
RSIM PROJECT

Jonas Klotz

Jan 23, 2023

CONTENTS

1	Setup	3
1.1	Data Set (Deep Globe Patches)	3
1.2	Word Embeddings	3
1.3	Run Example	3
2	Technical Documentation	5
2.1	Main	5
2.2	LAMP	5
2.3	Evaluation	5

This is the technical documentation of our project with [RSIM of TU BERLIN](#).

This is the technical documentation of our project with [RSIM of TU BERLIN](#).

First install the conda environment at `cluster_instructions/conda_env.yml` and activate it.

1.1 Data Set (Deep Globe Patches)

Then download the Deepglobe dataset from [KAGGLE](#). Place it in the project with the path `data/deepglobe`. Then run the patch-sampling pipeline: `data_pipeline/deepglobe/patch_sampling.py`. This creates LMDB files for the train, test and valid set at `data/deepglobe_patches/[train/test/valid]/`.

1.2 Word Embeddings

The folder `data/glove` contains the embeddings for the deepglobe-labels for embeddings spaces of 50 and 300. If other embeddings are necessary, or a different dataset is used. These can be created by modifying the `src/wordembedding/glove.py` and running it. To use this download the glove txt files from [Stanford](#). Currently `d = [50,100,200,300]` are available there. For different embedding size, retrain the glove model.

1.3 Run Example

The most important parameters are contained in this exemplary run. For all parameters check out `src/config_args.py`, for our parameterized runs check out `cluster_instructions/<model>_<loss>.sh`

```
python main.py -model CbMLC -loss weighted_bce -optim sgd -d_model 50 -lr 0.0001
-add_noise 0.1 -sub_noise 0.1
```


TECHNICAL DOCUMENTATION

For the documentation of specific methods consider the docstrings and inline comments of these methods.

2.1 Main

The main method is structured as follows:

1. Argument Parsing
2. Data Loading
3. Model Preparation and loading
4. Optimizer and Loss Setup
5. CUDA Setup
6. Training or Predict

2.2 LAMP

The Model can be found in the `src/lamp/Models.py`

Note: In the LAMP directory we only contributed the RESNETs in the `Decoders.py` and `Models.py`

2.3 Evaluation

To reconstruct our evaluation process, use the `plots/plot_training.ipynb` file. The averaged results are attached in the results directory as CSV files.