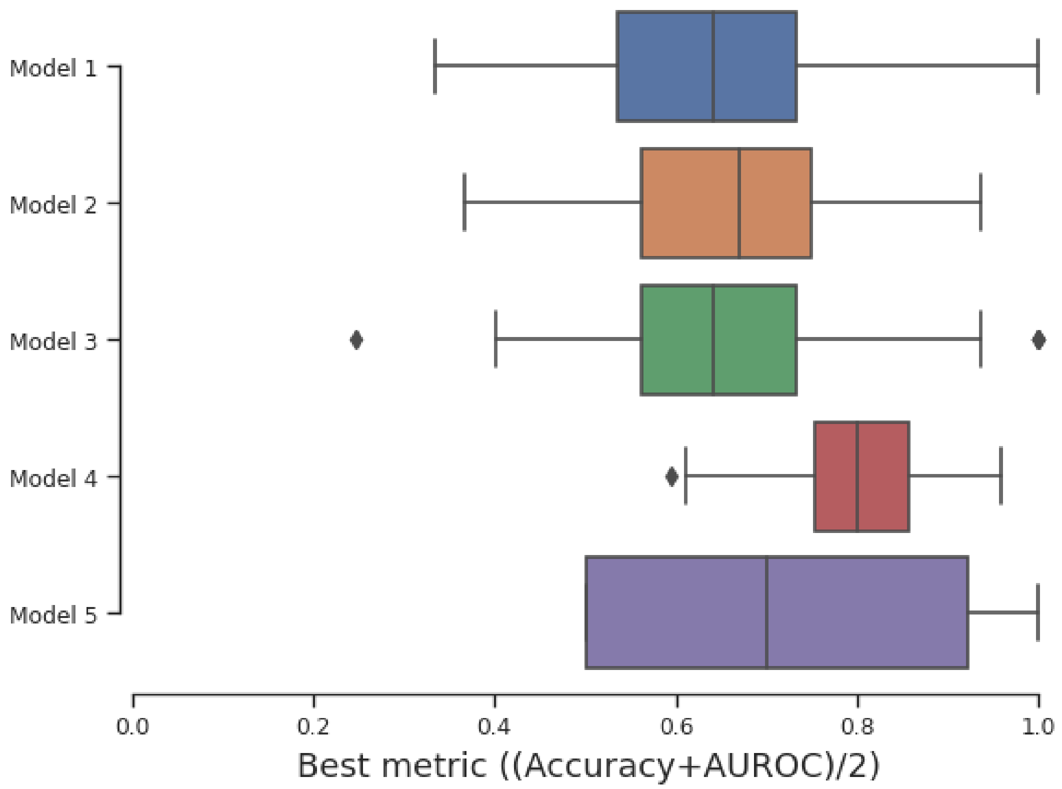


Fig. X. shows a comparison of the best metrics (computed on validation sets as (accuracy+AUROC)/2) on all the folds between the five models. The boxplots represent the distribution of the best metrics of all the folds for each model. From Fig. X., it appears that there is no clear difference between Model 1, 2 and 3 best metrics, that have respectively a mean of m_1=0.644, m_2=0.660, m_3=0.644, and a standard-deviation of s_1=0.151, s_2=0.148, s_3=0.144. However, Model 4 and 5 appear different from the others, and have respectively a mean of m_4=0.794, m_5=, and a standard-deviation of s_4=0.078, s_5=.



To confirm these observations, I performed statistical tests. Since I was interested in the difference between two conditions (simple versus multitask model, ASD only versus ASD + other diagnoses, multitask gender versus multitask age between 10 and 15), I aimed at performing a paired T-test to be able to compare the observations over the folds in pairs.

Verifying the assumption of normality on the difference between the pairs of samples, all the differences passed the Shapiro-Wilk test, except for the Model 4 and Model 5 compared to Model 1 (see Appendix). Thus, I performed an alternative non-parametric test - the Mann-Whitney U-test - on these pairs. The results, provided in **Table X.** show that there is no significant difference between Model 1, Model 2 and Model 3 (p_{m1-m2}=0.435, p_{m1-m3}=0.984, p_{m2-m3}=0.199) but there is a significant difference between Model 1 and Model 4 ($p_{m1\_m4}=0.0158 < 5\%$), and a difference between Model 1 and Model 5 ($p_{m1\_m5} < 10^{-13}$).

