


# Exoformer: Accelerating Bayesian atmospheric retrievals with transformer neural networks

L. Pagliaro<sup>1,\*</sup> , T. Zingales<sup>1,2</sup> , G. Piotto<sup>1,2</sup> , I. Giovannini<sup>1,2</sup> , and G. Mantovan<sup>3,2</sup> 

<sup>1</sup> Dipartimento di Fisica e Astronomia, Università degli Studi di Padova, Vicolo dell'Osservatorio 3, 35122 Padova, Italy

<sup>2</sup> INAF, Osservatorio Astronomico di Padova, Vicolo dell'Osservatorio 5, 35122 Padova, Italy

<sup>3</sup> Centro di Ateneo di Studi e Attività Spaziali "G. Colombo" – Università degli Studi di Padova, Via Venezia 15, 35131 Padova, Italy

Received 25 November 2025 / Accepted 9 March 2026

## ABSTRACT

Computationally expensive and time-consuming Bayesian atmospheric retrievals pose a significant bottleneck for the rapid analysis of high-quality exoplanetary spectra from present and next generation space telescopes, such as JWST and Ariel. As these missions demand more complex atmospheric models to fully characterize the spectral features they uncover, they will benefit from data-driven analysis techniques such as machine and deep learning. We introduce and detail a novel approach that uses a transformer-based neural network (Exoformer) to rapidly generate informative prior distributions for atmospheric transmission spectra of hot Jupiters. We demonstrate the effectiveness of Exoformer using both simulated observations and real JWST data of WASP-39b and WASP-17b within the TauREx retrieval framework, leveraging the nested sampling algorithm. By replacing standard uniform priors with Exoformer-derived informative priors, our method accelerates nested-sampling retrievals by factor of 3–8 in the tested cases, while preserving the retrieved parameters and best-fit spectra. Crucially, we ensure that the retrieved parameters and the best-fit models remain consistent with results from classical methods. Furthermore, we confirm the statistical consistency of the two retrieval approaches by comparing their log-Bayesian evidence, obtaining absolute values of each Bayes factor  $|\Delta \log Z| < 5$ , i.e., with no strong preference following common scales for either model. This hybrid approach significantly enhances the efficiency of atmospheric retrieval tools without compromising their accuracy, paving the way for more rapid analysis of complex exoplanetary spectra and enabling the integration of more realistic atmospheric models.

**Key words.** methods: data analysis – methods: numerical – methods: statistical – planets and satellites: atmospheres – planets and satellites: fundamental parameters

## 1. Introduction

The advent of the James Webb Space Telescope (JWST) (Gardner et al. 2006) and forthcoming Ariel (Tinetti et al. 2022) space missions opens the door to unprecedented spectroscopic observations of exoplanetary atmospheres. The study of atmospheric compositions plays a crucial role in understanding how planets form and evolve, as well as in detecting molecular signatures of life. Many tools based on Bayesian statistics have been developed to retrieve molecular abundances, temperatures, and many other atmospheric parameters from spectroscopic observations (e.g., TauREx (Al-Refaie et al. 2021), NEMESIS (Irwin et al. 2008), and petitRADTRANS (Mollière et al. 2019); a more complete list can be found in MacDonald & Batalha (2023)). The classical approach involves a Bayesian framework to obtain the posterior distributions of atmospheric parameters given an observed spectrum. These tools prove to be effective with low-resolution observations of the HST and Spitzer telescopes. However, high-quality data from JWST and Ariel show a completely new set of spectral features that can be described only with more complex atmospheric models (Rocchetto et al. 2016), considering for example atmospheres as multidimensional structures where physical phenomena such as convection and chemical disequilibrium occur. The resulting increase in parameters

describing the atmospheric model yields a strong computational bottleneck and challenges in achieving convergence to a solution.

In recent years, an increasing number of machine learning and deep learning algorithms have been implemented in the exoplanetary field, ranging from transit detection in light curves (e.g., McCauliff et al. (2015), Shallue & Vanderburg (2018)) to atmospheric characterization (e.g., Yip et al. (2021), Himes et al. (2022), Vasist et al. (2023)). These algorithms are based on a data-driven approach, which means that they can automatically learn to solve a given task by iteratively extracting multiple features from a large dataset (Janiesch et al. 2021). Machine learning encompasses a broad range of algorithms derived from different learning paradigms (e.g., decision trees (Quinlan 1986), support vector machines (Cortes & Vapnik 1995), and clustering (Kaufman 2005)). Deep learning by contrast relies on the artificial neuron – a mathematical function that employs nonlinear transformations through weighted inputs to process and learn from data – and its variants (e.g., convolution and attention). Individual neurons are typically grouped into distinct layers that are then repeated and connected to form more complex architectures, called deep neural networks. Deep neural networks are used extensively in complex tasks (such as images, texts, and sequential data processing), as they outperform machine learning in extracting hidden features and patterns from domains with large, high-dimensional data (LeCun et al. 2015; Janiesch et al. 2021).

\* Corresponding author: [leonardo.pagliaro@phd.unipd.it](mailto:leonardo.pagliaro@phd.unipd.it)

Bayesian retrievals require millions of atmospheric forward models per single observation, becoming extremely slow and computationally intensive when complex atmospheric models with a large number of parameters are taken into account. Therefore, with the thousands of spectra expected from new telescopes, the efficiency of repeated analysis will be significantly impacted. The integration of deep learning emerges as a promising solution to address the computational challenges of high-dimensional Bayesian atmospheric retrievals. There are multiple pathways through which we can achieve this goal: replacing the Bayesian tool (e.g., Zingales & Waldmann (2018)) to directly provide parameter posterior distributions, substituting the radiative transfer code with a surrogate model (e.g., Himes et al. (2022)), or generating informative priors (e.g., Hayes et al. (2020)) in place of uniform priors.

In our work, we adopted a state-of-the-art deep-learning architecture, the transformer, whose architecture was proposed by Vaswani et al. (2017) in the context of human language modeling to overcome the limitations of convolutional neural networks (CNN; (LeCun et al. 1989)) and long-short term memory (LSTM; Hochreiter & Schmidhuber (1997)) networks. Building on the success of transformers in predicting stellar parameters – as demonstrated in recent studies (e.g., Pan et al. (2024), Zhang et al. (2024)) – we propose a transformer-based approach for analyzing spectroscopic data of exoplanetary atmospheres. Just as stellar spectra exhibit interconnected emission and absorption features across the spectral domain, a transformer architecture is well suited to capture these complex relationships within exoplanetary atmospheric data. We developed a tool that can retrieve the approximated posterior distributions of six atmospheric parameters using the Monte Carlo (MC) dropout technique (Gal & Ghahramani 2016). In Section 2, we describe the transformer architecture and its fundamental operation, self-attention. In Section 3 we introduce Exoformer, our transformer-based tool, and describe its structure. In Section 4, we describe the training process of Exoformer and how uncertainties are estimated in the neural network. In Sections 5 and 6, we show the results of applying Exoformer to simulated and real JWST transmission spectra.

## 2. Transformer neural networks

Transformers are a type of neural network designed to learn useful representations of sequential data through a mechanism called self-attention. In fact, unlike other architectures such as CNN and LSTM, transformers have the ability to capture long-range dependencies and correlations in sequences. In the following paragraphs, we describe the most important components of a transformer algorithm.

### 2.1. Self-attention mechanism

Self-attention is the core mechanism of the transformer architecture: it allows the model to relate each element of a sequence to all the other elements of the same sequence. In general, the  $i$ -th output  $\mathbf{a}_i \in \mathbb{R}^D$  (where  $D$  is the embedding dimension) of the self-attention operation is a weighted sum of the  $N$  inputs  $\mathbf{x}_1, \dots, \mathbf{x}_N$ , with  $\mathbf{x}_j \in \mathbb{R}^D$ :

$$\mathbf{a}_i = \sum_{j=1}^N A_{ij} \mathbf{x}_j, \quad (1)$$

where  $\mathbf{A} \in \mathbb{R}^{N \times N}$  is the attention matrix, whose elements are normalized between  $[0, 1]$  and their sum equals to 1.

Vaswani et al. (2017) applied the self-attention mechanism to deep learning by introducing a set of learnable matrices to compute the attention matrix and the attention output. Given the sequence of  $N$  inputs  $\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{R}^D$ , each vector is first linearly transformed by three distinct learnable matrices  $\mathbf{W} \in \mathbb{R}^{D \times D}$ :

$$\mathbf{q}_i = \mathbf{x}_i \mathbf{W}_Q, \quad (2)$$

$$\mathbf{k}_i = \mathbf{x}_i \mathbf{W}_K, \quad (3)$$

$$\mathbf{v}_i = \mathbf{x}_i \mathbf{W}_V. \quad (4)$$

The three resulting vectors  $\mathbf{q}_i, \mathbf{k}_i, \mathbf{v}_i \in \mathbb{R}^D$  (called query, key, and value, respectively) are then used to obtain the  $A_{ij}$  element of the attention matrix through

$$A_{ij} = \text{softmax}(\mathbf{q}_i \mathbf{k}_j^T) = \frac{\exp(\mathbf{q}_i \mathbf{k}_j^T)}{\sum_{n=1}^N \exp(\mathbf{q}_i \mathbf{k}_n^T)} \quad (5)$$

and the self-attention output given by

$$\mathbf{a}_i = \sum_{j=1}^N A_{ij} \mathbf{v}_j. \quad (6)$$

In Eq. (5) every query  $\mathbf{q}_i$  is compared to all  $N$  keys  $\mathbf{k}_j$  to find the combinations with the highest correlation by using the geometric properties of the dot product. Equation (6) instead creates a new representation of the value  $\mathbf{v}_i$ , where the information about the pairs of vectors with the highest attention scores is stored.

