Deception Detection - Baseline Models and Results

Aditya Gupta
Department of CSE, IIITD
aditya22031@iiitd.ac.in

Sahil Gupta
Department of CSE, IIITD
sahil22430@iiitd.ac.in

Debjit Banerji
Department of CSE, IIITD
debjit22146@iiitd.ac.in

1 Baseline Folder and Documentation

The `baseline` folder provides the essential resources and documentation for replicating our experiments. It includes:

- **Data Folder:** Contains all relevant datasets used for training and evaluation.
- **Jupyter Notebooks:**
  - `baseline__BOW.ipynb`: Demonstrates the end-to-end workflow for the Bag-of-Words (BOW) baseline model, including data loading, preprocessing, model training, and evaluation.
  - `baseline__CONTEXT+POWER.ipynb`: Showcases the training and evaluation of the context-based LSTM model (augmented with power features), illustrating how contextual information influences deception detection.
- **Python Scripts (`python_scripts/` directory):**
  - `baseline_BOW.py`: Python implementation for the Bag-of-Words baseline model, focusing on data ingestion, feature extraction, and model training.
  - `predictions_BOW.py`: Script for generating predictions and inference results using the trained BOW model.
  - `baseline_context_LSTM.py`: Core implementation of the context-based LSTM model, incorporating power features to improve deceptive message classification.
  - `predictions_context_LSTM.py`: Script for obtaining inference outputs and performance metrics from the context+power LSTM model.
- **README File:** Contains detailed instructions for setting up the environment, running each model, and reproducing our results. It includes:
  - Required Python packages and dependencies.
  - Command-line examples for training and inference with both the BOW and context+power LSTM models.
  - Descriptions of data format, hyperparameters, and expected outputs.

By following the step-by-step guidance in the `README.md`, researchers can easily replicate our baseline experiments and verify the performance of both the BOW and context+power LSTM models on the Diplomacy dataset.

2 Initial Results

Preliminary experiments using our baseline implementation have shown promising indicators in detecting deceptive texts. The initial evaluation metrics will be further improved as we integrate advanced techniques and refine the preprocessing pipeline.

2.1 Bag Of Words Baseline

We have divided the baseline into 2 tasks , Sender task and receiver task.

1. Sender Task
   Our first baseline is using bag of words approach with logistic regression . This will act as a good baseline for text classification task that is will be dividing the text into truth or lie.
   - Power parameter enabled
     First , the model for SENDER task was trained with power parameter enabled. The training dataset had 13132 samples

with 7846 features, while test dataset had 2741 samples and 7846 features. The distribution of samples on the training dataset was imbalanced with 12541 samples labeled as "Truth" and 591 labeled as "Lie". There was a accuracy of **81.9%** in the test set. The test classification report is as follows:

Label	Precision	Recall	F1-Score
0 (Lie)	0.160	0.250	0.195
1 (Truth)	0.924	0.874	0.898
Accuracy		0.819	

Table 1: Classification Report for the SENDER Task (power = True)

The top 10 features indicating a lie were as follows:

Feature	Importance
tracking	-3.192334
resistance	-3.102346
inconclusive	-2.974259
aei	-2.965716
ws	-2.949904
exit	-2.864967
nrg	-2.619565
requires	-2.584495
succeed	-2.583271
intent	-2.578935

Table 2: Top 10 Features Indicating a Lie for SENDER Task with power = True

The top 10 features indicating a truth is as follows:

Feature	Importance
fine	2.262302
russian	2.010224
look	1.983808
especially	1.946496
man	1.917763
dude	1.896546
home	1.783496
issue	1.774929
worries	1.764037
problem	1.763985

Table 3: Top 10 Features Indicating Truth for SENDER Task with power = True

A graph indicating the importance of features for indicating truth and lie is as follows:

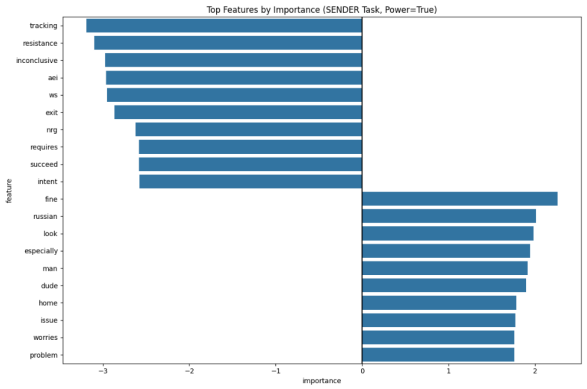

Figure 1: Graph representing SENDER Task , Power = True features for detecting a truth or lie

