

```
In [9]: print('Baseline')

# This is so that you don't have to restart the kernel everytime you edit hmm.py

%load_ext autoreload
%autoreload 2
```

Baseline

The autoreload extension is already loaded. To reload it, use:
 %reload_ext autoreload

```
In [ ]:
```

```
In [10]: from preprocess import load_arguments, load_label, load_values_from_json, combine_columns
from model import train_bert_model
import os

# Load arguments
df_arguments = load_arguments('data/arguments-training.tsv')

# Load json
human_values = load_values_from_json('data/value-categories.json')

df_labels = load_label("data/labels-training.tsv", human_values["1"])
df_full_level = combine_columns(df_arguments, df_labels)
train_arguments, valid_arguments = split_arguments(df_full_level)

train_arguments.info()
valid_arguments.info()
```

```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 4176 entries, 3401 to 3031
Data columns (total 24 columns):
 #   Column           Non-Null Count Dtype  
--- 
 0   Argument ID      4176 non-null  object  
 1   Conclusion       4176 non-null  object  
 2   Stance            4176 non-null  object  
 3   Premise           4176 non-null  object  
 4   Achievement       4176 non-null  int64  
 5   Benevolence: caring 4176 non-null  int64  
 6   Benevolence: dependability 4176 non-null  int64  
 7   Conformity: interpersonal 4176 non-null  int64  
 8   Conformity: rules     4176 non-null  int64  
 9   Face               4176 non-null  int64  
 10  Hedonism           4176 non-null  int64  
 11  Humility            4176 non-null  int64  
 12  Power: dominance    4176 non-null  int64  
 13  Power: resources    4176 non-null  int64  
 14  Security: personal  4176 non-null  int64  
 15  Security: societal  4176 non-null  int64  
 16  Self-direction: action 4176 non-null  int64  
 17  Self-direction: thought 4176 non-null  int64  
 18  Stimulation         4176 non-null  int64  
 19  Tradition           4176 non-null  int64  
 20  Universalism: concern 4176 non-null  int64  
 21  Universalism: nature  4176 non-null  int64  
 22  Universalism: objectivity 4176 non-null  int64  
 23  Universalism: tolerance 4176 non-null  int64  
dtypes: int64(20), object(4)
memory usage: 815.6+ KB
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1044 entries, 2 to 5210
Data columns (total 24 columns):
 #   Column           Non-Null Count Dtype  
--- 
 0   Argument ID      1044 non-null  object  
 1   Conclusion       1044 non-null  object  
 2   Stance            1044 non-null  object  
 3   Premise           1044 non-null  object  
 4   Achievement       1044 non-null  int64  
 5   Benevolence: caring 1044 non-null  int64  
 6   Benevolence: dependability 1044 non-null  int64  
 7   Conformity: interpersonal 1044 non-null  int64  
 8   Conformity: rules     1044 non-null  int64  
 9   Face               1044 non-null  int64  
 10  Hedonism           1044 non-null  int64  
 11  Humility            1044 non-null  int64  
 12  Power: dominance    1044 non-null  int64  
 13  Power: resources    1044 non-null  int64  
 14  Security: personal  1044 non-null  int64  
 15  Security: societal  1044 non-null  int64  
 16  Self-direction: action 1044 non-null  int64  
 17  Self-direction: thought 1044 non-null  int64  
 18  Stimulation         1044 non-null  int64  
 19  Tradition           1044 non-null  int64  
 20  Universalism: concern 1044 non-null  int64

```

```
21 Universalism: nature      1044 non-null    int64