Stacking the  $N$  input vectors into a column matrix  $\mathbf{X} \in \mathbb{R}^{N \times D}$ , we can rewrite Eq. (6) in a simpler way:

$$\mathbf{a} = \text{softmax}(\mathbf{Q} \mathbf{K}^T) \mathbf{V}, \quad (7)$$

where  $\mathbf{Q} \in \mathbb{R}^{N \times D}$ ,  $\mathbf{K} \in \mathbb{R}^{N \times D}$ ,  $\mathbf{V} \in \mathbb{R}^{N \times D}$  are the query, key, and value matrices for the entire sequence.

Moreover, Vaswani et al. (2017) improved the self-attention numerical stability by scaling the argument of the softmax with the square root of the embedding dimension:

$$\mathbf{a} = \text{softmax}\left(\frac{\mathbf{Q} \mathbf{K}^T}{\sqrt{D}}\right) \mathbf{V}. \quad (8)$$

### 2.2. Multi-head self-attention

Multiple self-attention mechanisms are typically applied in parallel to capture further relationships between the input vectors of a sequence. This approach introduced by Vaswani et al. (2017) is called multi-head self-attention.

Consider  $H$  self-attention heads  $\mathbf{a}_h$ , each with a different set of queries  $\mathbf{Q}_h$ , keys  $\mathbf{K}_h$ , and values  $\mathbf{V}_h$ : every head acts on a fraction  $D/H$  of the input embedding, allowing the extraction of correlations from different parts of the embedding. The output of all self-attention heads are then concatenated, and a final linear transformation  $\mathbf{W}_o \in \mathbb{R}^{D \times D}$  is applied:

$$\mathbf{mhsa} = \text{Concat}(\mathbf{a}_1, \dots, \mathbf{a}_h) \mathbf{W}_o. \quad (9)$$

This multiheaded attention mechanism is introduced to further enhance the performance of single self-attention by capturing deeper and more specific correlations in input data.

### 2.3. Positional encoding

By definition, the self-attention mechanism is invariant under permutation of the input sequence  $\mathbf{X}$ : by permuting the columns of  $\mathbf{X}$ , we permute all its representations across the mechanism in the same way. However, positional information is a key aspect in spectroscopy, because it is related to the different absorption or emission features of molecules.

To fix the problem, we can add a vector  $\mathbf{p}_i \in \mathbb{R}^D$  to the column vector  $\mathbf{x}_i$ ,

$$\mathbf{x}_i \rightarrow \mathbf{x}_i + \mathbf{p}_i, \quad (10)$$

where  $\mathbf{p}_i$  can be a customized mathematical function or a parameter learned during training. The vector  $\mathbf{p}_i$  encodes the position information inside the embedding, before self-attention is applied: for this reason, we refer to this mechanism as positional encoding.

Vaswani et al. (2017) presented a positional encoding based on a combination of sine and cosine functions:

$$\mathbf{p}_i(2j+1) = \cos\left(\frac{i}{10000^{2j/D}}\right), \quad (11)$$

$$\mathbf{p}_i(2j) = \sin\left(\frac{i}{10000^{2j/D}}\right), \quad (12)$$

where  $j$  indicates the elements of the vector. This representation ensures a unique position is associated with each element of the input sequence, while maintaining the output in a fixed range.

### 2.4. Encoder block

Sections 2.2 and 2.3 introduced the two main elements of the transformer architecture. In our work, we used only the encoder part of the architecture described in Vaswani et al. (2017). The transformer encoder was built with a series of encoder blocks, each containing the following layers (dashed box in Figure 1):

1. multi-head self-attention;
2. a residual (skip) connection around the **mhsa** output, where the output  $\mathbf{Y}$  is written as

$$\mathbf{Y} = \mathbf{X} + \mathbf{mhsa}(\mathbf{X}); \quad (13)$$

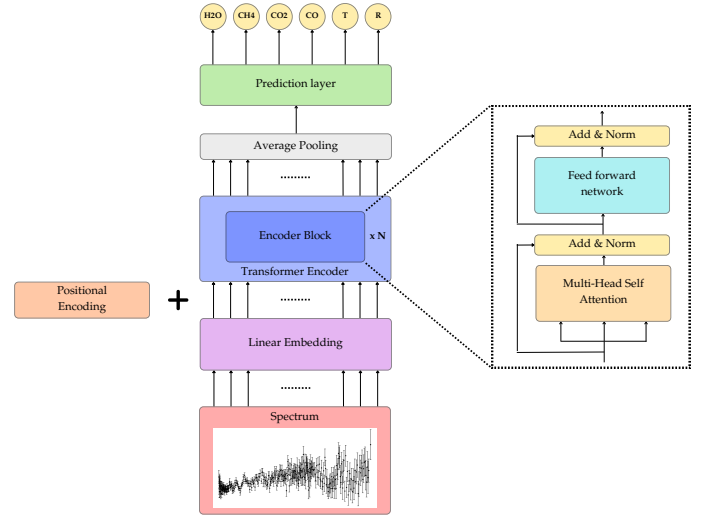
3. layer normalization that standardizes the skip-connection output to zero mean and unit variance, improving the transformer's numerical stability;
4. a feed-forward neural network applied to each output vector of layer normalization;
5. an additional skip-connection and final layer normalization.

In general, the encoder block is repeated multiple times inside a transformer encoder, receiving as input the output of the previous encoder block. This sequence of encoders enables the extraction of further correlations within the data.

## 3. Exoformer

We now present our transformer based tool, **Exoformer**, used to infer the values of six atmospheric parameters based on the data provided by the publicly available training dataset<sup>1</sup> by Zingales & Waldmann (2018). The transformer encoder forms the core of our model. However, other types of layers were integrated to extract the six predicted atmospheric parameters from an input transmission spectrum. Fig. 1 illustrates **Exoformer**'s structure: the shaded boxes indicate different layers, each with a specific functionality.

<sup>1</sup> <https://osf.io/6dxps/files/osfstorage>



**Fig. 1.** Schematic of the **Exoformer** architecture. Each box represents a layer described in Section 3. Inside the dashed box, the layers forming a single encoder block are indicated. Multiple encoder blocks repeated sequentially form the transformer encoder.

### 3.1. Linear embedding

An input transmission spectrum with 515 spectral points first passes through a linear embedding layer. There, each spectral point is transformed into a 128D vector using a multilayer perceptron (MLP), which yields a  $515 \times 128$  2D output. The first dimension represents the wavelength grid, while the new axis encodes preliminary spectral features captured by the MLP during the learning process. Lastly, the positional encoding vector is added to the embedding vector.

### 3.2. Transformer encoder

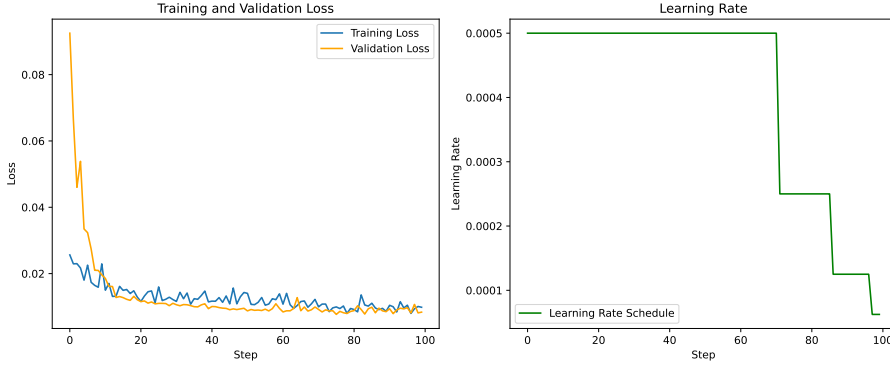
The resulting vector is then injected into the transformer encoder. This component is characterized by multiple transformer layers arranged in series and captures the long-range relations between the embeddings. **Exoformer** uses five transformer encoder blocks, each comprising eight multi-head self-attention layers and a feed-forward network with 1024 hidden units. Since the output of each transformer layer has the same dimensions as the input, the resulting vector from the transformer encoder has a  $515 \times 128$  shape.

### 3.3. Prediction layer

The last layer of **Exoformer** is used to predict the values of the six atmospheric parameters. To do this, we first applied four average pooling layers along the embedding dimension. The resulting vector provides an average representation of each embedding and is finally passed to a two-layer MLP, which produces six scalar outputs.

## 4. Methods

In this section, we discuss in detail the training procedure for **Exoformer**. We adopted the standard methodology for developing a deep learning tool, which includes using a representative dataset of real-world scenarios, applying data normalization for numerical stability, finding hyperparameters for the best



**Fig. 2.** Left plot: training and validation losses as a function of the training step. Right plot: learning rate trend as determined by the learning rate schedule applied during training.

**Table 1.** Boundary values of the seven atmospheric parameters used to generate the training dataset in Zingales & Waldmann (2018).

| Parameter                 | Lower bound | Upper bound |
|---------------------------|-------------|-------------|
| $\log \text{H}_2\text{O}$ | -8          | -1          |
| $\log \text{CH}_4$        | -8          | -1          |
| $\log \text{CO}$          | -8          | -1          |
| $\log \text{CO}_2$        | -8          | -1          |
| $M_p$                     | $0.8 M_J$   | $2.0 M_J$   |
| $R_p$                     | $0.8 R_J$   | $1.5 R_J$   |
| $T_{iso}$                 | 1000 K      | 2000 K      |

model, and performing uncertainty estimation for comparison with Bayesian tools.

