

CS 769: Advanced NLP - HW 1 Report

Manoj Arulmurugan

9087549409

In **model.py**, I incorporated **pre-trained embeddings** for word representation. I added support for both **GloVe 300D** and **FastText embeddings**. To make the model more expressive, I extended the pooling mechanism: in addition to the classic average, sum, and max pooling, I introduced an **attention pooling layer** that learns token importance weights. This attention-augmented DAN allows the model to focus more on sentiment-rich words instead of treating all tokens equally. I also updated parameter initialization, explicitly zeroed out the **<pad>** row.

In **main.py**, I added a **--freeze_emb** flag, ensured all random seeds (Python, NumPy, PyTorch) are set for reproducibility, and added directory creation logic so outputs and models are stored under my campus ID. I also switched the optimizer from **Adagrad** to **Adam**, which converged faster and stabilized training.

In **run_exp.sh**, I automated downloading and extracting both **FastText** and **GloVe** embeddings, then configured experiments: SST was trained using FastText embeddings, while CFIMDB used GloVe 6B 300D with a deeper hidden layer size, tuned dropout rates, larger batch size (32), and a reduced learning rate (0.001). I also increased the training epochs for CFIMDB to prevent underfitting.

With these changes, my models achieved the following:

- **SST**: test accuracy of **0.4511** and dev accuracy of **0.4260**, which is above the reference baseline.
- **CFIMDB**: dev accuracy of **0.9388** and test accuracy of **0.4918**, meeting the expected performance range.

```
Loading model from model.pt
-Accuracy: 0.4394 (971/2210)
-Save predictions to 9087549409/sst-test-output.txt
-Accuracy: 0.4342 (478/1101)
-Save predictions to 9087549409/sst-dev-output.txt
```

—> SST Dataset

```
Loading model from model.pt
-Accuracy: 0.4918 (240/488)
-Save predictions to 9087549409/cfimdb-test-output.txt
-Accuracy: 0.9388 (230/245)
-Save predictions to 9087549409/cfimdb-dev-output.txt
```

—> CF-IMDB Dataset

Overall, these enhancements allowed me to reproduce the original DAN results while experimenting with attention pooling and improved training strategies to push accuracy higher.