PromiseEval - SemEval Task 6

Muhammad Khubaib Mukaddam
School of Science and Engineering
Habib University
Karachi Pakistan
mk07218@st.habib.edu.pk

Ayesha Enayat
School of Science and Engineering
Habib University
Karachi Pakistan
ayesha.enayat@sse.habib.edu.pk

***Abstract*—In this research, we explore the application of Natural Language Processing techniques and Deep Learning models for the task of Promise Verification. This process involves several subtasks, including promise classification, evidence classification, evidence verification, and timeline verification. Initially, BERT was utilized for sequence classification, but improvements were achieved by switching to DeBERTa, which resulted in better performance. Additionally, contrastive learning was incorporated alongside classification loss to enhance the model's ability to differentiate between positive and negative pairs of data. For evidence verification, oversampling techniques were used to address class imbalances in the dataset, particularly for the 'Misleading' class. For the timeline verification subtask, BART was chosen over BERT, achieving superior results in comparison. We detail the methodology, experiments, and results, highlighting the effectiveness of these approaches for tackling the challenges of promise verification in text data.**

I. Introduction

In a world where promises are essential in shaping perceptions and form the basis of decision making, the integrity of commitments made by politicians, corporate leaders, and public figures needs to be scrutinized. These promises can range from environmental sustainability to social responsibility and governance ethics, significantly affecting the general public's trust and welfare. Recognizing this critical role of transparency and accountability, the SemEval's task 6: Promise Eval, aims to accurately assess a company's commitment and adherence to its ESG promises. To this end, a collection of ESG reports has bee done by the organizers of the tasks.

This task could significantly improve the transparency of organizations and public figures, compelling them to adhere more closely to their commitments, which in turn, would foster greater trust and credibility among stakeholders and the general public. Using the results of this task, stakeholders—including consumers, investors, and the general public—could make more informed decisions based on the verifiable actions and commitments of organizations and leaders. Ultimately, this task aims to contribute to a more tangible progress in addressing environmental issues, promoting social justice, and ensuring ethical governance, leading to a more sustainable and equitable world.

II. Problem Definition

The primary objective of this project is **Promise Verification**. Given a report or a part of a report from a company, the goal is to identify and verify promises made within that report. Specifically, I aim to determine whether a statement qualifies as a promise based on three key criteria:

- The statement must be related to Environmental, Social, and Governance (ESG) criteria **(required)**.
- The statement should outline a principle, commitment, or strategy that the company intends to uphold **(required)**.
- The statement should be supported by at least one piece of evidence (optional).

The Promise Verification process follows a pipeline approach, with multiple subtasks:

1) **Promise Classification**: Initially, I classify whether a statement constitutes a promise based on the criteria above.
2) **Evidence Verification**: If a promise is mentioned in the report, I need to evaluate whether it also contains evidence that supports the promise.
3) **Evidence Classification**: If evidence is mentioned for the promise, I evaluate its nature—whether the evidence is misleading, clear, or falls into another category.
4) **Timeline Verification**: If a promise is identified, I verify whether the timeline of the promise has been fulfilled or determine when it is expected to be fulfilled.

This structured approach allows for a comprehensive assessment of promises, ensuring that they are both identifiable and verifiable within the scope of ESG-related commitments.

PromiseEval: Multinational, Multilingual, Multi-Industry Promise Verification
SemEval-2025
Summer 2025 (co-located with a major NLP conference)
ESG

Fig. 1. Promise Eval

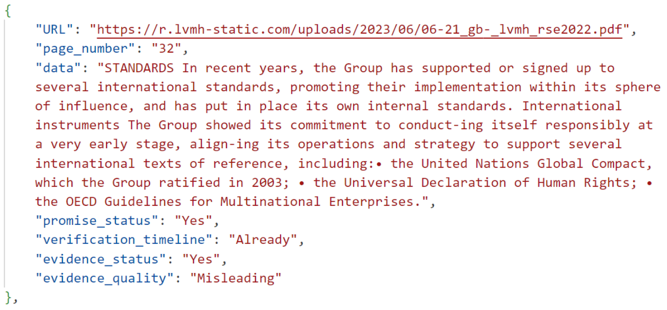
III. Data Description

The dataset used for the Promise Verification task consists of company reports, primarily focusing on Environmental, Social, and Governance (ESG) commitments. Each entry in the dataset provides detailed information regarding specific ESG-related statements. Below is an outline of the dataset structure, including key fields and preprocessing steps undertaken.

