ETHOS

AN ARGUMENT STRENGTH ANALYZER

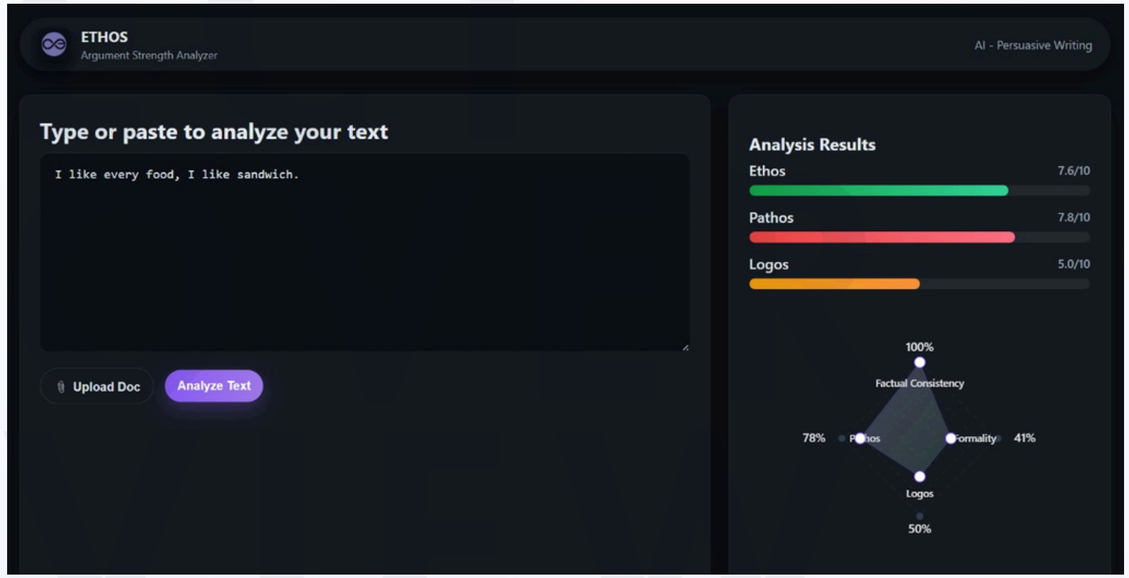
DES 646 PROJECT 2025/26-I

AUTOMATION AFICIANADOS

INTRODUCTION

To empower users, from students to professionals, to master the art of persuasion in their written arguments. We do this by providing actionable, AI-driven feedback that meticulously evaluates credibility (ethos), emotional impact (pathos), and logical coherence (logos). Our goal is to help writers refine their message, connect more deeply with their audience, and communicate with greater clarity and impact, ultimately enabling them to achieve their goals more effectively.

PROJECT OVERVIEW



Aim:

Help users improve their persuasive writing through actionable AI-driven feedback.

Core Objectives:

- Analyze credibility, emotion, and logic in text.
- Provide visual breakdowns (bars & graphs).
- Encourage iterative improvement.

METHODOLOGY

Model Selection

- Ethos: Formality detection + factual consistency (RoBERTa & Princeton NLP models)
- Pathos: Emotion classification (GoEmotions)
- Logos: Logical coherence (RoBERTa-MNLI)

Backend Pipeline

- Python + Django wraps inference logic
- Output as structured JSON → served to frontend

