Idea Factory Intensive Program #2

딥러닝 홀로서기

#11

이론강의/PyTorch실습/코드리뷰

딥러닝(Deep Learning)에 관심이 있는 학생 발굴을 통한
딥러닝의 이론적 배경 강의 및 오픈소스 딥러닝 라이브러리 PyTorch를 활용한 실습

Today`s Time Schedule

Assignment #1 Review

How to Parameterize Entire Code

How to Run Code with GPU!

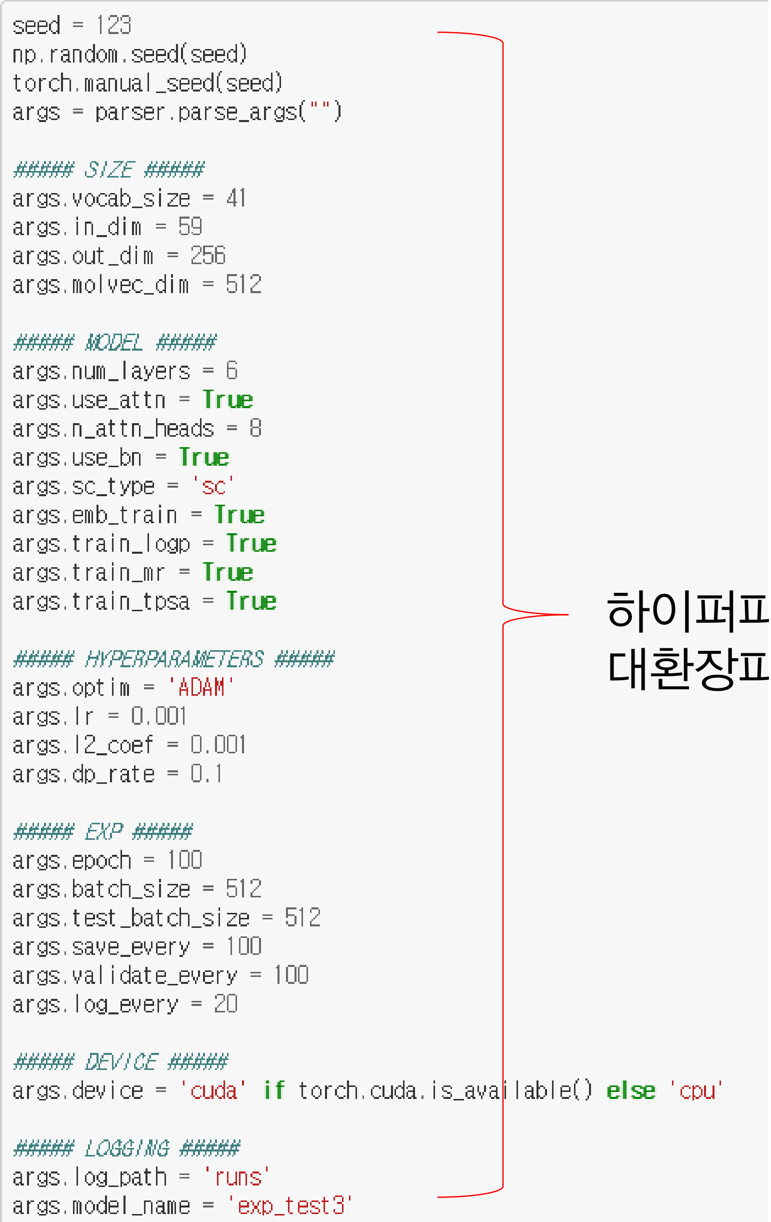
1 hour?

How to Overcome Overfitting — 1 hour

Big Wave: Hyperparameter Tuning — 2 hour

How to Parametrize Entire Code (And why?)

What is Parametrization?