A. Dataset Structure

Each record in the dataset includes the following fields:

- **URL**: A link to the source document, providing context and allowing for traceability.
- **page_number**: The page in the document where the statement is located.
- **data**: The textual content of the statement, which may contain potential promises.
- **promise_status**: A binary label indicating whether the statement contains a promise ("Yes") or not ("No").
- **verification_timeline**: Specifies the timeline of the promise's fulfillment, indicating if it has already been fulfilled or is yet to be.
- **evidence_status**: A binary indicator of whether evidence supporting the promise is present ("Yes") or not ("No").
- **evidence_quality**: Assesses the quality of any provided evidence, categorized as "Clear," "Misleading," or "Not Clear."



```
{
    "URL": "https://r.lvmh-static.com/uploads/2023/06/06-21_gb-_lvmh_rse2022.pdf",
    "page_number": "32",
    "data": "STANDARDS In recent years, the Group has supported or signed up to
    several international standards, promoting their implementation within its sphere
    of influence, and has put in place its own internal standards. International
    instruments The Group showed its commitment to conduct-ing itself responsibly at
    a very early stage, align-ing its operations and strategy to support several
    international texts of reference, including:• the United Nations Global Compact,
    which the Group ratified in 2003; • the Universal Declaration of Human Rights; •
    the OECD Guidelines for Multinational Enterprises.",
    "promise_status": "Yes",
    "verification_timeline": "Already",
    "evidence_status": "Yes",
    "evidence_quality": "Misleading"
},
```

Fig. 2. Example of Data

B. Dataset Size and Preprocessing

I combined the two datasets provided to us: "PromiseEval Trainset English" and "PromiseEval Sample Trainset English", which comprises of **600 records** in total. Preprocessing steps included:

- **Data Loading**: I began by loading the dataset from JSON files, sub-sequently converting it to a CSV format for consistency
- **Standardization of Labels**: "Yes" and "No" values were converted to binary format (1 and 0) for consistency in model training.
- **Data Cleaning**: Minor adjustments were made to ensure all text fields were uniformly formatted, including stripping whitespace from textual entries.
- **Verification Timeline:** Already, Less than 2 years, 2 to 5 years, More than 5 years, N/A
- **Evidence Quality:** Clear, Not Clear, Misleading, N/A
- **Label Breakdown:** I used these labels for sub task 3 and 4, and broke down the problem to a multi label classification. This approach was a bit more convenient and easier to follow through.

This dataset provides a structured approach to assess ESG promises by capturing essential attributes related to promises, timelines, and evidence, which are critical for the Promise Verification pipeline.

IV. Related Work

In recent years, several research papers have explored methodologies related to classification and verification tasks in deep learning and natural language processing (NLP). This section highlights the most relevant work in the field and demonstrates the evolution of approaches that inform our own methodology.

- **A Survey of Methods for Addressing Class Imbalance in Deep-Learning Based Natural Language Processing** [1] performs a comprehensive survey that systematically categorizes methods to address class imbalance in NLP. It provides a detailed analysis on various sampling strategies, data augmentation techniques, staged learning, and the different types of weights that can be used to resolve class imbalances. This paper is relevant to our work, particularly in the context of dealing with class imbalance in the task of evidence verification, where oversampling techniques were employed to handle the class imbalance, specifically for the 'Misleading' class.
- **ClaimVer: Explainable Claim-Level Verification and Evidence Attribution of Text Through Knowledge Graphs** [2] focuses on claim-level verification, which contrasts with traditional methods that operate at the sentence or paragraph level. By leveraging knowledge graphs and the attribution score property, the **ClaimVer** framework overcomes the limitations of previous fact-checking systems, which struggled with one-to-one mappings. This paper's approach of verifying individual claims with specific supporting evidence closely aligns with our work on promise verification, where each promise and its associated evidence need to be verified independently for accuracy and granularity.
- **What is the Real Intention behind this Question? Dataset Collection and Intention Classification** [3] introduces the novel concept of classifying **implicit negative intentions** in questions, which has been largely overlooked in existing NLP studies. The paper presents the **Question Intention Dataset**, aimed at detecting both explicit and implicit negative intentions in questions. It utilizes a **TF-IDF-based dictionary** in conjunction with Transformer models like RoBERTa, and highlights the importance of **polarity classification** for simplifying intention detection. This concept of intention classification

and dataset collection informed our understanding of implicit cues in textual evidence, aiding in the classification and verification of evidence in the Promise Verification task.
- **MarsEclipse team at SemEval-2023** [4] achieved significant results in multi-lingual and multi-label framing detection. Their use of contrastive learning in a multi-label text classification task enabled the model to effectively identify positive and negative example pairs, pulling similar frames closer together while pushing different frames apart. This innovative use of contrastive learning in text classification was particularly influential in shaping our approach to evidence verification, where contrastive loss was incorporated alongside classification loss to improve model performance in distinguishing misleading evidence from accurate evidence.