22 Universalism: objectivity 1044 non-null    int64
23 Universalism: tolerance   1044 non-null    int64
dtypes: int64(20), object(4)
memory usage: 203.9+ KB
```

Random Baseline

```
In [11]: from pprint import pprint

from baseline import random_prediction
from metrics import compute_metrics, print_metrics
from utils import extract_true_labels


random_prediction_values = random_prediction(human_values['1'], train_arguments)

metrics = compute_metrics((random_prediction_values,
                           extract_true_labels(train_arguments, human_values['1'])), k=100)
pprint(metrics)

Random Prediction
Added predictions for 4176 arguments
{'accuracy_thresh': 0.27721503376960754,
 'f1-score': {'Achievement': 0.403,
              'Benevolence: caring': 0.401,
              'Benevolence: dependability': 0.254,
              'Conformity: interpersonal': 0.082,
              'Conformity: rules': 0.356,
              'Face': 0.135,
              'Hedonism': 0.078,
              'Humility': 0.157,
              'Power: dominance': 0.16,
              'Power: resources': 0.182,
              'Security: personal': 0.524,
              'Security: societal': 0.447,
              'Self-direction: action': 0.408,
              'Self-direction: thought': 0.297,
              'Stimulation': 0.107,
              'Tradition': 0.197,
              'Universalism: concern': 0.476,
              'Universalism: nature': 0.124,
              'Universalism: objectivity': 0.29,
              'Universalism: tolerance': 0.227,
              'avg-f1-score': 0.265},
 'marco-avg-f1score': 0.265}
```

All 1 predictions

```
In [12]: from baseline import all_ones

one_values = all_ones(human_values['1'], train_arguments)
```

```
metrics = compute_metrics((one_values,
                           extract_true_labels(train_arguments, human_values['1'])), True)
pprint(metrics)

Predicting all 1
Added predictions for 4176 arguments
{'accuracy_thresh': 0.17182710766792297,
 'f1-score': {'Achievement': 0.427,
              'Benevolence: caring': 0.452,
              'Benevolence: dependability': 0.258,
              'Conformity: interpersonal': 0.081,
              'Conformity: rules': 0.376,
              'Face': 0.135,
              'Hedonism': 0.08,
              'Humility': 0.153,
              'Power: dominance': 0.164,
              'Power: resources': 0.193,
              'Security: personal': 0.547,
              'Security: societal': 0.477,
              'Self-direction: action': 0.407,
              'Self-direction: thought': 0.304,
              'Stimulation': 0.114,
              'Tradition': 0.2,
              'Universalism: concern': 0.55,
              'Universalism: nature': 0.129,
              'Universalism: objectivity': 0.301,
              'Universalism: tolerance': 0.242,
              'avg-f1-score': 0.28},
 'marco-avg-f1score': 0.28}
```

```
In [ ]: bert_model_evaluation = train_bert_model(train_arguments, os.path.join('model/', 'praj')
print(bert_model_evaluation['eval_f1-score'])

0% | 0/4176 [00:00<?, ?ex/s]
0% | 0/1044 [00:00<?, ?ex/s]
0% | 0/5 [00:00<?, ?ba/s]
0% | 0/2 [00:00<?, ?ba/s]
```

```
using `logging_steps` to initialize `eval_steps` to 500
PyTorch: setting up devices
The default value for the training argument `--report_to` will change in v5 (from all
installed integrations to none). In v5, you will need to use `--report_to all` to get
the same behavior as now. You should start updating your code and make this info disa
ppear :-).