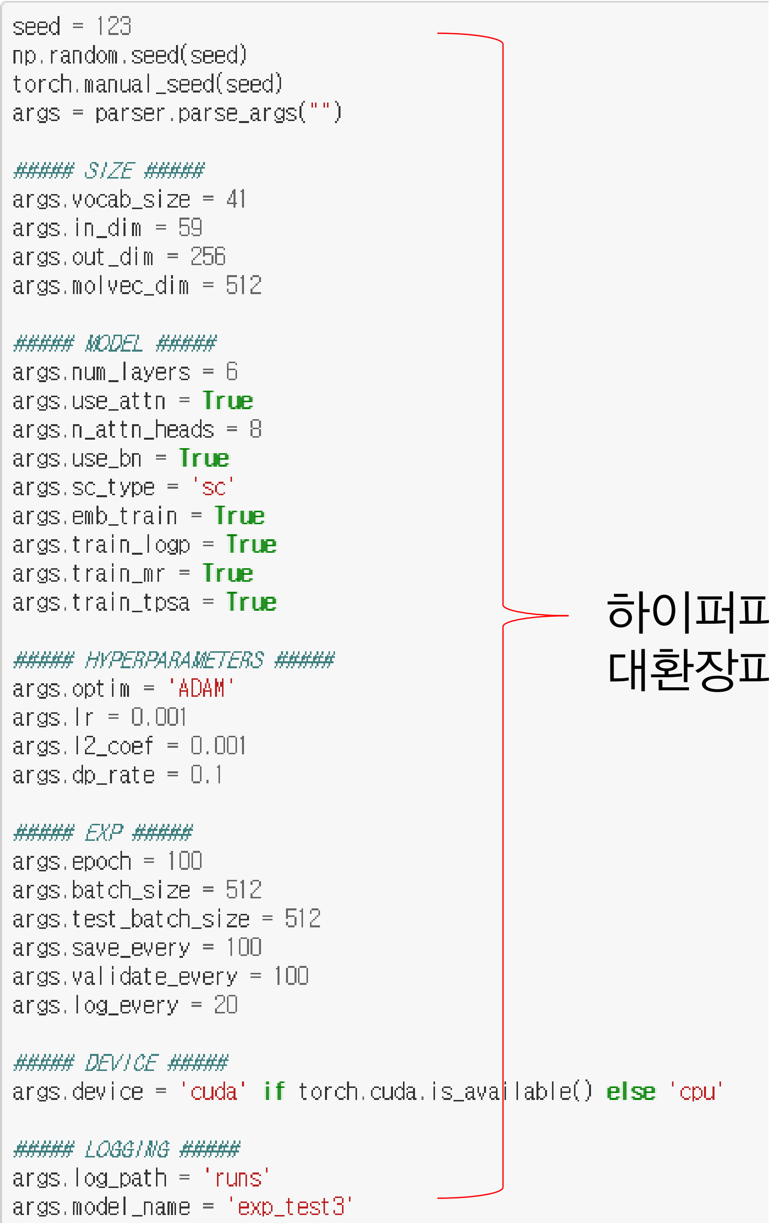


```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True

##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

하이퍼파라미터
대환장파티

https://github.com/heartcored98/COMET-pytorch/blob/master/pretraining_GCN.ipynb

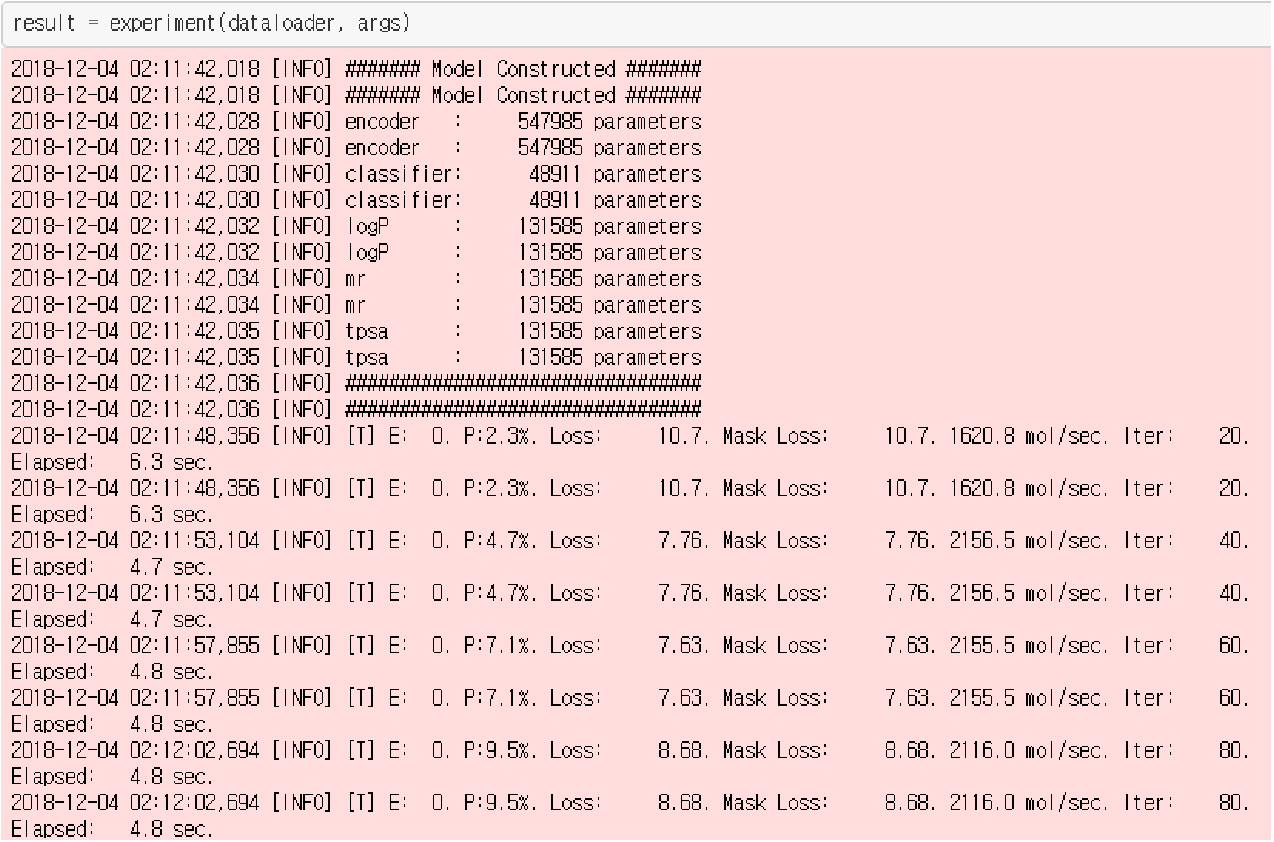
What is Parametrization?



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True

##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

하이퍼파라미터
대환장파티

실험 조건(args)을 input으로 받으면
알아서 모델 생성부터 실험 결과 리포팅까지..!