These works collectively contribute to our understanding of how deep learning models can be applied to the verification of textual claims, evidence, and intentions. Our approach builds on these existing methodologies, incorporating techniques such as contrastive learning, and data augmentation to tackle the challenges in Promise Verification and evidence classification.

V. Methodology

This project focuses on the Promise Verification task, divided into four subtasks: Promise Classification, Timeline Verification, Evidence Classification, and Evidence Verification. Each subtask demanded a tailored approach involving state-of-the-art NLP models and specific strategies to address challenges such as data imbalance and threshold optimization. Below is a detailed description of the methodology for each subtask. For the baseline approach, I selected **BERT for sequence classification** as the baseline model as it is able to capture contextual nuances in text, making it highly effective for natural language understanding tasks. BERT's architecture, enables it to understand the context of words within sentences, making it suitable for detecting the language used in ESG promises and statements. The pre-trained BERT model can be fine-tuned on a task-specific dataset, allowing it to learn from our labeled data (e.g., promise vs. non-promise). After further exploration, I tried different models and techniques aiming to improve the results.

A. Subtask 1: Promise Classification

The goal of this subtask was to classify whether a statement constitutes a promise. Initially, a `BERT`-based model was used for sequence classification, but the results were suboptimal. To improve performance, the model was upgraded to `DeBERTa`, a transformer architecture known for its superior contextual understanding and language representation. DeBERTa's robust contextual modeling capabilities outperformed BERT in handling nuanced language, making it an ideal choice for this subtask.
Additionally, I introduced **Contrastive Loss** alongside the classification loss. This technique helped the model better distinguish between similar and dissimilar promises, enhancing overall classification accuracy. Incorporating contrastive loss added an extra layer of learning, allowing the model to learn fine-grained differences in promise statements, and improving classification for borderline cases.

To further fine-tune the model's predictions, I manually adjusted the decision threshold. This step was crucial as the default threshold of 0.5 did not consistently yield the best results. By empirically determining the optimal threshold, I was able to significantly improve performance metrics.

```
def compute_loss(self, model, inputs, return_outputs
    =False, num_items_in_batch=None):

    labels = inputs.get(”labels”)
    outputs = model(**inputs, output_hidden_states=
        True)
    classification_loss = outputs.loss

    embeddings = outputs.hidden_states[-1][:, 0, :]
    positive_pairs, negative_pairs = create_pairs(
        embeddings, labels)

    contrastive_loss = 0

    if positive_pairs:
        pos_emb1 = torch.stack([p[0] for p in
            positive_pairs])
        pos_emb2 = torch.stack([p[1] for p in
            positive_pairs])
        cx =  contrastive_loss_fn(pos_emb1, pos_emb2
            , torch.ones(pos_emb1.size(0)).to(
            pos_emb1.device))
        contrastive_loss += cx

    if negative_pairs:
        neg_emb1 = torch.stack([n[0] for n in
            negative_pairs])
        neg_emb2 = torch.stack([n[1] for n in
            negative_pairs])
        cp = contrastive_loss_fn(neg_emb1, neg_emb2,
            -torch.ones(neg_emb1.size(0)).to(
            neg_emb1.device))
        contrastive_loss += cp

    total_loss = classification_loss + 0.1 *
        contrastive_loss
    return (total_loss, outputs) if return_outputs
        else total_loss
```

Listing 1. Contrastive Loss Method for Classification

B. Subtask 2: Evidence Verification:

For this binary classification task, the goal was to evaluate whether a statement containing a promise also included evidence to support it. Given its similarity to Subtask 1 in terms of task formulation, I utilized the `DeBERTa` model here as well, leveraging its contextual embedding capabilities for binary classification. with the same training pipeline. The combination of classification loss and a well-defined tokenization process allowed the model to achieve high accuracy. Since the dataset for this task was relatively balanced and straightforward, no additional augmentation or threshold adjustment was required.