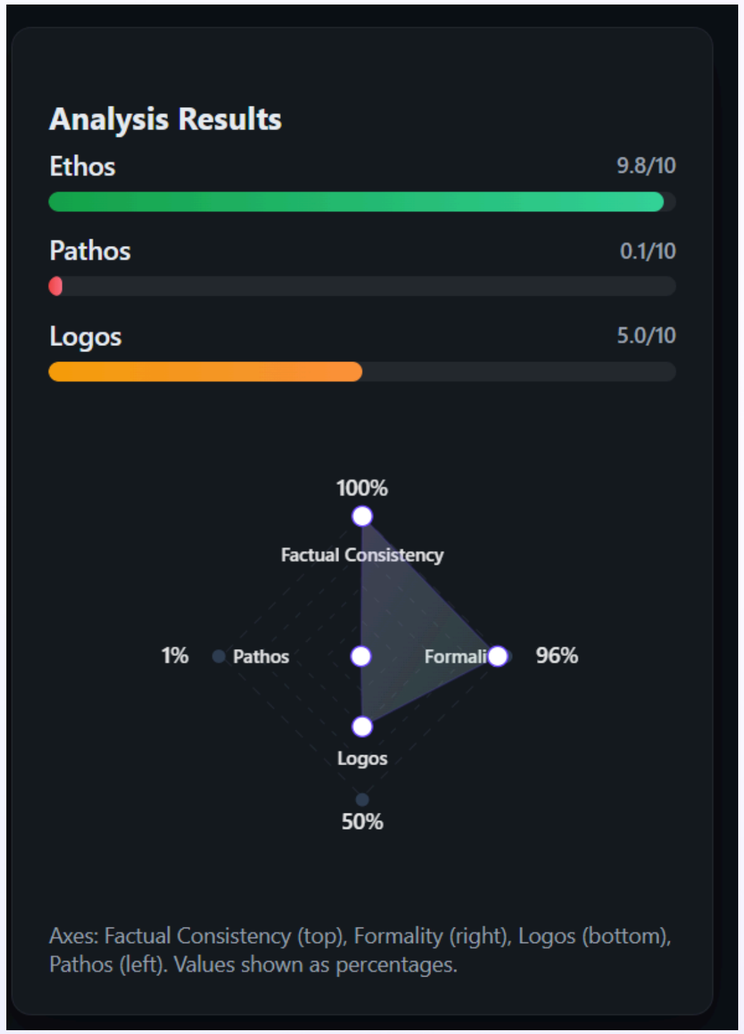
```python
# ----------------------------
def compute_emotion_scores(text: str, emotion_model=None, verbose: bool = False) -> Dict[str, float]:
    """
    Compute emotion distribution for input text.
    Returns dict: emotion label → average score across sentences.
    """
    from collections import defaultdict
    logging.set_verbosity_error()
    if emotion_model is None:
        emotion_model = pipeline(
            "text-classification",
            model="bhadresh-savani/distilbert-base-uncased-emotion",
            top_k=None
        )

    sentences = core.split_sentences(text)
    if not sentences:
        return {}

    # Use defaultdict to handle unexpected labels dynamically
    all_scores = defaultdict(float)

    for s in sentences:
        results = emotion_model(s)[0]  # list of dicts
        for r in results:
            label = r["label"].lower()
            score = float(r["score"])
            all_scores[label] += score
        if verbose:
            print(f"[DEBUG] Sentence: {s}")
            print({r["label"]: round(r["score"],3) for r in results})
```

