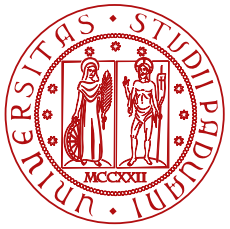

Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Master Thesis in ICT for Interent and Multimedia

Application of Integrated Gradients Explainability to Sociopsychological Semantic Markers

Master Candidate

Ali Aghababaei

Student ID 2071412

Supervisor

Prof. Tomasso Erseghe

University of Padova

Academic Year
2024/2025

To my parents, Azar and Hassan,
and my sister, Afagh,
who have always supported me with unconditional love and belief.

Abstract

Classification of textual data in terms of sentiment or more nuanced sociopsychological markers, such as agency, objectification, or fear, is now a widely adopted practice in natural language processing (NLP). These classifications are often applied at the sentence or document level, offering only a high-level view of the model's interpretation. This thesis aims to bridge that gap by leveraging the Integrated Gradients (IG) method to capture classification output at the *word level*, revealing which specific words contribute to the decision-making process of transformer-based models such as RoBERTa.

We begin by applying IG to a high-data scenario using BERTAgent, a validated agency detection model, to evaluate IG's effectiveness in identifying agentic language. This involves an extensive comparison of attribution techniques, including SHAP (SHapley Additive exPlanations), Sequential Integrated Gradients (SeqIG), and DeepLIFT (Deep Learning Important FeaTures), using explainability metrics such as comprehensiveness, sufficiency, and approximation error. The results guide the optimal configuration of IG parameters and culminate in a visualization system that renders word-level attributions in a form readable and meaningful to social scientists.

In the second part of the work, we turn to small, manually labeled datasets to investigate the role of IG in low-resource settings. Here, we employ an overfitting-based training strategy to amplify word-class associations, enabling the extraction of salient keywords tied to sociopsychological constructs like shame, fear, and objectification. The method provides interpretable insights aligned with expert annotations and offers a potential pathway for dictionary construction and theory-driven text analysis.

Overall, this thesis demonstrates that Integrated Gradients can significantly enhance the interpretability of NLP models by making classification processes transparent at the lexical level. This not only improves model accountability but also deepens our understanding of how language encodes psychological and social meaning.

Sommario

Contents

List of Figures

List of Tables

List of Algorithms

List of Code Snippets

List of Acronyms

IG Integrated Gradients

SeqIG Sequential Integrated Gradients

SHAP SHapley Additive exPlanations

DeepLIFT Deep Learning Important FeaTures

NLP Natural Language Processing

AI Artificial Intelligence

BPE Byte-Pair Encoding

TF-IDF Term Frequency–Inverse Document Frequency

LIME Local Interpretable Model-agnostic Explanations

RNNs Recurrent Neural Networks

CNNs Convolutional Neural Networks

LSTMs Long Short-Term Memory Networks

BERT Bidirectional Encoder Representations from Transformers

RoBERTa A Robustly Optimized BERT Pretraining Approach

PLMs Pretrained Language Models

XAI Explainable Artificial Intelligence

LIWC Linguistic Inquiry and Word

MLM Masked Language Modeling

NSP Next Sentence Prediction

POS Part-of-Speech

NER Named Entity Recognition

AE Approximation Error

TF-IDF Term Frequency–Inverse Document Frequency

1 Introduction

1.1 Background and Motivation

In recent years, the classification of textual data has emerged as a powerful tool for analyzing sentiment and, more importantly, for capturing nuanced sociopsychological constructs such as agency, fear, shame, or objectification. These markers provide valuable insights into how individuals communicate emotions, intentions, and social roles through language. With the advent of deep learning and transformer-based models such as Bidirectional Encoder Representations from Transformers (BERT) [1] and A Robustly Optimized BERT Pretraining Approach (RoBERTa) [2], it has become possible to perform these classifications with impressive accuracy across a wide range of applications, from public health monitoring to political discourse analysis [3].

However, the power of these models comes with a cost: interpretability. While they are highly effective in assigning sentence- or document-level labels, they often fail to reveal which specific words drive these decisions. This lack of transparency presents a significant limitation, especially in domains like social psychology and computational social science, where understanding *why* a model makes a prediction can be as important as the prediction itself [4]. As a result, there has been a growing demand for methods that offer explainability at the word level, enabling researchers to trace back classification outcomes to the lexical cues that caused them.

To address this challenge, this thesis explores the application of the Integrated

Gradients (IG) method [5], an explainability technique that attributes the output of a neural model to its input features via a principled gradient-based approach. Originally developed for image classification tasks, IG has gained traction in the Natural Language Processing (NLP) domain due to its completeness property and computational efficiency [6]. Unlike attention-based explanations, IG provides mathematically grounded attribution scores, making it especially suitable for applications requiring reliable and faithful explanations [7].

The core motivation of this work stems from the need to move beyond surface-level predictions and towards models that can transparently link specific lexical choices to underlying psychological constructs. In particular, we focus on *agency*, the capacity for goal-directed action and intentionality, as a test case. Agency is a cornerstone of human psychology, closely tied to motivation, self-efficacy, and collective behavior [8, 9]. Yet, its expression in language is subtle and context-dependent, making it a challenging target for both classification and interpretation.

Building upon previous work that introduced BERTAgent [10], a RoBERTa-based model trained to detect agency in text, this thesis extends the investigation by applying IG to uncover which words most significantly contribute to the model's predictions. The goal is not only to validate IG's utility but also to build a bridge between advanced machine learning techniques and psychologically interpretable analysis.

1.2 Research Objectives and Questions

The primary goal of this thesis is to improve the interpretability of NLP models applied to the classification of sociopsychological semantic markers. While existing models such as RoBERTa demonstrate high accuracy in identifying constructs like sentiment or agency, they often fail to provide transparent explanations for their predictions. This lack of interpretability becomes particularly problematic when these models are used in sensitive domains such as social psychology, where understanding the underlying rationale behind a classification is as important as the classification itself.

To address this gap, the thesis leverages the IG method as a means of providing word-level attribution for classification decisions. In doing so, it seeks to uncover the specific lexical items that contribute most strongly to a model's output.

The work explores both large-scale, pre-trained NLP models (e.g., BERTAgent for agency detection) and smaller, manually labeled datasets where overfitting is encouraged to surface semantically relevant keywords.

More formally, this thesis is guided by the following research objectives:

- To evaluate the effectiveness of the Integrated Gradients method in attributing classification outcomes to individual words in a sentence.
- To compare IG with alternative explainability techniques (e.g., SHapley Additive exPlanations (SHAP), Sequential Integrated Gradients (SeqIG), Deep Learning Important FeaTures (DeepLIFT)) using quantitative metrics such as comprehensiveness, sufficiency, and approximation error.
- To apply IG in both high-resource and low-resource settings and analyze its capacity to extract meaningful word-level insights across these contexts.
- To develop a framework for visualizing IG outputs in a form that is accessible and informative for social science researchers.
- To assess the potential of IG-based analysis for identifying new lexical indicators and expanding existing dictionaries used in psychological research.

Based on these objectives, the following research questions are posed:

- Which words contribute most significantly to the classification of sociopsychological markers such as agency in transformer-based models?
- How does IG compare to other interpretability methods in terms of fidelity and usefulness for NLP applications in social contexts?
- Can overfitting on small labeled datasets, followed by IG analysis, yield meaningful and interpretable class-specific lexical markers?
- How can IG outputs be best visualized to support human interpretation and theory-driven linguistic analysis?

1.3 Contribution of This Thesis

This thesis builds directly upon the findings of a published research paper that applied IG to sociopsychological marker classification [11]. While the original paper demonstrated the feasibility of IG for word-level attribution in a targeted application, namely, agency detection using BERTAgent, this thesis expands that work in multiple important directions, aiming for a broader and more detailed academic contribution suitable for a master's thesis.

The main contributions of this thesis are:

- **Comprehensive Methodological Expansion:** The thesis offers an in-depth exploration of the IG technique, including formal definitions, implementation details, parameter sensitivity analysis, and practical considerations for applying it to NLP tasks. Particular attention is given to choices of baseline, number of integration steps, and IG's completeness property.

- **Systematic Evaluation of Alternatives:** Beyond IG, the thesis evaluates competing interpretability methods such as SHAP, DeepLIFT, and SeqIG. Their performance is rigorously compared using quantitative evaluation metrics (e.g., comprehensiveness, sufficiency, approximation error) across multiple datasets.

- **Unified Framework for Large and Small Datasets:** A novel methodological framework is introduced that extends the IG approach to low-resource scenarios, where small datasets are labeled manually. This includes an intentional overfitting strategy to amplify class-specific signal, allowing for the discovery of distinguishing lexical markers even with limited data.

- **Visualization and Human-Centered Interpretability:** A user-oriented rendering of IG outputs is developed, using SpaCy, an open-source NLP library in Python, to resolve multi-token alignment and improve readability. This visual tool helps domain experts, especially in psychology and linguistics, to interpret model decisions at the word level with higher fidelity.

- **Theoretical and Practical Insights for Social Psychology:** The findings offer linguistically grounded insights into constructs such as agency, shame, and objectification, with implications for both psychological theory and applied NLP. The thesis also discusses how IG-based outputs can inform the development of domain-specific dictionaries and classification tools.

- **Reproducible Research and Open Resources:** All relevant code, models, and datasets used for experimentation have been structured in a reproducible and documented format, aligning with open science principles and allowing others to build upon this work.

In sum, the thesis does not merely replicate the results of the original paper but provides a significantly expanded investigation into the methodological robustness, practical usability, and interdisciplinary relevance of word-level explainability in NLP.

1.4 Thesis Structure

This thesis is organized into six chapters, each addressing a different aspect of the research question, methodology, and findings.

- **Chapter 1 – Introduction:** Provides the background, motivation, and objectives of the study. It outlines the specific research questions and the main contributions of the thesis.
- **Chapter 2 – Literature Review:** Surveys the existing work on sentiment classification, sociopsychological markers, and Explainable Artificial Intelligence (XAI) methods in NLP. It highlights the limitations in current approaches and identifies the research gap addressed by this thesis.
- **Chapter 3 – Methodology:** Details the overall research design, datasets used, and tools implemented. It introduces the Integrated Gradients method and its alternatives, and explains the evaluation metrics adopted for performance benchmarking.
- **Chapter 4 – Experiments and Applications:** Describes the practical application of IG to both large and small datasets. It discusses parameter tuning, algorithmic comparisons, visualization techniques, and real-world evaluations, including annotated highlights for plausibility testing.
- **Chapter 5 – Discussion:** Interprets the main findings, providing a comparative analysis of different explainability techniques and their relevance to sociopsychological research. It also reflects on the implications of the results from a theoretical and applied perspective.
- **Chapter 6 – Conclusion and Future Work:** Summarizes the key results of the thesis, acknowledges its limitations, and proposes avenues for future research, particularly in developing psychologically grounded tools for interpretable Artificial Intelligence (AI).

Each chapter is designed to build upon the previous one, offering a coherent progression from conceptual foundation to empirical analysis and critical reflection. This structure ensures that both technical depth and interdisciplinary relevance are addressed throughout the thesis.

2

Literature Review

2.1 Explainability in Natural Language Processing

Explainability has become a central concern in modern NLP, particularly as models grow in complexity and opacity. While traditional machine learning models allowed for some degree of interpretability through feature inspection or rule-based logic, state-of-the-art deep learning architectures, especially transformers, have made the interpretability task significantly more challenging [4, 12].

In the context of NLP, explainability is especially relevant because language carries both explicit and implicit meaning. Tasks such as sentiment analysis, hate speech detection, and sociopsychological profiling involve latent constructs that are difficult to model without providing some insight into what textual elements influence predictions. In response, the field has increasingly turned to XAI methods, which aim to bridge this interpretability gap by attributing outputs to inputs in a human-understandable form [13, 7].

Among the many proposed techniques, gradient-based methods such as IG [5], DeepLIFT [14], and SHAP [15] have gained traction for their ability to assign importance scores to input tokens. IG, in particular, is noted for its desirable *completeness property*, which ensures that the sum of word attributions equals the model's output difference relative to a baseline [5]. This property, combined with the method's computational feasibility and compatibility with transformer models, has positioned IG as a preferred choice for many NLP

explainability tasks.

However, the adoption of explainability tools remains uneven across domains. In high-stakes areas like finance, healthcare, and social science, regulatory demands and ethical concerns have pushed for greater model transparency [4, 12]. Still, explainability in NLP is not only about compliance, it also provides scientific value, enabling a deeper understanding of the linguistic markers that underlie complex psychological constructs.

2.2 Sentiment and Sociopsychological Marker Analysis

Sentiment analysis has long served as a foundational task in NLP, offering insights into the emotional valence of textual data. Traditional methods such as dictionary-based approaches, e.g., Linguistic Inquiry and Word (LIWC) [16], provided early tools for analyzing text affectively. However, these methods often lacked contextual nuance, struggling with negation, sarcasm, or syntactic complexity. The emergence of deep learning, and particularly transformer-based models, has markedly improved sentiment classification performance, enabling context-aware interpretation and transfer learning [3, 17].

Beyond sentiment, researchers have increasingly turned their attention to more complex sociopsychological markers, such as emotions (e.g., fear, joy), mental states (e.g., depression, shame), and traits (e.g., agency, objectification). These constructs are more semantically rich and less explicitly labeled in text, making them challenging yet important targets for automated detection. For instance, recent work has focused on identifying emotional content in tweets [18], offensive language [19], or even subtle features like self-dehumanization [20].

One of the most notable contributions to this area is the development of BERTAgent [10], a fine-tuned RoBERTa model trained to predict levels of agency in language. Agency, defined as the capacity for intentional and goal-oriented behavior [8], plays a central role in both individual psychology and collective action [21]. The ability to detect agency at scale offers valuable tools for analyzing political mobilization, health communication, and social discourse more broadly.

However, while models like BERTAgent achieve strong predictive accuracy, they operate largely as black boxes. This makes it difficult to assess which

linguistic elements drive model decisions, an issue particularly problematic for constructs like agency, which are abstract and can be expressed in varied and context-dependent ways. This thesis therefore positions word-level attribution methods like IG as a means to bridge this interpretability gap and provide meaningful insights into how models recognize sociopsychological content.

2.3 Transformer-Based Language Models

The introduction of transformer architectures has fundamentally reshaped natural language processing. Unlike previous models such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), or Long Short-Term Memory Networks (LSTMs), transformers rely entirely on self-attention mechanisms to model contextual dependencies across input sequences [22]. This innovation allows transformers to capture long-range linguistic relationships without relying on recurrence, leading to substantial gains in both accuracy and scalability.

Among the most influential transformer-based models is BERT, introduced by Devlin et al. [1]. BERT's bidirectional training and masked language modeling objective enabled significant improvements in various NLP tasks, including question answering, sentiment classification, and named entity recognition. Building upon BERT, RoBERTa was later proposed as an improved variant that eliminates the Next Sentence Prediction objective and leverages more training data and larger mini-batches [2]. Empirical studies confirm that RoBERTa outperforms BERT in most downstream applications, including sentiment and emotion detection [23].

These Pretrained Language Models (PLMs) are typically fine-tuned on specific tasks with relatively small labeled datasets, allowing for high performance even in low-resource settings. However, while PLMs have become the de facto standard for state-of-the-art NLP, their interpretability remains limited. Transformer models often encode latent representations that are difficult to unpack or attribute to specific input features, making it challenging to understand their decisions, particularly in tasks involving nuanced psychological constructs.

This limitation has motivated the integration of explainability tools, such as IG, to extract word-level attributions from models like BERT and RoBERTa. For example, BERTAgent [10], a RoBERTa-based model specifically trained to

detect agency, combines the expressive power of PLMs with task-specific annotation, enabling deeper insight into goal-oriented language. Nonetheless, interpretability still lags behind performance, highlighting the need for structured approaches like the one proposed in this thesis.

2.4 Word-Level Attribution Techniques

Transformer-based models such as BERT and RoBERTa have achieved impressive accuracy across a wide range of NLP tasks. However, their inner workings often remain opaque, especially when interpreting decisions at the word level. This opacity has driven the development of attribution methods designed to explain model outputs by highlighting the contribution of individual input tokens.

One of the most influential techniques is *IG*, introduced by Sundararajan et al. [5]. IG attributes the prediction of a model to its input features by computing the integral of the gradients along a straight path from a baseline input to the actual input. It satisfies the desirable *completeness* property, meaning the sum of attributions equals the difference in model output between the input and the baseline. This makes IG particularly suitable for models where interpretability is essential.

