

BCILattice

Model Catalog Reference

Edge GPT Sequence Transformer Models

bcinexus.io/docs/models

Categories:	4
Model Classes:	15
Methods Documented:	18
Source:	pipeline_catalog.csv
URL:	bcinexus.io/docs/models

Auto-generated from BCILattice pipeline catalog · 2026

Edge Models

MobileEEGNet

CLASS

```
from nm_models.edge import MobileEEGNet
MobileEEGNet()
Module: nm_models.edge
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

QuantizedEEGNet

CLASS

```
from nm_models.edge import QuantizedEEGNet
QuantizedEEGNet()
Module: nm_models.edge
Role: both
Output: model_obj:Any
Methods:
    .convert_to_int8()
    -> self:QuantizedEEGNet
    .export_onnx()
    -> path:str
    .forward()
    -> output:Any
    .prepare_quantization()
    -> self:QuantizedEEGNet
```

TinyEEGNet

CLASS

```
from nm_models.edge import TinyEEGNet
TinyEEGNet()
Module: nm_models.edge
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

GPT Models

BioGPTBCI

CLASS

```
from nm_models.gpt_models import BioGPTBCI
BioGPTBCI()
Module: nm_models.gpt_models
Role: both
Output: model_obj:Any
```

Methods:

`.forward()`
-> output:Any

GPT2BCI

CLASS

`from nm_models.gpt_models import GPT2BCI`

`GPT2BCI()`

Module: nm_models.gpt_models
Role: both
Output: model_obj:Any

Constructor Parameters:

`gpt2_size` dropdown default: gpt2

Methods:

`.forward()`
-> output:Any

LaBraMBCI

CLASS

`from nm_models.gpt_models import LaBraMBCI`

`LaBraMBCI()`

Module: nm_models.gpt_models
Role: both
Output: model_obj:Any

Methods:

`.forward()`
-> output:Any

LargeGPT

CLASS

`from nm_models.gpt_models import LargeGPT`

`LargeGPT()`

Module: nm_models.gpt_models
Role: both
Output: model_obj:Any

Methods:

`.forward()`
-> output:Any

MiniGPT

CLASS

`from nm_models.gpt_models import MiniGPT`

`MiniGPT()`

Module: nm_models.gpt_models
Role: both
Output: model_obj:Any

Methods:

`.forward()`
-> output:Any

TinyGPT

CLASS

```
from nm_models.gpt_models import TinyGPT
TinyGPT()
Module: nm_models.gpt_models
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

Sequence Models

BiLSTMClassifier

CLASS

```
from nm_models.sequence import BiLSTMClassifier
BiLSTMClassifier()
Module: nm_models.sequence
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

GRUClassifier

CLASS

```
from nm_models.sequence import GRUClassifier
GRUClassifier()
Module: nm_models.sequence
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

LSTMClassifier

CLASS

```
from nm_models.sequence import LSTMClassifier
LSTMClassifier()
Module: nm_models.sequence
Role: both
Output: model_obj:Any
Methods:
    .forward()
    -> output:Any
```

MambaSSM

CLASS

```
from nm_models.sequence import MambaSSM
MambaSSM()
Module:          nm_models.sequence
Role:            both
Output:          model_obj:Any

Methods:
  .forward()
  -> output:Any
```

TCNClassifier

CLASS

```
from nm_models.sequence import TCNClassifier
TCNClassifier()
Module:          nm_models.sequence
Role:            both
Output:          model_obj:Any

Methods:
  .forward()
  -> output:Any
```

Transformer Models

PatchTST

CLASS

```
from nm_models.transformers import PatchTST
PatchTST()
Module:          nm_models.transformers
Role:            both
Output:          model_obj:Any

Methods:
  .forward()
  -> output:Any
```