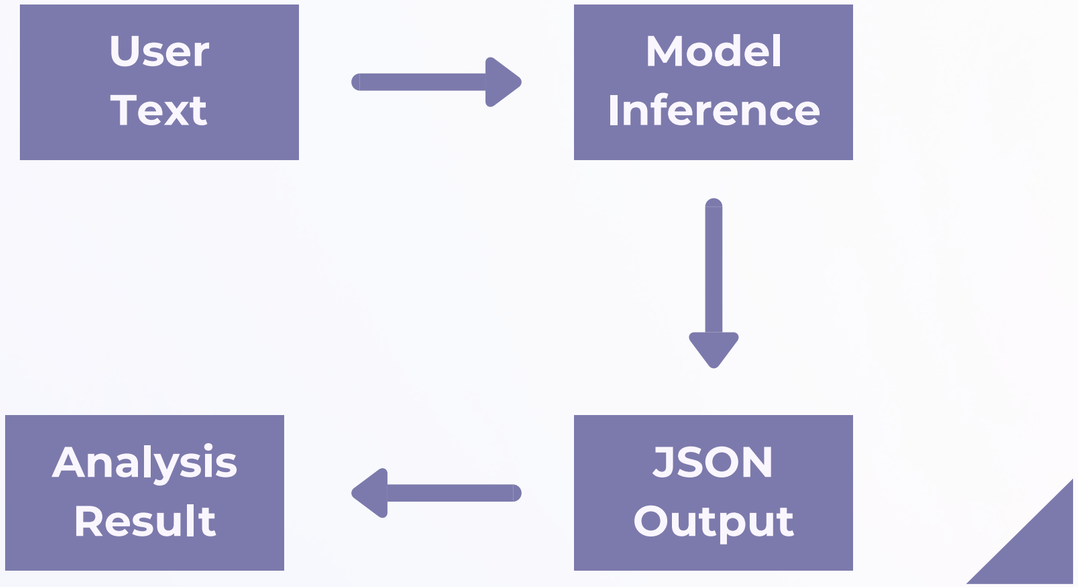
SCORING & VISUALIZATION



- Each score normalized to 1–10 scale
- Overall metrics:
  - **Ethos:** credibility, factual consistency, formality
  - **Pathos:** emotional engagement
  - **Logos:** logical coherence