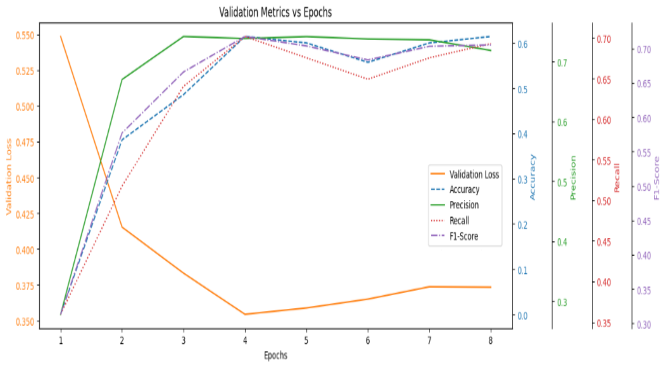


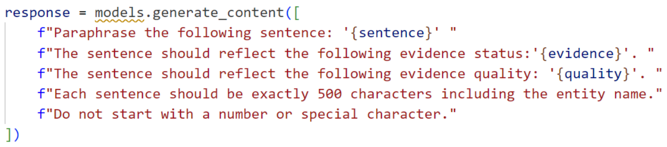
Fig. 3. Training Plot for subtask2

C. Subtask 3: Evidence Classification

The goal of this task was to classify the nature of the evidence (e.g., misleading, clear, or another category). Initially, I used `BERT`, but later transitioned to `DeBERTa`, which provided better results due to its enhanced representation capabilities.

A significant challenge in this task was class imbalance, particularly the underrepresentation of the `Misleading` class. To address this, I utilized the `Gemini API` for data augmentation. This API enabled the generation of synthetic samples for the 'Misleading' class which effectively increased the number of samples in the underrepresented class. This oversampling technique ensured the model had sufficient data to learn from and improved it's capacity to generalize across minority classes.

Standard classification loss was used, along with careful validation to ensure that the augmented data did not introduce noise or compromise the quality of predictions.



```
response = models.generate_content([
    f"Paraphrase the following sentence: '{sentence}' "
    f"The sentence should reflect the following evidence status:'{evidence}'. "
    f"The sentence should reflect the following evidence quality: '{quality}'. "
    f"Each sentence should be exactly 500 characters including the entity name."
    f"Do not start with a number or special character."
])
```

Fig. 4. Prompt for Data augmentation

D. Subtask 4: Timeline Verification

In this subtask, the aim was to verify whether the timeline of a promise was fulfilled or when it was expected to be fulfilled. Initially, I used `BERT`, but transitioned to `BART`, a model better suited for tasks requiring sequence-to-sequence learning. This change yielded improved results, particularly in handling temporal information within text.

The use of `BART` allowed for better processing of sequential dependencies, making it a superior choice for tasks involving timeline verification. The model's ability to generate and evaluate sequences provided a significant advantage in this subtask.

```
from transformers import
    BartForSequenceClassification, AutoTokenizer

checkpoint = "facebook/bart-large"
tokenizer = AutoTokenizer.from_pretrained(checkpoint
    )
model = BartForSequenceClassification.
    from_pretrained(checkpoint, num_labels=6)

model
```

Listing 2. Loading BART Model for Sequence Classification

VI. Evaluation and Results

The performance of both the baseline and final models was evaluated using the **F1 score**, a metric well-suited for classification tasks due to its ability to balance precision and recall. The evaluation encompassed four key subtasks within the **Promise Verification Pipeline**: **Promise Classification**, **Evidence Classification**, **Evidence Quality**, and **Verification Timeline**. The results, as summarized in Table I, provide a comprehensive comparison of the baseline and final models' performance.

TABLE I
Comparison of F1 Scores for Baseline and Final Models Across the Four Subtasks

Subtask	Baseline F1 (%)	Final F1 (%)	Improvement (%)
Subtask 1	76.67	77.50	+0.83
Subtask 2	72.80	80.00	+7.20
Subtask 3	43.81	59.80	+15.99
Subtask 4	25.78	41.20	+15.42