Several enhancements to IG have been proposed for NLP. *SeqIG* modifies the baseline dynamically, replacing only one word at a time with a MASK token [24], while keeping the rest of the sentence unchanged. Though more computationally expensive, it offers fine-grained attribution. *Discretized IG*, proposed by Sanyal and Ren [25], replaces the straight interpolation path with a nonlinear trajectory more reflective of the discrete nature of token embeddings. Despite theoretical advantages, it shows limited improvement on transformer-based models like RoBERTa.

Alternative attribution methods include *DeepLIFT* [14] and *GradientSHAP* [15], both of which build upon gradient-based logic to assign feature importance. DeepLIFT propagates activation differences from a reference input to the actual input, even when gradients are zero. GradientSHAP, on the other hand, blends IG with Shapley values, averaging attributions across multiple baselines.

Comparative evaluations reveal that while methods like SHAP and DeepLIFT provide faster computations, IG often achieves superior performance in terms

of comprehensiveness and sufficiency [13]. This has motivated its application in socially meaningful domains, where it is critical not only to make predictions but to understand what drives them.

2.5 Gaps in the Existing Literature

While significant progress has been made in both the development of powerful language models and the emergence of explainability techniques, several key gaps remain in the literature, particularly at the intersection of interpretable NLP and sociopsychological analysis.

First, most explainability research focuses on general NLP tasks such as sentiment analysis, text classification, or question answering [4, 13], with relatively little attention given to complex, abstract constructs like agency, shame, or objectification. These constructs are more nuanced than simple binary sentiment and often rely on subtle lexical cues, making them harder to model and explain. There is a lack of systematic investigations into how interpretability tools perform in these psychologically grounded domains.

Second, many existing studies evaluate attribution methods using synthetic or benchmark datasets that may not reflect the linguistic diversity or psychological depth of real-world text [7]. Few works assess whether the words identified as important by attribution methods align with human judgments or expert annotations in a meaningful way. This limits their utility in domains such as social science and psychology, where interpretability must be both technically valid and semantically credible.

Third, current attribution methods are usually evaluated in high-resource contexts, assuming access to large, labeled datasets. However, in practical applications, particularly those involving domain-specific psychological constructs, such datasets are often unavailable. There is a lack of approaches tailored to small-scale or sparse data settings where overfitting may paradoxically be exploited to highlight class-specific linguistic patterns.

Finally, although techniques like IG offer strong theoretical properties, they are not always presented in ways that are usable or interpretable by non-technical users. Existing implementations often produce token-level outputs that are difficult to map back to natural language constructs, especially in the presence of subword tokenization. More effort is needed to build human-centered visual-

ization tools that make these attributions readable, interpretable, and useful in applied research.

This thesis addresses these gaps by applying, evaluating, and extending word-level attribution techniques, primarily IG, within the context of sociopsychological text classification. It focuses on bridging the gap between computational precision and theoretical insight, particularly in domains where interpretability is not optional but essential.

3 Methodology

3.1 Research Design

The methodological foundation of this thesis is built on the application and evaluation of explainability techniques in NLP, with a specific focus on sociopsychological semantic markers. The core objective is to assess how well these techniques, especially IG, can attribute classification decisions to individual words, thereby offering interpretable insights into model behavior.

This chapter is structured around the experimental pipeline used throughout the thesis. It begins with the description of the datasets employed, followed by an overview of the main tools and models used in the analysis: RoBERTa as the base transformer model, SpaCy for token alignment and linguistic processing, and BERTAgent for agency classification. The chapter then details each explainability method applied, including their theoretical rationale, implementation specifics, and strengths or limitations. Finally, it outlines the evaluation metrics and methodological considerations that shaped the design of experiments.

The research follows a dual-path strategy:

- In high-resource settings, pre-trained models (e.g., BERTAgent) are used on large labeled datasets to examine how IG performs under well-optimized classification conditions.
- In low-resource scenarios, small manually labeled datasets are used with overfitting as an intentional strategy to force model memorization, enabling IG to expose key lexical features that define different classes.

This design allows for a comprehensive understanding of IG's behavior

across real-world NLP use cases and provides insight into how explainability tools can be leveraged not just for auditing models, but also for enhancing theoretical understanding in social psychology.

3.2 Datasets

This thesis relies on five distinct datasets to evaluate the behavior and robustness of attribution methods across different types of textual content. The datasets were selected to represent a broad range of linguistic styles, text lengths, and psychological contexts.

3.2.1 #covid19

A collection of Twitter posts made during the early months of the COVID-19 pandemic, filtered using the hashtag #covid19. The dataset consists of 1000 tweets per week, selected based on engagement (likes, retweets, replies), spanning February to April 2020 [26]. These tweets were kept in their original form to preserve the full context and character limit constraints, resulting in longer sentence structures on average.

The following tweets demonstrate how public discourse around COVID-19 varies in agentic tone:

High Agency Example

"Support makes everyone stronger. And in our fight against COVID, we need each other's support as a society. Let us help each other more, and win this fight collectively! #covid #strongertogether"

Low Agency Example

"Delhi made a mistake notifying the domicile rules during this time of crisis and I made a mistake getting sucked into this debate at a time when it is wrong to let politics distract from the COVID fight."

3.2.2 IMDB

A widely-used dataset for sentiment analysis, containing 50,000 movie reviews balanced evenly between positive and negative sentiment [27]. To suit the sentence-level analysis focus of this thesis, the reviews were split into individual sentences.

An example sentence from the IMDB dataset illustrating different levels of agency:

High Agency Example

"One gets the impression that the sole message is the only way you can succeed and make positive gain is if you accept Jesus as your personal savior."

Low Agency Example

"The set pieces are clumsily set up if at all and are badly executed, it is just awful on every front apart from the music maybe, I do not remember thinking the music stinks apart from the songs."

3.2.3 Slurs

Drawn from a 2022 questionnaire study [20], this dataset includes 336 narratives describing episodes of verbal aggression experienced by women. Each entry was labeled by psychological experts according to aggressor gender, relationship type, and perceived emotional and objectification levels. Due to its thematic richness, the dataset is especially suitable for studying nuanced sociopsychological markers such as shame and self-objectification.

Examples from the slurs dataset show how narrations can vary in agency expression:

High Agency Example

"So I move forward a little and, a little at a time, I try to enter."

Low Agency Example

"I feel disrespected and discredited."

3.2.4 Snippets

A set of very short social media posts (typically 2–5 words) taken from the "Affect in Tweets" dataset used in SemEval 2018 [18]. This dataset provides insight into attribution behavior in highly compressed texts, where every token carries significant semantic weight.

Even short posts can express varying levels of agency:

High Agency Example

"I unfollowed without hesitation."

Low Agency Example

"People like that irritate my soul."

3.2.5 Highlights

The highlights dataset consists of 802 short texts (100–125 words), each written by a native English speaker on a socially relevant issue that they personally identified as important. Authors were prompted to *"review the list of socially relevant issues below* [1] *and select the one that resonates most with you and you believe requires mobilizing others to take action in order to address it."* After completing their texts, the authors were asked to highlight the portions they felt were most effective in encouraging collective action.

To enrich the dataset and validate these self-assessments, the same highlighting task was also completed by 2,347 independent external readers, with each text receiving evaluations from at least two readers (mean = 2.4, SD = 0.76). These reader annotations help capture a broader perception of persuasive content.

This dataset is used in the present thesis to evaluate the plausibility of IG-based attributions by comparing model-identified high-agency segments with human-highlighted regions. All data referenced in this study were collected before December 3rd, 2025 [28].

[1] The list included issues such as climate change, racism, homelessness, gender inequality, access to education, and public health.

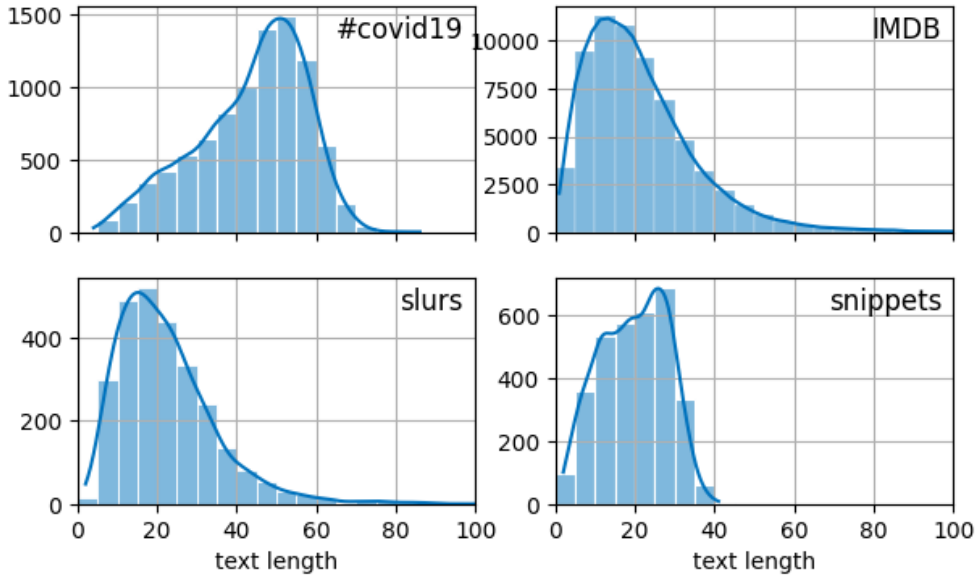

Figure 3.1: Text length distribution across datasets: the histograms show the number of samples binned by sentence/token length for each dataset.

To ensure correct input to the classifiers, the *IMDB* and *slurs* datasets, which contain longer texts, were split into sentences to control for text length. A superficial cleaning step was applied to all datasets, removing mentions, HTML links, special characters, and redundant whitespace. The resulting token length distribution for each dataset is illustrated in Figure 3.1. As shown, *IMDB* and *slurs* have a comparable sentence-level length distribution, primarily active in the 5–40 token range. In contrast, the #covid19 dataset exhibits higher average length, with most samples ranging from 30–70 tokens. The *snippets* dataset, by design, contains the shortest texts, typically in the lowest range of 0–40 tokens.

To ensure a balanced and computationally tractable test set, subsets of each dataset were sampled across their full length spectrum. Specifically, 1600 tweets were selected from #covid19, evenly distributed over the [25, 64] length range (40 samples per length). From *IMDB*, 1500 sentences were sampled in the range [1, 49] (30 samples per length), while 1335 samples were taken from *slurs*, uniformly covering lengths from 5 to 40 (about 40 per length). Finally, 300 samples were chosen from *snippets*, targeting lengths between 2 and 14 (about 20 per length). This strategy ensures that each dataset contributes meaningfully across a diverse range of text sizes, facilitating robust comparisons.

3.3 Tools: RoBERTa, SpaCy, BERTAgent

The following tools were central to the experiments and analysis conducted in this thesis. Each tool was chosen for its role in either text classification, linguistic preprocessing, or interpretability enhancement. This section provides an in-depth overview of the theoretical foundation, implementation details, and use cases of RoBERTa, SpaCy, and BERTAgent.

3.3.1 RoBERTa: A Robustly Optimized BERT Pretraining Approach

RoBERTa is a transformer-based language model developed by Liu et al. [2], building on the success of BERT introduced by Devlin et al. [1]. Like BERT, RoBERTa is trained on the Masked Language Modeling (MLM) objective but removes the Next Sentence Prediction (NSP) objective and introduces several key improvements aimed at increasing performance and robustness.

Transformer Architecture RoBERTa follows the standard transformer encoder architecture described by Vaswani et al. [22]. The core of this architecture is the self-attention mechanism, which allows the model to compute contextualized representations of each word by attending to all other words in the input sequence.

Given a sequence of input tokens $x = (x_1, x_2, \ldots, x_n)$, each token is first embedded into a vector $e_i \in \mathbb{R}^d$, and these embeddings are passed through multiple layers of transformer blocks. Each block includes: - Multi-head self-attention:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

where Q, K, and V are the query, key, and value matrices derived from the input embeddings. - Position-wise feedforward networks:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

Key Improvements Over BERT RoBERTa introduces several modifications to the original BERT pretraining protocol:

- Removal of the NSP objective, which was found to be unnecessary and even detrimental to some downstream tasks.
- Dynamic masking during pretraining rather than static masking, improving the diversity of learned representations.
- Larger mini-batch sizes and longer training durations on significantly more data.
- Training on an aggregate of corpora including BookCorpus, CC-News, OpenWebText, and STORIES.

Tokenization and Subword Units RoBERTa, like BERT, uses the Byte-Pair Encoding (BPE) algorithm to tokenize input text into subword units. This is crucial for handling rare or unknown words while maintaining a manageable vocabulary size. For example, the word "unmotivated" might be tokenized into:

```
["un", "##motiv", "##ated"]
```

This subword decomposition is critical when analyzing model attribution outputs, as IG will assign values at the subword level unless handled post hoc (as discussed in Section 4.4).

Fine-Tuning for Downstream Tasks RoBERTa is designed for transfer learning. Once pretrained, the model can be fine-tuned on a specific classification or regression task by appending a task-specific output head (e.g., a fully connected layer with a softmax activation for classification). This fine-tuning updates all parameters based on the task-specific loss function:

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

where C is the number of classes, y_i is the true label, and $\hat{y}_i$ is the predicted probability for class i.

Input Processing and Embedding Access To interpret model outputs using IG, the token embeddings must be extracted directly from RoBERTa's embedding layer. The sentence is first tokenized using HuggingFace's `AutoTokenizer`, which returns three key components:

- `input_ids`: a tensor of token IDs for the sentence, including special tokens (`[CLS]`, `[SEP]`),
- `attention_mask`: a binary mask indicating real tokens (1) versus padding (0),
- `offset_mapping`: character spans for each token, used for alignment with SpaCy tokens.

```
encodings = tokenizer(
    sentence,
    return_tensors="pt",
    padding=False,
    truncation=True,
    add_special_tokens=True,
    return_attention_mask=True,
    return_offsets_mapping=True
)
input_ids = encodings['input_ids'].to(device)
attention_mask = encodings['attention_mask'].to(device)
```

Code 3.1: Encoding input sentence with HuggingFace tokenizer

Embeddings are then retrieved using the model's embedding layer:

```
input_embeddings = model.roberta.embeddings.word_embeddings(input_ids
    ).detach()
input_embeddings.requires_grad = True
```

Code 3.2: Extracting input embeddings from RoBERTa

Use in This Thesis In this work, RoBERTa serves as the base architecture for all classification tasks. Pretrained weights were loaded via the HuggingFace Transformers library, and the model was either used directly (e.g., via BERTAgent) or fine-tuned on smaller datasets using `RobertaForSequenceClassification`. During experiments with Integrated Gradients and other attribution methods, the internal layers of RoBERTa provided the gradient flows necessary to compute word-level contributions.

3.3.2 SpaCy: Linguistic Annotation and Token Alignment

SpaCy is a high-performance NLP library developed by Explosion AI [29]. Designed for speed, robustness, and practical usability, SpaCy is widely used for

tasks such as tokenization, Part-of-Speech (POS) tagging, syntactic dependency parsing, Named Entity Recognition (NER), and sentence segmentation.

In this thesis, SpaCy is not used as a classifier, but rather as a **linguistic processing engine**. Its role is to bridge the gap between RoBERTa's subword-level outputs and human-readable tokens by:

- Producing sentence-level token boundaries.
- Identifying syntactic dependencies (e.g., negations, phrasal verbs).
- Providing alignment between RoBERTa subword tokens and complete words.

Tokenization and Alignment Unlike RoBERTa, which uses BPE for tokenization into subword units, SpaCy performs rule-based tokenization, yielding linguistically interpretable tokens. For instance, the sentence:

"These people are not lazy at all!"

would be tokenized by SpaCy into:

```
["These", "people", "are", "not", "lazy", "at", "all", "!"]
```

Whereas RoBERTa might split `"lazy"` into subword pieces like:

```
["la", "##zy"]
```

To visualize attribution maps at the word level, it is necessary to **recombine RoBERTa subword scores** into full tokens. This is done by:

- Mapping BPE subwords back to SpaCy tokens using alignment based on character offsets.
- Aggregating attribution scores for subwords belonging to the same word (typically by summing or averaging).