#### 4.1. Dataset

To train Exoformer, we relied on the dataset described in Zingales & Waldmann (2018). The dataset contains  $10^7$  atmospheric transmission spectra of hot Jupiters, generated using the analytical forward model of TauREx 3 (Al-Refaie et al. 2021). Each spectrum was parameterized with seven atmospheric parameters: four molecule abundances ( $\text{H}_2\text{O}$ ,  $\text{CH}_4$ ,  $\text{CO}$ , and  $\text{CO}_2$ ), isothermal temperature  $T_{iso}$ , and the planet’s mass  $M_p$  and radius  $R_p$ . The atmospheric forward model, in addition to the absorption contributions of the four molecules, includes Rayleigh scattering and collisionally induced absorptions (CIAs) from  $\text{H}_2\text{-H}_2$  and  $\text{H}_2\text{-He}$  pairs. Table 1 summarizes the upper and lower boundaries of the seven parameters. All spectra were binned to a custom wavelength grid, ranging from 0.3 to 50  $\mu\text{m}$ , with 515 spectral points. This allowed us to cover not only the main absorption features of the four previously mentioned molecules but also the principal band passes of JWST and Ariel. For our regression problem, we used six of these seven available parameters:  $\text{H}_2\text{O}$ ,  $\text{CH}_4$ ,  $\text{CO}$ , and  $\text{CO}_2$  abundances, the isothermal temperature  $T_{iso}$ , and the planet’s radius  $R_p$ .

We then divided the entire dataset into three distinct subsets: training, validation, and test datasets. The training set was used to train the network, the validation set monitored the learning progress, and the test set identified the model with the best performance. The proportion assigned to each set is as follows: 90% for training, 9% for testing, and 1% for validation. Following modern best practices for large-scale datasets ( $10^7$  spectra), a 1% validation split provided a statistically significant sample size of  $10^5$  instances. This proportion ensured a reliable estimate of model performance while optimizing computational efficiency during the training process.

**Table 2.** Planetary parameters for the test case planet used as input for the TauREx forward model.

| Planet                    |         |
|---------------------------|---------|
| $\log \text{H}_2\text{O}$ | -4      |
| $\log \text{CH}_4$        | -5      |
| $\log \text{CO}$          | -5      |
| $\log \text{CO}_2$        | -6      |
| $M_p$                     | $1 M_J$ |
| $R_p$                     | $1 R_J$ |
| $T_{iso}$                 | 1000 K  |

**Notes.** The host star is a Sun-like, as used by Zingales & Waldmann (2018) in the training dataset.

#### 4.2. Spectra preprocessing

Because transmission spectra exhibit varying transit depth magnitudes, we need to scale each spectrum to a consistent range. This prevents Exoformer from assigning higher importance to inputs with higher raw values – a situation we want to avoid to ensure the tool remains unbiased. Consequently, we applied the normalization scheme introduced in Zingales & Waldmann (2018): every spectrum was divided into 14 bands, and the spectral points in each interval were normalized between 0 and 1 using the maximum and minimum values of those intervals. Figure 3 illustrates how a spectrum is transformed after applying the normalization scheme. The second panel depicts the 14 wavelength bands as vertical dashed lines, while the third shows that, within each subinterval defined by these bands, the spectral point with the highest transit depth normalizes to 1 and the point with the lowest transit depth normalizes to 0.

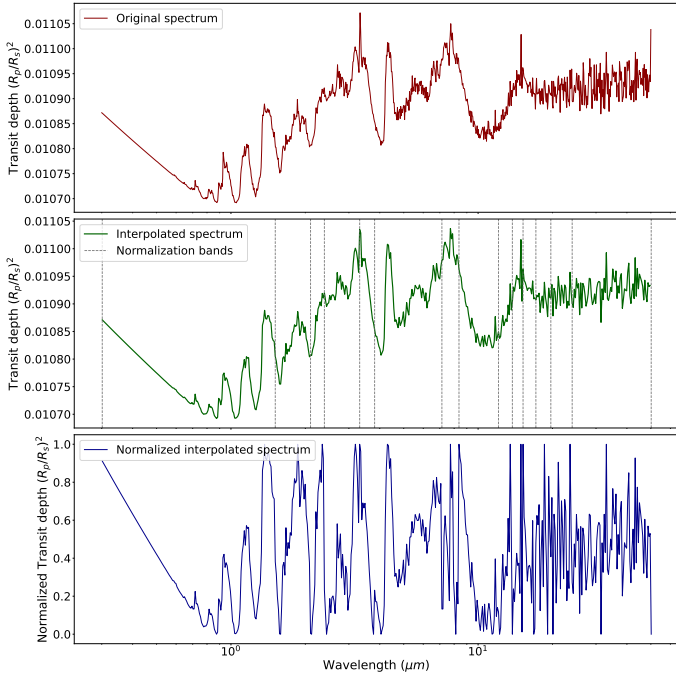
To mitigate similar biases in the outputs, atmospheric parameters values were also normalized between 0 and 1. We applied min-max scaling using the bounds in Table 2 for each parameter  $x$ :

$$z = \frac{(x - \min)}{(\max - \min)}, \quad (14)$$

where  $z$  is the scaled  $x$  parameter, while min and max are the lower and upper boundaries, respectively.

#### 4.3. Training

Exoformer was trained on the training dataset using the AdamW optimizer (Loshchilov & Hutter 2017) with mini-batches of size 64, initial learning rate of  $0.5 \cdot 10^{-4}$ , and weight decay of  $1 \cdot 10^{-4}$ . To prevent Exoformer training from stagnating



**Fig. 3.** Preprocessing phases on the test planet spectrum in Table 2. Upper plot: analytical spectrum computed using the TauREx forward model. Middle plot: analytical spectrum binned to the custom grid and normalization bands. Bottom plot: interpolated spectrum after normalization.

in the loss landscape, we used a scheduler that reduces the learning rate by a factor of 0.5 if the validation loss does not improve over ten consecutive steps (Fig. 2, right panel). Since Exoformer was used for a regression problem, we chose the mean squared error (MSE) between the real and predicted atmospheric parameters as training loss. In the left plot of Fig. 2 training and validation losses are shown as functions of training steps, illustrating model convergence and generalization capability. The best model hyperparameters were found using the Optuna (Akiba et al. 2019) Python library, with root mean squared error (RMSE) as the performance metric to minimize.

#### 4.4. Uncertainty estimation

Traditional deep-learning regression approaches focus on generating a single predicted value for each output parameter. To obtain a measure of the output uncertainty, which is crucial for our work, techniques such as MC dropout (Gal & Ghahramani 2016) are used. These techniques allow multiple outputs to be sampled during the inference phase, effectively providing a distribution of possible predictions. Our choice to use MC dropout is motivated by its computational efficiency and ease of implementation. In Exoformer the MC dropout mechanism was applied to all layers containing dropout.

Monte Carlo dropout arises from the need to predict the posterior distribution of an output  $y^*$  given some unseen data  $x^*$  with a neural network  $f$  defined by a set of weights  $\mathbf{W}$ ,

$$\mathbf{y}^* = f(\mathbf{x}^*, \mathbf{W}), \quad (15)$$

trained on a dataset  $D = \{\mathbf{X}, \mathbf{Y}\} = \{x_1, \dots, x_N, y_1, \dots, y_N\}$ . The expression of the posterior distribution is given by Bayesian inference:

$$P(y^*|x^*, D) = \int P(y^*|x^*, \mathbf{W})P(\mathbf{W}|D) d\mathbf{W}. \quad (16)$$

Gal & Ghahramani (2016) found that the previously defined neural network is equivalent to an approximation of the Gaussian process when dropout regularization is applied. This allows

rewriting Eq. (16) as

$$P(y^*|x^*, D) \approx q(y^*|x^*) \approx \mathcal{N}(f(x^*, \mathbf{W})). \quad (17)$$

In this way, we can compute the predictive mean and variance by running  $N_{step}$  forward passes of the neural network with dropout regularization kept active during inference time (after the training phase, when the neural network is used for predictions). The resulting expressions for mean and variance are, respectively,

$$\text{Mean}_{q(y^*|x^*)}(y^*) = \mathbb{E}_{q(y^*|x^*)}(y^*) \approx \frac{1}{N_{step}} \sum_{i=1}^{N_{step}} f(x^*, \mathbf{W}_i), \quad (18)$$

$$\begin{aligned} \text{Var}_{q(y^*|x^*)}(y^*) &\approx \frac{1}{N_{step}} \sum_{i=1}^{N_{step}} f(x^*, \mathbf{W}_i)^T f(x^*, \mathbf{W}_i) \\ &\quad - \mathbb{E}_{q(y^*|x^*)}(y^*)^T \mathbb{E}_{q(y^*|x^*)}(y^*). \end{aligned} \quad (19)$$

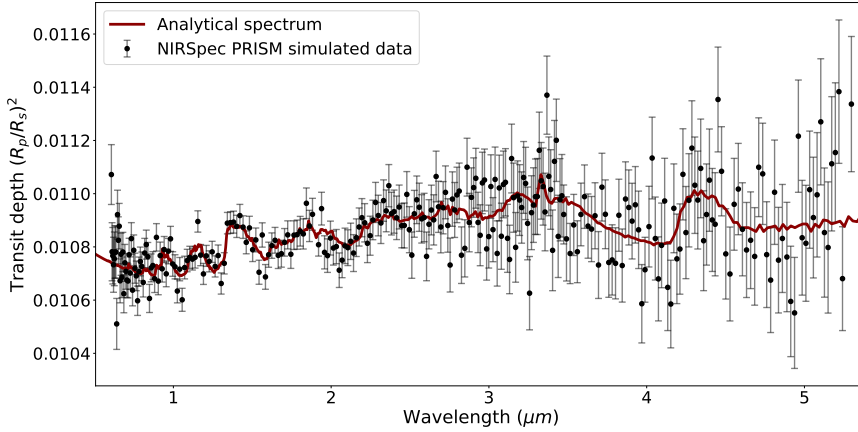
In other words, the variance includes not only the uncertainty of the model but also the uncertainty from the training data.

