Replication

Note, due to the randomness of a CNN, replicating the exact results is unlikely but you will get similar results, you can look through the individual results from the plots, or results array to see the highs and lows.

Preprocessing:

Text Preprocessing (stemming, stopword removal, etc)

Control variables:

```
LR = 1e-3
BATCH_SIZE = 50
DROPOUT = 0.5
MAX_DOC_LEN = 1000
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [128, 128]
NUM_EPOCHS = 20
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = None
```

Project = ‘tensorflow’
Num_iters = 10
Binary average
Then for each different test, enabling and disabling the features:

- `Pp.clean_str`
- `Pp.remove_stopwords` and the line above
- `preprocess="stem"` or `lemmatize`

Max Token Length:

Control:

```
LR = 1e-3
BATCH_SIZE = 50
DROPOUT = 0.5
MAX_DOC_LEN = ?
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [128, 128]
NUM_EPOCHS = 20
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = None
```

Project = ‘tensorflow’
Num_iters = 10
only clean string enabled
Binary average

Then change the MAX_DOC_LEN to 500, 200, 100, 75, 50, 25

Hyperparameter tuning:

Control:
Num_iters = 10
With clean string enabled
Binary average

Default parameters

```
LR = 1e-3
BATCH_SIZE = 50
DROPOUT = 0.5
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
```

```
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [128, 128]
NUM_EPOCHS = 20
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = None
```

Project = ‘pytorch’

Post Hyperparameter tuned:

```
LR = 1.5e-3
BATCH_SIZE = 50
DROPOUT = 0.34
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [512, 256]
NUM_EPOCHS = 25
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = 0.10568
```

Project = ‘pytorch’

Experiments

Positive Classification Performance:

Same across projects:
Binary average - change on baseline too
Num_iters = 10
Clean string enabled

Tensorflow

```
LR = 9.4e-4
BATCH_SIZE = 50
DROPOUT = 0.49
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
N_FILTERS = [458, 425]
FILTER_SIZES = [2, 3]
NUM_EPOCHS = 29
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = None
```

Project = ‘tensorflow’

Pytorch

```
LR = 1.5e-3
BATCH_SIZE = 50
DROPOUT = 0.34
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
```

```
FILTER_SIZES = [3, 4]
N_FILTERS = [512, 256]
NUM_EPOCHS = 25
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = 0.10568
```

Project = ‘pytorch’

Keras

```
LR = 1.3e-3
BATCH_SIZE = 50
DROPOUT = 0.071
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [249, 127]
NUM_EPOCHS = 21
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = 0.2
```

Project = ‘keras’

MXNet

```
LR = 1e-3
BATCH_SIZE = 50
DROPOUT = 0.2
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
```

```
FILTER_SIZES = [2, 4]
N_FILTERS = [102, 466]
NUM_EPOCHS = 14
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = 0.2
```

Project = ‘incubator-mxnet’

Macro Average

Control:
Binary average - change on baseline too
Project = ‘keras’
Num_iters = 10
Clean string enabled

```
LR = 1.3e-3
BATCH_SIZE = 50
DROPOUT = 0.071
MAX_DOC_LEN = 73
TEST_SIZE = 0.2
MAX_VOCAB = 10000
HIDDEN_SIZE = []
POOL_SIZE = 2
FILTER_SIZES = [3, 4]
N_FILTERS = [249, 127]
NUM_EPOCHS = 21
EMBEDDING_TYPE = 1
FREEZE_EMBEDDINGS = False
THRESHOLD = 0.2
```