loading configuration file config.json from cache at C:\Users\akshi/.cache\huggingfac
e\hub\models--prajjwal1--bert-small\snapshots\0ec5f86f27c1a77d704439db5e01c307ea11b9d
4\config.json
Model config BertConfig {
    "_name_or_path": "prajjwal1/bert-small",
    "attention_probs_dropout_prob": 0.1,
    "classifier_dropout": null,
    "hidden_act": "gelu",
    "hidden_dropout_prob": 0.1,
    "hidden_size": 512,
    "id2label": {
        "0": "LABEL_0",
        "1": "LABEL_1",
        "2": "LABEL_2",
        "3": "LABEL_3",
        "4": "LABEL_4",
        "5": "LABEL_5",
        "6": "LABEL_6",
        "7": "LABEL_7",
        "8": "LABEL_8",
        "9": "LABEL_9",
        "10": "LABEL_10",
        "11": "LABEL_11",
        "12": "LABEL_12",
        "13": "LABEL_13",
        "14": "LABEL_14",
        "15": "LABEL_15",
        "16": "LABEL_16",
        "17": "LABEL_17",
        "18": "LABEL_18",
        "19": "LABEL_19"
    },
    "initializer_range": 0.02,
    "intermediate_size": 2048,
    "label2id": {
        "LABEL_0": 0,
        "LABEL_1": 1,
        "LABEL_10": 10,
        "LABEL_11": 11,
        "LABEL_12": 12,
        "LABEL_13": 13,
        "LABEL_14": 14,
        "LABEL_15": 15,
        "LABEL_16": 16,
        "LABEL_17": 17,
        "LABEL_18": 18,
        "LABEL_19": 19,
        "LABEL_2": 2,
        "LABEL_3": 3,
        "LABEL_4": 4,
        "LABEL_5": 5,
```

```
"LABEL_6": 6,
"LABEL_7": 7,
"LABEL_8": 8,
"LABEL_9": 9
},
"layer_norm_eps": 1e-12,
"max_position_embeddings": 512,
"model_type": "bert",
"num_attention_heads": 8,
"num_hidden_layers": 4,
"pad_token_id": 0,
"position_embedding_type": "absolute",
"transformers_version": "4.24.0",
"type_vocab_size": 2,
"use_cache": true,
"vocab_size": 30522
}

loading weights file pytorch_model.bin from cache at C:\Users\akshi/.cache\huggingface\hub\models--prajjwal1--bert-small\snapshots\0ec5f86f27c1a77d704439db5e01c307ea11b9d4\pytorch_model.bin
Some weights of the model checkpoint at prajjwal1/bert-small were not used when initializing BertForSequenceClassification: ['cls.seq_relationship.weight', 'cls.predictions.transform.LayerNorm.bias', 'cls.predictions.transform.dense.weight', 'cls.seq_relationship.bias', 'cls.predictions.transform.LayerNorm.weight', 'cls.predictions.decode_r.weight', 'cls.predictions.transform.dense.bias', 'cls.predictions.decoder.bias', 'cls.predictions.bias']
- This IS expected if you are initializing BertForSequenceClassification from the checkpoint of a model trained on another task or with another architecture (e.g. initializing a BertForSequenceClassification model from a BertForPreTraining model).
- This IS NOT expected if you are initializing BertForSequenceClassification from the checkpoint of a model that you expect to be exactly identical (initializing a BertForSequenceClassification model from a BertForSequenceClassification model).
Some weights of BertForSequenceClassification were not initialized from the model checkpoint at prajjwal1/bert-small and are newly initialized: ['classifier.bias', 'classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
C:\Users\akshi\anaconda3\envs\py39\lib\site-packages\transformers\optimization.py:30
6: FutureWarning: This implementation of AdamW is deprecated and will be removed in a future version. Use the PyTorch implementation torch.optim.AdamW instead, or set `no_deprecation_warning=True` to disable this warning
    warnings.warn(
***** Running training *****
    Num examples = 4176
    Num Epochs = 20
    Instantaneous batch size per device = 8
    Total train batch size (w. parallel, distributed & accumulation) = 8
    Gradient Accumulation steps = 1
    Total optimization steps = 10440
    Number of trainable parameters = 28773908
You're using a BertTokenizerFast tokenizer. Please note that with a fast tokenizer, using the `__call__` method is faster than using a method to encode the text followed by a call to the `pad` method to get a padded encoding.