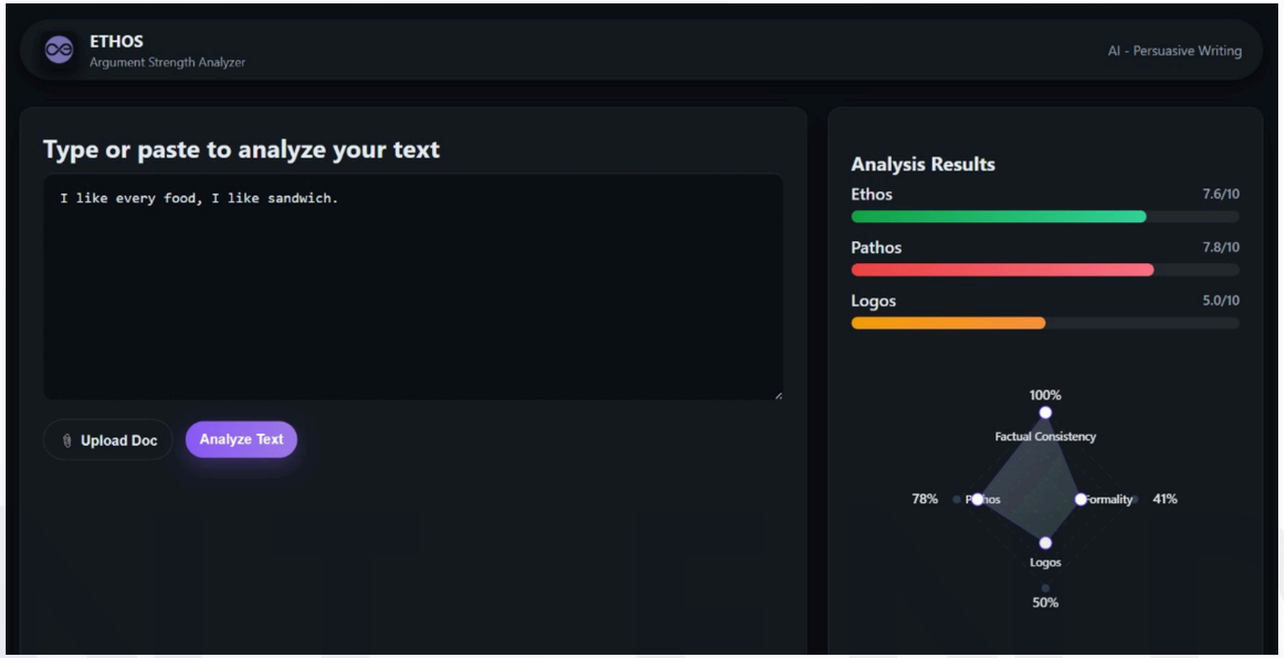
SCORING

SYSTEM ARCHITECTURE

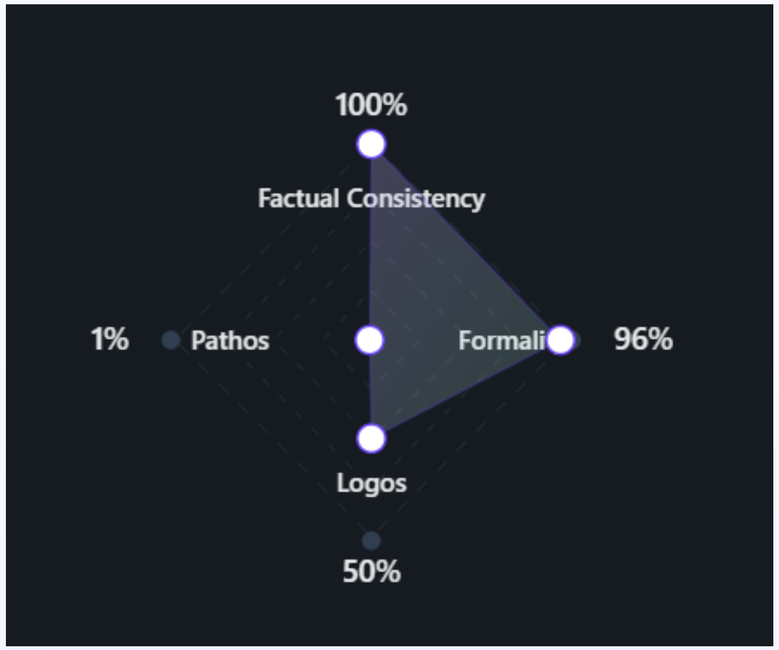


FRONT END DESIGN

- HTML, CSS, JS frontend (fully responsive)
- Real-time updates from backend JSON
- Interactive visual elements:
  - Dynamic score bars
  - Quadrilateral “Personality Map”
  - Sentence-level breakdown