```
result = experiment(dataloader, args)
```

```
2018-12-04 02:11:42,018 [INFO] ####### Model Constructed #######
2018-12-04 02:11:42,018 [INFO] ####### Model Constructed #######
2018-12-04 02:11:42,028 [INFO] encoder   :     547985 parameters
2018-12-04 02:11:42,028 [INFO] encoder   :     547985 parameters
2018-12-04 02:11:42,030 [INFO] classifier:      48911 parameters
2018-12-04 02:11:42,030 [INFO] classifier:      48911 parameters
2018-12-04 02:11:42,032 [INFO] logP      :     131585 parameters
2018-12-04 02:11:42,032 [INFO] logP      :     131585 parameters
2018-12-04 02:11:42,034 [INFO] mr        :     131585 parameters
2018-12-04 02:11:42,034 [INFO] mr        :     131585 parameters
2018-12-04 02:11:42,035 [INFO] tpsa      :     131585 parameters
2018-12-04 02:11:42,035 [INFO] tpsa      :     131585 parameters
2018-12-04 02:11:42,036 [INFO] #################################
2018-12-04 02:11:42,036 [INFO] #################################
2018-12-04 02:11:48,356 [INFO] [T] E:  0. P:2.3%. Loss:     10.7. Mask Loss:     10.7. 1620.8 mol/sec. Iter:    20.
Elapsed:   6.3 sec.
2018-12-04 02:11:48,356 [INFO] [T] E:  0. P:2.3%. Loss:     10.7. Mask Loss:     10.7. 1620.8 mol/sec. Iter:    20.
Elapsed:   6.3 sec.
2018-12-04 02:11:53,104 [INFO] [T] E:  0. P:4.7%. Loss:     7.76. Mask Loss:     7.76. 2156.5 mol/sec. Iter:    40.
Elapsed:   4.7 sec.
2018-12-04 02:11:53,104 [INFO] [T] E:  0. P:4.7%. Loss:     7.76. Mask Loss:     7.76. 2156.5 mol/sec. Iter:    40.
Elapsed:   4.7 sec.
2018-12-04 02:11:57,855 [INFO] [T] E:  0. P:7.1%. Loss:     7.63. Mask Loss:     7.63. 2155.5 mol/sec. Iter:    60.
Elapsed:   4.8 sec.
2018-12-04 02:11:57,855 [INFO] [T] E:  0. P:7.1%. Loss:     7.63. Mask Loss:     7.63. 2155.5 mol/sec. Iter:    60.
Elapsed:   4.8 sec.
2018-12-04 02:12:02,694 [INFO] [T] E:  0. P:9.5%. Loss:     8.68. Mask Loss:     8.68. 2116.0 mol/sec. Iter:    80.
Elapsed:   4.8 sec.
2018-12-04 02:12:02,694 [INFO] [T] E:  0. P:9.5%. Loss:     8.68. Mask Loss:     8.68. 2116.0 mol/sec. Iter:    80.
Elapsed:   4.8 sec.
```

https://github.com/heartcored98/COMET-pytorch/blob/master/pretraining_GCN.ipynb

Why Parametrization?

- Manage various experiment in one-place

Why Parametrization?

- Manage various experiment in one-place
- Prevent possible situation where hyperparameter is fixed inside the model

Why Parametrization?

- Manage various experiment in one-place

- Prevent possible situation where hyperparameter is fixed inside the model

- Re-usability

Why Parametrization?

- Manage various experiment in one-place
- Prevent possible situation where hyperparameter is fixed inside the model
- Re-usability
- Readability

Why Parametrization?

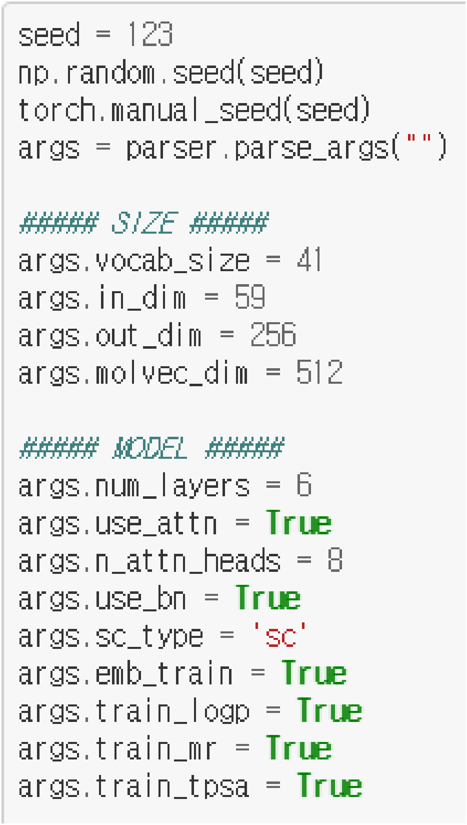
- Manage various experiment in one-place
- Prevent possible situation where hyperparameter is fixed inside the model
- Re-usability
- Readability
- Auto hyperparameter optimizer could conduct various experiments instead of us

Okay, Which parameter?

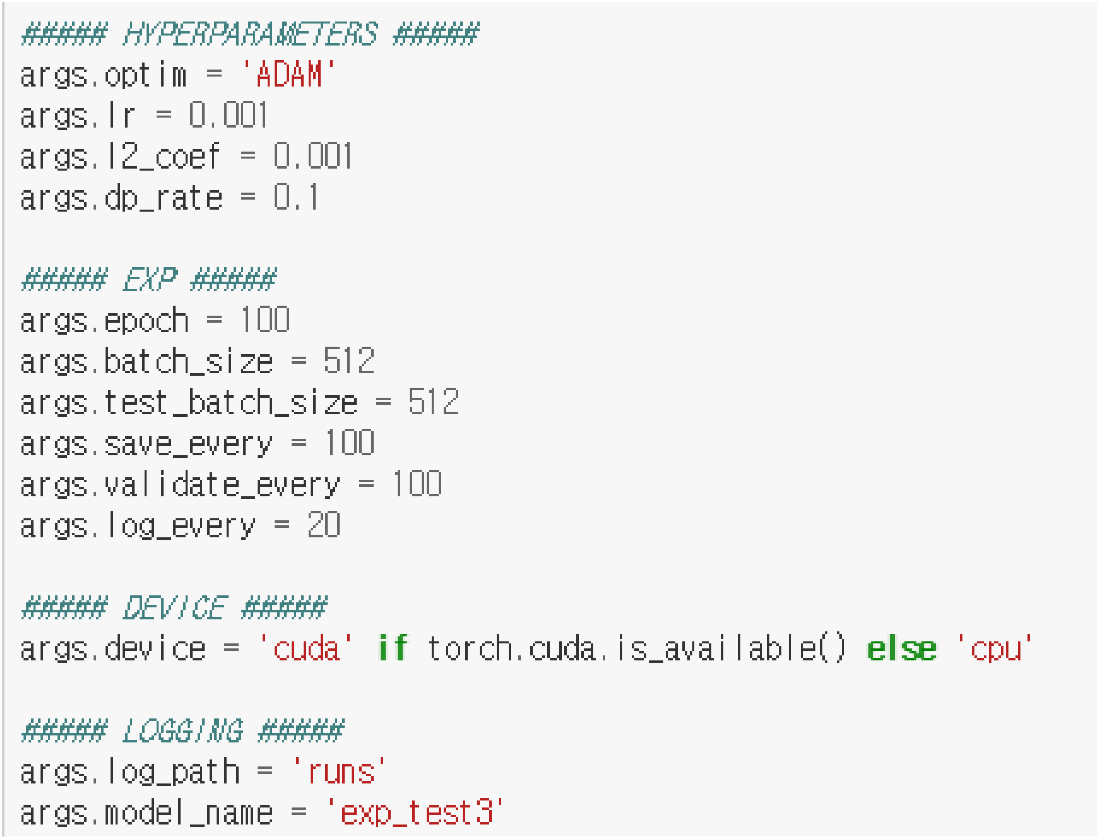
Okay, Which parameter?

Hyperparameter!

Hyperparameter

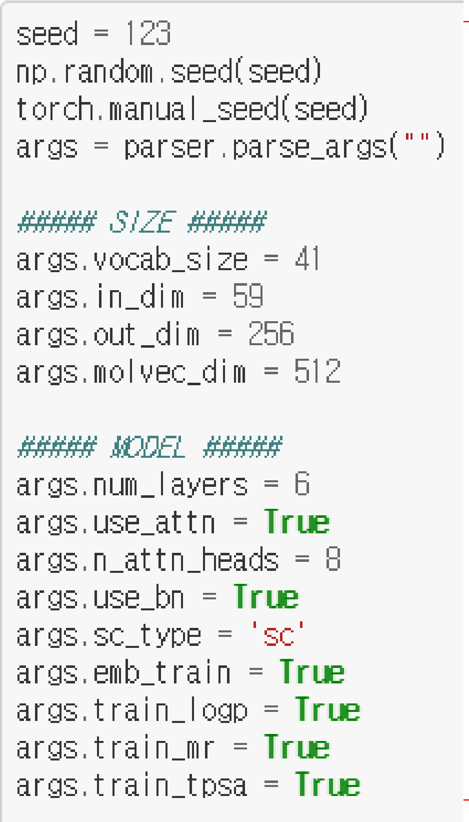