## 5. Results

In this section, we analyze the performance of Exoformer trained on the Zingales & Waldmann (2018) dataset. We evaluated its effectiveness by assessing its performance on a simulated JWST observation of a hot Jupiter and confirmed its robustness using unseen data. Subsequently, we compared its results with those from a Bayesian retrieval tool applied to real JWST observational data.

### 5.1. Retrieval on simulated observations

To evaluate the retrieval performance of Exoformer using realistic data, we used the atmospheric model from TauREx 3 (Al-Refaie et al. 2021) to simulate the transmission spectrum of a hot Jupiter. The model includes the absorption contributions from the four chemical species mentioned above (with cross



**Fig. 4.** Simulated NIRSpect PRISM observation of the transmission spectrum in Fig. 3. The observational data points (black dots) are binned to the native resolution of NIRSpect PRISM ( $R = 100$ ) and superimposed on the original (red line) TauREx analytical spectrum.

sections from ExoMol (Tennyson et al. 2024)), an isothermal temperature profile, Rayleigh scattering, and CIA contributions. A summary of the reference values for the planet is provided in Table 2. We then used Pandexo (Batalha et al. 2017), a tool for simulating JWST spectroscopic observations of exoplanetary atmospheres to create a more realistic observation of the test exoplanet. The simulated observation includes a noise floor of 30 ppm, a transit duration of four hours, and a single transit of the planet. For this simulation, we selected the NIRSpect instrument operating in PRISM mode, which covers a wavelength range of 0.7–5.0  $\mu\text{m}$  at a native resolution of 100. In Fig. 4 the analytical spectrum generated with TauREx (red line) and the NIRSpect Prism observation simulation (dots with error bars) are shown.

To account for uncertainties in real observations, we assumed Gaussian-distributed errors. Doing so, we generated a set of noisy spectra  $\mathbf{x}_i(\lambda)$  by sampling  $N_{\text{sample}}$  times from a normal distribution. The mean value of the distribution for each spectral point corresponds to the observed transit depth at wavelength  $\lambda_j$ , while the standard deviation is equal to the associated error at  $\lambda_j$ . We interpolated each noisy spectrum to the Exoformer grid using a cubic scheme, setting all points outside the instrument coverage to zero. Thanks to the correlations between the spectral features (captured during training and stored in the embedding), the transformer can still make predictions even when spectral data from wavelengths outside the instrument bands is missing. As a last step, we applied the normalization scheme described in Sect. 4.2.

We performed inference on  $N_{\text{sample}} = 500$  noisy spectra samples, applying MC dropout with  $N_{\text{step}} = 100$  and  $p_{\text{drop}} = 0.8$  to each. The  $N_{\text{step}}$  value was chosen to avoid excessive computational overhead during inference, while  $p_{\text{drop}}$  was treated as a hyperparameter and optimized to keep Exoformer consistent with the ground truth values of the atmospheric parameters. In total, we obtained a set of  $N_{\text{sample}}$  predictive distributions, each with  $N_{\text{step}}$  elements, for the seven parameters. Finally, these distributions were concatenated to compute the mean and  $1\sigma$  bounds of the parameters, as described in Gal & Ghahramani (2016). Figure 5 shows the retrieval results: Exoformer predictions are consistent (except for mass) with the ground truth values in Table 2 within the error bars. The distributions are Gaussian-like, as expected from the definition of MC dropout. The width of the distributions is controlled by the variance in Eq. (19), which is proportional to the dropout probability (Gal & Ghahramani 2016). Since we used a high value for this probability, we expect a high value for the  $1\sigma$  intervals. Another aspect is the increase of the distribution toward the edge of the boundary conditions, which arises from Exoformer’s prediction layer.

In fact, a ReLU activation function was applied to the last layer. Since its expression is  $\max(0, x)$ , it sets all negative values to zero without limiting positive values from the previous layer.

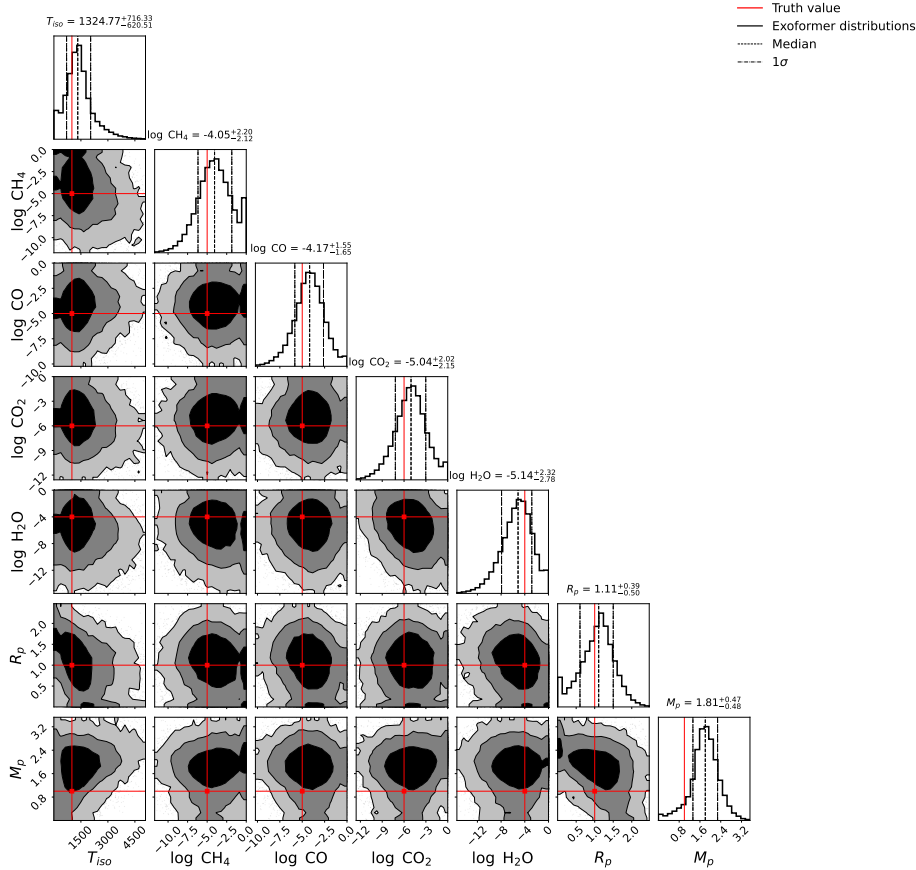
Planetary mass retrieval is challenging due to its degeneracy with other atmospheric parameters (Changeat et al. 2020), as distinct parameter combinations can produce similar spectra. Consequently, to mitigate the impact of the planetary mass uncertainty on the accuracy of atmospheric composition (Di Maio et al. 2023), we did not include the mass prediction from Exoformer and instead fixed its value in subsequent retrievals.

## 5.2. Comparison with Bayesian retrieval tools

We then tested Exoformer against a Bayesian retrieval tool on real JWST transmission spectra. For this comparison, we selected WASP-39b and WASP-17b, two hot Jupiter with parameters falling within the limits of the training dataset. We used the transmission spectrum obtained with the NIRSpect instrument in PRISM mode, reduced with the FIREFLY pipeline by Rustamkulov et al. (2023) for WASP-39b, and the transmission spectrum from NIRISS in SOSS mode for WASP-17b reduced by Louie et al. (2025).

We first performed the retrieval with TauREx, applying the same forward model used to generate the training dataset and the analytical spectrum of the test planet. The model includes six fitting parameters, corresponding to those retrieved by Exoformer. We assigned log-uniform prior ranges of  $10^{-10}$ – $10^{-1}$  to the molecular mixing ratios, whereas we applied uniform distributions for the priors of  $T_{\text{iso}}$  and  $R_p$ , with ranges 1000–2000 K and 0.8–1.5  $R_J$ , respectively. Following the procedure described in Section 5.1, we performed the analysis of WASP-39b and WASP-17b with Exoformer, obtaining a second set of posterior distributions.

The posterior distributions obtained with our tool agree with those obtained from a Bayesian framework within  $1\sigma$  for both planets (except for WASP-39b  $\text{CH}_4$ , which is compatible within  $1.5\sigma$ ; see Fig. A.1 and A.2 in green and blue, respectively). This shows that Exoformer’s retrieval capability is comparable to TauREx’s, although with slightly lower accuracy. However, we highlight an important trade-off between accuracy and computational speed: while TauREx yields precise results, its execution time is significantly longer than Exoformer’s, which accomplished the same task in a fraction of the time. TauREx required  $\approx 498$  h (WASP-39b) and  $\approx 86$  h (WASP-17b) with uniform priors to complete the retrievals on a single CPU core. By



**Fig. 5.** Posterior distributions and ground truth values (red lines) of the seven parameters. The retrieval was performed using Exoformer on the NIRSpec PRISM simulation. The dashed lines indicate the median of the distribution, while the dashed-dotted lines indicate the  $1\sigma$  intervals.

contrast, Exoformer inference to generate the priors itself took  $\sim 2$  minutes on an NVIDIA A2 GPU.