```

[9268/10440 2:10:43 < 16:32, 1.18 it/s, Epoch

17.75/20]

Step	Training Loss	Validation Loss	Accuracy Thresh	F1-score	Marco-avg-f1score
500	0.430600	0.380950	0.846169	{'Achievement': 0.27, 'Benevolence: caring': 0.05, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.0, 'Face': 0.0, 'Hedonism': 0.0, 'Humility': 0.0, 'Power: dominance': 0.0, 'Power: resources': 0.0, 'Security: personal': 0.59, 'Security: societal': 0.5, 'Self-direction: action': 0.02, 'Self-direction: thought': 0.0, 'Stimulation': 0.0, 'Tradition': 0.0, 'Universalism: concern': 0.52, 'Universalism: nature': 0.0, 'Universalism: objectivity': 0.0, 'Universalism: tolerance': 0.0, 'avg-f1-score': 0.1}	0.100000
1000	0.372600	0.354642	0.857711	{'Achievement': 0.58, 'Benevolence: caring': 0.28, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.05, 'Face': 0.0, 'Hedonism': 0.0, 'Humility': 0.0, 'Power: dominance': 0.0, 'Power: resources': 0.29, 'Security: personal': 0.68, 'Security: societal': 0.56, 'Self-direction: action': 0.35, 'Self-direction: thought': 0.46, 'Stimulation': 0.0, 'Tradition': 0.04, 'Universalism: concern': 0.51, 'Universalism: nature': 0.27, 'Universalism: objectivity': 0.0, 'Universalism: tolerance': 0.0, 'avg-f1-score': 0.2}	0.200000
1500	0.341400	0.341015	0.863171	{'Achievement': 0.56, 'Benevolence: caring': 0.37, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.22, 'Face': 0.0, 'Hedonism': 0.0, 'Humility': 0.0, 'Power: dominance': 0.0, 'Power: resources': 0.32, 'Security: personal': 0.68, 'Security: societal': 0.67, 'Self-direction: action': 0.46, 'Self-direction: thought': 0.55, 'Stimulation': 0.0, 'Tradition': 0.15, 'Universalism: concern': 0.66, 'Universalism: nature': 0.7, 'Universalism: objectivity': 0.1, 'Universalism: tolerance': 0.03, 'avg-f1-score': 0.27}	0.270000
2000	0.323300	0.334286	0.866188	{'Achievement': 0.58, 'Benevolence: caring': 0.42, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.27, 'Face': 0.0, 'Hedonism': 0.0, 'Humility': 0.0, 'Power: dominance': 0.0, 'Power: resources': 0.44, 'Security: personal': 0.7, 'Security: societal': 0.63, 'Self-direction: action': 0.59, 'Self-direction: thought': 0.57, 'Stimulation': 0.0, 'Tradition': 0.3, 'Universalism: concern': 0.62, 'Universalism: nature': 0.72, 'Universalism: objectivity': 0.17, 'Universalism: tolerance': 0.15, 'avg-f1-score': 0.31}	0.310000
2500	0.303000	0.330607	0.866715	{'Achievement': 0.6, 'Benevolence: caring': 0.44, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.35, 'Face': 0.0, 'Hedonism': 0.13, 'Humility': 0.0, 'Power: dominance': 0.04, 'Power: resources': 0.49, 'Security: personal': 0.69, 'Security: societal': 0.65, 'Self-direction: action': 0.54, 'Self-direction: thought': 0.59, 'Stimulation': 0.03, 'Tradition': 0.34, 'Universalism: concern': 0.69, 'Universalism: nature': 0.74, 'Universalism: objectivity': 0.17, 'Universalism: tolerance': 0.15, 'avg-f1-score': 0.34}	0.340000

Step	Training Loss	Validation Loss	Accuracy Thresh	F1-score	Marco- avg-f1score
				0.21, 'Universalism: tolerance': 0.19, 'avg-f1-score': 0.34}	
3000	0.288100	0.333998	0.864847	{'Achievement': 0.61, 'Benevolence: caring': 0.45, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.38, 'Face': 0.0, 'Hedonism': 0.18, 'Humility': 0.0, 'Power: dominance': 0.06, 'Power: resources': 0.51, 'Security: personal': 0.67, 'Security: societal': 0.65, 'Self-direction: action': 0.57, 'Self-direction: thought': 0.61, 'Stimulation': 0.1, 'Tradition': 0.41, 'Universalism: concern': 0.69, 'Universalism: nature': 0.77, 'Universalism: objectivity': 0.28, 'Universalism: tolerance': 0.21, 'avg-f1-score': 0.36}	0.360000
3500	0.272500	0.328541	0.867433	{'Achievement': 0.61, 'Benevolence: caring': 0.44, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.39, 'Face': 0.0, 'Hedonism': 0.13, 'Humility': 0.0, 'Power: dominance': 0.04, 'Power: resources': 0.49, 'Security: personal': 0.68, 'Security: societal': 0.66, 'Self-direction: action': 0.53, 'Self-direction: thought': 0.6, 'Stimulation': 0.03, 'Tradition': 0.41, 'Universalism: concern': 0.7, 'Universalism: nature': 0.77, 'Universalism: objectivity': 0.23, 'Universalism: tolerance': 0.18, 'avg-f1-score': 0.34}	0.340000