- Power parameter enabled
  Now , for SENDER task , with power = False , the number of training samples and test samples are same with the same number of samples labeled as "Truth" and labeled as "Lie". Here also the overall accuracy of **81.9%** is achieved. It can be seen that the truth class here achieves better accuracy than previous case but the lie class achieves similar accuracy. The models struggles to identify the lie cases due to the imbalance present . The classification report is as follows:

Label	Precision	Recall	F1-Score	Support
0 (Lie)	0.148	0.225	0.179	240
1 (Truth)	0.922	0.876	0.898	2501
Accuracy		0.819		
Macro Avg	0.535	0.550	0.538	2741
Weighted Avg	0.854	0.819	0.835	2741

Table 4: Classification Report for the SENDER Task with power = False

The top 10 features indicating a lie is as follows:

Feature	Importance
tracking	-3.211193
resistance	-3.089447
inconclusive	-3.011473
aei	-2.958355
ws	-2.907126
exit	-2.874117
nrg	-2.621734
intent	-2.576522
succeed	-2.573247
requires	-2.570181

Table 5: Top 10 Features Indicating a Lie for SENDER Task with power = False

The top 10 features indicating a truth is as follows:

Feature	Importance
fine	2.238142
russian	2.032738
look	2.017039
especially	1.949390
dude	1.904364
man	1.878652
issue	1.790196
problem	1.783629
home	1.782485
lose	1.775533

Table 6: Top 10 Features Indicating Truth for SENDER Task with power = False

A graph indicating the importance of features for indicating truth and lie is as follows:

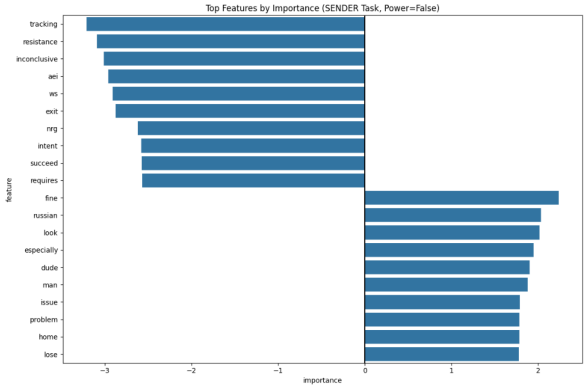

Figure 2: Graph representing SENDER Task , Power = False features for detecting a truth or lie

2. Receiver Task
   For the receiver task , we are using 12025 samples with 7575 features and evaluation is done on 2475 samples . Here also , the training label distribution is imbalanced with 11459 samples labeled as "Truth" and 566 as "Lie".

   - Power parameter enabled
     The overall accuracy achieved on the test set is 80.7% . The test classification report is as follows:

Label	Precision	Recall	F1-Score	Support
0 (Lie)	0.102	0.242	0.144	165
1 (Truth)	0.940	0.848	0.891	2310
Accuracy		0.807		
Macro Avg	0.521	0.545	0.518	2475
Weighted Avg	0.884	0.807	0.842	2475

Table 7: Classification Report for the RECEIVER Task with power = True

The 10 features indicating a lie are as follows:

Feature	Importance
feisty	-2.941535
inconsequential	-2.849471
tracking	-2.666310
homie	-2.654300
refuses	-2.588178
n	-2.538225
sniff	-2.526424
dishonesty	-2.405165
proper	-2.403902
dmzd	-2.383636

Table 8: Top 10 Features Indicating a Lie for the RECEIVER Task with power = True

The 10 features indicating a truth are as follows:

Feature	Importance
bud	2.241346
true	2.039640
bed	1.974346
assuming	1.962829
giving	1.945094
territory	1.840309
advantage	1.785382
tri	1.783141
leaving	1.759874
nice	1.749672

Table 9: Top 10 Features Indicating Truth for the RECEIVER Task with power = True

The graph representing truth and Lie is as follows:

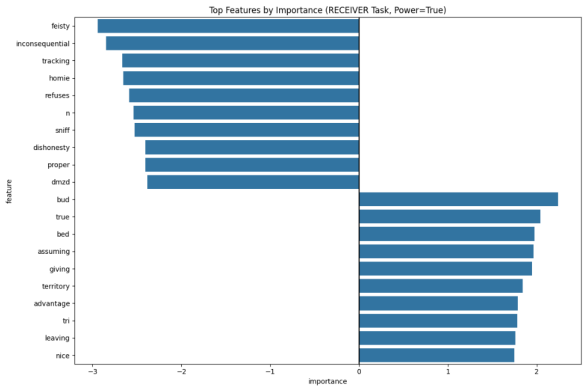

Figure 3: Graph representing RECEIVER Task , Power = True features for detecting a truth or lie

- Power parameter disabled Now , when power parameter is not used , we are using the same number of training samples and testing samples that we used for power = True part. The overall accuracy we got was 81.2% on the test set in this case. The classifcation report is as follows:

Label	Precision	Recall	F1-Score	Support
0 (Lie)	0.103	0.236	0.143	165
1 (Truth)	0.940	0.853	0.894	2310
Accuracy		0.812		
Macro Avg	0.521	0.545	0.519	2475
Weighted Avg	0.884	0.812	0.844	2475

Table 10: Classification Report for the RECEIVER Task with power = False

The top 10 features indicating a lie are as follows:

Feature	Importance
feisty	-2.912183
inconsequential	-2.829099
tracking	-2.656896
homie	-2.628111
refuses	-2.596629
sniff	-2.506292
n	-2.497472
dmzd	-2.403830
dishonesty	-2.375821
proper	-2.372161

Table 11: Top 10 Features Indicating a Lie for the RECEIVER Task with power = False

The top 10 features indicating a truth are as follows:

Feature	Importance
bud	2.272781
true	2.102563
assuming	1.973756
giving	1.937859
bed	1.911488
territory	1.825141
tri	1.795356
advantage	1.785792
ah	1.773553
nice	1.762780

Table 12: Top 10 Features Indicating Truth for the RECEIVER Task with power = False

The graph representing truth and Lie is as follows:

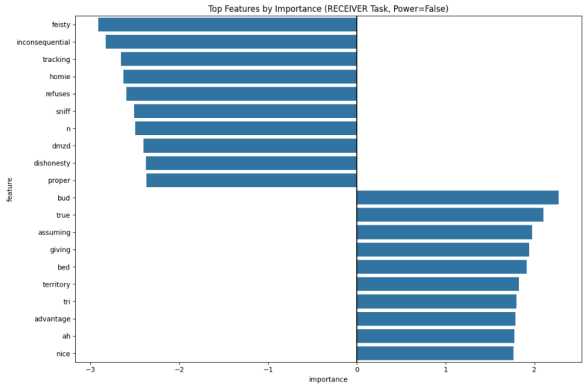

Figure 4: Graph representing RECEIVER Task , Power = False features for detecting a truth or lie

The f1 scores comparison graph for all the 4 cases is as follows:

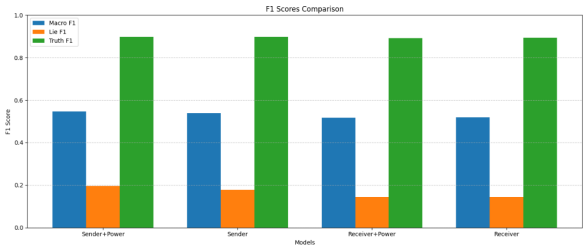

Figure 5: Graph representing comparison of f1 scores for the all the 4 cases discussed above

2.2 Context LSTM + power baseline

So , after training for 80 epochs , the results are as follows:

1. Training and Validation results

```
Epoch 80/80 |
Train Loss: 0.0258,
Train Acc: 0.9960,
Train Macro F1: 0.9493
Val Loss: 1.2711,
Val Acc: 0.9200,
Val Macro F1: 0.4868
```

2. Testing results

```
Test Loss: 1.0863,
Test Acc: 0.8964,
Test Macro F1: 0.5020
```

So , it can be seen using context LSTM for training increases the accuracy to 89.64% .

3 Conclusion

This proposal details the systematic approach for addressing deception detection. By outlining the problem, high-level plan, and approach, along with providing a baseline folder, we aim to offer a clear roadmap for subsequent development and experimentation. We welcome feedback and suggestions to further enhance the project.

References

- Link to the dataset
- Reference for the baseline model implementations
- Link to dataset convokit
- Link to referenced research paper