The retrieval performed on the two transmission spectra serves as a robustness test for Exoformer. Real observations often contain atmospheric phenomena unseen during the training phase. For example, WASP-39b’s atmosphere contains strong traces of  $\text{SO}_2$  and  $\text{H}_2\text{S}$ , which originate from photochemical processes (Constantinou et al. 2023). These unknown chemical species can interfere with the target molecules. Furthermore, clouds and haze can significantly affect retrievals by reducing or eliminating absorption features across observed wavelengths (Lu 2023), ultimately resulting in biased measurements. Despite the challenges posed by real-world observations, the posterior distributions recovered by Exoformer remain consistent with those from TauREx, highlighting the robustness and reliability of our tool when applied to JWST spectroscopic data.

## 6. Informative priors

The results obtained with Exoformer, as detailed in Section 5.2, present an opportunity to explore a hybrid approach that combines both the robustness and accuracy of Bayesian methods with the speed of deep learning, potentially enhancing the performance of existing Bayesian tools. Bayesian algorithms benefit from informative prior distributions, which accelerate convergence by constraining the probability within specific parameter space regions (Gelman et al. 2017).

In fact, by reducing the volume of the prior distribution  $V_{\text{prior}}$  over the volume of the posterior distribution  $V_{\text{posterior}}$ , the Kullback–Leibler (KL) divergence between the distributions is

reduced (Petrosyan & Handley 2022):

$$D_{KL}(\text{posterior} \parallel \text{prior}) \approx \ln\left(\frac{V_{\text{prior}}}{V_{\text{posteriors}}}\right). \quad (20)$$

Because the time complexity  $T$  of the nested sampling algorithm is proportional to the KL divergence between priors and posteriors (Petrosyan & Handley 2022) then

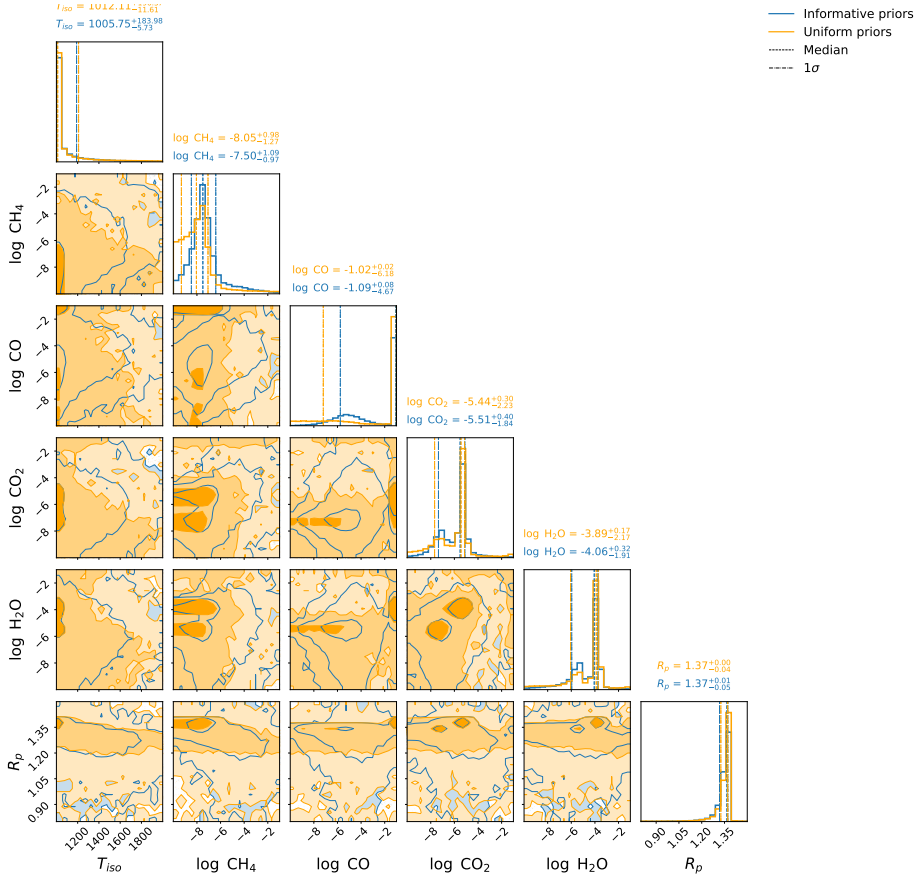
$$T \propto \ln\left(\frac{V_{\text{prior}}}{V_{\text{posterior}}}\right). \quad (21)$$

So when we restrict the priors space using informative priors, the overall effect is a reduction of the run-time of the algorithm.

### 6.1. WASP-39b and WASP-17b

To assess the speedup a Bayesian retrieval could achieve with informative priors, we reexamined the WASP-39b and WASP-17b transmission spectra. We transformed Exoformer’s posterior distributions into informative priors for the nestle plugin, which can also be used with the multinest plugin (Feroz et al. 2009). To keep the informative prior distributions consistent with the uniform priors used in the retrievals, we limited the informative priors to the same boundaries presented in Section 5.2 for all parameters. The two retrieval methodologies returned a set of posterior distributions compatible with one another within  $1\sigma$  (Figs. 6 and 7). However, we observe significant improvement in the computational times (Table 3) for both planets: the speedup is close to eight times for WASP-39b and three times for WASP-17b.

The logarithmic Bayes factors (Kass & Raftery 1995; Trotta 2007) for the two planets show different values (Table 3). These



**Fig. 6.** Corner plot for the WASP-39b retrieval. The posterior distributions obtained with informative priors are shown in blue, while those obtained with uniform priors are shown in orange. The dashed lines indicate the median of the distributions, while the dashed-dotted lines indicate the  $1\sigma$  intervals. All the parameters from the two retrievals are compatible within  $1\sigma$ .

**Table 3.** Summary table for the six atmospheric retrievals performed in this work (two on real JWST data and four on simulated observations).

| Planet      | Timescales    |                   | Speedup | log-Bayesian evidence |                     | log-Bayes factor |
|-------------|---------------|-------------------|---------|-----------------------|---------------------|------------------|
|             | Uniform prior | Informative prior |         | Uniform prior         | Informative Prior   |                  |
| WASP-39b    | 498 h         | 64 h              | 7.8     | $668.04 \pm 0.14$     | $669.20 \pm 0.14$   | -1.16            |
| WASP-17b    | 86 h          | 27 h              | 3.2     | $413.52 \pm 0.10$     | $408.64 \pm 0.11$   | 4.87             |
| Simulated 1 | 44 h          | 36 h              | 1.2     | $2185.67 \pm 0.11$    | $2185.21 \pm 0.11$  | 0.46             |
| Simulated 2 | 71 h          | 65 h              | 1.1     | $2175.64 \pm 0.11$    | $2174.42 \pm 0.11$  | 1.22             |
| Simulated 3 | 33 h          | 27 h              | 1.2     | $2189.13 \pm 0.10$    | $2191.779 \pm 0.09$ | -2.649           |
| Simulated 4 | 23 h          | 20 h              | 1.1     | $2183.214 \pm 0.08$   | $2181.244 \pm 0.09$ | 1.97             |

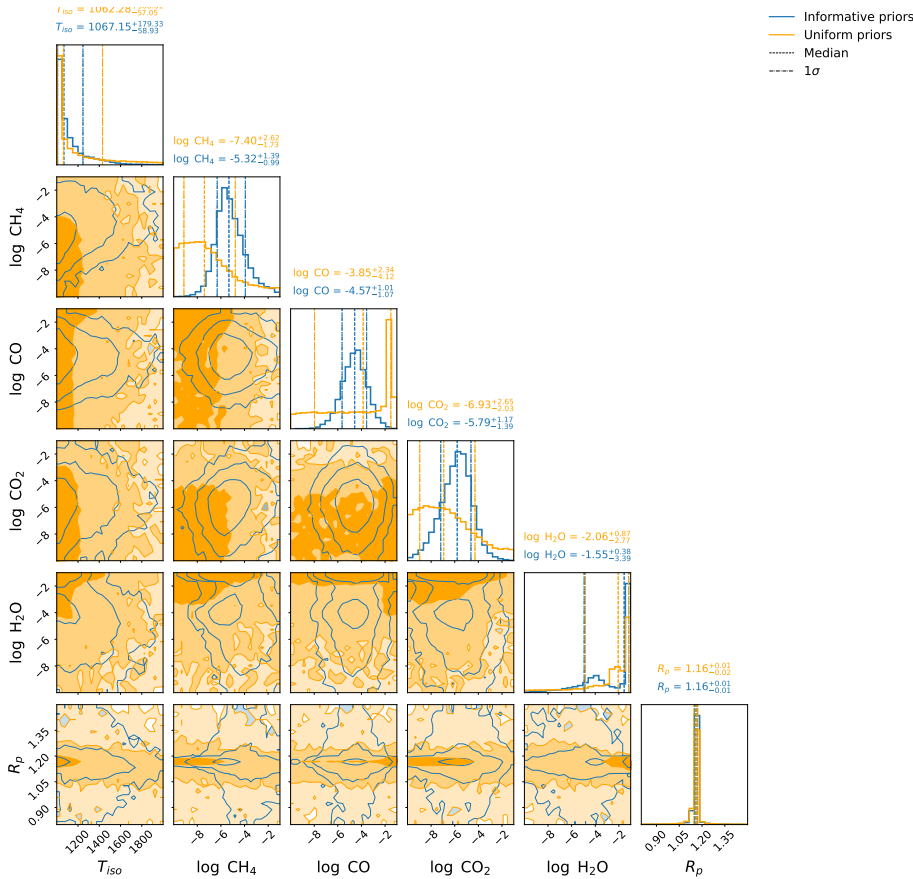
**Notes.** The first three columns show the computational timescales for the retrievals with uniform and informative priors, along with the corresponding speedup. The last two columns show the log-Bayesian evidences for the same retrievals and the corresponding log-Bayes factor (computed following [Kass & Raftery \(1995\)](#); [Trota \(2007\)](#)) as the difference between evidences with uniform and informative priors).

factors were computed as the difference between the evidence of the model with uniform priors and that with informative priors. For the WASP-39b retrieval, a log-Bayes factor  $|\log B| = 1.16 < 2$  ([Kass & Raftery 1995](#); [Trota 2007](#)) indicates a weak preference for the model with informative priors. Indeed, the two retrievals show very similar best-fit models (Fig. 8), with no evident differences. Furthermore, both models exhibit a clear deviation from observed data in the same wavelength range. This fitting yields higher transit depth values, explaining why the CO posterior distribution is pushed toward the upper boundary of the prior space. In the WASP-17b retrieval, a log-Bayes factor of  $2 < |\log B| = 4.87 < 5$  moderately favors ([Kass & Raftery 1995](#); [Trota 2007](#)) the model found using uniform priors. Again, Fig. 9 shows very similar best-fit models, with an appreciable difference only in the range 2.25–2.5  $\mu\text{m}$ . This results in a greater abundance of H<sub>2</sub>O for the uniform priors retrieval (Fig. 7).