Syntactic Dependencies SpaCy's dependency parser is based on transition-based parsing using neural networks. It assigns each word a grammatical role in the sentence, defining head–dependent relationships. This is critical when handling constructs like negations or passive voice.

For example, in:

"She is not motivated."

SpaCy would annotate "not" as a **negation modifier** (`neg`) of "motivated", allowing us to associate the attribution not only with the negation term, but with the target verb. This is essential when performing IG-based visualizations, where a simple word-level heatmap could misrepresent meaning if negations are not properly contextualized.

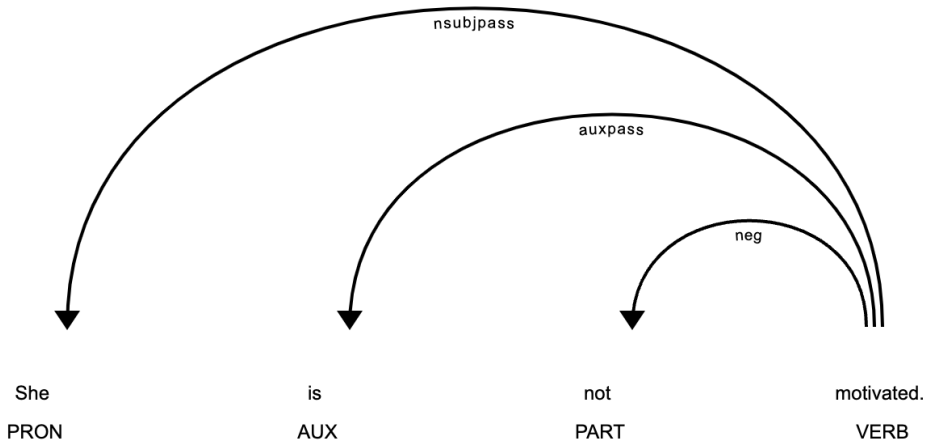

Figure 3.2: Dependency parse of the sentence "She is not motivated." using SpaCy. The model identifies "not" as a `neg` modifier of "motivated", with "is" as an auxiliary verb (`auxpass`).

Negation and Phrase Correction Logic One of the preprocessing contributions of this thesis is a **negation-aware recomposition strategy**. This involves:

- Detecting negation modifiers (`neg`) using SpaCy's dependency tree.
- Identifying phrasal elements such as auxiliaries (`aux`), passive markers (`auxpass`), or adjectival complements (`acomp`).
- Merging relevant tokens to ensure that IG attribution highlights the correct semantic chunk.

For instance, the phrase:

"not be lazy"

would be treated as a **single conceptual unit**, with its total attribution score reflecting the combination of "not", "be", and "lazy".

Implementation Details The version used in this thesis is `spaCy v3.5`, and the `en_core_web_sm` model was employed for all linguistic annotations. Token alignments were performed using SpaCy's `Doc.char_span` function to map character spans from BPE back to human-readable tokens.

This enabled accurate IG attribution rendering and allowed semantic interpretations to be more trustworthy, especially in the context of socially nuanced language.

3.3.3 BERTAgent: A Model for Agency Detection

BERTAgent is a transformer-based classifier developed specifically to detect the sociopsychological marker of **agency** in textual data [10]. Built upon the RoBERTa architecture, it serves as a dedicated model for identifying linguistic expressions of intentionality, goal-orientation, and personal efficacy — features central to social and political psychology.

Definition of Agency In psychological literature, *agency* refers to the capacity for intentional and goal-directed action [8]. It encompasses constructs like motivation, autonomy, control, and purposeful behavior. From a linguistic perspective, agency is often encoded implicitly, through verbs, pronouns, and sentence structure, making its detection both important and nontrivial.

BERTAgent was designed to bridge this gap by quantifying the presence of agentic language in natural text. This makes it especially useful for applications such as political discourse analysis, mental health monitoring, and collective action research.

Model Architecture BERTAgent leverages the `RobertaForSequenceClassification` model from HuggingFace Transformers, fine-tuned on a custom dataset annotated for agency. The base architecture inherits the full RoBERTa encoder stack:

- Input token embeddings are processed through 12 transformer layers (in the base version).
- The output corresponding to the `[CLS]` token is passed to a classification head: a linear layer followed by a sigmoid (for regression).
- The output is a scalar value representing the degree of agency in the input sentence.

Training Dataset and Labeling Strategy The training data for BERTAgent was generated from a curated lexicon of agency-related terms, derived from prior psychological research [30, 31]. Sentences containing these terms were extracted and annotated by a panel of expert raters. This resulted in a robust labeled dataset spanning various degrees of agentic expression.

The model was trained using the standard binary cross-entropy or mean squared error loss, depending on whether a regression or classification output was targeted:

$$\mathcal{L} = -\left(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right) \quad \text{(for binary classification)}$$

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad \text{(for regression)}$$

Use in This Thesis In this thesis, BERTAgent serves two roles:

1. **As a benchmark classifier for high-agency detection.** It is used on datasets like `#covid19`, `IMDB`, and `snippets` to evaluate how well attribution methods like IG identify agentic features.
2. **As a controlled model for IG evaluation.** Since BERTAgent outputs are continuous (in the regression variant), they are ideal for computing precise attribution scores using IG's completeness property:

   $$F(x) - F(x_0) = \sum_{i=1}^{n} a_i(x)$$

   where $F(x)$ is the model output for the input x, x_0 is the baseline input, and $a_i(x)$ is the attribution for token i.

In Section 4, BERTAgent is used to generate heatmaps of word-level agency via Integrated Gradients, and the results are compared to human-annotated highlights. Its regression output enables fine-grained interpretability, making it a crucial component of the thesis methodology.

3.4 Explainability Methods: IG, GradientShap, SeqIG, DeepLIFT

To interpret the predictions made by transformer-based NLP models, this thesis adopts several post hoc explainability techniques. These methods assign attribution scores to input features (i.e., tokens or subwords), quantifying their contribution to the final prediction. The techniques explored include IG, SeqIG, GradientShap, and DeepLIFT. This section outlines the theoretical basis, mathematical formulation, and practical role of each.

3.4.1 Integrated Gradients (IG)

IG [5] is a gradient-based attribution method designed to satisfy two desirable axioms: *Sensitivity* and *Completeness*. It estimates the contribution of each input feature by integrating gradients along the path from a baseline input to the actual input.

Given a model $F : \mathbb{R}^n \to \mathbb{R}$ and input vector x, let x_0 be a baseline (typically all zeros or a neutral embedding). The IG attribution for the i-th input feature is:

$$\text{IG}_i(x) = (x_i - x_{0,i}) \cdot \int_{\alpha=0}^{1} \frac{\partial F(x_0 + \alpha \cdot (x - x_0))}{\partial x_i} d\alpha$$

In practice, the integral is approximated numerically via a Riemann sum with m steps:

$$\text{IG}_i(x) \approx (x_i - x_{0,i}) \cdot \frac{1}{m} \sum_{k=1}^{m} \frac{\partial F(x_0 + \frac{k}{m} \cdot (x - x_0))}{\partial x_i}$$

Completeness property:

$$\sum_{i=1}^{n} \text{IG}_i(x) = F(x) - F(x_0)$$

In this thesis, IG is computed using the Captum library, with various baselines tested (zero, mean, MASK, PAD) and steps m ranging from 50 to 1000. See Section 4 for empirical comparisons.

Integrated Gradients with Captum This thesis uses the Captum library [32] to compute IG. The key method is `IG.attribute()`, which accepts the following parameters:

- `inputs`: token-level embeddings, not input IDs.
- `baselines`: a baseline embedding tensor, created per method (see below).
- `additional_forward_args`: a tuple containing the attention mask.
- `n_steps`: number of integration steps (e.g., 50, 100, 300).
- `target`: the regression target index; set to 0 since output is scalar.
- `return_convergence_delta`: False, as we are not using convergence tracking.

```
attributions = IG.attribute(
    inputs=input_embeddings,
    baselines=baseline_embeds,
    additional_forward_args=(attention_mask,),
    internal_batch_size=128,
    n_steps=n_steps,
    target=0,
    return_convergence_delta=False
)
```

Code 3.3: Captum IG attribution call

Baseline Embedding Strategies The choice of baseline affects attribution significantly. This thesis explores four options:

- **Zero baseline**: all tokens set to zero vectors.
- **Mean baseline**: each token set to the mean embedding across the vocabulary.
- **Mask baseline**: token IDs replaced with `[MASK]` and embedded.
- **Pad baseline**: token IDs replaced with `[PAD]` and embedded.

The baseline is created based on the input tensor length:

```
if base_token == 'pad':
    baseline_ids = torch.tensor([[tokenizer.pad_token_id] * input_ids.size(1)]).to(device)
    baseline_embeds = model.roberta.embeddings.word_embeddings(baseline_ids).detach()
```

```
elif base_token == 'mask':
    baseline_ids = torch.tensor([[tokenizer.mask_token_id] *
    input_ids.size(1)]).to(device)
    baseline_embeds = model.roberta.embeddings.word_embeddings(
    baseline_ids).detach()
elif base_token == 'zero':
    baseline_embeds = torch.zeros_like(input_embeddings).to(device)
elif base_token == 'mean':
    baseline_embeds = mean_embedding.unsqueeze(0).expand_as(
    input_embeddings)
```

Code 3.4: Building various baselines

To ensure that the [CLS] and [SEP] token do not affect attribution due to their structural role, their embeddings are copied from the original input into the baseline:

```
special_token_mask = (input_ids == tokenizer.cls_token_id) | (
    input_ids == tokenizer.sep_token_id)
baseline_embeds[special_token_mask] = input_embeddings[
    special_token_mask]
```

Code 3.5: Retaining special tokens

Embedding-Level Forward Wrapper Because Captum's IG method requires access to the embeddings, a wrapper is used to pass `inputs_embeds` into the model:

```
class ModelWrapper(torch.nn.Module):
    def __init__(self, model):
        super().__init__()
        self.model = model

    def forward(self, input_embeddings, attention_mask):
        return self.model(inputs_embeds=input_embeddings,
    attention_mask=attention_mask)[0]
```

Code 3.6: Model wrapper

3.4.2 Sequential Integrated Gradients (SeqIG)

SeqIG [24] is a variant of IG designed for NLP, where token interactions are discrete and interdependent. Instead of interpolating the entire input at once,

SeqIG isolates and perturbs one token at a time, replacing it with a MASK while keeping others fixed. This allows the model to isolate the contextual impact of each word more precisely.

SeqIG sacrifices the completeness property but often yields more intuitive explanations in short texts (see Section 4). It is significantly more computationally expensive, as it must re-run IG for each token individually.

3.4.3 SHAP (GradientShap)

SHAP [15] is a family of attribution methods grounded in cooperative game theory. It assigns each input feature a Shapley value, representing its average marginal contribution across all feature subsets. GradientShap combines IG with sampling techniques to approximate Shapley values efficiently.

Mathematically, the SHAP value ϕ_i for input x_i is:

$$\phi_i = \mathbb{E}_{S \subseteq N \setminus \{i\}} \left[\frac{|S|!(|N| - |S| - 1)!}{|N|!} (F(x_{S \cup \{i\}}) - F(x_S)) \right]$$

In practice, GradientShap samples multiple baselines and applies IG to estimate these values, producing smooth attribution maps.

SHAP values satisfy the following desirable properties:

- **Additivity**: the attributions sum to the difference in output.
- **Symmetry**: equal features get equal attribution.
- **Dummy**: features with no effect get zero attribution.

3.4.4 DeepLIFT

DeepLIFT [14] is a backpropagation-based method that compares the activation of each neuron to a reference baseline and propagates contribution scores rather than gradients. This avoids issues with vanishing gradients and enables explanation even in flat activation regions.

Let x be the input and x_0 the reference. The contribution C_i of input x_i is:

$$C_i = \frac{\Delta y}{\Delta x_i} \cdot \Delta x_i$$

where $\Delta y = F(x) - F(x_0)$ and $\Delta x_i = x_i - x_{0,i}$.

DeepLIFT propagates these contributions through the network by computing multipliers analogous to gradients but relative to a reference.

3.5 Evaluation Metrics: Comprehensiveness, Sufficiency, Approximation Error

To assess the quality and reliability of attribution methods in NLP, it is important to use quantitative metrics that reflect how faithfully these methods explain model behavior. In this thesis, we adopt three primary evaluation metrics commonly used in the explainability literature: **comprehensiveness**, **sufficiency**, and **approximation error**. Each of these provides a different lens through which to assess whether the attributions correspond to meaningful and influential parts of the input.

3.5.1 Comprehensiveness

Comprehensiveness measures how much the prediction score drops when the most important words (as identified by the attribution method) are removed from the input [33]. If the words with the highest attribution scores truly drive the model's decision, removing them should lead to a significant decrease in the output score.

Formally, for an input x and a fraction f of top-attributed tokens removed, the comprehensiveness $C_f(x)$ is defined as:

$$C_f(x) = F(x) - F(x_f^c)$$

where $F(x)$ is the model output for the original input, and $F(x_f^c)$ is the output after removing the top f fraction of tokens (i.e., those with the highest attribution values in absolute terms).

A higher $C_f(x)$ value indicates that the removed tokens significantly contributed to the prediction, which implies that the attribution method identified important tokens.

3.5.2 Sufficiency

Sufficiency is the complement of comprehensiveness. It measures how much of the original prediction score can be retained when only the most attributed tokens are kept and the rest are removed [33]. A good attribution method should be able to identify a small subset of words that are sufficient to maintain the model's prediction.

It is defined as:

$$S_f(x) = F(x) - F(x_f^s)$$

where x_f^s is a reduced input that retains only the top f fraction of tokens (again, by attribution score).

A lower $S_f(x)$ value is better, indicating that the retained words are sufficient to reconstruct the original prediction.

3.5.3 Approximation Error

Approximation Error (AE) quantifies how well the sum of the attribution scores approximates the model output difference between the input and the baseline. This metric is particularly important for methods like IG, which are designed to satisfy the **completeness** property:

$$\sum_{i=1}^{n} a_i(x) \approx F(x) - F(x_0)$$

The AE is computed as a relative error:

$$\text{AE}(x) = \left| \frac{\sum_{i=1}^{n} a_i(x) - (F(x) - F(x_0))}{F(x) - F(x_0)} \right|$$

where: - $a_i(x)$ is the attribution for token i, - $F(x)$ is the model prediction on input x, - $F(x_0)$ is the prediction for the baseline input x_0.

A smaller AE indicates that the method is numerically stable and correctly approximates the output change.

Implementation Context All three evaluation metrics, comprehensiveness, sufficiency, and approximation error, were implemented through custom PyTorch functions tailored to the structure of the BERTAgent model. These func-

tions operate directly on token-level attributions and model outputs, rather than relying on built-in tools such as Captum. For both sufficiency and comprehensiveness, multiple values of f (e.g., 10%, 20%, 30%) were tested to understand the trade-off between number of words used and interpretability quality. Approximation Error was calculated only for IG method that provides additive attributions.

These metrics are used in Chapter 4 to compare attribution methods across datasets and interpretability configurations.

Adaptation for Regression-Based Models In standard classification tasks, comprehensiveness and sufficiency are often evaluated by observing changes in the predicted class probability after masking a subset of input features. However, in this thesis, BERTAgent is implemented as a *regressor* that outputs a continuous agency score rather than class probabilities. As such, the evaluation metrics must be adapted accordingly.

First, instead of selecting tokens with the highest *signed* attribution values, we select the top k% tokens with the highest **absolute** attribution values. This is because both strongly positive and strongly negative attributions indicate influence on the regression output.

Second, the prediction difference is computed using the absolute difference in scalar model output:

$$\text{Comprehensiveness}(x) = \left|F(x) - F(x_k^c)\right|, \quad \text{Sufficiency}(x) = \left|F(x) - F(x_k^s)\right|$$

where $F(x)$ is the original model output, and $F(x_k^c)$ and $F(x_k^s)$ are the outputs after masking the top k% most influential tokens (removed or retained, respectively).