4000	0.261700	0.329824	0.866188	{'Achievement': 0.6, 'Benevolence: caring': 0.37, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.37, 'Face': 0.0, 'Hedonism': 0.13, 'Humility': 0.0, 'Power: dominance': 0.06, 'Power: resources': 0.49, 'Security: personal': 0.7, 'Security: societal': 0.67, 'Self-direction: action': 0.51, 'Self-direction: thought': 0.55, 'Stimulation': 0.03, 'Tradition': 0.38, 'Universalism: concern': 0.69, 'Universalism: nature': 0.75, 'Universalism: objectivity': 0.28, 'Universalism: tolerance': 0.26, 'avg-f1-score': 0.34}	0.340000
4500	0.246700	0.333564	0.865517	{'Achievement': 0.6, 'Benevolence: caring': 0.47, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.44, 'Face': 0.0, 'Hedonism': 0.12, 'Humility': 0.0, 'Power: dominance': 0.08, 'Power: resources': 0.47, 'Security: personal': 0.69, 'Security: societal': 0.68, 'Self-direction: action': 0.53, 'Self-direction: thought': 0.59, 'Stimulation': 0.07, 'Tradition': 0.44, 'Universalism: concern': 0.71, 'Universalism: nature': 0.75, 'Universalism: objectivity': 0.22, 'Universalism: tolerance': 0.25, 'avg-f1-score': 0.36}	0.360000
5000	0.237800	0.334196	0.865661	{'Achievement': 0.61, 'Benevolence: caring': 0.47, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.39, 'Face': 0.0, 'Hedonism': 0.18, 'Humility': 0.02, 'Power: dominance': 0.1, 'Power: resources': 0.47, 'Security: personal': 0.68, 'Security: societal': 0.67, 'Self-direction: action': 0.56, 'Self-direction: thought': 0.62, 'Stimulation': 0.12,	0.370000

Step	Training Loss	Validation Loss	Accuracy Thresh	F1-score	Marco- avg-f1score
				{'Tradition': 0.38, 'Universalism: concern': 0.71, 'Universalism: nature': 0.77, 'Universalism: objectivity': 0.29, 'Universalism: tolerance': 0.27, 'avg-f1-score': 0.37}	
5500	0.226600	0.336163	0.864416	{'Achievement': 0.61, 'Benevolence: caring': 0.44, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.09, 'Conformity: rules': 0.38, 'Face': 0.03, 'Hedonism': 0.23, 'Humility': 0.02, 'Power: dominance': 0.1, 'Power: resources': 0.44, 'Security: personal': 0.68, 'Security: societal': 0.67, 'Self-direction: action': 0.55, 'Self-direction: thought': 0.61, 'Stimulation': 0.12, 'Tradition': 0.44, 'Universalism: concern': 0.7, 'Universalism: nature': 0.76, 'Universalism: objectivity': 0.33, 'Universalism: tolerance': 0.3, 'avg-f1-score': 0.38}	0.380000
6000	0.216600	0.340302	0.865326	{'Achievement': 0.61, 'Benevolence: caring': 0.47, 'Benevolence: dependability': 0.0, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.39, 'Face': 0.03, 'Hedonism': 0.18, 'Humility': 0.02, 'Power: dominance': 0.08, 'Power: resources': 0.48, 'Security: personal': 0.69, 'Security: societal': 0.68, 'Self-direction: action': 0.55, 'Self-direction: thought': 0.61, 'Stimulation': 0.09, 'Tradition': 0.37, 'Universalism: concern': 0.69, 'Universalism: nature': 0.76, 'Universalism: objectivity': 0.31, 'Universalism: tolerance': 0.28, 'avg-f1-score': 0.36}	0.360000
6500	0.211700	0.343114	0.861207	{'Achievement': 0.61, 'Benevolence: caring': 0.46, 'Benevolence: dependability': 0.03, 'Conformity: interpersonal': 0.0, 'Conformity: rules': 0.41, 'Face': 0.08, 'Hedonism': 0.22, 'Humility': 0.04, 'Power: dominance': 0.13, 'Power: resources': 0.48, 'Security: personal': 0.68, 'Security: societal': 0.66, 'Self-direction: action': 0.53, 'Self-direction: thought': 0.61, 'Stimulation': 0.2, 'Tradition': 0.43, 'Universalism: concern': 0.7, 'Universalism: nature': 0.78, 'Universalism: objectivity': 0.31, 'Universalism: tolerance': 0.3, 'avg-f1-score': 0.38}	0.380000
7000	0.202400	0.344509	0.862356	{'Achievement': 0.61, 'Benevolence: caring': 0.46, 'Benevolence: dependability': 0.04, 'Conformity: interpersonal': 0.04, 'Conformity: rules': 0.44, 'Face': 0.05, 'Hedonism': 0.22, 'Humility': 0.04, 'Power: dominance': 0.12, 'Power: resources': 0.46, 'Security: personal': 0.68, 'Security: societal': 0.65, 'Self-direction: action': 0.54, 'Self-direction: thought': 0.6, 'Stimulation': 0.2, 'Tradition': 0.45, 'Universalism: concern': 0.69, 'Universalism: nature': 0.77, 'Universalism: objectivity': 0.3, 'Universalism: tolerance': 0.31, 'avg-f1-score': 0.38}	0.380000
7500	0.196200	0.345777	0.864559	{'Achievement': 0.61, 'Benevolence: caring': 0.46, 'Benevolence: dependability': 0.01, 'Conformity: interpersonal': 0.04, 'Conformity: rules': 0.42, 'Face': 0.08, 'Hedonism': 0.18, 'Humility': 0.04, 'Power: dominance': 0.1, 'Power: resources': 0.48, 'Security:	0.380000