This tendency of the retrievals toward the model obtained with uniform priors – despite the similar fit with the model using informative priors – can be explained by the definition of Bayesian evidence itself. In fact, when two models both fit the data well, broader priors in one of them (such as those generated using uniform distributions) can yield larger evidence values ([Trota 2008](#)), thereby increasing the absolute value of the Bayes factor.

Figure 6 (for CH<sub>4</sub>) and Fig. 7 lastly demonstrate the regularization effects of the Gaussian-like priors of Exoformer. As a result, the posterior distributions are smoother and more regular than those obtained with uniform priors ([Lorente et al. 2023](#)).

It is crucial to note that these two retrievals do not provide new solutions for the atmospheric characterization of the two selected exoplanets, but rather demonstrate a method that is consistent with – and effectively enhances – Bayesian retrievals.



**Fig. 7.** Corner plot for the WASP-17b retrievals, with labels as in Figure 6. The two retrievals are compatible with one another within  $1\sigma$ , with more regular posterior distributions from the informative priors.

This is because the incompleteness of our atmospheric model can induce degenerate solutions. In the WASP-39b transmission spectrum, our atmospheric model (Section 4.1) does not completely capture the prominent  $\text{CO}_2$  feature at  $\sim 4.3 \mu\text{m}$  (Rustamkulov et al. 2023) (Fig. 8) and shows a significant discrepancy with the observed data between 4.5 and 5.5  $\mu\text{m}$ . Similarly, the strong  $\text{H}_2\text{O}$  features at  $\sim 1.4$ ,  $\sim 1.8$ , and above 2.5  $\mu\text{m}$  in WASP-17b (Louie et al. 2025) (Fig. 9) are not captured by our model. The introduction of a radiative-convective thermochemical equilibrium (RCTE) model (as in Rustamkulov et al. (2023) and Louie et al. (2025)) constrains the sampler to converge to more realistic solutions (green lines in Figs. 8 and 9).

Retraining Exoformer with a more sophisticated physical model is feasible but beyond the scope of this work, as it would require both implementing the new model in TauREx and generating a new, large-scale dataset. Our goal rather is to show that this hybrid approach is fully compatible and consistent with traditional retrievals using uniform priors, while also significantly reducing computational time.

## 6.2. Simulated planets

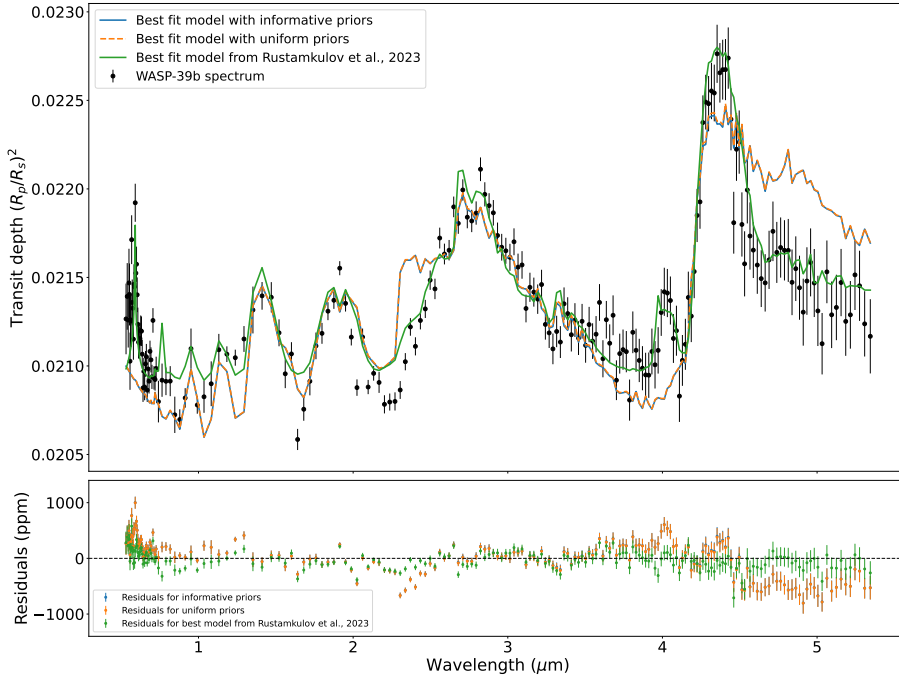
We also tested our strategy of combining Exoformer and TauREx with four additional simulated NIRSpect PRISM spectra, generated as described in Section 5.1. The last four rows of Table 3 summarize the log-Bayes factors obtained from the retrievals on these simulated observations, indicating no strong evidence (Kass & Raftery 1995; Trotta 2007) in favor of either retrieval method; thus, we can consider both implementations equivalent. However, in terms of computational timescales (Table 3), we notice a different behavior compared to the results from real observations: the retrievals with uniform priors are only slightly slower than those with informative priors.

The small improvement can be interpreted in the context of Bayesian inference. In our case, the four simulated transmission spectra were generated with the same forward model (Section 4.1) used in the retrieval process, with realistic instrumental noise added through Pandexo. Thus, we describe the simulated observations with exactly the number of parameters necessary to explain the physical processes behind their formation. Following Trotta (2008), our parameters are already well constrained by the data (the effective number of parameters, given by the Bayesian complexity (Spiegelhalter et al. 2002) equals the number of free parameters), so any additional information provided by the informative priors contributes only marginally to the model’s predictivity. As a result, informed prior knowledge yields a minor speedup compared to using uniform priors.

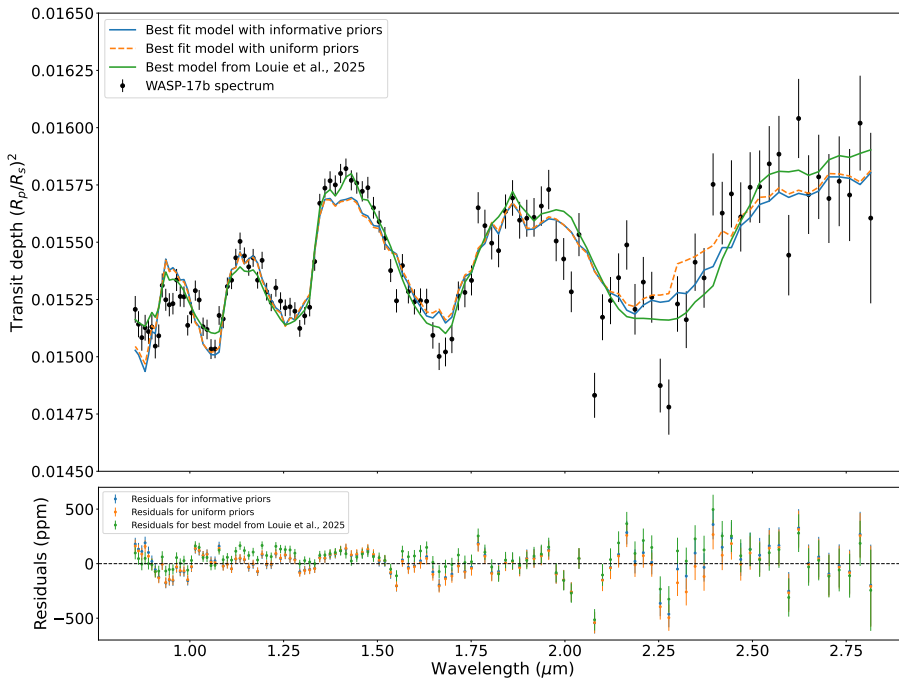
In contrast, JWST observations of real exoplanets require additional parameters to explain the additional spectral features (see Figs. 8 and 7), which are not provided by the model used in our retrieval. As a consequence, in this case, the retrieval performance is controlled by the parameter space available from the priors, and so how much the Bayesian algorithm has to explore.

## 7. Discussion and conclusions

Atmospheric retrievals frequently rely on uninformative priors, such as uniform distributions, due to limited knowledge of molecular abundances and temperatures in exoplanetary atmospheres. This approach forces Bayesian frameworks, such as nested sampling, to explore the entire parameter space, creating a computational bottleneck that becomes increasingly predominant as the number of fitting parameters increases. However,



**Fig. 8.** Best-fit models for the WASP-39b NIR-Spec PRISM observation obtained using uniform (orange line) priors, informative (blue line) priors, and the 1D RCTE model (green line) from Rustamkulov et al. (2023). Both the uniform and informative models miss important spectral features that are instead captured by the RCTE model.