The approximation error is calculated the same way as in classification models:

$$\text{AE}(x) = \left| \frac{\sum_i a_i(x) - (F(x) - F(x_0))}{F(x) - F(x_0)} \right|$$

Implementation Snippet The following PyTorch code illustrates how sufficiency, comprehensiveness, and approximation error were computed for regression outputs:

```python
def _eval_sufficiency(self, forward_fn, input_embed, attention_mask, attr, topk):
    original_output = forward_fn(input_embed, attention_mask).item()
    abs_attr = torch.abs(attr)
    topk_indices = torch.topk(abs_attr, int(abs_attr.shape[0] * topk / 100), sorted=False).indices
    mask = torch.zeros_like(input_embed[0][:, 0]).bool()
    mask[topk_indices] = 1
    masked_input_embed = input_embed[0][mask].unsqueeze(0)
    masked_attention_mask = attention_mask[0][mask].unsqueeze(0)
    logits_perturbed = forward_fn(masked_input_embed, masked_attention_mask).item()
    return abs(original_output - logits_perturbed)

def _eval_comprehensiveness(self, forward_fn, input_embed, attention_mask, attr, topk):
    original_output = forward_fn(input_embed, attention_mask).item()
    abs_attr = torch.abs(attr)
    topk_indices = torch.topk(abs_attr, int(abs_attr.shape[0] * topk / 100), sorted=False).indices
    mask = torch.ones_like(input_embed[0][:, 0]).bool()
    mask[topk_indices] = 0
    masked_input_embed = input_embed[0][mask].unsqueeze(0)
    masked_attention_mask = attention_mask[0][mask].unsqueeze(0)
    logits_perturbed = forward_fn(masked_input_embed, masked_attention_mask).item()
    return abs(original_output - logits_perturbed)

def _approximation_error(self, model_output, baseline_output, attr_sum):
    output_diff = model_output - baseline_output
    return abs((attr_sum - output_diff) / output_diff)
```

Code 3.7: Evaluation metrics adapted for regression with BERTAgent

These changes ensure the evaluation metrics remain meaningful and consistent with the regression objective of BERTAgent.

3.6 Methodological Challenges and Design Considerations

Despite the growing availability of interpretability tools and pretrained transformer models, their integration into sociopsychological research workflows remains nontrivial. This section outlines the main challenges encountered in the design and execution of the thesis methodology and discusses the strategies adopted to overcome them.

3.6.1 Attribution at the Embedding Level

One of the first design decisions involved determining where to apply the IG method. IG is classically applied to input features, but in transformer-based NLP models, these are token embeddings, not raw text. This necessitated accessing the `inputs_embeds` interface in HuggingFace's Transformers library. A custom wrapper class was implemented to allow IG to operate directly on the model's embedding layer, bypassing the default input ID path. This added technical complexity but ensured that gradient attribution correctly captured the semantic contribution of each token.

3.6.2 Baseline Design for IG

Choosing a baseline is critical in IG, as it represents the "absence" of input. In NLP, this is inherently ambiguous: should absence mean silence (zero vector), a placeholder (mask), an empty structure (pad), or a neutral average (mean embedding)? Each of these introduces different biases. This thesis tested multiple baseline types—`zero`, `pad`, `mask`, and `mean`—and retained the original `[CLS]` and `[SEP]` token embeddings in all baseline inputs to eliminate artificial attribution due to structural tokens. This design improves interpretability while preserving IG's completeness property.

3.6.3 Special Token Handling and Alignment

Because transformer tokenization often splits words into subword units (e.g., "unmotivated" $\rightarrow$ ["un", "##motiv", "##ated"]), it was necessary to merge subword attributions to recover meaningful word-level scores. Additionally, special

tokens like `[CLS]`, `[SEP]`, and `[PAD]` can introduce attribution noise. These were excluded from interpretation or forcibly assigned zero attributions. Mapping from RoBERTa tokens back to SpaCy tokens was performed using offset mappings, and multiple overlapping subwords were aggregated by span alignment (detailed in Section 4.4).

3.6.4 Attribution Evaluation for Regression Outputs

Most interpretability metrics are defined for classification tasks and rely on class probability changes. Since BERTAgent outputs a continuous regression score, both **comprehensiveness** and **sufficiency** had to be redefined. This was done by:

- Selecting top-k% tokens based on absolute attribution values, recognizing that both strongly positive and negative attributions indicate influence.
- Measuring output change as the absolute difference between the original and perturbed regression scores.

This adaptation ensures that the metrics remain meaningful in a scalar-output setting.

3.6.5 Interpretability vs. Faithfulness Trade-offs

While IG is theoretically faithful due to its completeness property, variants such as Sequential IG (SeqIG) and GradientShap sometimes produce more intuitive results in practice. However, these come with trade-offs:

- SeqIG is computationally expensive and loses completeness.
- GradientShap's reliance on sampled baselines introduces variance.
- DeepLIFT fails in saturation zones of activations common in transformer models.

To manage these trade-offs, this thesis benchmarked each method across three datasets using multiple faithfulness metrics and documented interpretability outcomes through visual and numerical analysis.

3.6.6 Hardware and Memory Constraints

Running IG over large datasets, especially with hundreds of interpolation steps and multi-token baselines, posed significant memory challenges. To mitigate this:

- Experiments were batched and run with reduced internal batch sizes.
- Attribution was computed only on a curated subset of examples with controlled token lengths.
- GPU memory was freed between forward passes using `torch.cuda.empty_cache()`.

These constraints influenced the design of evaluation subsets (detailed in Section 4) and guided the selection of representative samples for analysis.

4

Experiments and Applications

4.1 Overview of Experimental Setup

The experiments in this thesis are organized to evaluate and compare interpretability methods, primarily IG, in their ability to explain model predictions related to sociopsychological constructs, such as agency. The primary model used is BERTAgent, a regression-based classifier fine-tuned to detect agency in natural language text. Each experiment focuses on a different aspect of attribution quality, interpretability, and model transparency.

This chapter begins by optimizing the IG configuration through a detailed analysis of baseline selection and step size tuning (Section 4.2). It then moves into comparative evaluation of other attribution methods (e.g., SHAP, DeepLIFT, and SeqIG), visualization strategies, and validation of interpretability using human annotations. Special attention is also given to scenarios involving limited data, where attribution can be used to support dictionary expansion and extract class-specific keywords.

4.2 Optimization of Integrated Gradients Configuration

Before applying IG across datasets, it was necessary to select an optimal configuration for the baseline and number of integration steps N. These parameters significantly affect both the numerical stability of IG and the interpretability of

its attributions. This section evaluates these parameters using four criteria: comprehensiveness, sufficiency, model output alignment, and approximation error. All experiments were performed using the BERTAgent model on selected samples from four datasets.

4.2.1 Baseline Comparison

Figure 4.1 compares four baseline strategies—`zero`, `mask`, `pad`, and `mean`—in terms of comprehensiveness (top row) and sufficiency (bottom row) across four datasets. Attributions were computed using IG with $N = 300$ steps.

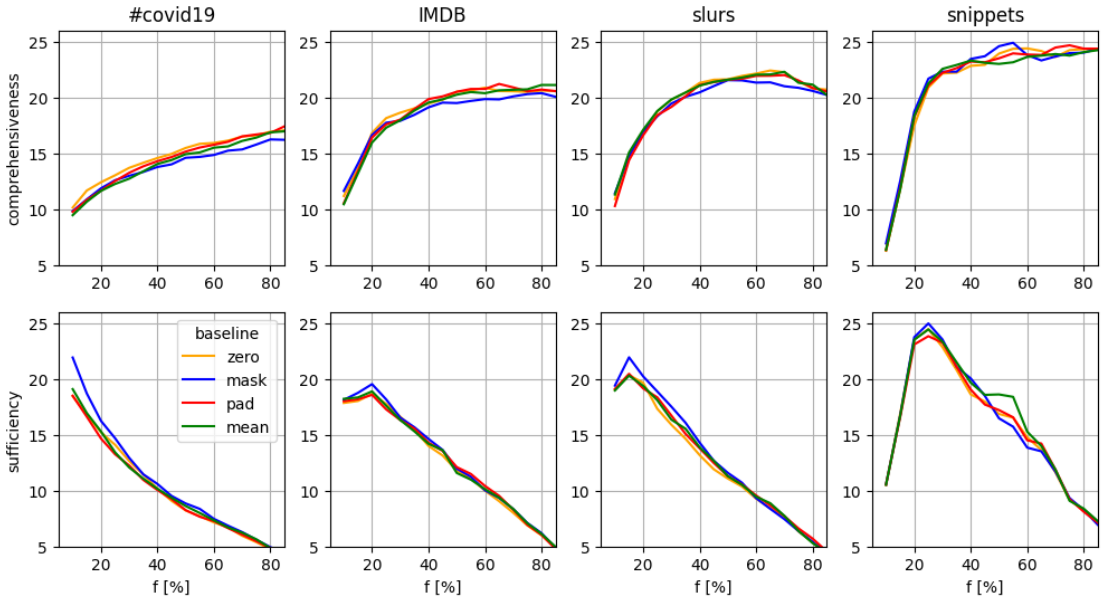

Figure 4.1: Comprehensiveness and sufficiency in IG (with $N = 300$ steps) as a function of the fraction f in the four datasets, and for different baseline choices.

Computation Details. For each sentence in the dataset (e.g., Snippets), we apply IG using a selected baseline and extract the attribution scores for all input tokens. We then calculate comprehensiveness and sufficiency for multiple values of f, specifically, for $f \in \{10\%, 15\%, \ldots, 85\%\}$. This yields an array of scores for each sample. Each element in the array reflects the drop (for comprehensiveness) or retention (for sufficiency) in the model's output when only the top $f\%$ most attributed tokens are either removed or retained.

Once the per-sample arrays are computed, we average the values across all samples for each f level. These mean values are then used to generate the

aggregate plots shown in Figure 4.1, capturing the overall effect of each baseline on attribution reliability.

While all baselines behave similarly in general, some clear trends emerge. The `mask` baseline (in blue) consistently performs worst, yielding the lowest comprehensiveness and highest sufficiency, particularly at smaller values of f. Although omitted for readability, error analysis confirms that this difference is statistically significant. In contrast, the `zero`, `pad`, and `mean` baselines all perform comparably well, and can be considered reliable choices for IG. These findings suggest that the `mask` baseline introduces artifacts or model biases that distort attribution reliability.

4.2.2 Baseline Neutrality and Model Output Behavior

Comprehensiveness and sufficiency are helpful for assessing how well a baseline highlights influential tokens, but they do not account for how the baseline affects the model's absolute output. According to IG's completeness property, the sum of the attributions should ideally match the difference between the original model output and the baseline output, i.e., $F(x) - F(x_0)$. This implies that the baseline output $F(x_0)$ should ideally be as close to zero as possible to ensure that the attribution sum directly reflects the model's prediction $F(x)$.

To analyze this aspect, Figure 4.2 shows the value of the model's output at the baseline input $F(x_0)$ as a function of the input length (number of tokens k), both in raw form (top panel) and normalized by k (bottom panel). Normalization helps interpret the average influence per token, particularly in longer texts.

From the figure, it is evident that the `mask` baseline (blue line) results in relatively constant output values across input lengths. However, it is consistently outperformed by the `zero` baseline (yellow), which yields the lowest and most stable values of $F(x_0)$ across nearly the entire spectrum of token lengths. This indicates that the `zero` baseline best satisfies the completeness constraint by keeping the baseline output close to zero.

The `pad` (red) and `mean` (green) baselines exhibit similar behavior to each other: both show a peak in baseline output at lower k values and increasing saturation for longer inputs. The `mean` baseline, in particular, reaches the highest output values as k increases. While these two baselines perform comparably in shorter texts, they are both outperformed by the `zero` baseline across almost all input lengths, except for a narrow region in $k \in [10, 20]$, where all methods

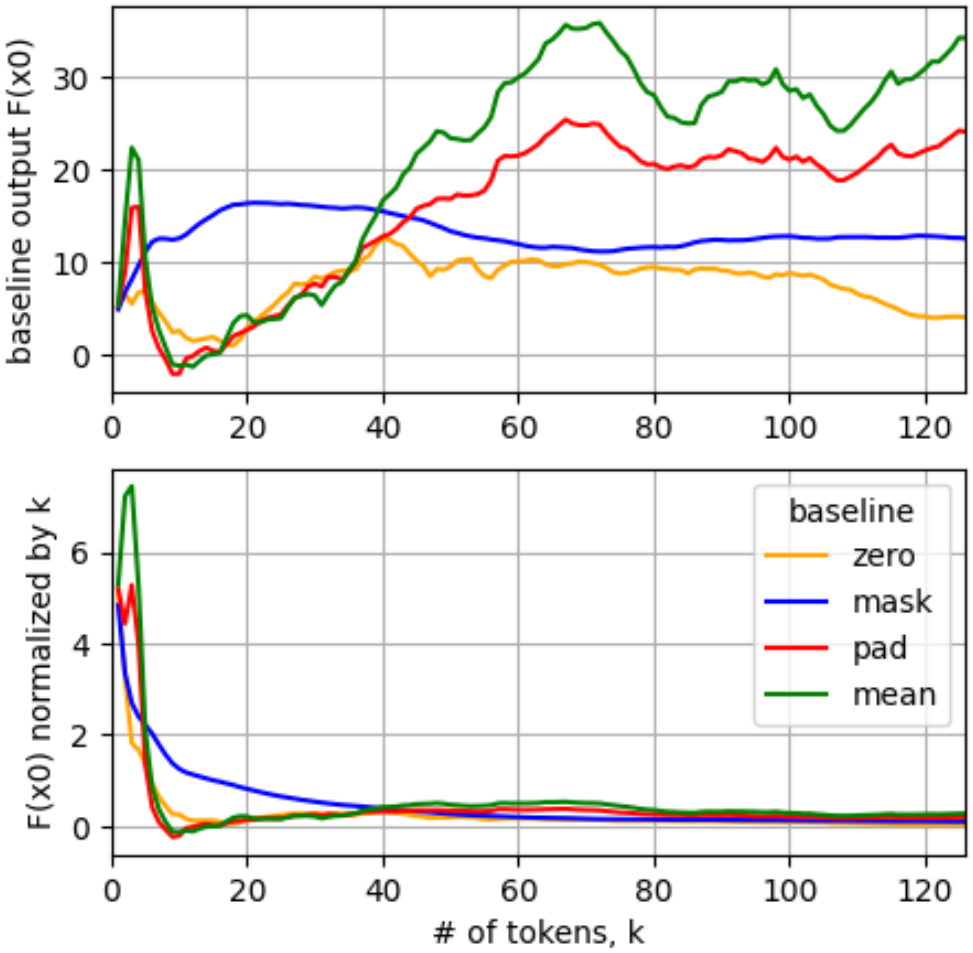

Figure 4.2: Baseline output $F(x_0)$ (top) and normalized output $F(x_0)/k$ (bottom) as a function of the number of tokens k in IG (with $N = 300$ steps), and for different baselines.

converge to similarly low $F(x_0)$ values.

Conclusion. Due to its consistently low and stable baseline outputs across varying input lengths, the `zero` baseline is identified as the most neutral and theoretically grounded choice. It is therefore selected as the default for all attribution experiments throughout this thesis.

4.2.3 Step Size Sensitivity

An important parameter in IG is the number of interpolation steps N, which determines how finely the input space is sampled along the path from the

baseline to the input. A larger N improves the approximation of the integral, but increases computational cost.

Figure 4.3 shows comprehensiveness and sufficiency on the Snippets dataset using the `zero` baseline, for three different values of $N \in \{50, 100, 300\}$. As seen in the figure, the curves are nearly identical across all values of N. This confirms that comprehensiveness and sufficiency are not sensitive to N, at least for $N \geq 50$, because these metrics are driven by the ranking of attribution scores, which tends to remain stable even if the magnitude of scores changes.

Although only the Snippets dataset is shown, this trend was observed consistently across all datasets and baselines, and thus does not warrant further visual comparison.