Step	Training Loss	Validation Loss	Accuracy Thresh	F1-score	Marco- avg-f1score
8000	0.193200	0.347192	0.862117	{'Achievement': 0.68, 'Security: societal': 0.66, 'Self-direction: action': 0.53, 'Self-direction: thought': 0.58, 'Stimulation': 0.23, 'Tradition': 0.42, 'Universalism: concern': 0.7, 'Universalism: nature': 0.77, 'Universalism: objectivity': 0.31, 'Universalism: tolerance': 0.31, 'avg-f1-score': 0.38}	0.380000
8500	0.186100	0.350327	0.861398	{'Achievement': 0.62, 'Benevolence: caring': 0.46, 'Benevolence: dependability': 0.07, 'Conformity: interpersonal': 0.04, 'Conformity: rules': 0.44, 'Face': 0.08, 'Hedonism': 0.18, 'Humility': 0.02, 'Power: dominance': 0.13, 'Power: resources': 0.46, 'Security: personal': 0.67, 'Security: societal': 0.66, 'Self-direction: action': 0.55, 'Self-direction: thought': 0.61, 'Stimulation': 0.2, 'Tradition': 0.4, 'Universalism: concern': 0.7, 'Universalism: nature': 0.76, 'Universalism: objectivity': 0.3, 'Universalism: tolerance': 0.28, 'avg-f1-score': 0.38}	0.390000
9000	0.184600	0.351487	0.860153	{'Achievement': 0.61, 'Benevolence: caring': 0.45, 'Benevolence: dependability': 0.07, 'Conformity: interpersonal': 0.08, 'Conformity: rules': 0.44, 'Face': 0.08, 'Hedonism': 0.22, 'Humility': 0.04, 'Power: dominance': 0.13, 'Power: resources': 0.46, 'Security: personal': 0.67, 'Security: societal': 0.66, 'Self-direction: action': 0.55, 'Self-direction: thought': 0.6, 'Stimulation': 0.22, 'Tradition': 0.45, 'Universalism: concern': 0.68, 'Universalism: nature': 0.76, 'Universalism: objectivity': 0.28, 'Universalism: tolerance': 0.28, 'avg-f1-score': 0.39}	0.390000

```
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-500
Configuration saved in model/prajjwal1/bert-small\checkpoint-500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-500\special_tokens_map.json
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-1000
Configuration saved in model/prajjwal1/bert-small\checkpoint-1000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-1000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-1000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-1000\special_tokens_map.json
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-1500
Configuration saved in model/prajjwal1/bert-small\checkpoint-1500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-1500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-1500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-1500\special_tokens_map.json
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-2000
Configuration saved in model/prajjwal1/bert-small\checkpoint-2000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-2000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-2000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-2000\special_tokens_map.json
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-2500
Configuration saved in model/prajjwal1/bert-small\checkpoint-2500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-2500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-2500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-2500\special_tokens_map.json
***** Running Evaluation *****
  Num examples = 1044
  Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-3000
Configuration saved in model/prajjwal1/bert-small\checkpoint-3000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-3000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-3000\tokenizer_config.json
```

```
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-3000\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-3500
Configuration saved in model/prajjwal1/bert-small\checkpoint-3500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-3500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-3500\tokenizer_c
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-3500\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-4000
Configuration saved in model/prajjwal1/bert-small\checkpoint-4000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-4000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-4000\tokenizer_c
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-4000\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-4500