Paired T-test	**T**	**dof**	**Alterna-tive**	**p_val**	**CI95%**	**cohen-d**	**BF10**	**power**
Model 1 - Model 2	0.783	99	2-sided	0.435	[-0.02, 0.05]	0.103	0.149	0.176
Model 1 - Model 3	-0.0206	99	2-sided	0.984	[-0.04, 0.04]	0.003	0.111	0.050
Model 2 - Model 3	1.293	99	2-sided	0.199	[-0.01, 0.04]	0.109	0.248	0.189
MW U-test	**U-val**		**Alternative**	**p_val**	**RBC**	**CLES**		
Model 1 - Model 4	8081.0		2-sided	5,17.10^{-14}	-0.616	0.808		
Model 1 - Model 5	5981.5		2-sided	0.0158	-0.193	0.598		

Table X. Statistical tests on the best metrics of the validation fold results between the models.

Inference on test set:

	Model 1	**Model 2**	**Model 3**	**Model 4**	**Model 5**
Accuracy	50%	52.1%	47.9%	67.1%	**85.2%**
AUROC	0.581	0.551	**0.599**	**0.599**	0.368
(Acc. + AUROC)/2	0,541	0,536	0,539	**0,635**	0,61
Specificity	0.442	0.808	0.327	0.776	**0.902**
Sensitivity	0.571	0.167	**0.667**	0.386	0.0

Table X introduces the results obtained on the test sets for each model. Accuracy, AUROC, Specificity and Sensitivity are provided. Model 5 has the highest accuracy (85,2%), while Model 3 and 4 show the highest AUROC score (0.599). The highest specificity (0.902) is achieved by Model 5 while the highest sensitivity (0.667) by Model 3. It can also be observed that there is a drop in the metric (Accuracy+AUROC)/2. The best metric is reached by Model 4 (0,635).

Data description is provided on **table X**. It can be seen that the datasets that were used to build and test Models 4 and 5 are much more imbalanced (~26% of ASD for Model 4; ~6% of ASD for Model 5) than the datasets used to build Models 1,2,3 (~45% of ASD). Looking at **Figure X** and at the statistical test results on **Table X**, it would seem that Model 4 works better than Models 1,2,3. However, since the data is imbalanced, the accuracy metric is biassed and tends to be higher than for balanced data. Indeed, it can be observed on Table X that the results of Model 5 on the test set are the lowest sensitivity (0) but the highest accuracy (85,2%). It means that no autistic people are well predicted but a few other non autistic people are predicted ASD. Thus, we can not deduce that Model 4 is the best model because the metrics are not comparable between Model 1,2,3, Model 4, and Model 5.

Visualisation - Interpretation

All along the Transformer encoder, the time series of the 200 regions of the Craddock atlas (add ref) have been transformed and represented in a different way than initially. Each time series is represented by 16 features at the end of the Transformer encoder part of the model, just before the Fully Connected Block. I computed the Pearson correlations between the regions under this implicit representation by 16 features.

I translated the 200 regions into the Yeo network atlas that is simpler to represent. I plotted chord diagrams and brain representations of the correlations just computed in the Yeo referential. Results as chord diagrams are introduced in **Figure X.** Results as “glass brains” are drawn in Figure X, Figure X, Figure X, Figure X and Figure X.