**Fig. 9.** Best-fit models for the WASP-17b NIRISS SOSS observation obtained using uniform (orange line) and informative (blue line) priors compared to the best-fit model by Louie et al. (2025) (green line). The residuals show that our two models are consistent, differing only in the 2.25–2.5  $\mu\text{m}$  wavelength range. As for WASP-39b, our atmospheric model cannot describe all spectra features, such as the strong  $\text{H}_2\text{O}$  features at  $\sim 1.4$ ,  $\sim 1.8$ , and above 2.5  $\mu\text{m}$ .

when prior knowledge is available, we can avoid excessive sampling in regions of the parameter space with improbable values (Ashton et al. 2022). Deep learning models such as Exoformer address these challenges by extracting useful information from the training data and constraining the range of possible values for the fitting parameters. These tools rapidly compute parameter distributions – typically in minutes – that a Bayesian framework can use to focus on high-probability regions of the parameter space, resulting in a more efficient and smoother inference than with uniform priors (Gelman et al. 2017).

A key drawback of this strategy is the unrepresentative prior problem (Chen et al. 2019, 2023) that affects nested sampling-based algorithms. As the prior distribution moves away from the true value of the parameter, the likelihood remains almost

flat within the prior space (Chen et al. 2019, 2023). By definition, the nested sampling algorithm selects live points from priors with higher likelihood values at each iteration (Skilling 2006). Consequently, on a flat likelihood surface, the time spent searching for higher-likelihood values is significantly greater. This results in slow convergence timescales or – in the worst case – trapping of the algorithm in low-likelihood regions, yielding incorrect posterior distributions (Chen et al. 2019, 2023). Our tool, however, demonstrates robustness by deriving the correct fitting parameters (within  $1\sigma$ ), thereby minimizing the risk of exploring incorrect regions of the prior space.

Our strategy of computing informative priors through a transformer-based tool and combining it with a Bayesian retrieval proved effective in reducing the computational

timescales of atmospheric retrievals. We achieved a three to eight time speedup compared to classical retrievals performed using uniform prior distributions, while maintaining consistency with the retrieved parameters and the best-fit models.

The transformer architecture at the foundation of our tool has been effective in regression and classification tasks for sequential data. However, further improvements to the dataset are necessary to prepare Exoformer for large future atmospheric surveys such as Ariel, which will give us deeper insights into atmospheres for a wide population of exoplanets (Edwards et al. 2019). For WASP-39b and WASP-17b, our model fails to fit all spectral features, indicating the absence of molecules and physical processes observed by JWST. When describing populations of hot Jupiters, these failures can be addressed by including additional molecular compounds typically found in hot Jupiters (sulfur compounds such as SO<sub>2</sub> and H<sub>2</sub>S (Zahnle et al. 2009), or oxidized compounds such as TiO, VO, and SiO (Désert et al. 2008)), multidimensional effects (e.g., (Helling et al. 2021; Zingales et al. 2022)), or different chemical states (e.g., chemical disequilibrium and photochemistry (Tsai et al. 2021)).

Another aspect to consider is the type of exoplanets described by the dataset: the hot giant planets in our dataset constitute only a small fraction of the broader exoplanet family. It has been well established that small exoplanets are the most abundant class (Petigura et al. 2013). However, their atmospheres occupy different regions of the parameter space (smaller masses and radii, higher bulk densities, and different physical processes; e.g., Madhusudhan et al. (2016)) than those of giant exoplanets.

A comprehensive deep learning tool for future survey analysis should also be able to analyze the atmospheric spectra of these small planets. The advantage of our data-driven approach is that the architecture remains fixed while training Exoformer on new datasets. The updated network state can then be saved and applied to the appropriate data.

The custom wavelength grid, derived from Zingales & Waldmann (2018), represents another critical point. Although the grid covers the wavelength domain of JWST filters, its resolution was initially designed for HST observations. Consequently, it is no longer adequate for application with high-resolution JWST data and for future observations with Ariel. In fact, during the interpolation, some spectral information could be lost because of the lower number of points in the grid compared to the observations. A new custom grid should be developed with a sufficiently fine set of wavelength points to match the typical resolution of JWST instruments. In this way, we will help Exoformer extract more spectral features from real JWST observations, further increasing its accuracy.

## Data availability

The Exoformer and ExoformerPriors TauREx plugin are available on GitHub respectively at Exoformer and ExoformerPriorsTauREx.

*Acknowledgements.* This publication was produced while attending the PhD program in Astronomy at the University of Padova, Cycle XXXIV, with the support of a scholarship co-financed by the Ministerial Decree no. 118 of 2nd March 2023, 760 based on the NRRP – funded by the European Union – NextGenerationEU – Mission 4 Component 1 – CUP C96E23000340001. GPI and GMA acknowledge support by the Space It Up project funded by the Italian Space Agency, ASI, and the Ministry of University and Research, MUR, under contract no. 2024-5-E.0 – CUP no. I53D24000060005. TZi acknowledges support from CHEOPS ASI-INAF agreement no. 2019-29-HH.0, NVIDIA Academic

Hardware Grant Program for the use of the Titan V GPU card and the Italian MUR Departments of Excellence grant 2023–2027 “Quantum Frontiers”.

## References

- Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. 2019, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*
- Al-Refai, A. F., Changeat, Q., Waldmann, I. P., & Tinetti, G. 2021, *AJ*, 151, 37
- Ashton, G., Bernstein, N., Buchner, J., et al. 2022, *Nat. Rev. Methods Primers*, 2 [arXiv:2205.15570] [stat]
- Batalha, N. E., Mandell, A., Pontoppidan, K., et al. 2017, *PASP*, 129, 064501
- Changeat, Q., Keyte, L., Waldmann, I. P., & Tinetti, G. 2020, *ApJ*, 896, 107
- Chen, X., Feroz, F., & Hobson, M. 2023, *Bayesian Anal.*, 18
- Chen, X., Hobson, M., Das, S., & Gelderblom, P. 2019, *Stat. Comput.*, 29, 835
- Constantinou, S., Madhusudhan, N., & Gandhi, S. 2023, *ApJ*, 943, L10
- Cortes, C., & Vapnik, V. 1995, *Mach. Learn.*, 20, 273
- Di Maio, C., Changeat, Q., Benatti, S., & Micela, G. 2023, *A&A*, 669, A150
- Désert, J.-M., Vidal-Madjar, A., Lecavelier Des Etangs, A., et al. 2008, *A&A*, 492, 585
- Edwards, B., Mugnai, L., Tinetti, G., Pascale, E., & Sarkar, S. 2019, *AJ*, 157, 242
- Feroz, F., Hobson, M. P., & Bridges, M. 2009, *MNRAS*, 398, 1601
- Gal, Y., & Ghahramani, Z. 2016, in *PMLR*, 48, Proceedings of The 33rd International Conference on Machine Learning, ed. M. F. Balcan, & K. Q. Weinberger (New York, New York, USA: PMLR), 1050
- Gardner, J. P., Mather, J. C., Clampin, M., et al. 2006, *Space Sci. Rev.*, 123, 485
- Gelman, A., Simpson, D., & Betancourt, M. 2017, *Entropy*, 19, 555
- Hayes, J. J. C., Kerins, E., Awiphan, S., et al. 2020, *MNRAS*, 494, 4492
- Helling, Ch., Worters, M., Samra, D., et al. 2021, *A&A*, 648, A80
- Himes, M. D., Harrington, J., Cobb, A. D., et al. 2022, *Planet. Sci. J.*, 3, 91
- Hochreiter, S., & Schmidhuber, J. 1997, *Neural Comput.*, 9
- Irwin, P., Teanby, N., De Kok, R., et al. 2008, *JQSRT*, 1136
- Janiesch, C., Zschech, P., & Heinrich, K. 2021, *Electron. Mark.*, 31, 685
- Kass, R. E., & Raftery, A. E. 1995, *J. Am. Stat. Assoc.*, 90, 773
- Kaufman, L. 2005, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley Series in Probability and Mathematical Statistics (Hoboken, NJ: Wiley)
- LeCun, Y., Boser, B., Denker, J., et al. 1989, in *Advances in Neural Information Processing Systems*, 2 (Morgan-Kaufmann)
- LeCun, Y., Bengio, Y., & Hinton, G. 2015, *Nature*, 521, 436
- Llorente, F., Martino, L., Curbelo, E., López-Santiago, J., & Delgado, D. 2023, *WIREs Comput. Stat.*, 15, e1595
- Loshchilov, I., & Hutter, F. 2017, in *International Conference on Learning Representations*
- Louie, D. R., Mullens, E., Alderson, L., et al. 2025, *AJ*, 169, 86
- Lu, L. 2023, *Highl. Sci. Eng. Technol.*, 38, 90
- MacDonald, R. J., & Batalha, N. E. 2023, *RNAAS*, 7, 54
- Madhusudhan, N., Agúndez, M., Moses, J. I., & Hu, Y. 2016, *Space Sci. Rev.*, 205, 285
- McCaulliff, S. D., Jenkins, J. M., Catanzarite, J., et al. 2015, *ApJ*, 806, 6
- Mollière, P., Wardenier, J. P., Van Boekel, R., et al. 2019, *A&A*, 627, A67
- Pan, J.-S., Ting, Y.-S., & Yu, J. 2024, *MNRAS*, 528, 5890
- Petigura, E. A., Howard, A. W., & Marcy, G. W. 2013, *PNAS*, 110, 19273
- Petrosoyan, A., & Handley, W. 2022, *MaxEnt 2022*
- Quinlan, J. R. 1986, *Mach. Learn.*, 1, 81
- Rocchetto, M., Waldmann, I. P., Venot, O., Lagage, P.-O., & Tinetti, G. 2016, *ApJ*, 833, 120
- Rustamkulov, Z., Sing, D. K., Mukherjee, S., et al. 2023, *Nature*, 614, 659
- Shallue, C. J., & Vanderburg, A. 2018, *AJ*, 155, 94
- Skilling, J. 2006, *Bayesian Anal.*, 1
- Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. 2002, *J. R. Stat. Soc. Ser. B Methodol.*, 64, 583
- Tennyson, J., Yurchenko, S. N., Zhang, J., et al. 2024, *JQSRT*, 326, 109083
- Tinetti, G., Eccleston, P., Luefvinger, T., et al. 2022, arXiv e-prints [arXiv:2104.04824]
- Trotta, R. 2007, *MNRAS*, 378, 72
- Trotta, R. 2008, *Contemp. Phys.*, 49, 71
- Tsai, S.-M., Malik, M., Kitzmann, D., et al. 2021, *ApJ*, 923, 264
- Vasist, M., Rozet, F., Absil, O., et al. 2023, *A&A*, 672, A147
- Vaswani, A., Shazeer, N., Parmar, N., et al. 2017, *Adv. Neural Inform. Process. Syst.*, 30
- Yip, K. H., Changeat, Q., Nikolaou, N., et al. 2021, *ApJ*, 162, 195
- Zahnle, K., Marley, M. S., Freedman, R. S., Lodders, K., & Fortney, J. J. 2009, *ApJ*, 701, L20
- Zhang, M., Wu, F., Bu, Y., et al. 2024, *A&A*, 683, A163
- Zingales, T., & Waldmann, I. P. 2018, *AJ*, 156, 268
- Zingales, T., Falco, A., Pluriel, W., & Leconte, J. 2022, *A&A*, 667, A13