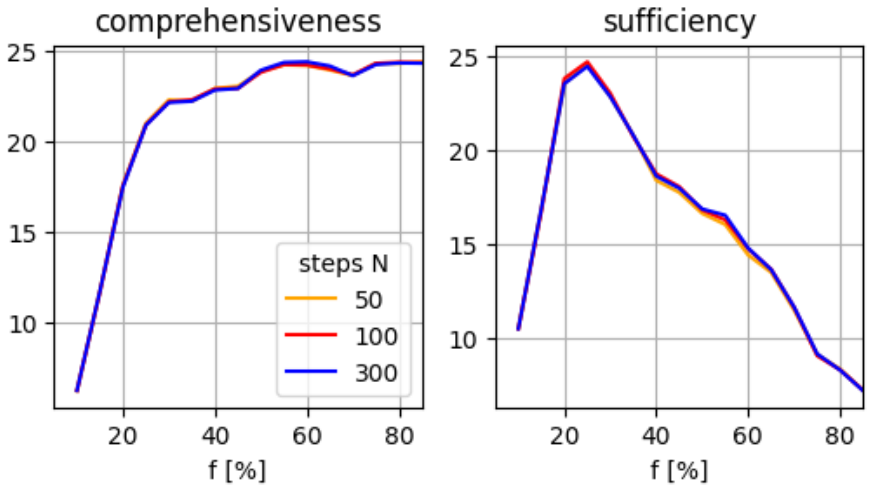

Figure 4.3: Comprehensiveness and sufficiency in IG for the Snippets dataset with `zero` baseline, comparing step sizes $N \in \{50, 100, 300\}$. Curves are almost perfectly overlapping.

4.2.4 Approximation Error Analysis

To more reliably evaluate the role of N, we turn to the approximation error, defined as the difference between the sum of attributions and the actual model output difference $F(x) - F(x_0)$. This metric reflects how well IG adheres to the completeness property (see 3.5.3).

Figure 4.4 shows the distribution of approximation error for different values of N and baseline types, across two settings: the combined datasets (top panel) and the Snippets dataset (bottom panel). Each subplot is a box plot, displaying the 25%, 50%, and 75% percentiles, along with outliers.

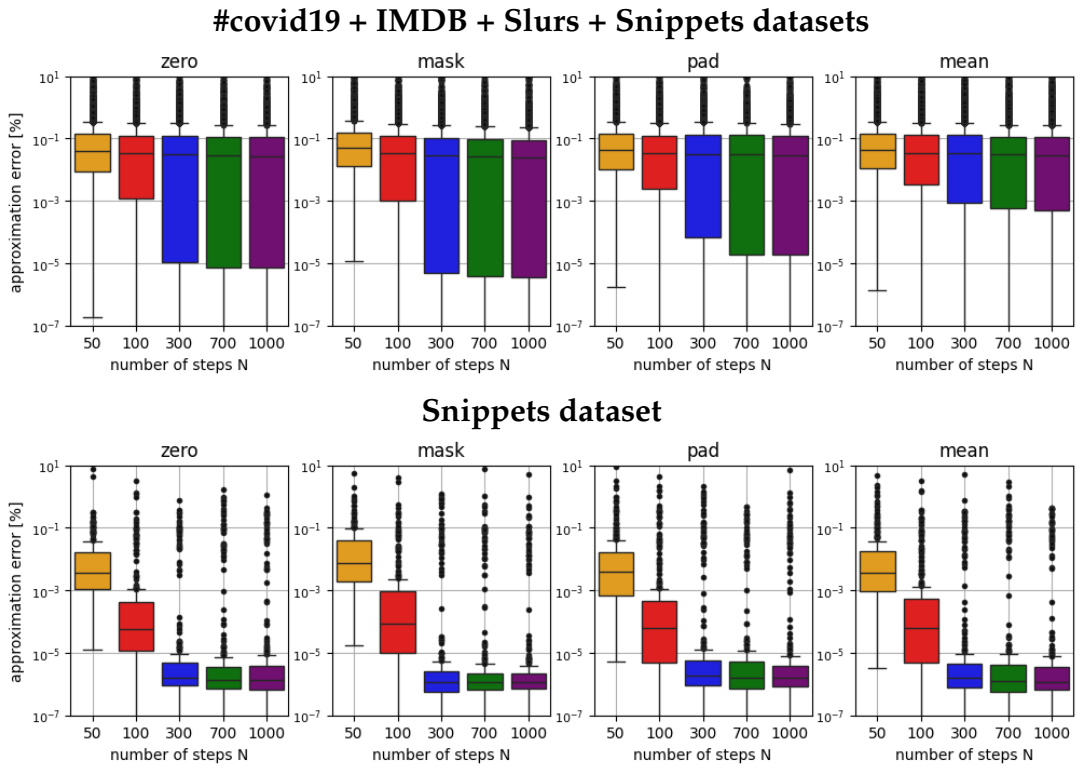

Figure 4.4: Approximation error statistics in IG for varying $N \in \{50, 100, 300, 700, 1000\}$ and baseline types. Top: All datasets combined. Bottom: Snippets dataset only.

From the top panel, we observe that the `zero` and `mask` baselines yield the lowest approximation errors. The `pad` baseline performs slightly worse, while the `mean` baseline exhibits the highest error rates across all step sizes.

In both panels, performance improves significantly when increasing N from 50 to 300, but tends to saturate beyond $N = 300$, particularly for `zero` and `mask`. This effect is especially visible in short texts, where outliers become more prominent at lower N values. These results validate $N = 300$ as a reliable and efficient trade-off between accuracy and computation.

Conclusion. Based on the evidence from Figures 4.1–4.4, the most robust configuration for IG in this thesis is:

- **Baseline:** `zero`,
- **Integration Steps:** $N = 300$,
- **Attribution Source:** input embeddings via HuggingFace `inputs_embeds`.

This setting achieves reliable attribution behavior, minimal baseline influence, and low approximation error, especially for short or mid-length inputs. Additional discussion of interpretability and applications follows in Chapter 5.

4.3 Comparison of Attribution Techniques

To evaluate the strengths and limitations of various attribution techniques in providing word-level explanations, we compare four methods: IG, SeqIG, GradientSHAP, and DeepLIFT. These techniques were assessed using the BERTAgent model on four diverse datasets: #covid19, IMDB, Slurs, and Snippets. Two standard metrics, comprehensiveness and sufficiency, are used to quantify attribution faithfulness (see Section 3.5). In both cases, we vary the top-f% of attributed tokens selected, from 10% to 85%.

4.3.1 Aggregate Performance Comparison

Figure 4.5 presents a side-by-side comparison of four attribution methods, IG, SeqIG, GradientSHAP, and DeepLIFT, on four datasets: #covid19, IMDB, Slurs, and Snippets. Comprehensiveness (top row) and sufficiency (bottom row) are reported for each dataset as a function of the fraction f of top-attributed tokens retained or removed.

The methods vary significantly in their ability to isolate meaningful input features:

- **Sequential Integrated Gradients (SeqIG)** shows the highest overall comprehensiveness and the lowest sufficiency scores across datasets, particularly on short-form inputs like the Snippets dataset. Its iterative integration over masked tokens enables more precise attribution, but this comes with a steep computational cost.

- **Integrated Gradients (IG)** performs nearly as well as SeqIG, especially on longer texts such as IMDB. It captures token influence via a single path integration and is significantly more efficient than SeqIG. It consistently shows strong performance in both metrics, demonstrating robustness across datasets.

- **GradientSHAP** provides moderate performance. As a sampling-based method, it tends to introduce variance and underperforms in comprehensiveness while showing average sufficiency values. Its lack of fine-grained attribution makes it less suited for token-level analysis in compact texts.

- **DeepLIFT** exhibits the weakest results, with the lowest comprehensiveness and highest sufficiency. Its behavior suggests a failure to consistently identify impactful input features, especially in short or linguistically sparse examples.

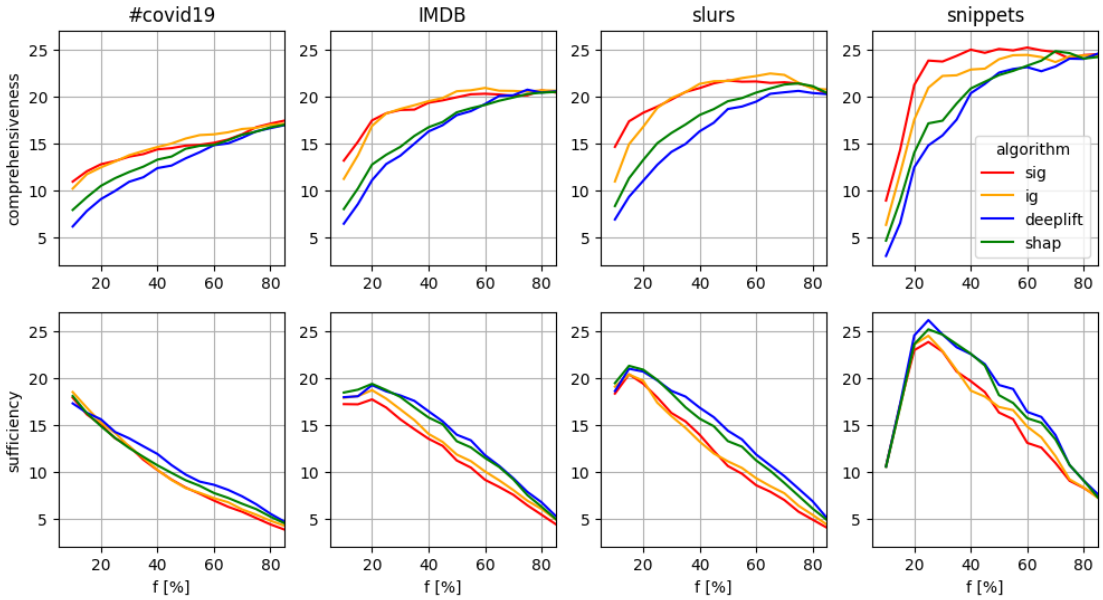

Figure 4.5: Comprehensiveness and sufficiency in different attribution methods as a function of token fraction f, for each dataset.

Interestingly, while differences across methods are clearly visible at lower values of f, they tend to converge as f increases. This is an expected consequence of token accumulation: when a larger portion of the input is retained or removed, the marginal impact of the top-ranked tokens is diluted. Thus, attribution differences become less pronounced at higher token fractions.

4.3.2 Token Length Sensitivity

To further investigate the effect of input length on attribution quality, we fix $f = 20\%$ and analyze comprehensiveness and sufficiency as a function of input length, averaged over the four datasets. The resulting curves are shown in Figure 4.6.

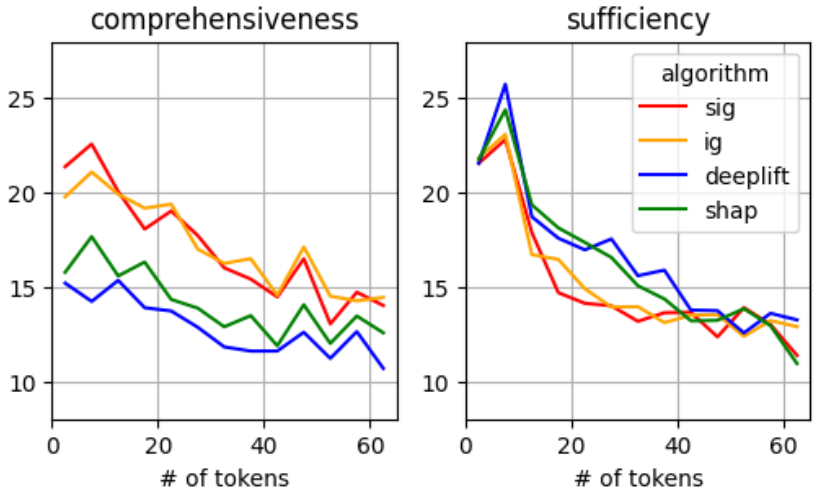

Figure 4.6: Comprehensiveness and sufficiency at $f = 20\%$ as a function of input length. All datasets are included.

Several trends emerge:

- All methods exhibit a noticeable decline in comprehensiveness as input length increases. This reflects the fact that in longer sequences, the influence of any single token becomes less dominant, making it harder for attribution methods to isolate impactful elements.
- A similar degradation is observed in sufficiency, but the drop-off rates vary by method. In particular, DeepLIFT and GradientSHAP deteriorate more rapidly than IG or SeqIG.
- **SeqIG and IG maintain the highest performance** across lengths, with SeqIG showing a slight advantage in very short inputs (less than 10 tokens). For longer sequences, their scores converge, further emphasizing the strong baseline reliability of IG.

These results underscore the importance of considering input length when choosing an attribution technique. While SeqIG offers a marginal gain for short, information-dense inputs, the computational trade-off becomes less justifiable for longer or batched sequences.

Implementation Details. All attribution methods were implemented using the Captum library. IG and SeqIG used a `zero` embedding baseline. IG was configured with $N = 300$ integration steps for all datasets, while SeqIG was run with $N = 300$ for the Snippets dataset and $N = 50$ elsewhere to reduce computational overhead. DeepLIFT and GradientSHAP were executed with their standard reference implementations.

To quantify computational cost, Table 4.1 reports the average time (in seconds) required to compute attributions per sentence for each method on the Snippets dataset using $N = 50$. As expected, DeepLIFT and GradientSHAP were the most efficient, each requiring under 0.05 seconds per sample. IG required nearly 10 times more, while SeqIG was by far the most computationally expensive, approximately 10 times slower than IG and over 100 times slower than DeepLIFT or GradientSHAP.

Table 4.1: Average runtime (in seconds per sample) for each attribution method on the Snippets dataset with $N = 50$.

Method	**Time (s)**
`DeepLIFT`	0.0456
`GradientSHAP`	0.0467
`IG`	0.4542
`SeqIG`	4.6552

Conclusion. In conclusion, SeqIG yields the most accurate and faithful attributions, particularly in shorter texts, but incurs a high computational cost. IG emerges as the most practical and effective method overall, offering a favorable balance between performance and efficiency. GradientSHAP provides acceptable but noisy results, while DeepLIFT appears less suitable for tasks that require fine-grained token-level interpretability. These findings validate the decision to adopt IG as the primary attribution method in this thesis and emphasize the importance of aligning attribution choice with both computational and linguistic requirements.

4.4 Rendering and Visualization of Word-Level Outputs

Once the optimal configuration of IG, with a `zero` baseline and $N = 300$ steps, was selected, several post-processing steps were necessary to make the attribution results human-readable and linguistically interpretable. This section describes how the raw IG outputs were rendered into word-level heatmaps, allowing effective analysis of the linguistic factors contributing to perceived

agency.

4.4.1 Subword Merging via SpaCy Alignment

The RoBERTa tokenizer operates at the subword level, which often splits single words into multiple fragments. For instance:

These people are lazy and un mot ivated

Here, the word "unmotivated" is split into three tokens (`un`, `mot`, `ivated`), each with its own attribution value. To produce clean and interpretable visualizations, we align RoBERTa subwords to the more human-readable tokenization provided by SpaCy. Attribution scores of subwords that map to the same SpaCy token are summed and rendered as one:

These people are lazy and unmotivated

This process eliminates attribution fragmentation and better matches human reading patterns.

This functionality is implemented in the method:

```
    def _spacy_map(self, text, token_ids, offsets, attributions):
        """Map RoBERTa tokens and attributions to SpaCy tokens."""
        roberta_tokens = self.tokenizer.convert_ids_to_tokens(
token_ids.tolist())
        filtered_tokens = [
            (token.replace('G', ''), attribution, offset)
            for token, attribution, offset in zip(roberta_tokens,
attributions, offsets)
            if token not in ['<pad>', '<s>', '</s>']
        ]
        filtered_tokens, filtered_attributions, filtered_offsets =
zip(*filtered_tokens)

        doc = self.nlp(text)
        spacy_tokens = [token.text for token in doc]
        spacy_offsets = [(token.idx, token.idx + len(token.text)) for
 token in doc]

        spacy_attributions = np.zeros(len(spacy_tokens))
        spacy_index, roberta_index = 0, 0
```

```
        while spacy_index < len(spacy_offsets) and roberta_index < len(filtered_offsets):
            spacy_start, spacy_end = spacy_offsets[spacy_index]
            roberta_start, roberta_end = filtered_offsets[roberta_index]
            if spacy_start <= roberta_start < spacy_end:
                spacy_attributions[spacy_index] += filtered_attributions[roberta_index]
                if roberta_end <= spacy_end:
                    roberta_index += 1
                else:
                    spacy_index += 1
            elif roberta_start < spacy_start:
                roberta_index += 1
            else:
                spacy_index += 1

        return {
            'tokens': spacy_tokens,
            'attributions': spacy_attributions,
            'f_tokens': np.array(filtered_tokens),
            'f_attributions': np.array(filtered_attributions),
        }
```

Code 4.1: Token alignment and aggregation with SpaCy

The function parses the original sentence with SpaCy, matches character offsets between RoBERTa and SpaCy tokens, and merges overlapping attributions into a single SpaCy token score.