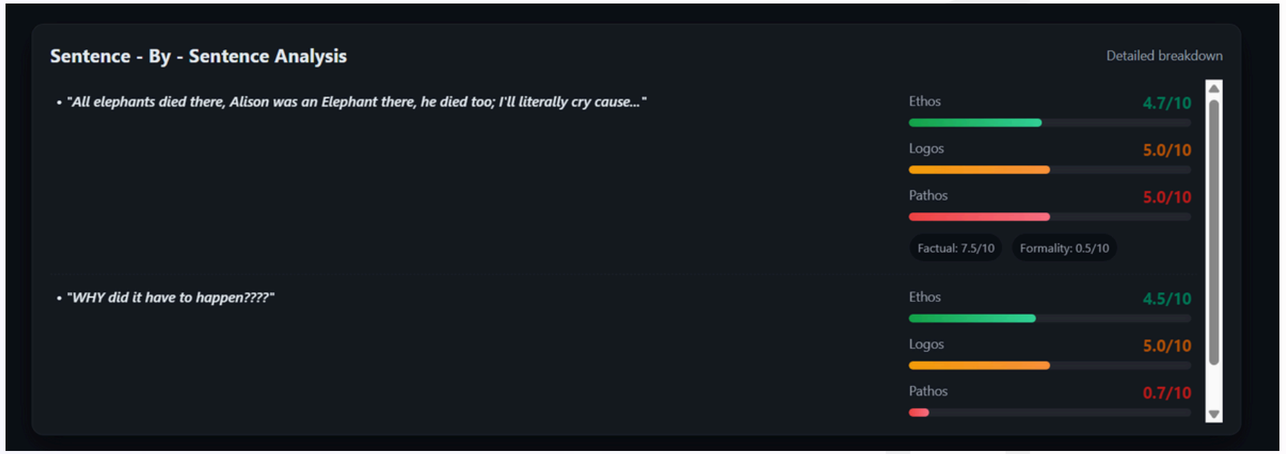
ARGUMENT TRAIT QUADRILATERAL



- Represents 4 traits:
  - Factual Consistency
  - Formality
  - Logos
  - Pathos
- Polygon visualization shows relative strengths.
- Designed using SVG dynamically updated with user data.

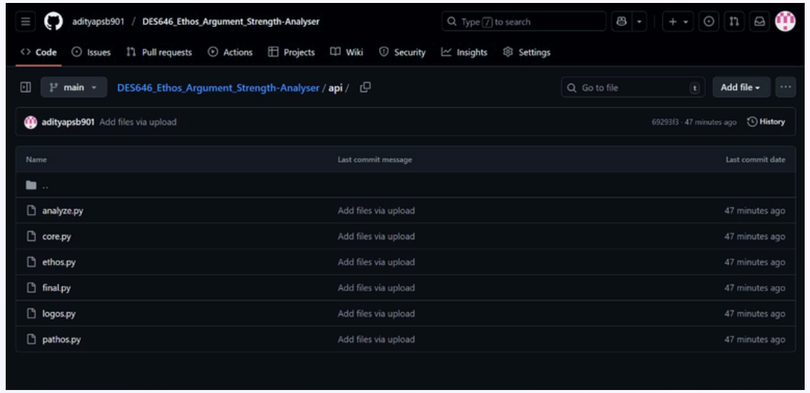
DETAILED BREAKDOWN

- Sentence-level Ethos, Pathos, Logos visualization.
- Helps pinpoint weak areas in text.
- Provides local and global interpretability.



DETAILED BREAKDOWN

GITHUB REPOSITORY



- All code and assets are hosted on GitHub for version control and collaboration.
- Repository includes:
  - Backend API (Django + Python)
  - Frontend (HTML, CSS, JS)
  - Model inference scripts (Ethos, Pathos, Logos)
  - Dataset preprocessing and evaluation notebooks
- Ensures transparency, reproducibility, and easy deployment.

CODE IMPLEMENTATION & INTEGRATION

- Backend Highlights:
  - Python-based pipeline for processing text input.
  - JSON response structure for Ethos, Pathos, Logos metrics.
  - spaCy used for linguistic preprocessing.
- Frontend Highlights:
  - Dynamic visualization of scores and quadrilateral map.
  - Responsive layout using HTML, CSS, JS.
  -

```json
{
  "overall": {"ethos": 0.83, "logos": 0.75, "pathos": 0.71},
  "sentencewise": [
    {
      "sentence": "The Eiffel Tower was completed in 1889.",
      "ethos": {"score": 0.85, "factual_consistency": 0.9, "formality": 0.78},
      "logos": 0.74,
      "pathos": 0.64
    },
    {
      "sentence": "It inspires millions of tourists every year.",
      "ethos": {"score": 0.81, "factual_consistency": 0.83, "formality": 0.76},
      "logos": 0.72,
      "pathos": 0.78
    }
  ]
}
```

```python
def compute_formality(text: str, formality_model=None, verbose: bool = False) -> float:
    """
    Compute an average formality score (0 to 1) using the pretrained formality ranker.
    Higher → more formal writing.
    Automatically inverts scores for 'informal' predictions.
    """
    if formality_model is None:
        formality_model = pipeline(
            "text-classification",
            model="s-nlp/roberta-base-formality-ranker",
            top_k=1
        )

    sentences = core.split_sentences(text)
    if not sentences:
        return 0.0

    scores = []
    for s in sentences:
        result = formality_model(s)[0][0]
        label = result["label"].lower()
        score = float(result["score"])

        # Invert if the predicted label is 'informal'
        if label == "informal":
            score = 1 - score

        score = max(0.0, min(1.0, score))
        scores.append(score)

        if verbose:
            print(f"[{label}] {s} → adjusted_score={score:.3f}")
```