```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```



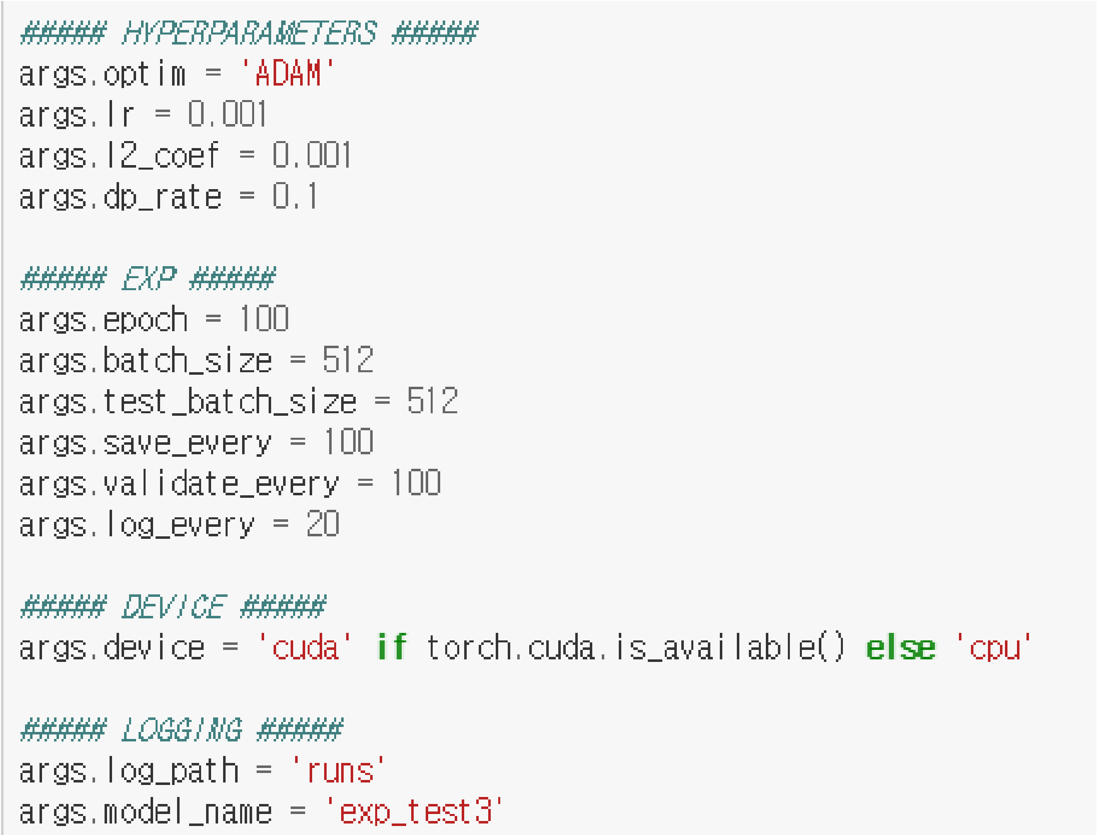
```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

Hyperparameter



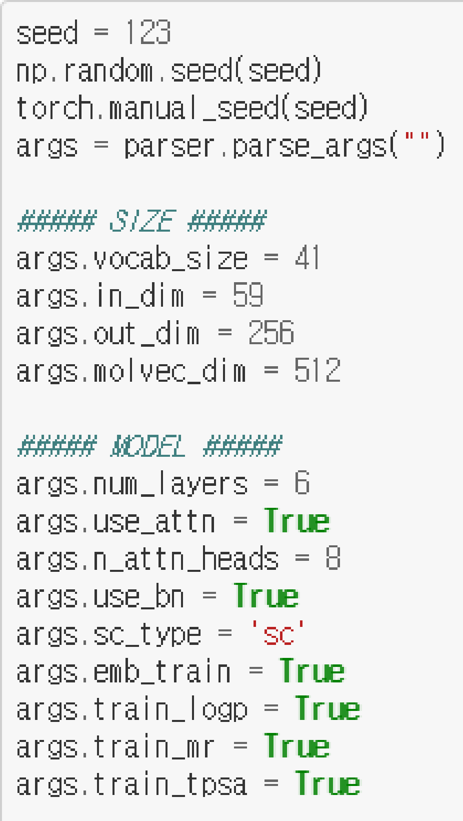
```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Determine Model