VII. Analysis and Insights

- **Promise Classification:** The final model achieved a slight improvement of +0.83% over the baseline, reaching an F1 score of 77.50%. This result demonstrates the robustness of the initial approach in identifying promises. The small improvement may suggest that the subtask's inherent simplicity or dataset limitations restrict further gains. However, the stable performance highlights the reliability of the classification techniques employed, with potential avenues for improvement including leveraging more diverse datasets or incorporating semantic embeddings.
- **Evidence Classification:** A substantial improvement of +7.20% was observed in the final model, which achieved an F1 score of 80.00%. This progress indicates that the refined training strategies, such as the use of contrastive learning and newer models such as `DeBERTa`, contributed significantly to the model's ability to discern supporting evidence. The results suggest that the model effectively generalizes over nuanced variations in evidence presence.
- **Evidence Quality:** The final model's performance saw a significant increase of +15.99%, with the F1 score

rising from 43.81% to 59.80%. This improvement reflects the model's enhanced ability to evaluate evidence clarity, which was achieved through better representation learning techniques. Despite the improvement, this subtask remains challenging due to the subjective nature of evidence quality assessment and the limited granularity of annotations. Eventhough the data augmentation technique did prove fruitful, there is still room for improvement.

- **Verification Timeline:** The most notable improvement was in this subtask, where the final model achieved an increase of +15.42%, bringing the F1 score to 41.20%. Using BART and technique of few shot learning, the score did improve, however, the overall low score indicates that the model struggles to capture complex temporal dependencies.

VIII. Discussion

- The significant improvements in **Evidence Classification**, **Evidence Quality**, and **Verification Timeline** demonstrate the value of iterative development and the inclusion of sophisticated techniques like contrastive learning, few-shot learning, and augmented datasets.
- The relatively smaller gain in **Promise Classification** may point to task saturation, where the current methods are already close to the upper performance bound for the dataset used. It could also suggest that this subtask is less sensitive to the advanced techniques applied in the final model.
- While the improvements are promising, the relatively low scores for **Evidence Quality** and **Verification Timeline** highlight areas that require further exploration, particularly in terms of data diversity, annotation quality, and advanced modeling techniques.

IX. Conclusion

The evaluation results reflect the effectiveness of the enhancements introduced in the final model. While significant progress was made, especially in the more complex subtasks, there remains ample opportunity for further refinement. Future work will focus on addressing the limitations identified, including dataset size and diversity, and exploring innovative modeling approaches to achieve even greater performance across all subtasks.

X. Future Work

The results obtained in this research highlighted several areas that need further exploration and refinement. One key avenue for future work involves enhancing the dataset quality and diversity. While the current dataset provided a strong foundation, its limited size and potential biases may have constrained the model's ability to generalize across all subtasks. Expanding the dataset to include more diverse and representative samples, along with incorporating finer-grained annotations for subtasks like Evidence Quality, could lead to substantial improvements.

Another area of focus is the integration of advanced modeling techniques to address the challenges observed in subtasks with lower performance, such as Verification Timeline and Evidence Quality. Leveraging models such as sequence-based architectures or graph neural networks, could enhance the model's ability to understand complex relational patterns. Additionally, incorporating pre-training strategies tailored to the subtasks, such as domain-specific language models or multitask learning frameworks, may provide a more robust foundation for downstream tasks.

Overall, future work should focus on addressing the identified limitations, leveraging state-of-the-art advancements in natural language processing, and ensuring the model's scalability and applicability in real-world scenarios.

References

[1] S. Henning, W. Beluch, A. Fraser, and A. Friedrich, “A Survey of Methods for Addressing Class Imbalance in Deep-Learning Based Natural Language Processing,” Jan. 2023, *doi:* https://doi.org/10.18653/v1/2023.eacl-main.38.

[2] P. Prabhu, S. Dammu1, H. Naidu, M. Dewan, Y. Kim, T. Roosta, A. Chadha, C. Shah, “ClaimVer: Explainable Claim-Level Verification and Evidence Attribution of Text Through Knowledge Graphs” Jan. 2023, *doi:* https://doi.org/10.48550/arXiv.2403.09724

[3] Maryam Sadat Mirzaei, Kourosh Meshgi, and Satoshi Sekine, “What is the Real Intention behind this Question? Dataset Collection and Intention Classification,” Jan. 2023, *doi:* https://doi.org/10.18653/v1/2023.acl-long.761.

[4] P. Heinisch, M. Plenz, A. Frank, and P. Cimiano, ”ACCEPT at SemEval-2023 Task 3: An Ensemble-based Approach to Multilingual Framing Detection,” in *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, 2023, pp. 1358–1365. [Online]. https://aclanthology.org/2023.semeval-1.187