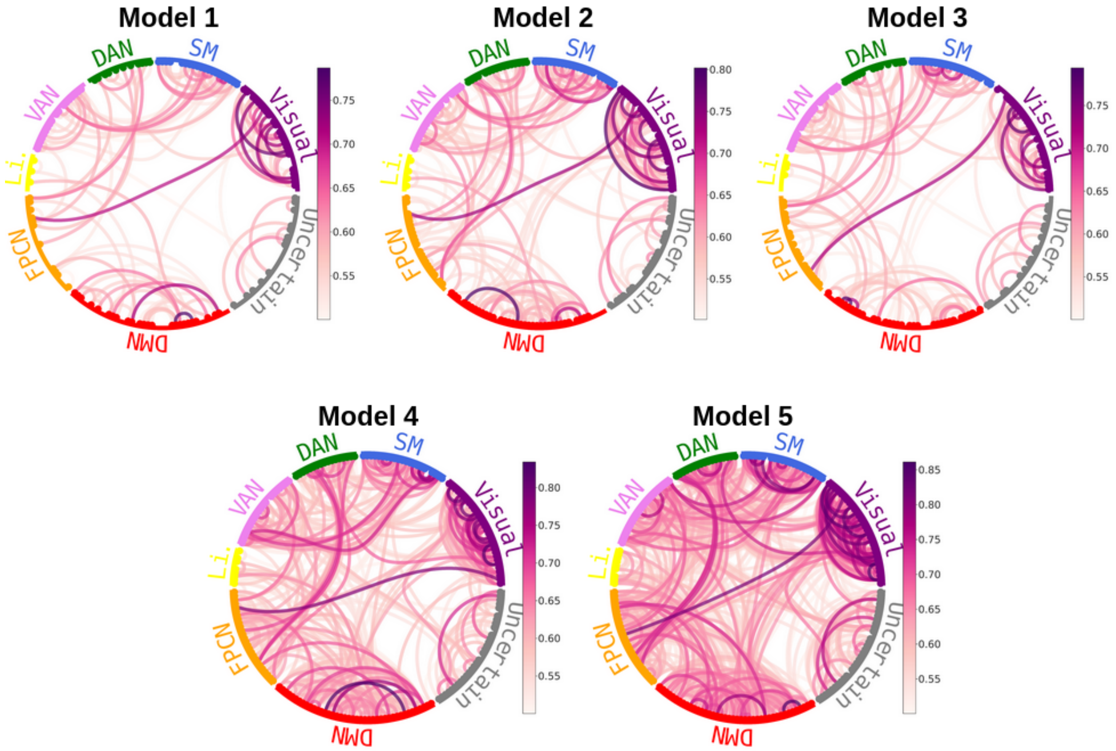
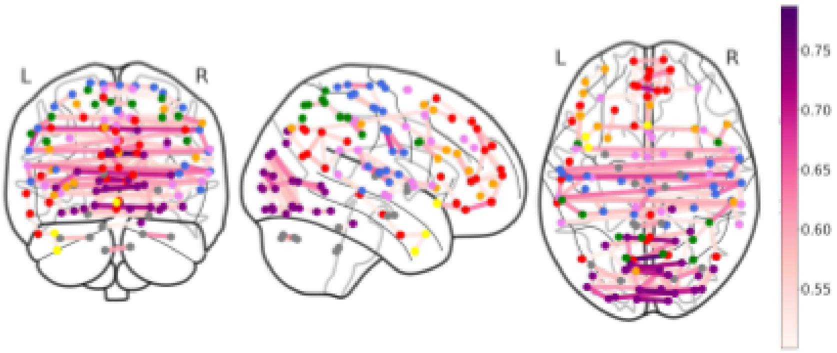
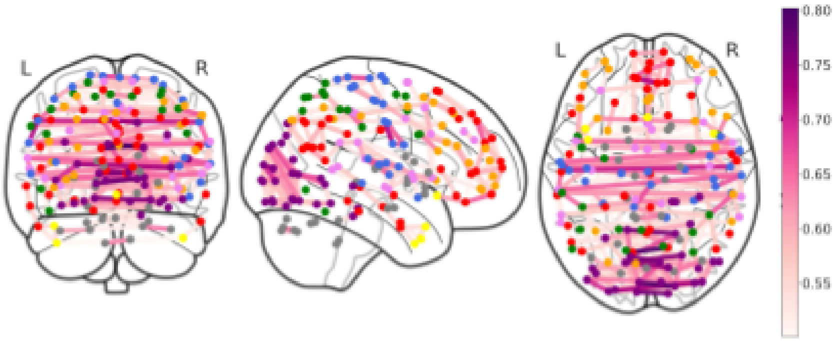


Figure X. Chord representations of the correlations between the implicit representations of 200 regions by 16 features after the last layer of the Transformer encoder part of each model. The regions were translated into the Yeo Network (add ref) to simplify the plot: “Visual” (purple) is for the Visual Network; “SM” (blue) is for the Somatomotor Network; “DAN” (green) is for the Dorsal Attention Network; “VAN” (violet) is for the Ventral Attention Network; “Li.” (yellow) is for the Limbic Network; “FPCN” (orange) is for the Frontoparietal Control Network; “DMN” (red) is for the Default Mode Network; “Uncertain” (grey) is for regions in the CC200 atlas that did not match any region in the Yeo atlas. Inside the circles, I represented as pink lines with varying intensity levels the correlations between 0.5 and 1 (no negative correlation was found to have an absolute value greater than 0.5). Add ref