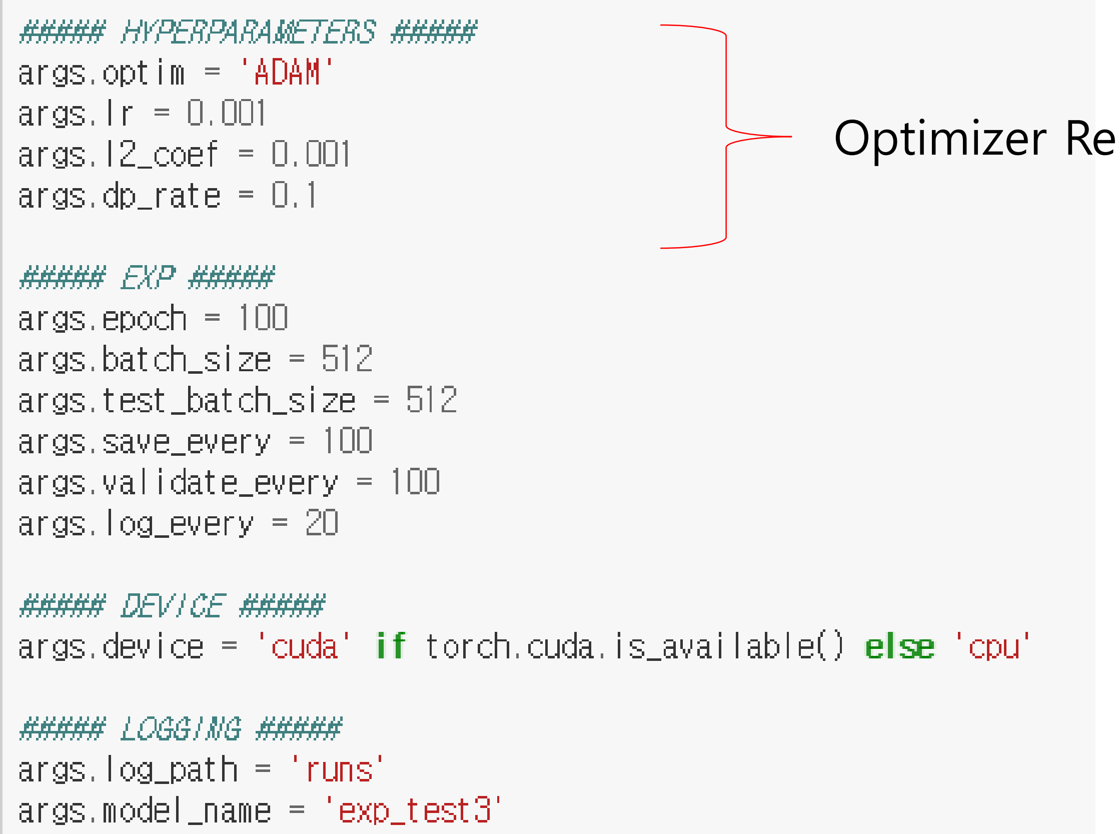
```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

Hyperparameter



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

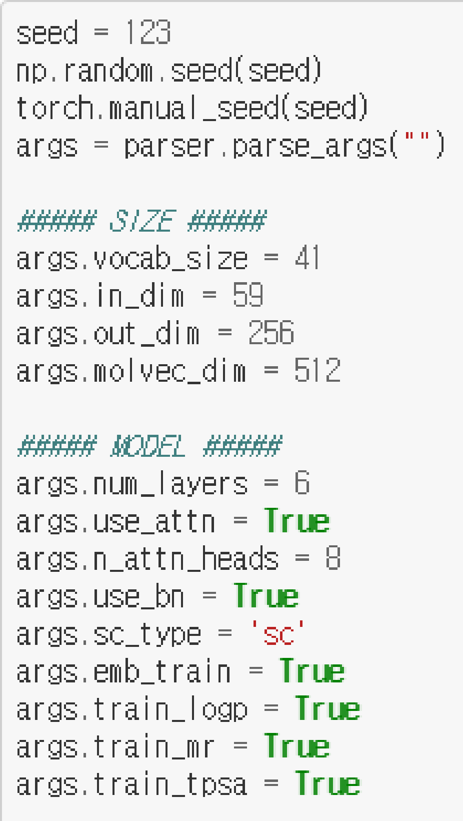
Determine Model