4.4.2 Negation-Aware Attribution Aggregation

Negations play a critical role in shaping agency. For example, in the phrase "not lazy", the word "not" typically receives a strong attribution because it inverts the meaning of "lazy". However, in human interpretation, "not lazy" should be understood as a positive agency indicator, not split into isolated attributions.

Thus, we leverage SpaCy's dependency parser to detect negations (`neg`) and combine their attribution with the word being negated (the syntactic head). Additionally, if the head is an auxiliary verb (`auxpass`) or has an adjectival complement (`acomp`), these tokens are also included in the attribution span. For

instance, the following sentence initially produces:

These people are not lazy at all !

To faithfully capture meaning, the attribution from "not" is redistributed to the full phrase "not lazy", which carries the agentic implication:

These people are not lazy at all !

This significantly improves interpretability and ensures the correct meaning is visually emphasized.

This is implemented in the method:

```
    def _group_negations(self, text, attributions, show=False):
        """Group negations to adjust attributions."""
        doc = self.nlp(text)
        for token in doc:
            if token.dep_ == "neg":
                related_tokens = [token.i, token.head.i]
                for child in token.head.children:
                    if child.dep_ in ["acomp", "prt"]:
                        related_tokens.append(child.i)
                if token.head.dep_ == 'auxpass':
                    related_tokens.append(token.head.head.i)
                if show:
                    print(doc[related_tokens])
                attributions[related_tokens] = attributions[
related_tokens].sum()
        return attributions
```

Code 4.2: Negation-aware attribution recomposition

The method uses SpaCy's dependency parser to detect negation scopes and recomposes them into single attribution units for visualization.

4.4.3 Filtering Attribution Sign Mismatch

In some cases, certain words receive attribution values that conflict with the overall agency score of the sentence. For instance, in a sentence that globally expresses positive agency:

This person is far from being lazy !

Here, the words "person" and "lazy" receive negative attributions, which may confuse interpretation. To prevent this, we zero out attributions that are inconsistent with the overall sign of the agency score:

This person is far from being lazy !

This step improves clarity and prevents over-interpretation of contextual noise.

4.4.4 Normalization to Agency Score

To allow visual comparisons across examples, the final attributions are scaled so that the maximum value matches the sentence-level agency score $F(x)$. This ensures that:

- Sentences with low agency appear with lighter highlights.
- Sentences with high agency show more intense colors.

For example, a sentence with high agency may be rendered as:

Join our next #fridays4future strike !

Compared to earlier examples, this sentence uses lighter shades of green, reflecting its lower agency score even though attribution is present.

Implementation. Both the filtering of polarity-inconsistent attributions and the normalization to the sentence-level agency score are handled within a single function in the `AgencyIG` class:

```
def _map_ag(self, attributions, agency, text, show=False):
    """Map and normalize attributions with agency scores."""
    # Adjust attribution sum to match agency score
    adjusted_attributions = attributions + (agency - attributions.sum
    ()) / len(attributions)

    # Recompose negations before final scoring
    adjusted_attributions = self._group_negations(text,
    adjusted_attributions, show=show)

    # Filter by polarity and normalize
    if agency > 0:
```

```
        adjusted_attributions = np.clip(adjusted_attributions, 0, 1)
        return adjusted_attributions / adjusted_attributions.max() *
    agency
    else:
        adjusted_attributions = np.clip(adjusted_attributions, -1, 0)
        return adjusted_attributions / adjusted_attributions.min() *
    agency
```

Code 4.3: Filtering and normalizing attributions with agency

This function performs three critical post-processing steps:

- **Polarity filtering:** Clipping attribution signs to retain only values consistent with the agency score.
- **Normalization:** Scaling attribution values so that the most influential token reflects the full magnitude of the predicted agency.
- **Residual correction:** Adjusting attribution totals to exactly match $F(x)$, satisfying the completeness constraint.

Together, these mechanisms enhance the fidelity and interpretability of the IG output, ensuring that both quantitative accuracy and linguistic coherence are preserved in the final visualizations.

4.4.5 Rendering for HTML and LaTeX

The final output is rendered for either HTML or LaTeX formats. The core rendering function is:

```
    def render_ig(self, sentence_index, latex=False):
        """Render integrated gradients for a specific sentence in
    HTML or LaTeX."""
        row = self.df.iloc[sentence_index]
        text, agency, tokens, attributions = (
            row['text'], row['agency'], row['token'], row['
    attribution']
        )
        print(f"index: {sentence_index}\n")
        if latex:
            display(self._display_tex(tokens, self._map_ag(
    attributions, agency, text)))
        else:
            display(self._display_html(tokens, self._map_ag(
    attributions, agency, text)))
```

Code 4.4: Rendering IG output

And the LaTeX string formatting is handled by:

```
    def _texstr(self, string, color='white'):
        """Render a string with a LaTeX colorbox."""
        tex_str = f"\\colorbox[HTML]({color[1:]})({string})"
        return tex_str.replace('(', '{').replace(')', '}')
```

Code 4.5: LaTeX token color rendering

Summary

Through token merging, negation-aware recomposition, polarity filtering, and agency-based normalization, this rendering pipeline transforms raw IG output into coherent visual explanations.

4.5 Evaluation Through Human-Annotated Highlights

To further validate the plausibility and explanatory power of the IG-based attribution method, we conducted an evaluation using a dedicated dataset of human-annotated highlights. This dataset, referred to as the *highlights dataset*, contains 802 texts written by English native speakers in response to a specific prompt designed to motivate action. As such, the texts naturally exhibit high levels of agency, making them an ideal testbed for assessing alignment between model attributions and human-perceived agency cues.

For each text, the original author and two or three expert annotators were asked to highlight the portion(s) of the sentence they believed carried the strongest "call-to-action", the part most relevant for motivating readers. We hypothesize that these highlights should correspond to the most agentic words, as measured by Integrated Gradients.

4.5.1 Experimental Setup

To make the evaluation tractable and focused, each text was split into individual sentences, and only those that were at least partially highlighted were retained. For each sentence, we computed the following measures:

- The overall agency level a, as predicted by the BERTAgent model.

- The highlight fraction f_h, defined as the proportion of tokens highlighted by a human ($f_h \in [0, 1]$).
- The highlight agency a_h, defined as the sum of attribution scores over only the human-highlighted tokens. This was computed using the IG method, with attribution vectors post-processed via the rendering pipeline described in Section 4.4, including polarity alignment and normalization to the agency score a.
- The maximum agency $a_{\max}$, defined as the sum of the top-k IG scores over any contiguous token span of length equal to the highlight span. This simulates a "best-case" highlight in terms of agency signal.
- A baseline noise estimate, computed by averaging IG scores over randomly selected token subsets of the same length as the highlights, repeated across multiple seeds.

By construction, $a_h \leq a_{\max} \leq a$ for positively agentic texts, and the reverse holds for negatively agentic ones.

4.5.2 Results and Analysis

Figure 4.7 presents six panels summarizing the relationship between human highlights and IG-based agency signals across various highlight lengths.

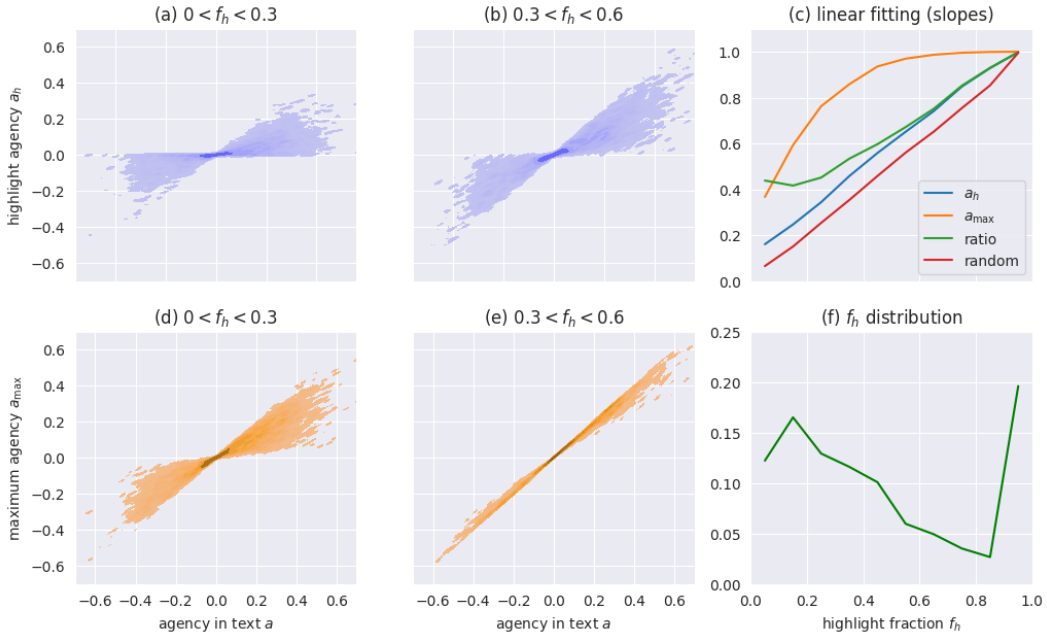

Figure 4.7: Highlights evaluation: (a, b) highlight agency a_h vs. sentence agency a; (d, e) $a_{\max}$ vs. a; (c) linear fitting slopes across f_h; (f) empirical distribution of f_h.

Highlight alignment improves with size. Panels (a) and (b) show a_h as a function of a, stratified by highlight length $f_h \in (0, 0.3]$ and $f_h \in (0.3, 0.6]$ respectively. While short highlights (a) display moderate alignment, longer highlights (b) yield tighter correlations, suggesting that human annotations better capture agentic content when annotators are permitted to select larger spans.

Max-agency highlights track model output closely. Panels (d) and (e) plot $a_{\max}$ against a in the same f_h bins. These show much stronger linear correlation than their a_h counterparts, especially in the lower range, demonstrating that the model's most highly attributed tokens align well with its own global prediction.

Slope analysis and interpretability. Panel (c) reports slope values from linear regressions of a_h and $a_{\max}$ against a, evaluated across a sequence of highlight bins $f_h \in (\frac{k}{10}, \frac{k+1}{10}]$. The curve labeled "ratio" shows the ratio $a_h/a_{\max}$ as a proxy for how representative the human-highlighted words are. Impressively, even for the shortest highlights ($f_h < 0.2$), this ratio exceeds 0.4, implying that humans consistently identify some of the most agentic words—even in minimal selections. A "random" baseline (in red) is also plotted, demonstrating that human highlights outperform chance across all ranges.

Highlight fraction distribution. Panel (f) reveals that most human highlights are short ($f_h < 0.3$), reinforcing that even when annotators are selective, they tend to capture highly meaningful agency-bearing words.

4.5.3 Illustrative Example

It is important to note that, especially at low values of f_h, human highlights tend to capture not only the most agentic words, but also the most semantically meaningful and cohesive parts of the sentence. As a result, highlights may not always coincide exactly with the words carrying the highest individual attributions.

This is not a flaw of the IG method, but rather reflects the natural tendency of annotators to prioritize phrases that are both impactful and interpretable. Consequently, we should not expect a_h to reach the full value of $a_{\max}$, especially when only a few tokens are selected.

A representative example is shown below. The sentence is rendered with IG attributions (as background color) and human highlights (underlined):

If we all work together ,
we can make a difference !

In this case, the expert annotator highlights the latter clause—"we can make a difference!"—which they perceive as the most agentic and motivational portion of the sentence. However, some high-attribution tokens (such as "work together") are left unmarked, because the reader focused on the overall intent and rhetorical structure rather than individual token strength.

This kind of selection behavior helps explain why a_h remains consistently below $a_{\max}$ even for highly agentic texts, while still capturing a substantial proportion of agency, often over 40% of $a_{\max}$, as shown in Figure 4.7(c).

Far from being a limitation, this reveals the human strategy in persuasive communication: to highlight semantically cohesive, action-oriented expressions that balance agency with clarity. The fact that IG attributions track closely with these highlights supports both the interpretability and sociopsychological validity of the method.

4.5.4 Implications for Sociopsychological Research

These findings are consistent with theoretical expectations. Prior work in collective action psychology highlights the role of agency (or efficacy) in motivating action. Specifically, the perception of personal or collective efficacy has been shown to significantly predict intention to act [34]. In that study, agency accounted for 11.56% of the variance in collective action intentions ($r = 0.34$). More recent work by Formanowicz et al. [35] extends this notion, demonstrating that messages with agentic language are more effective at mobilization, particularly in digital and collective contexts.

The observed alignment between human highlights and IG scores thus provides not only computational validation of our method but also evidence for its sociopsychological utility. Language users appear to recognize and highlight agentic segments of persuasive messages, and our model's attributions track these cues effectively. This supports both the cognitive plausibility of IG and its

practical value in analyzing persuasive or mobilizing discourse.

The ability to detect and quantify agentic signals at the token level, grounded in both model reasoning and human interpretation, positions this method as a promising tool for broader applications in behavioral analysis, activism framing, and engagement modeling.

4.6 IG in Small Dataset Contexts: Overfitting and Label Extraction

As established in earlier sections, IG effectively highlights the most influential words for model predictions. While our previous applications targeted large-scale datasets, many real-world problems, especially in sociopsychological research, operate in small-data regimes. In these contexts, IG can be used not only for interpretation but for label exploration, lexicon seeding, and schema refinement.

4.6.1 Overfitting for Feature Extraction

In contexts where labeled data is limited, we adopt a deliberate overfitting approach for extracting class-defining features at the word level. Rather than aiming for generalization, the objective is to force a powerful model, namely, RoBERTa, to memorize the training set. This allows us to leverage IG to identify the most influential tokens associated with each class label.

Methodological Motivation. As described in the original formulation [11], the goal is not to build a general-purpose classifier, but to create an interpretable model that reproduces known class associations with perfect fidelity. Formally, the process is as follows:

- We train a text classification model on a small dataset, using all available examples as the training set (i.e., no train/test split is applied).
- The model is trained until it reaches 100% accuracy on the training set, deliberately overfitting to the data.
- Once the model is perfectly memorized, IG is applied to extract per-token attributions for each class.

This enables us to expose the specific lexical choices the model relies on to distinguish between classes. The approach is especially useful when datasets contain nuanced sociopsychological markers (e.g., shame, fear, objectification), where traditional rule-based methods may fail to uncover class-specific patterns.

Training Setup. We use the `RobertaForSequenceClassification` model from HuggingFace Transformers, implemented in PyTorch. The model is initialized from a pretrained `roberta-base` checkpoint and fine-tuned on the labeled dataset using cross-entropy loss.

The training is performed as follows:

- **Tokenizer:** RoBERTa tokenizer with padding and truncation to 128 tokens.
- **Input:** Raw text is tokenized and passed as input IDs and attention masks.
- **Optimizer:** AdamW optimizer with a learning rate of $1e{-}5$ and linear learning rate decay.
- **Loss:** Categorical cross-entropy loss over the class logits.
- **Device:** Training is done on a single NVIDIA GPU (e.g., RTX A6000).
- **Epochs:** Training continues for approximately 20 epochs, or until training accuracy reaches 100%.
- **Batch size:** Typically set to 16 or 32, depending on GPU memory constraints.

The HuggingFace PyTorch code structure is standard:

```
from transformers import RobertaTokenizer,
    RobertaForSequenceClassification
from transformers import Trainer, TrainingArguments
from datasets import Dataset

tokenizer = RobertaTokenizer.from_pretrained('roberta-base')
model = RobertaForSequenceClassification.from_pretrained('roberta-
    base', num_labels=num_classes)

def preprocess(example):
    return tokenizer(example['text'], truncation=True, padding='
    max_length')

dataset = Dataset.from_dict(data).map(preprocess)

training_args = TrainingArguments(
```

```
    output_dir='./results',
    per_device_train_batch_size=16,
    num_train_epochs=20,
    learning_rate=1e-5,
    logging_steps=10,
    evaluation_strategy="no",
    save_strategy="no",
)

trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=dataset,
)

trainer.train()
```

Code 4.6: Fine-tuning RoBERTa for overfitting-based attribution extraction

Why Overfit? Unlike conventional ML practice, where overfitting is seen as a flaw, here it is a feature. By achieving perfect training accuracy, we ensure that the model's internal representation has fully absorbed the discriminative patterns in the data. IG then enables us to back out those exact patterns in the form of attribution maps.

This strategy is particularly useful when the goal is **explainability rather than generalization**, and where data collection at scale is not feasible due to annotation costs or domain constraints. As will be shown in Section 4.6.5, this approach allows us to extract candidate keywords for dictionary construction or concept modeling.