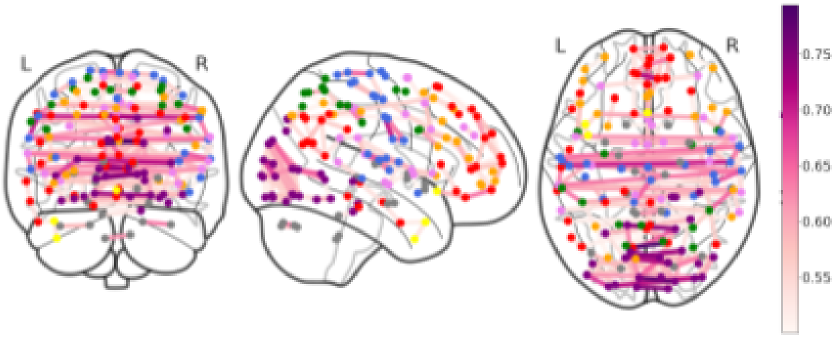
Model 1



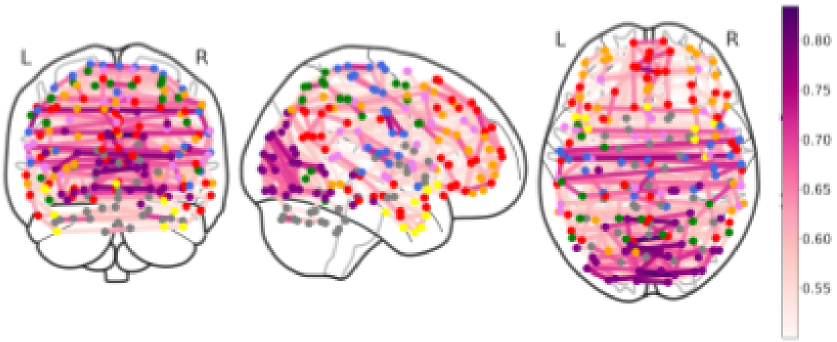
Model 2



Model 3



Model 4



Model 5

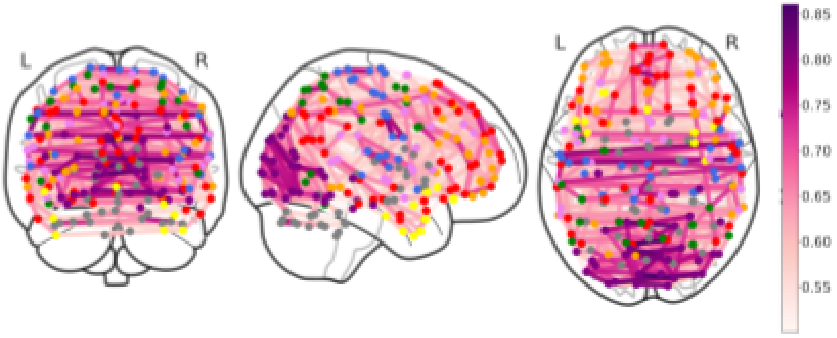
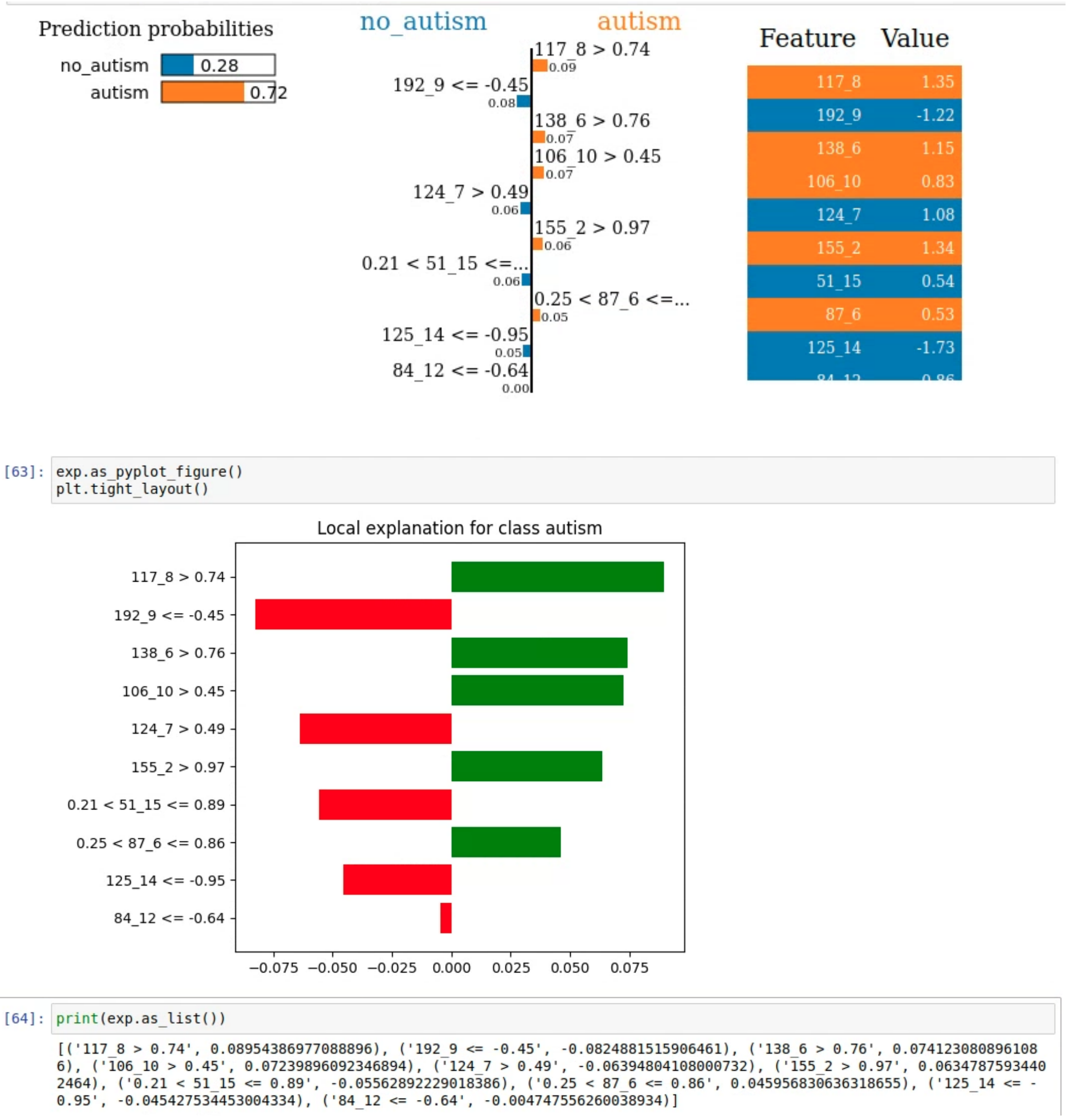


Figure X: Brain representations of the correlations between the implicit representations of 200 regions by 16 features after the last layer of the Transformer encoder part of each model. The regions were translated into the Yeo Network (add ref) to simplify the plot. Each CC200 region is represented by dot points and coloured in function of their correspondence with the Yeo atlas: Purple is for the Visual Network; Blue is for the Somatomotor Network; Green is for the Dorsal Attention Network; Violet is for the Ventral Attention Network; Yellow is for the Limbic Network; Orange is for the Frontoparietal Control Network; Red is for the Default Mode Network; Grey is for regions in the CC200 atlas that did not match any region in the Yeo atlas. I represented as pink lines with varying intensity levels the correlations between 0.5 and 1 (no negative correlation was found to have an absolute value greater than 0.5).