```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

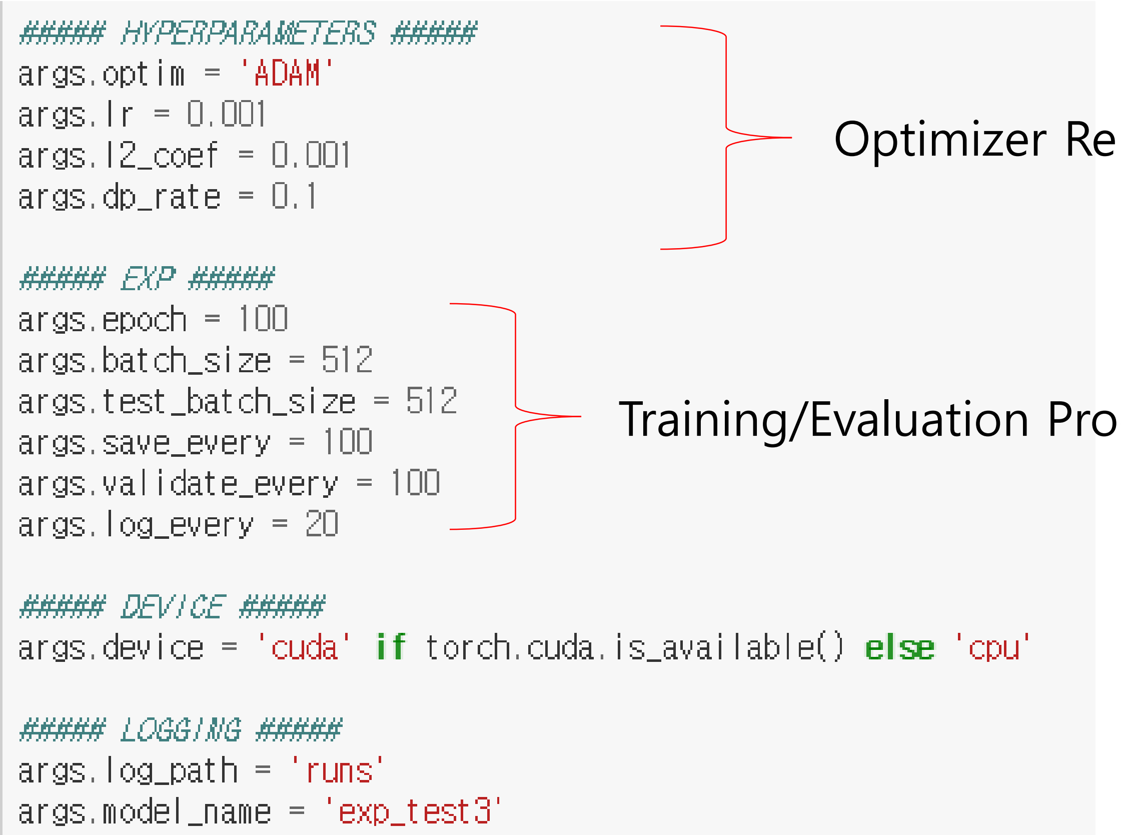
Optimizer Related

Hyperparameter



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Determine Model

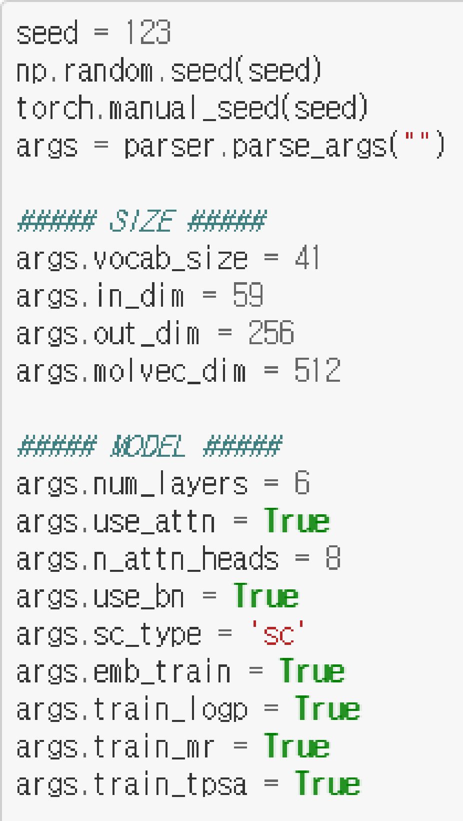


```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

Optimizer Related

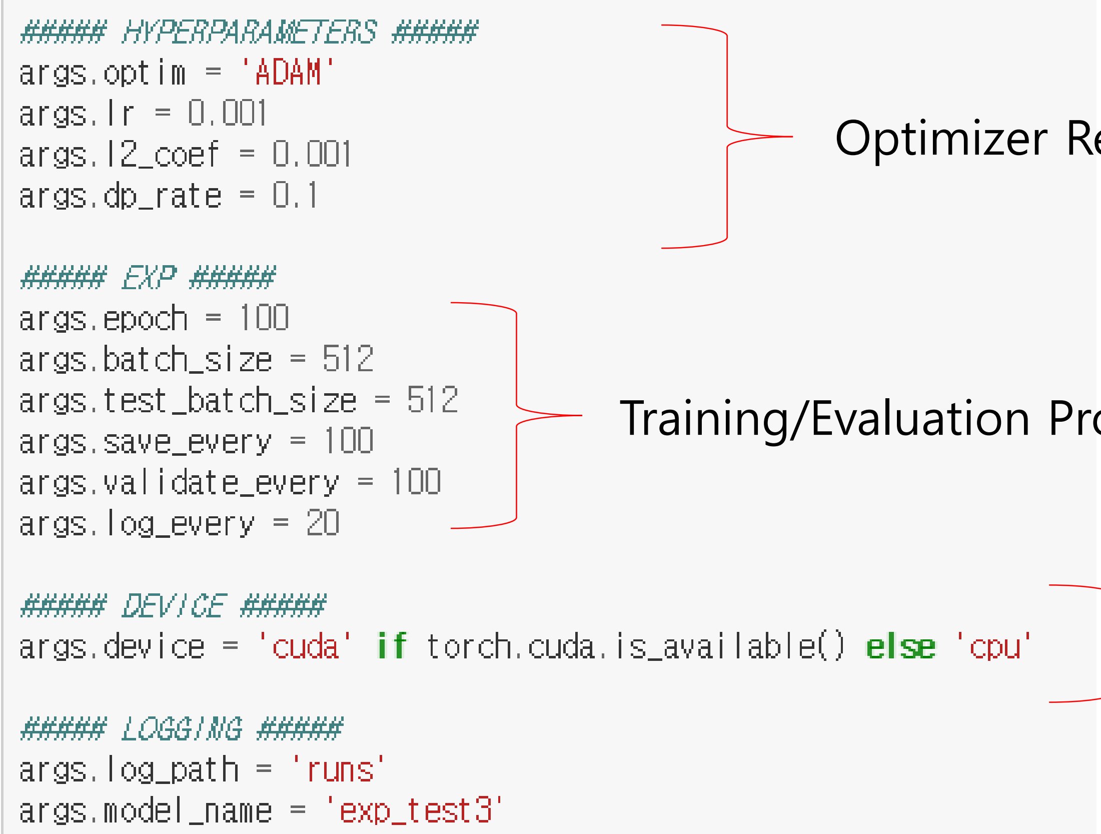
Training/Evaluation Process Related

Hyperparameter



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Determine Model



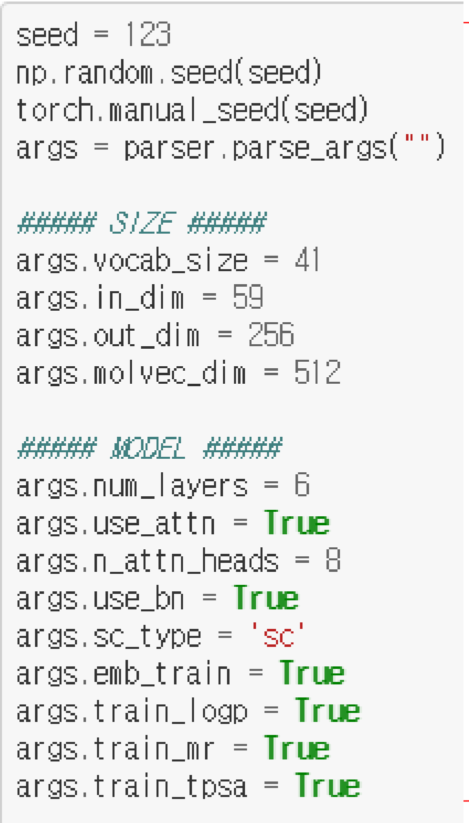
```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