4.6.2 Case Study: The Slurs Dataset

We apply this method to the *slurs* dataset, a corpus of 336 first-person female narratives of verbal harassment. Each story is labeled by experts with the following annotations:

- **Aggressor gender:** man, woman, both, unspecified;
- **Relationship:** partner/family, friend, acquaintance, unknown;
- **Emotions:** shame, fear (rated 1–7);

- **Objectification:** perceived degree of sexual objectification.

The resulting class-specific IG scores are shown in Table 4.2, with darker shades indicating stronger attribution. Note: unlabeled or ambiguous cases were included in training but excluded from the IG summary.

4.6.3 Label-Specific Insights

Aggressor Gender. In the "man" class, IG surfaces gendered terms such as *male*, *boy*, *brother*, *boyfriend*, *man*, and *husband*. These validate that the classifier internalized gender semantics. In contrast, for the "woman" class, terms like *mother*, *she*, and *girlfriend* dominate, often accompanied by verbs like *obsess* or *charge*, reflecting linguistic differentiation between aggressor and victim. The "both" and "unspecified" classes tend to reflect neutral or context-ambiguous tokens (e.g., *couple*, *person*, *voice*).

Relationship Type. Partner-related terms (*partner*, *boyfriend*, *husband*) dominate the family class, while *employer*, *advisor*, and *supervisor* are associated with acquaintances. Unknown or distant aggressors yield generic or socially distant labels like *salesperson*, *customer*, and *stranger*.

Fear and Shame. High levels of fear correlate with terms like *afraid*, *abuse*, *violence*, and *unprovoked*. Shame correlates with *mortify*, *insecurity*, and *degrade*. Notably, these terms only appear in the high-class bins, showing that IG reflects emotional gradience.

Sexual Objectification. In high-objectification examples, IG highlights sexually explicit or body-focused terms like *cock*, *slut*, *blowjob*, and *skirt*. These align well with theoretical definitions of objectification as the reduction of a person to their sexual utility [36, 37].

4.6.4 Interpretability and Practical Utility

This method yields interpretable, psychologically grounded explanations for model behavior, even in underpowered classification settings. Unlike pre-compiled dictionaries, IG captures contextual and emergent linguistic markers

Table 4.2: Keywords identified by IG for various sociopsychological labels in the *slurs* dataset. Darker colors indicate stronger attribution values.

class	#docs	keywords
GENDER (of the aggressor) - classifier 4 classes		
both	5	couple husband walk dog people block aggressive woman front ask voice person think partner hate house come yell friend stand
man	189	male boy brother boyfriend man lad guy gentleman husband dad notice target accord annoying misogynistic boss sudden supervisor entirely kid
woman	92	obsess secretary waitress sister girl girlfriend nurse mother woman charge supermarket staff granddaughter bedroom restaurant owner intervene treat same
NA	50	
RELATIONSHIP (between the victim and the aggressor) - classifier 5 classes		
acquaintance	62	employer boss mate advisor thesis supervisor roommate master neighbour neighbor colleague university classmate single use client catechism exercise factory guy
friend	26	friend friendship boyfriend classmate university college good classroom mutual back believe impulsively m talk take call still due tell year
partner/family	62	daughter engage partner date boyfriend husband dad sister ex brother girlfriend mother obsess guy spend then conversation pregnant mum
unknown person	166	forum salesperson checkout store customer counter gentleman motorist aged overtook pizza winning service catch vaxxer exchange colleague citizen bar nearby
NA	20	
SHAME (level of shame evaluated by authors) - classifier 7 classes		
low (1-2)	124	confrontational incense overtook unprovoked excuse chap offer prostitute scooter chauvinist mutual mph junk derisive ban butcher annoying polite impatient honk
medium (3-5)	112	cleavage pushy desk oppose sketchy embarrasment embarrsse hurtful reach reschedule division effort bitchy pizza report particularly tipsy offensive tonight wickedness
high (6-7)	100	assign heartbroken infatuation mortify speechless normal insecurity inattentive dad university behaviour girlfriend compulsive syllabus humiliate heavily breathe terrible salesperson fear
FEAR (level of fear evaluated by authors) - classifier 7 classes		
low (1-2)	108	topic report rude assign employer join cheat amusing inattentive blowjob oratory division item offense sell team skirt awkward jest random
medium (3-5)	138	heartbroken temper unqualified dickhead spend bully exactly loose bias overtook reckless threaten witness misogynistic useless nightclub insecure entitle outfit space
high (6-7)	90	bouncer violence violent unprovoked scare scary afraid pizza chase nervousness abusive avoid ground youngish aggressive horn terrify demeanour memorable nightclub
OBJECTIFICATION (level of objectification evaluated by experts) - classifier 7 classes		
low (1-2)	124	foolish grumpy incompetent excuse attribute slacker intelligent ungrateful error log handicapped slur retard misunderstand housekeep cockroach suitcase disgusting omission manoeuvre
medium (3-5)	92	buggy confrontational muzzle roundabout reprimand telepass cocksucke whore fight slut savage spineless huff foul agitate ex threaten sexism wound idiotic
high (6-7)	120	blowjob cock cunt infatuation pushy slut skirt moron whore cheat wet slag humiliation nightclub sweatshirt pizza real heart dick jealous
SELF OBJECTIFICATION (level of self objectification evaluated by experts) - classifier 9 classes		
low (-6,-2)	96	handicapped arm mood offer attribute overtook infant health drunk walker mental sheep staff intervene sibling landlord criticize yesterday bar refuse
medium (-2,2)	147	cook dick elevator shit thick pizza compulsive macho forum employer tender system unprovoked ride challenge deny seat loose wheelchair key
high (2,6)	93	blowjob cock small hire pregnant tipsy salesperson single bed client hurl cleavage cockroach strength overtake superior dog sleep shoe skirt

specific to the dataset. This makes it especially valuable in early-stage annotation workflows or when working with evolving sociolinguistic phenomena (e.g., online harassment slang).

In the next section, we explore how this approach generalizes across datasets to support dictionary expansion and lexicon development.

4.6.5 Salient Keywords and Dictionary Expansion

A central contribution of this thesis is the demonstration that explainability tools such as IG, when applied in conjunction with deliberate model overfitting, can serve as a highly effective mechanism for discovering semantically relevant and discriminative keywords. This has direct applications for dictionary construction and concept modeling in domains like sentiment, emotion, and sociopsychological trait analysis.

From Attributions to Keywords. Once the overfit **RobertaForSequenceClassification** model has memorized the labeled dataset (as discussed in Section 4.6.1), IG is applied to compute word-level attribution scores for each input example. Because the model has been trained to full accuracy, the attributions can be interpreted as an **explanation of what features the model relied on to distinguish between classes**.

For each class or target label, we collect all input texts and compute their corresponding IG attributions. Then, for every token in those texts, we aggregate the attribution values (e.g., via sum or mean) and rank the tokens by their overall importance to the classification decision.

This process yields a ranked list of **salient tokens per class**, effectively functioning as a data-driven dictionary that reflects the discriminative language used in the dataset.

Visualization of Class-Specific Lexicons. Table 4.2 (see previous section) illustrates the outcome of this pipeline. It reports the top-scoring tokens extracted for each label in the *slurs* dataset, spanning several sociopsychological dimensions including:

- **Gender of aggressor:** tokens like `boyfriend`, `husband`, `girl`, `nurse`, etc.
- **Relationship to aggressor:** e.g., `employer`, `friend`, `partner`, `stranger`.

- **Shame, fear, objectification:** affective terms like `mortify`, `fear`, `cock`, `slut`, etc., stratified by severity level.
- **Self-objectification:** tokens related to body image, sexuality, and internalization (e.g., `pregnant`, `skirt`, `cockroach`, `inferior`).

Importantly, the extracted tokens are not only consistent with psychological theory (e.g., objectification theory [36]) but also highlight novel candidates that may not appear in hand-curated dictionaries. For instance, terms like `pizza`, `nightclub`, or `skirt` acquire sociocultural connotations in context, which makes them relevant to nuanced labels such as self-objectification.

Interpretability and Transparency. Compared to classical methods such as Term Frequency–Inverse Document Frequency (TF-IDF) or Local Interpretable Model-agnostic Explanations (LIME), the IG-based pipeline used here offers two major advantages:

1. It is directly tied to the model's prediction function, ensuring faithfulness to the underlying classification logic.
2. It works at the token level and captures context-sensitive attributions, enabling fine-grained interpretability.

These properties make it particularly well-suited for generating **interpretable, evidence-based keyword lists**, which can subsequently be used for:

- Constructing domain-specific lexicons.
- Enriching annotation guidelines.
- Seeding features for downstream tasks (e.g., psychological profiling, content moderation, thematic analysis).

Limitations and Considerations. While the approach is powerful, several caveats must be kept in mind:

- The output tokens are strongly influenced by the tokenizer (e.g., RoBERTa's BPE), and alignment to natural words requires careful preprocessing (as described in Section 4.4).
- Attribution scores are sensitive to baseline choice and IG parameters such as the number of steps N.
- Overfitting on small datasets may capture dataset-specific quirks; interpretability is achieved at the cost of generalization.

Nonetheless, when used properly, this approach offers a scalable alternative to manual lexicon construction, blending the rigor of neural modeling with the clarity of token-level interpretability.

5 Discussion

5.1 Interpretation of IG Outputs

The previous chapters introduced and evaluated the use of IG as an explainability tool for token-level attribution in text classification and regression models, particularly in the context of sociopsychological markers such as agency, shame, fear, and objectification. In this section, we reflect on the interpretation of the resulting IG scores, how they behave, what they reveal, and how they should be read both technically and conceptually.

Faithfulness and Completeness. IG enjoys the formal property of *completeness*, meaning the sum of attributions across all input tokens (when using a suitable baseline) equals the model's output minus the baseline output. This property anchors the attributions in a mathematically sound framework, but as shown in Chapter 4, its practical realization depends heavily on the choice of baseline and number of steps. In cases where the baseline output $F(x_0)$ is near zero and N is sufficiently large (e.g., $N = 300$), IG faithfully reflects the distribution of influence across input tokens.

Positive and Negative Contributions. In our regression setting (e.g., agency prediction), the attribution of each token can be positive or negative depending on whether the token increases or decreases the predicted score. This bidirectional interpretation is essential, especially in domains where valence matters.

For example, in a sentence like:

"These people are not lazy at all."

The token `not` contributes positively to agency, counteracting the negative implication of `lazy`. Our negation-aware IG implementation (see Section 4.4) merges the contributions of syntactically related tokens to prevent misinterpretation from isolated values.

Sufficiency vs. Comprehensiveness. We evaluated IG explanations using two complementary metrics: *sufficiency* (how well top-attributed tokens alone can reproduce the model's output) and *comprehensiveness* (how much removing top tokens degrades the output). These helped validate whether IG identifies the most impactful words. As observed in Figure 4.1, sufficiency and comprehensiveness scores correlated strongly with attribution strength for well-chosen baselines (e.g., zero or mean). However, their utility diminished for hyperparameter tuning (e.g., step count), where approximation error proved more informative.

Limitations of Interpretability. While IG provides a theoretically grounded explanation method, it still faces challenges in practice. Attributions are sensitive to:

- **Baseline embeddings:** Different baselines yield different attribution distributions, especially in models like RoBERTa with no explicit padding token semantics.
- **Tokenization artifacts:** Byte-pair encodings can split meaningful words into subwords, requiring careful alignment (solved here via SpaCy integration).
- **Contextual dependence:** IG attributes importance based on the gradient of output w.r.t. input embeddings, but this gradient is influenced by context. Therefore, identical tokens may receive different attributions depending on sentence structure and surrounding words.

Interpreting Agency in Practice. In the specific case of agency prediction, IG reveals not just the linguistic cues but also the psychological framing encoded in the model. For instance, verbs like `make`, `organize`, or `refuse` receive strong

positive attributions, while adjectives like `weak` or verbs like `hesitate` tend to receive negative ones. This aligns with theoretical expectations from sociopsychological literature on agency and self-efficacy.

Taken together, these findings suggest that IG provides a robust framework for token-level interpretability, provided that the underlying model is stable, well-calibrated, and paired with suitable visualization and preprocessing steps.

5.2 Visualizations and Case Examples

To further demonstrate the interpretability benefits of IG, we present a set of real sentence-level examples with word-level attributions rendered via the methodology described in Section 4.4. These examples illustrate how IG captures the contribution of individual words toward the agency prediction, and how structural elements like negation, intensity, or syntactic construction influence attribution.

High Agency Statement. In the sentence below, the subject makes a confident, affirming declaration:

I 'm absolutely motivated .

This sentence received a high agency score ($a = 0.91$), and the strongest attribution is given to the verb "motivated" and its intensifier "absolutely." The visual rendering shows a dark green highlight over "motivated," reflecting its dominant contribution, while the rest of the sentence supports the positive framing with lighter tones.

Low Agency Statement. Contrast this with a semantically opposite expression:

Lazy and unmotivated

This statement, with an agency score of $a = -0.88$, showcases strong negative attributions over both "lazy" and "unmotivated." These tokens are colored in intense magenta/pink tones, visually marking them as dominant drivers of the model's low-agency prediction.

Negation and Inversion. In the example below, negation plays a pivotal role:

I 'm not motivated .

Although the lexical content includes "motivated," the presence of "not" completely inverts the attribution. Both "not" and "motivated" receive equal, strongly negative attribution, consistent with the negation-aware aggregation described in Section 4.4. The sentence's agency score of $a = -0.88$ aligns with this interpretation.

Ambiguity and Wordplay. In this case, attribution captures subtle semantic inversion through phrasing:

He is a hardly working individual

Here, the phrase "hardly working" is a play on "hard-working," leading to an agency score of $a = -0.75$. The attribution system assigns strong negative importance to "hardly," successfully identifying the semantic flip even though the sentence superficially appears positive. This demonstrates IG's sensitivity to negation, adverbial modifiers, and subtle context shifts.

Structural Negation. Finally, we observe the effect of negation within a compound predicate:

Not coordinated activity

The attribution strongly focuses on the "Not coordinated" phrase, consistent with the sentence's meaning of disorganization and lack of agency. The agency score is $a = -0.68$. Even though "activity" is neutral in itself, its lack of coordination flips the overall prediction to a negative domain.

Interpretability in Action. These examples highlight the value of IG not only for model auditing but also for linguistic interpretation. The visualization makes the model's internal logic legible: we see which words drive a classification decision, how semantic polarity propagates through context, and how intensifiers or negations shape model understanding. This token-level interpretability is critical when applying NLP to complex human constructs like agency, where surface form may belie deeper structure.

5.3 Sociopsychological and Practical Relevance

The ability to detect and quantify sociopsychological markers from language, particularly agency, has profound implications across disciplines including psychology, communication, digital humanities, and AI ethics. This section discusses the broader significance of the work presented in this thesis, linking technical outcomes to psychological theory and real-world utility.

Agency as a Core Construct. Agency, in psychological literature, refers to an individual's perceived ability to act intentionally and influence their environment. It plays a pivotal role in theories of motivation, self-efficacy, and social cognition [38, 34]. Linguistically, agency manifests through verbs, modality, and syntax that signal control, intentionality, or capacity to act. The IG-based methodology developed here allows us to decompose that construct at the token level, identifying which words contribute to the perception of agency and how they are modulated by context (e.g., negation, intensification, or narrative framing).

Our findings confirm that high-agency language tends to involve assertive verbs (e.g., *lead*, *organize*, *refuse*) and agentive constructions, while low-agency language correlates with passivity, helplessness, or negated structures (e.g., *not motivated*, *uncoordinated*, *give up*). These patterns are consistent with prior work in social psychology and lend empirical support to the linguistic signature of agency as captured by neural models.

Validation Through Human Highlights. As shown in Section 4.5, IG attributions were evaluated against human-annotated highlights in texts designed to mobilize or persuade. The correspondence between human intuition and model explanations suggests that agency is not only a theoretical construct but also a pragmatic feature that individuals reliably recognize and emphasize. This validates the IG approach not only as technically sound but also psychologically meaningful.

Applications in Communication and Mobilization. Understanding agency in language has direct implications for persuasive communication, political speech, and social campaigns. For instance, the ability to detect low-agency phrasing in

a tweet or speech may help identify rhetorical strategies that induce passivity, while high-agency constructions can reveal calls to action and empowerment.

