NLP - Deception Detection [End Sem Presentation]

NLP_grp43

Harsh Rajput - 2022201

Krishna Shukla - 2022254

Varun Kumar - 2022563

INDRAPRASTHA INSTITUTE *of* INFORMATION TECHNOLOGY **DELHI**

Introduction

- Strategy game *Diplomacy* involves negotiation, alliances, and betrayal.
- Goal: Classify in-game messages as **truthful** or **deceptive**.
- Importance: Helps analyze player behavior and improve AI strategies.
- Approach: Build an NLP model with enhanced **feature engineering** and **modeling techniques** for better accuracy.

Literature Review

- **Deception in Diplomacy** *(Peskov et al.)*
  - Dataset: 17,289 annotated messages from 12 games.
  - Dual-annotation: Sender's intent vs. receiver's perception.
  - Key Insight: Deception blends truth and lies to build trust.
  - ML models using **linguistic cues**, **context**, and **power dynamics** matched human-level performance.
- **Political Agreement & Disagreement** *(Davoodi et al.)*
  - Dataset: State bills + legislature, district & donor data.
  - Task: Predict voting outcomes using bill text & context.
  - Model: Shared relational embeddings.
  - Highlight: **Contextual info** boosts accuracy in political alignment prediction.

Dataset

- **Source**: 17,289 annotated messages from 12 online *Diplomacy* games (Peskov et al.)
- **Game Context**: WWI-era strategy game where players form/break alliances; success depends on negotiation & deception (no randomness).
- **Annotations**:
  - **Sender Label**: Actual intent (truthful/deceptive)
  - **Receiver Label**: Perceived truthfulness (true/false/NA)
- **Data Split**: 9 games (train), 1 (validation), 2 (test)
- **Message Fields**:
  - Sender, Receiver, Message Text
  - Labels: `sender_labels`, `receiver_labels`
  - Metadata: Game ID, Year, Season, Message Index, Game Score, Score Delta

Methodology

- **Approach**: Supervised learning to classify messages as **truthful** or **deceptive**
- **Pipeline**: Language Representation → Classification Model

1. Feature Representations (Encoders):

- **RoBERTa** – Robust transformer, strong NLP performance
- **BERT** – Contextualized embeddings from bidirectional transformers
- **MiniLM** – Lightweight transformer, fast & efficient
- **GloVe** – Static word embeddings (mean of token vectors)

For transformers: used **[CLS]** token embedding

2. Classification Models:

- **Logistic Regression** – Simple, interpretable baseline
- **SVM** – Effective in high-dimensional settings
- **Random Forest** – Captures nonlinear patterns via ensemble trees
- **MLP** – Neural network for complex relationships

Evaluation Metric

The primary evaluation metric for this task is accuracy, which measures the proportion of correctly classified messages.

$$\text{Accuracy} = \frac{\text{Correctly classified messages}}{\text{Total messages}}$$

To address the issue of class imbalance, we use the macro- averaged F1 score as a more reliable metric, particularly for evaluating the detection of ACTUAL_LIE.

$$\text{F1}_{\text{macro}} = \frac{1}{2}(\text{F1}_{\text{truth}} + \text{F1}_{\text{lie}})$$

Results

Model	Log. Reg.	RF	SVM	MLP
RoBERTa	48.5	52.5	48.5	54.4
BERT	55.6	52.0	51.0	51.0
MiniLM	48.5	52.5	48.5	48.5
GloVe	48.5	48.5	48.5	48.5

Learning & Conclusion

- **Real-World NLP Exposure**: Tackled strategic deception in human communication using complex, annotated game dialog.
- **Handling Dialog Data**: Learned to preprocess and structure annotated datasets capturing both sender and receiver perspectives.
- **Class Imbalance**: Addressed skewed data distribution; used macro F1 and adjusted training strategies for fair evaluation.
- **Model Experimentation**: Explored traditional (GloVe) and transformer-based (BERT, RoBERTa, MiniLM) embeddings.
- **Classifier Insights**: Compared Logistic Regression, SVMs, Random Forests, and MLPs across different embeddings.
- **Evaluation & Fairness**: Gained experience in designing robust experiments and interpreting model performance in sensitive NLP tasks.

Thank You