Appendix A: Additional figures

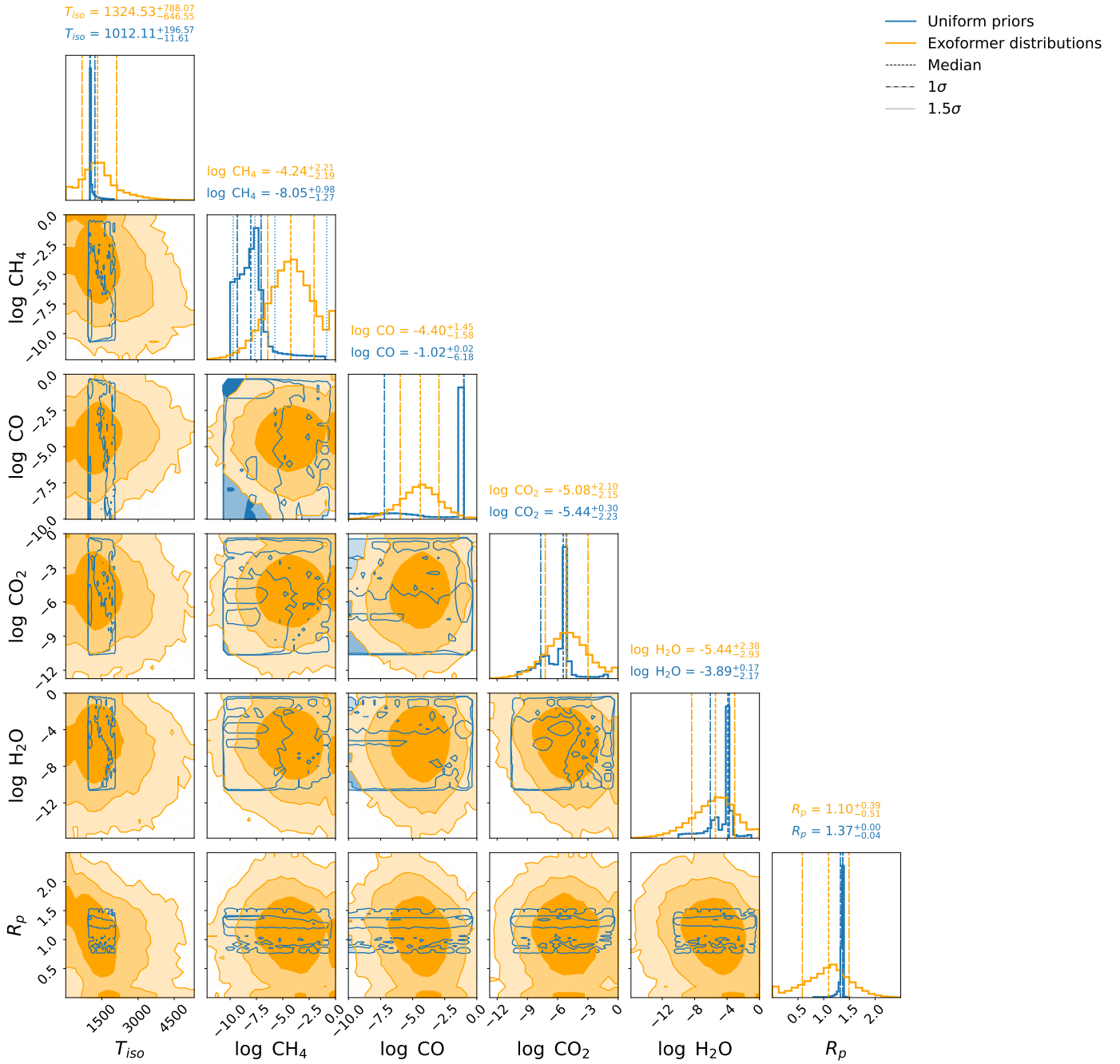


Fig. A.1: Corner plot for the WASP-39b retrieval. In blue we show the posterior distributions retrieved with TauREx using uniform priors, while in orange we show the distribution obtained using Exoformer. The distributions are compatible with one another within  $1\sigma$ , while for  $CH_4$  within  $1.5\sigma$ .

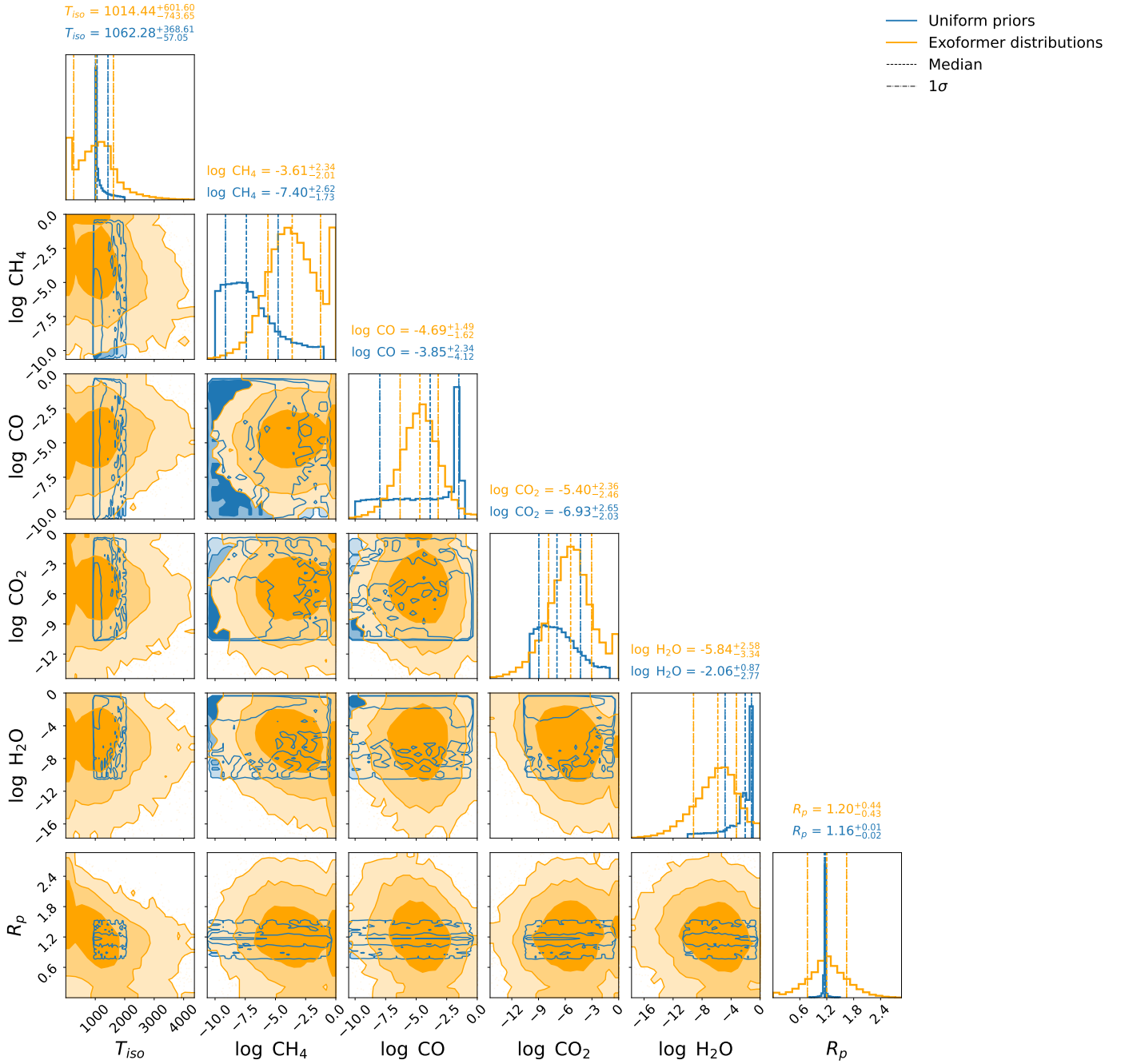


Fig. A.2: Corner plot for the WASP-17b retrieval. In blue we show the posterior distributions retrieved with TauREx using uniform priors, while in orange we show the distribution obtained using Exoformer. The distributions are compatible with one another within  $1\sigma$ .