In addition to the visualisations, I implemented the algorithm LIME (add ref). LIME is a model-agnostic method that interprets the decision locally of any model that predicts probabilities. For a given observation passed into the algorithm, it provides an approximation of the decision taken on each feature of the observations (e.g. greater or lower than a threshold for continuous features, one modality rather than another for categorical features) and classifies the importance of each feature for this particular observation. The classment and the approximated decision function may help to explain what the algorithm does. I did not lead a full analysis on all the participants, for instance by separating people with a diagnosis and people with another diagnosis, or by segmenting the patients on age, gender, comorbidities. Many parameters should be optimised before being able to run LIME properly, including the parameters of the input data (e.g. normalisation function), of each Transformer model, and of LIME (e.g. the number of random observations to project in the space). This could be done in future work to better explain and interpret what features are important for each model to predict Autism (or age or gender in multi-task models). I report a view of the results on a participant having a diagnosis of Autism (and no other diagnosis according to data provided) on **Figure X.** It displays the 10 most important features that serve to predict the label “Autism” for this participant, and it explains why in terms of thresholds on each continuous variable.



Feature	Value
117_8	1.35
192_9	-1.22
138_6	1.15
106_10	0.83
124_7	1.08
155_2	1.34
51_15	0.54
87_6	0.53
125_14	-1.73
84_12	0.86

In [63]:
```
exp.as_pyplot_figure()
plt.tight_layout()
```

In [64]:
```
print(exp.as_list())
```

```
[('117_8 > 0.74', 0.08954386977088896), ('192_9 <= -0.45', -0.0824881515906461), ('138_6 > 0.76', 0.074123080896108
6), ('106_10 > 0.45', 0.07239896092346894), ('124_7 > 0.49', -0.06394804108000732), ('155_2 > 0.97', 0.0634787593440
2464), ('0.21 < 51_15 <= 0.89', -0.05562892229018386), ('0.25 < 87_6 <= 0.86', 0.045956830636318655), ('125_14 <= -
0.95', -0.045427534453004334), ('84_12 <= -0.64', -0.004747556260038934)]
```

Figure X. LIME algorithm run on one autistic participant data: it explains the decisions of the Fully Connected Layer of the Model considered (e.g. Model 1). The feature names are provided with numbers. For instance, “117_8” is for region 117 of CC200 atlas and encoding feature number 8 (among the 16 ones) after the Transformer encoder part of the model. For this feature, the value is strictly greater than 0.74 that is the threshold found by the LIME algorithm, meaning that it is consistent with a prediction of Autism for LIME. The two types of graphs are displayed in a Python 3 Jupyter Notebook.

Discussion

This study introduced the implementation of the Transformer algorithm (Vaswani et al., 2017) applied to brain time-series of rs-fMRI data in order to build a classification model of Autism. Several approaches were developed: Model 1 was a simple binary classifier on data that included people with a diagnosis of Autism and no other diagnosis, and people without any diagnosis (according to clinical data provided); Model 2 and 3 were built on the same data but a multitask approach was implemented, predicting Autism and gender for Model 2, and predicting Autism and people aged between 10-15 for Model 3; Model 4 and 5 were simple binary classifiers of Autism like Model 1, but included patients with other diagnoses than Autism, like ADHD, anxiety, depression, etc.
The main interest that led to preparing this study was to explore if a Transformer architecture can help in better representing brain time series and to better model brain activity of autistic people. Implementing multitask models that included gender prediction or age prediction was a way to better model what has been observed in the literature about ASD on gender difference (add ref) and on the evolution of the diagnosis with age (add ref). Hence, it was a way to explore if bringing this information to the optimisation of the model could improve the performance of Autism prediction. Adding data with other diagnoses was also a way to be more realistic in the modelling, since comorbidities are widely observed in autistic people (add refs) and since overlaps between psychiatric disorders have been observed in genetics, potential neuroimaging markers, and symptoms (add refs).

The results show that there is no model that can be considered as the best model to predict ASD. It was shown that the imbalance between the datasets added a bias to the accuracy metric that made the comparison between certain models unfair, thus, impossible. Future work could include a strategy to calibrate models trained on imbalanced datasets in order to build comparable results.
Models 1,2,3 were comparable, and it was interesting to see that the multitask approach did not seem to have an effect on the global performance of the model. However, and interestingly, specificities and sensitivities were very

different between these models (sp1 = 0.442, sp2 = 0.808, sp3 = 0.327; se1 = 0.571, se2 = 0.167, se3 = 0.667). These results show that the multitask approaches actually had an effect on the models which was expected. It also shows that the multitask models did not converge to a model with a stabler performance like for Model 1 (i.e. there was a greater gap between specificity and sensitivity of Model 2 or 3 than between Model 1 ones), nor do they converge to a higher performance that we would have hoped.