Optimizer Related

Training/Evaluation Process Related

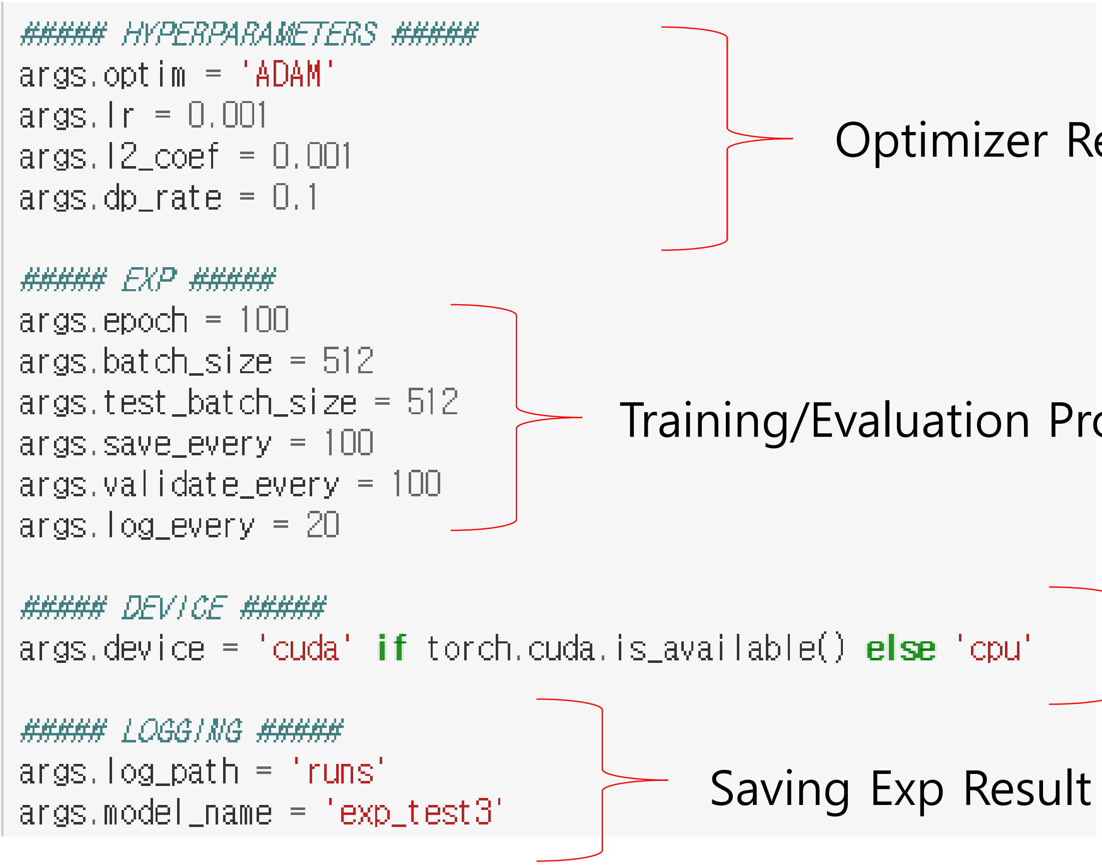
Device

Hyperparameter



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Determine Model



```
##### HYPERPARAMETERS #####
args.optim = 'ADAM'
args.lr = 0.001
args.l2_coef = 0.001
args.dp_rate = 0.1

##### EXP #####
args.epoch = 100
args.batch_size = 512
args.test_batch_size = 512
args.save_every = 100
args.validate_every = 100
args.log_every = 20

##### DEVICE #####
args.device = 'cuda' if torch.cuda.is_available() else 'cpu'

##### LOGGING #####
args.log_path = 'runs'
args.model_name = 'exp_test3'
```

Optimizer Related

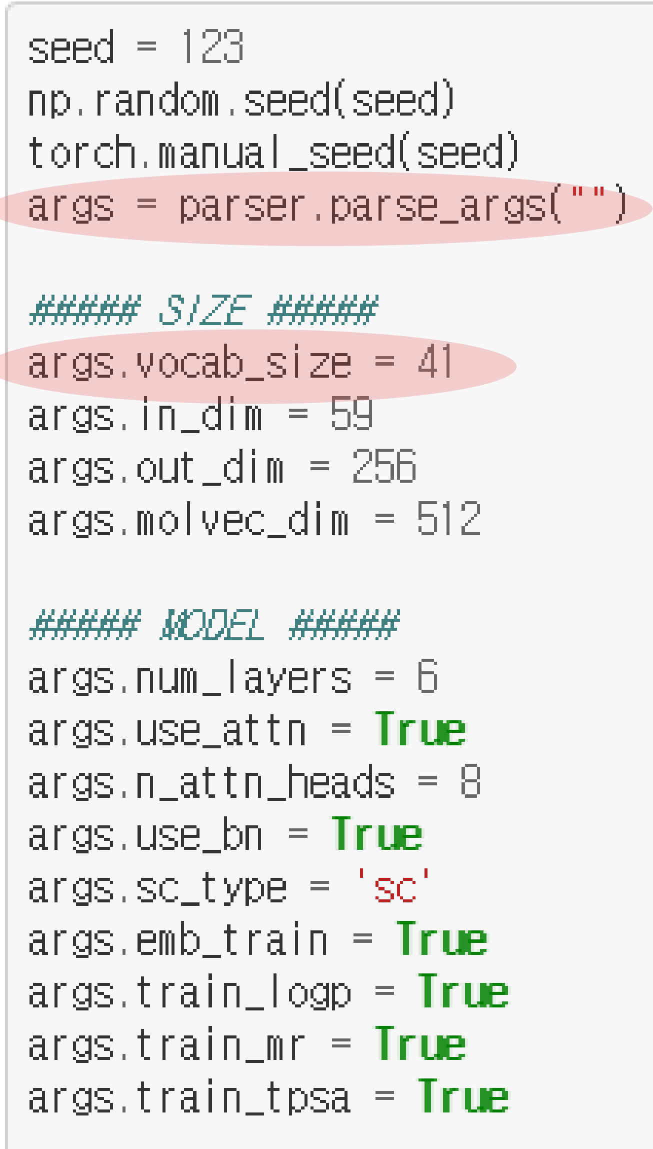
Training/Evaluation Process Related

Device

Saving Exp Result

Okay, Then.. How?