TOOLS & FRAMEWORKS

Frontend
HTML, CSS, JavaScript

Backend
Flask, Python

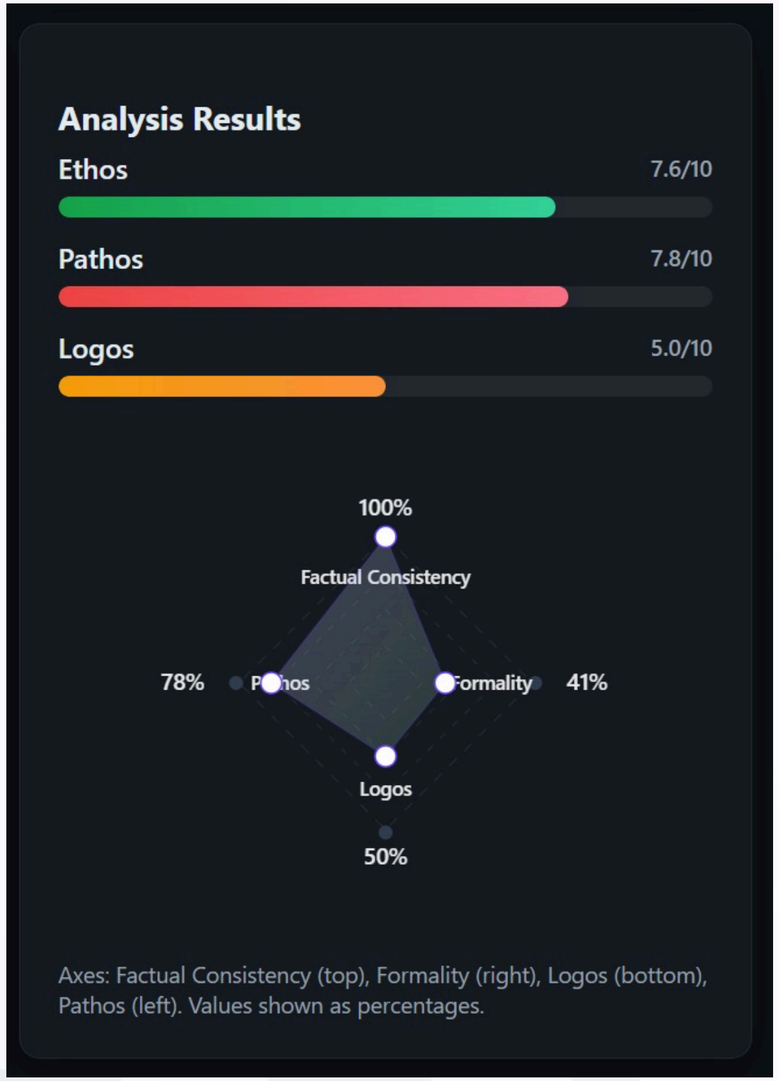
Models
Hugging Face Transformers (RoBERTa, GoEmotions)

Visualization
Custom SVG, responsive layout

Libraries
SpaCy, Torch, NumPy

RESULTS AND EVALUATION

- Scoring tested across multiple sample arguments.
- Consistent logical mapping to human interpretation.
- User testing (planned for 15–20 participants).



FUTURE WORK & CONCLUSION

- Integrate plagiarism & fact-checking APIs.
- Include more nuanced emotional dimension.
- Deploy on web server for public testing.
- Improve dashboard UX based on user feedback.

THANK YOU

Aditya Pratap Singh Baghel
240064

Harshit
240438

Pratham Gupta
240779

Anand Pachpol
240115