Configuration saved in model/prajjwal1/bert-small\checkpoint-4500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-4500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-4500\tokenizer_c
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-4500\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-5000
Configuration saved in model/prajjwal1/bert-small\checkpoint-5000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-5000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-5000\tokenizer_c
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-5000\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-5500
Configuration saved in model/prajjwal1/bert-small\checkpoint-5500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-5500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-5500\tokenizer_c
onfig.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-5500\special_token
s_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-6000
```

```
Configuration saved in model/prajjwal1/bert-small\checkpoint-6000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-6000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-6000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-6000\special_tokens_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-6500
Configuration saved in model/prajjwal1/bert-small\checkpoint-6500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-6500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-6500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-6500\special_tokens_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-7000
Configuration saved in model/prajjwal1/bert-small\checkpoint-7000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-7000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-7000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-7000\special_tokens_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-7500
Configuration saved in model/prajjwal1/bert-small\checkpoint-7500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-7500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-7500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-7500\special_tokens_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-8000
Configuration saved in model/prajjwal1/bert-small\checkpoint-8000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-8000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-8000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-8000\special_tokens_map.json
***** Running Evaluation *****
    Num examples = 1044
    Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-8500
Configuration saved in model/prajjwal1/bert-small\checkpoint-8500\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-8500\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-8500\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-8500\special_tokens_map.json
***** Running Evaluation *****
```

```
Num examples = 1044
Batch size = 8
Saving model checkpoint to model/prajjwal1/bert-small\checkpoint-9000
Configuration saved in model/prajjwal1/bert-small\checkpoint-9000\config.json
Model weights saved in model/prajjwal1/bert-small\checkpoint-9000\pytorch_model.bin
tokenizer config file saved in model/prajjwal1/bert-small\checkpoint-9000\tokenizer_config.json
Special tokens file saved in model/prajjwal1/bert-small\checkpoint-9000\special_tokens_map.json
```

```
In [15]: print(bert_model_evaluation)
```

```
{'eval_loss': 0.3503268361091614, 'eval_accuracy_thresh': 0.861398458480835, 'eval_f1-score': {'Achievement': 0.61, 'Benevolence: caring': 0.45, 'Benevolence: dependability': 0.07, 'Conformity: interpersonal': 0.08, 'Conformity: rules': 0.44, 'Face': 0.08, 'Hedonism': 0.22, 'Humility': 0.04, 'Power: dominance': 0.13, 'Power: resources': 0.46, 'Security: personal': 0.67, 'Security: societal': 0.66, 'Self-direction: action': 0.55, 'Self-direction: thought': 0.6, 'Stimulation': 0.22, 'Tradition': 0.45, 'Universalism: concern': 0.68, 'Universalism: nature': 0.76, 'Universalism: objectivity': 0.28, 'Universalism: tolerance': 0.28, 'avg-f1-score': 0.39}, 'eval_marco-avg-f1score': 0.39, 'eval_runtime': 18.0899, 'eval_samples_per_second': 57.712, 'eval_steps_per_second': 7.242, 'epoch': 20.0}
```