Various factors may explain these points. Multitask models were trained with a loss that was the sum of each task's loss, arbitrarily chosen to be balanced (alpha=0,5). Many other experiments could be led to optimise this alpha. In addition, there may not be a linear relationship between the two losses and looking for more complex and realistic ways to mix the two task's losses could be done in future work. In addition, the learning rate was fixed and common for the two tasks. It would be smarter to have two learning rates and to optimise those for each task separately.

A 100-folds cross-validation frame was used to train all the models. Even if it added implementation and technical challenges compared to a more traditional train-validation-test frame (a hundred times more experiments to launch), it also allowed more robust statistics in estimating the metrics of performance, that was a clear good point in this study methodology. A drawback was that less parameter optimisation was performed. And many parameters that are very likely to have a great impact on the results were not explored like the global architecture of the model (e.g. number of encoder layers, of fully connected layers, of self-attention heads, positional encoder function) and the training parameters (e.g. the learning rate, weight decay, alpha in multitask cases, loss). Future work should dedicate a longer time in optimising such parameters.

In particular, the size of the representation of time-series may actually be crucial in how the model performs. We chose it to be 16 here. In other words, extracted and cropped time series of size 100 frames were reduced to a size of 16 features by the model. The idea was to compress the information at the maximum. Nevertheless, this dimension may be too low (or too big) to represent the information present in 100 frames, as well as the relationships existing with other regions at resting-state. The size of the representation should be better adjusted.

In addition, we actually do not know if 100 frames is enough to represent long and short scale networks in the brain at resting state. Plus, even if about one or two thousands observations were used, this scale may be insufficient to feed a Transformer well. Indeed, Transformer algorithms need big datasets to converge well. In the context of ASD, that is characterised by a wide spectrum of behaviours and sensitivities, added to the high inter-individual variability of the brains, this scale may not be appropriate and much more data may be needed.

Regarding the type of data itself, many points can alter the success of the methodological approach suggested in this study. Our quality check method on the ABIDE 1 dataset was quite inclusive, there may be a few scans with artefacts in the dataset. Heavy preprocessing pipelines, including a registration step, may distort the information too much in the presence of mental conditions. Is resting-state fMRI data alone sufficient to build a robust model predicting and explaining Autism? It is very likely that it is not, and combining this modality with other modalities in neuroimaging (e.g. sMRI, PET, EEG, MEG) and in genetics may be more relevant to best model Autism. Other confounds like scanning acquisition parameters and the socio-environment of each participant may influence the outcomes too. It would be interesting to integrate such data to the training framework.

Finally, the interpretation method used is limited. Interpretability of Transformers is good when there is one layer and one self-attention head. But in a realistic case, there are many layers and self-attention heads like in our study. The representation after the last encoder layer was studied and plotted to extract relationships between regions. However, no consideration of the other intermediate representations was provided. In addition, implementing LIME seemed interesting to interpret the most important regions to predict Autism for the Fully Connected block. However, it is limited to one subject at a time and no generalisation to the whole dataset was performed. It is challenging to implement but could be interesting to perform in future work. The Fully Connected block does not offer many ways to be explained or interpreted.

Replacing this part of the model by other algorithms that are more interpretable may be interesting too (e.g. linear regression, SVM, decision tree).

Conclusion

This study introduced the implementation of the Transformer algorithm applied to brain time-series of rs-fMRI data in order to build a classification model of Autism. Several approaches were developed, including multitasking, but no clear difference between the various models was observed in the global performance. A 100-folds cross-validation frame was used to train all the models that allowed robust statistics in estimating the metrics of performance. An interpretation and explanation of each model understanding of the data and decisions to predict ASD was provided. In the discussion, many points were raised that should be improved in future work, including model and training parameter optimisation, and mixing rs-fMRI with other data modalities for instance.

Appendix:

Shapiro-Wilk test of normality:

```
        W     pval     normal
model
diff_gender_asd     0.987893     0.500361     True
diff_age_asd     0.986557     0.408545     True
diff_age_gender     0.990540     0.708221     True
diff_comorbs_asd     0.974202     0.046736     False
diff_comorbs_hbn_asd     0.970325     0.023409     False
```