Introduce Argparse

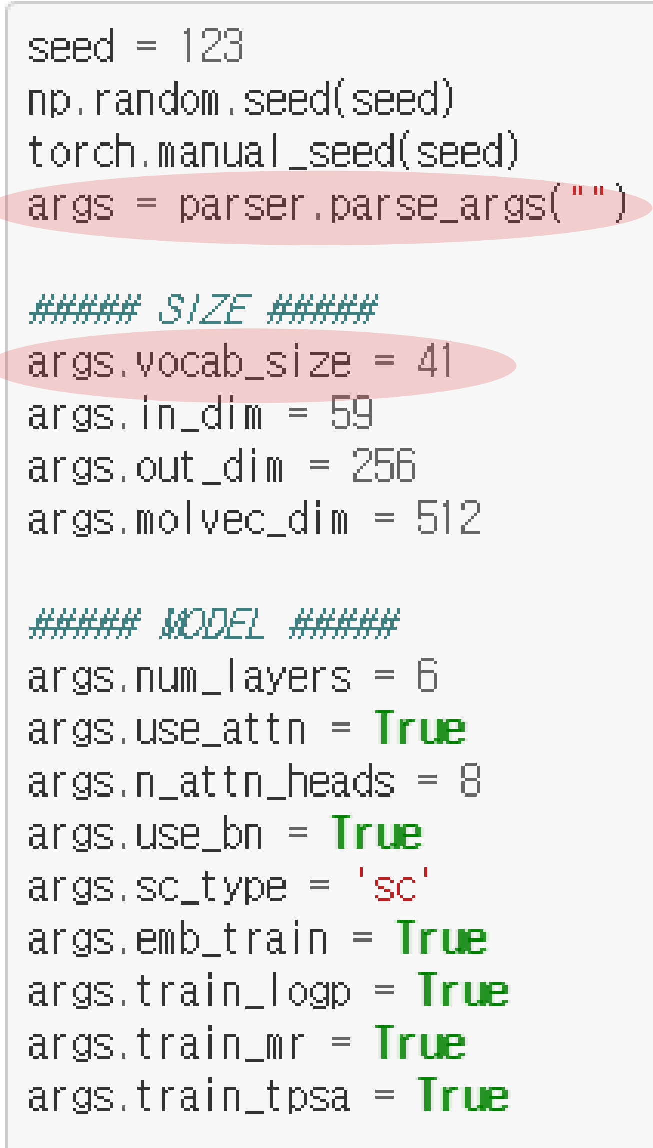


```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Argparse : Dictionary like object

Import argparse!
Let`s try below code to see what argparse can do!

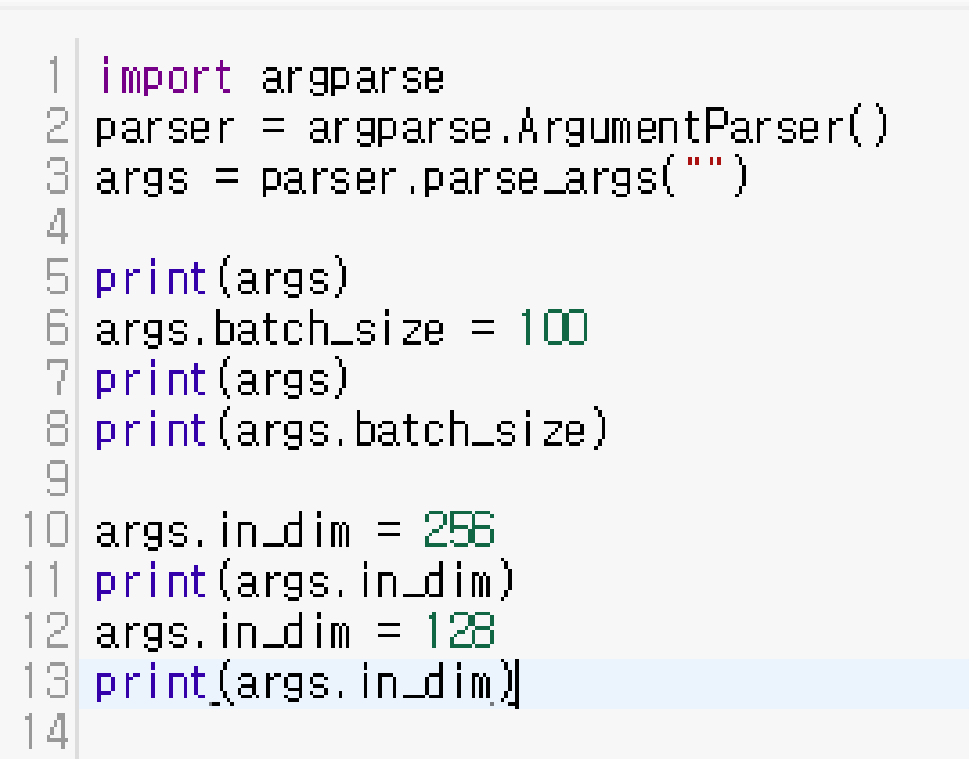
Introduce Argparse



```
seed = 123
np.random.seed(seed)
torch.manual_seed(seed)
args = parser.parse_args("")

##### SIZE #####
args.vocab_size = 41
args.in_dim = 59
args.out_dim = 256
args.molvec_dim = 512

##### MODEL #####
args.num_layers = 6
args.use_attn = True
args.n_attn_heads = 8
args.use_bn = True
args.sc_type = 'sc'
args.emb_train = True
args.train_logp = True
args.train_mr = True
args.train_tpsa = True
```

Argparse : Dictionary like object

Import argparse!
Let`s try below code to see what argparse can do!



```
import argparse
parser = argparse.ArgumentParser()
args = parser.parse_args("")

print(args)
args.batch_size = 100
print(args)
print(args.batch_size)

args.in_dim = 256
print(args.in_dim)
args.in_dim = 128
print(args.in_dim)

```

Experiment Manager?

Function. Experiment

- Construct Model
- Construct Loss Function
- Construct Optimizer
- Start Training and Report Progress
- Report Evaluation Result

Experiment Manager?

Function. Experiment

- Construct Model ← nn.ModuleList
- Construct Loss Function
- Construct Optimizer
- Start Training and Report Progress
- Report Evaluation Result

Experiment Manager?

Function. Experiment

- Construct Model
- Construct Loss Function
- Construct Optimizer
- Start Training and Report Progress
- Report Evaluation Result

Rest of parameterization is full of ‘if’ statement

Summary

- Do not let the hyperparameter value live inside the code
- Use Argparse to gather and manage hyperparameters