Recent findings [35] show that agentic language boosts collective action and engagement, especially in contexts of social change or advocacy. The methods presented here offer a systematic way to analyze such discourse, both retrospectively (to study impact) and prospectively (to inform content design).

Practical Utility and Automation. From a technological standpoint, explainable models for agency detection can be integrated into tools for:

- **Mental health and well-being:** Detecting low-agency language in user inputs for early signs of depression or burnout.
- **Education and coaching:** Providing feedback on self-reflective writing (e.g., journals, personal statements) to encourage empowered expression.
- **Media and journalism:** Analyzing tone in editorials, opinion pieces, or coverage to assess narrative framing.
- **Content moderation:** Identifying demeaning or disempowering language, especially in gendered or aggressive discourse.

These use cases emphasize the value of interpretable attribution methods: rather than treating agency as a black-box classification score, we provide actionable insights into what exactly makes a sentence empowering or passive.

Ethical Considerations. While promising, this approach must be used responsibly. The same tools that detect agency could be co-opted to engineer manipulative messages. Additionally, agency is a culturally and socially contextual concept, what constitutes empowered language in one setting may be inappropriate or misunderstood in another. Interpretability is thus not only a technical virtue but also a safeguard, allowing practitioners to interrogate and calibrate the models they deploy.

6 Conclusion and Future Work

6.1 Summary of Key Findings

This thesis proposed a detailed and interpretable approach for analyzing agency and related sociopsychological markers in text using IG. By combining explainable deep learning techniques with psychological insight and careful preprocessing, we addressed both the technical challenge of attribution and the conceptual challenge of interpretability in sociolinguistic contexts.

The work began with a review of related literature, establishing the importance of word-level attribution for psychological constructs such as agency. Building upon a previously published study [11], we significantly expanded the experimental framework and explanatory depth, producing a robust and versatile methodology.

Key findings of this thesis include:

- **Baseline Sensitivity:** We showed that the choice of baseline in IG critically affects attribution quality. Among various baselines (zero, pad, mean, mask), the `zero` embedding proved most neutral and consistent across datasets, satisfying completeness more reliably.
- **IG Step Optimization:** Through empirical comparison, we found that a step count of $N = 300$ offered a practical tradeoff between attribution precision and computational cost. Approximation error analysis confirmed its reliability across sentence lengths.
- **Rendering and Visualization:** A structured pipeline was implemented to align subword tokens with SpaCy tokens, handle negations, and normalize attribution magnitudes. These visualizations allowed human-

interpretable outputs that highlighted the agency-bearing components of a sentence.

- **Human Alignment:** Using a dataset of human-highlighted persuasive texts, we demonstrated that IG reliably attributes high agency to segments humans intuitively emphasize. This supports the psychological plausibility of the model's behavior.
- **Small Dataset Strategy:** We introduced a deliberate overfitting strategy on small, labeled datasets to extract class-defining keywords via IG. Applied to the "slurs" dataset, this approach yielded interpretable lexicons for gender, emotional impact, and objectification.
- **Practical and Cross-Disciplinary Impact:** We discussed how explainable IG outputs could be used for content moderation, persuasive messaging, psychological assessment, and digital humanities research, always with a focus on transparency and accountability.

Together, these contributions form a comprehensive framework for interpreting text classification outputs in a sociopsychologically meaningful way. The thesis not only validates IG as a technical tool but also contextualizes its outputs within human cognitive and communicative frameworks.

6.2 Limitations

While this thesis provides a detailed and practical framework for interpretable agency detection in text, several limitations must be acknowledged. These span technical, methodological, and conceptual domains, and they point to opportunities for refinement and future work.

Dependence on Baseline Selection. Integrated Gradients is known to be sensitive to the choice of baseline input. Although our experiments identified the `zero` embedding baseline as the most neutral and effective across datasets, this decision remains task-dependent and somewhat heuristic. In real-world applications where the semantic definition of "absence of input" is ambiguous (as is often the case in language), baseline selection may introduce bias or unintended artifacts.

Approximation Tradeoffs. The choice of number of integration steps (N) in IG directly affects the fidelity of the attribution. While $N = 300$ yielded a good compromise between completeness and computational efficiency, this value is

not universally optimal. Some longer or more complex sentences may require more integration steps for stable results, and approximation error can still impact the reliability of individual attributions.

Model-Specific Interpretations. The interpretability of IG outputs is inherently tied to the underlying model. This thesis focused on RoBERTa-based models, including a fine-tuned version for agency prediction (BERTAgent). The attributions revealed are specific to how these models internalize patterns, and may differ across architectures or training regimes. Thus, the findings should not be overgeneralized to "universal" linguistic patterns without appropriate model comparisons.

Overfitting as a Double-Edged Sword. In Section 4.6.1, we introduced a strategy of deliberate overfitting for small datasets in order to extract class-specific attributions. While effective for keyword discovery, this technique risks learning dataset artifacts rather than generalizable signals. In practical deployments, this method should be treated as a tool for qualitative exploration rather than robust classification.

Semantic Ambiguity and Polysemy. Language often contains semantically ambiguous or polysemous terms. While IG can assign different attributions to the same token in different contexts, it does not inherently "understand" these ambiguities. This limits its effectiveness when interpreting sentences that rely on sarcasm, idioms, or culturally embedded references, areas where human inference would be more nuanced.

Tokenization and Alignment Limitations. Although our visualization pipeline aligns subword tokens from RoBERTa with SpaCy tokens, this process is not perfect. Edge cases involving irregular spacing, punctuation, or BPE token splitting can cause subtle misalignments. These may introduce small inaccuracies in attribution visualization, especially for fine-grained interpretive tasks.

Generalizability Beyond Agency. While the methods developed are generalizable in principle to other psychological constructs (e.g., sentiment, dominance, empathy), all experiments in this thesis were focused on the construct of agency.

Further work is needed to validate the methodology across other abstract or affective variables.

6.3 Directions for Future Research and Applications

The methods and findings presented in this thesis open several promising avenues for future exploration, both within NLP and across interdisciplinary contexts. As interest in interpretable and psychologically grounded AI continues to grow, the following research directions are particularly relevant.

Expanding Beyond Agency. Although this thesis focused primarily on the sociopsychological construct of agency, the methodology developed, especially the token-level attribution and visualization pipeline, is applicable to other nuanced constructs such as empathy, assertiveness, entitlement, or dominance. Future work could explore how different psychological variables manifest linguistically, and whether similar interpretability techniques can reveal their structure with the same granularity.

Cross-Model Comparison. While this work focused on RoBERTa-based architectures, attribution behavior may vary significantly across models such as GPT-2, T5, or newer instruction-tuned LLMs. Comparing IG attributions across architectures can help isolate model-specific biases, assess attribution stability, and better understand how different models encode sociopsychological signals. Such comparisons would also help standardize explainability practices across platforms.

Multi-Language and Cultural Adaptation. The current pipeline is English-specific, relying on SpaCy and BPE alignment. However, agency and other constructs may be expressed differently across languages and cultures. Expanding this work to multilingual corpora, and adapting the preprocessing and visualization tools accordingly, would significantly broaden its applicability. This may also reveal cross-cultural variations in the linguistic markers of agency, enabling comparative sociolinguistic analysis.

Interactive and Real-Time Applications. With the current rendering tools and IG computation pipeline, it is feasible to develop interactive applications for

writing assistance, content moderation, or classroom education. For instance, a writing tool could visualize in real-time whether a sentence sounds agentic or passive, helping users express themselves more clearly or confidently. IG explanations could also be embedded into conversational agents to increase transparency and trust.

Improved Highlight Alignment and Human Feedback. The evaluation with human-highlighted persuasive texts (Section 4.5) suggests that IG aligns well with human judgments, but further work could refine this. Specifically, integrating human feedback into the IG attribution process, either through post-hoc weighting or model fine-tuning, could improve the clarity and relevance of attributions, especially in ambiguous or subtle examples. This could evolve into a hybrid human-in-the-loop interpretability framework.

Robustness and Attribution Calibration. An important future direction is improving the robustness of attribution scores. This includes normalizing across sentence lengths, calibrating attributions for multi-word expressions, and accounting for lexical frequency biases. Attribution methods like IG could be combined with techniques from causal inference or counterfactual reasoning to strengthen their reliability and interpretive value.

Ethical Use and Transparency Standards. Finally, the potential societal impact of interpretability tools calls for the development of usage guidelines and transparency standards. As IG can be used not only for explanation but also for manipulation or surveillance, future research should explore ethical safeguards, consent protocols, and interpretability audits. Cross-disciplinary collaboration between computer scientists, ethicists, and psychologists will be essential to ensure responsible deployment.

References

[1] Jacob Devlin et al. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).

[2] Yinhan Liu. “Roberta: A robustly optimized bert pretraining approach”. In: *arXiv preprint arXiv:1907.11692* (2019).

[3] Soujanya Poria et al. “Beneath the Tip of the Iceberg: Current Challenges and New Directions in Sentiment Analysis Research”. In: *IEEE Transactions on Affective Computing* 14.1 (2023), pp. 108–132. DOI: 10.1109/TAFFC.2020.3038167.

[4] Arwa Diwali et al. “Sentiment Analysis Meets Explainable Artificial Intelligence: A Survey on Explainable Sentiment Analysis”. In: *IEEE Transactions on Affective Computing* 15.3 (2024), pp. 837–846. DOI: 10.1109/TAFFC.2023.3296373.

[5] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. “Axiomatic attribution for deep networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 3319–3328.

[6] Masahiro Makino et al. “The Impact of Integration Step on Integrated Gradients”. In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*. Ed. by Neele Falk, Sara Papi, and Mike Zhang. St. Julian’s, Malta: Association for Computational Linguistics, Mar. 2024, pp. 279–289. URL: https://aclanthology.org/2024.eacl-srw.22.

[7] Jasmijn Bastings et al. “" Will You Find These Shortcuts?" A Protocol for Evaluating the Faithfulness of Input Salience Methods for Text Classification”. In: *arXiv preprint arXiv:2111.07367* (2021).

[8] Albert Bandura. “Social cognitive theory: An agentic perspective”. In: *Annual review of psychology* 52.1 (2001), pp. 1–26.

[9] Andrea E Abele et al. "Towards an operationalization of the fundamental dimensions of agency and communion: Trait content ratings in five countries considering valence and frequency of word occurrence". In: *Eur. J. Soc. Psych.* 38.7 (2008), pp. 1202–1217.

[10] Jan Nikadon et al. "BERTAgent: The Development of a Novel Tool to Quantify Agency in Textual Data". In: *accpeted in the Journal of Experimental Psychology: General* (2025). https://osf.io/preprints/psyarxiv/qw6u3. DOI: 10.31234/osf.io/qw6u3.

[11] Ali Aghababaei et al. *Application of integrated gradients explainability to sociopsychological semantic markers*. 2025. arXiv: 2503.04989 [cs.CL]. URL: https://arxiv.org/abs/2503.04989.

[12] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. "Explainable ai: A review of machine learning interpretability methods". In: *Entropy* 23.1 (2020), p. 18.

[13] Vojtěch Bartička et al. "Evaluating attribution methods for explainable nlp with transformers". In: *International Conference on Text, Speech, and Dialogue*. Springer. 2022, pp. 3–15.

[14] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. "Learning important features through propagating activation differences". In: *International conference on machine learning*. PMlR. 2017, pp. 3145–3153.

[15] Scott Lundberg and Su-In Lee. *A Unified Approach to Interpreting Model Predictions*. 2017. arXiv: 1705.07874 [cs.AI]. URL: https://arxiv.org/abs/1705.07874.

[16] James W Pennebaker. *Linguistic inquiry and word count: LIWC 2001*. 2001.

[17] Seungwan Seo et al. "Comparative study of deep learning-based sentiment classification". In: *IEEE Access* 8 (2020), pp. 6861–6875.

[18] Saif Mohammad et al. "Semeval-2018 task 1: Affect in tweets". In: *Proceedings of the 12th international workshop on semantic evaluation*. 2018, pp. 1–17.

[19] Taha Shangipour Ataei et al. "Pars-OFF: A Benchmark for Offensive Language Detection on Farsi Social Media". In: *IEEE Transactions on Affective Computing* 14.4 (2023), pp. 2787–2795. DOI: 10.1109/TAFFC.2022.3219229.

[20] Carmen Cervone et al. "The Words of (Non-) Humanity: Sexist Slurs Elicit Self-Dehumanization in Women". In: *Journal of Language and Social Psychology* 44.1 (2025), pp. 12–37.

[21] Andrea E Abele and Bogdan Wojciszke. "Communal and agentic content in social cognition: A dual perspective model". In: *Advances in experimental social psychology*. Vol. 50. Elsevier, 2014, pp. 195–255.

[22] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[23] Rui Mao et al. "The Biases of Pre-Trained Language Models: An Empirical Study on Prompt-Based Sentiment Analysis and Emotion Detection". In: *IEEE Transactions on Affective Computing* 14.3 (2023), pp. 1743–1753. DOI: 10.1109/TAFFC.2022.3204972.

[24] Joseph Enguehard. *Sequential Integrated Gradients: a simple but effective method for explaining language models*. preprint, https://arxiv.org/abs/2305.15853. 2023. arXiv: 2305.15853 [cs.CL]. URL: https://arxiv.org/abs/2305.15853.

[25] Soumya Sanyal and Xiang Ren. "Discretized integrated gradients for explaining language models". In: *arXiv preprint arXiv:2108.13654* (2021).

[26] Tomaso Erseghe et al. "Projection of Socio-Linguistic markers in a semantic context and its application to online social networks". In: *Online Social Networks and Media* 37 (2023), p. 100271.

[27] Andrew Maas et al. "Learning word vectors for sentiment analysis". In: *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*. 2011, pp. 142–150.

[28] Jan Nikadon et al. "Uncovering the Linguistic Drivers of Mobilization: A Theory- and Data-Driven Computational Approach". submitted to Computers and Human Behavior. 2025.

[29] Matthew Honnibal et al. *spaCy: Industrial-Strength Natural Language Processing in Python*. https://spacy.io. Version 3.5. 2023.

[30] Andrea E Abele and Bogdan Wojciszke. "Agency and communion from the perspective of self versus others." In: *Journal of personality and social psychology* 93.5 (2007), p. 751.

[31] Jochen E Gebauer et al. "Agency-communion and self-esteem relations are moderated by culture, religiosity, age, and sex: Evidence for the "self-centrality breeds self-enhancement" principle". In: *Journal of Personality* 81.3 (2013), pp. 261–275.

[32] Narine Kokhlikyan et al. *Captum: A unified and generic model interpretability library for PyTorch*. `https://captum.ai`. Accessed: 2025-05-15. 2020.

[33] Jay DeYoung et al. *ERASER: A Benchmark to Evaluate Rationalized NLP Models*. 2020. arXiv: `1911.03429 [cs.CL]`. URL: `https://arxiv.org/abs/1911.03429`.

[34] M. van Zomeren, T. Postmes, and R. Spears. "Toward an integrative social identity model of collective action: A quantitative research synthesis of three socio-psychological perspectives". In: *Psych. Bull.* 134.4 (2008), p. 504.

[35] Magdalena Formanowicz et al. "Mobilize is a Verb: The Use of Verbs and Concrete Language is Associated with Authors' and Readers' Perceptions of a Text's Action Orientation and Persuasiveness". In: *Personality and Social Psychology Bulletin* (2023).

[36] B. L. Fredrickson and T.-A. Roberts. "Objectification Theory: Toward Understanding Women's Lived Experiences and Mental Health Risks." In: *Psychology of Women Quarterly* 21.2 (1997), pp. 173–206. DOI: `10.1111/j.1471-6402.1997.tb00108.x`.

[37] Sarah J Gervais et al. "Seeing women as objects: The sexual body part recognition bias". In: *European Journal of Social Psychology* 42.6 (2012), pp. 743–753.

[38] Albert Bandura. "Self-efficacy: Toward a unifying theory of behavioral change". In: *Psychological Review* 84.2 (1977), pp. 191–215.

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Prof. Tomaso Erseghe, whose guidance, patience, and kindness have been invaluable throughout the course of this thesis. He has not only supported me academically but also instilled in me the mindset required to think critically and independently as a researcher.

I am also deeply thankful to the Department of Information Engineering at the University of Padova for providing me with this exceptional opportunity to study, grow, and broaden my knowledge.