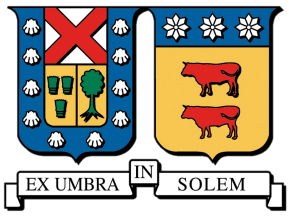

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA

Master's Dissertation

for obtaining the academic degree of
Master in Informatics Engineering

Counterfactual Explanations for Domain Specific NLP Systems with Applications to Radiology and Hate Speech

Domingo Benoit Cea

Composition of the Jury

Supervised by:	Ph.D. Ricardo Ñanculef	Universidad Técnica Federico Santa María, Valparaíso Chile
National evaluator:	Ph.D. Roberto Asín	Universidad Técnica Federico Santa María, Valparaíso Chile
International evaluator:	Ph.D. Giacomo Frisone	Universitá di Bologna, Italia
Commission chair:	Ph.D. Roberto Asín	Universidad Técnica Federico Santa María, Valparaíso Chile

2026

CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☐ Memoria o trabajo de título; ☒ Tesis de Postgrado;

Título del trabajo: Counterfactual Explanations for Domain Specific NLP Systems with Applications to Radiology and Hate Speech

Nombre del candidato(a): Domingo Benoit Cea

Carrera / Grado: Magíster en Ciencias de la Ingieneria Informática

Campus: Casa Central Valparaiso ; **Departamento**: Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Ricardo Ñanculef, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

☒ El trabajo **NO contiene información que amerite confidencialidad** y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

☐ 6 meses; ☐ 12 meses; ☐ 2 años; ☐ 3 años; ☐ 5 años; ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 27-04-26 ; **Firma:** _____

Estudiante o Candidato(a):

Fecha: 27-04-26 ; **Firma:** _____

Declaration of authorship

I hereby declare that I wrote this thesis on the subject

Counterfactual Explanations for Domain Specific NLP Systems with Applications to Radiology and Hate Speech

independently. I did not use any other aids, sources, figures or resources than those stated in the references. I clearly marked all passages that were taken from other sources and cited them correctly.

Furthermore I declare that – to my best knowledge – this work or parts of it have never before been submitted by me or somebody else at this or any other university.

Domingo Benoit Cea

Valparaíso, March 24, 2026 _____

Acknowledgements

The author acknowledges the financial support provided by the Departamento de Informática of Universidad Técnica Federico Santa María for attending and presenting a paper at the International Conference on Agents and Artificial Intelligence (ICAART 2026), held in Marbella, Spain, March 4–6, 2026. Additional travel support was generously provided by the author's advisor through ANID CCTVal (CIA250027).

Abstract

Counterfactual and contrastive explanations have emerged as promising approaches to interpretability in Natural Language Processing, offering clear and actionable insights into the decision boundaries of text classifiers. However, existing methods have been developed predominantly for English and rely on domain-agnostic minimality metrics that fail to capture the linguistic characteristics of specialized domains. In this thesis, we introduce MMiCE (Multilingual Minimal Contrastive Editing), an extension of the MiCE framework that addresses these limitations through three key contributions. First, we expand MiCE to multilingual settings by resolving critical reproducibility barriers associated with its original implementation and re-implementing it using modern, actively maintained libraries. Second, we propose an inverse gradient attribution strategy for multilabel classification tasks, enabling contrastive explanation generation in settings where the traditional one-vs-rest paradigm breaks down. Third, we incorporate MAUVE as a domain-adapted fluency metric within the edit search framework.

We evaluate MMiCE on three datasets spanning two languages and three domain-specific contexts: `IMDB` (English sentiment classification), `ChileanHate` (informal Chilean Spanish hate speech detection), and `42K_HCUCH` (Spanish radiology report classification). Our results demonstrate that MMiCE substantially outperforms both the original MiCE framework in terms of edit minimality, and Polyjuice as a counterfactual baseline, with a flip-score difference exceeding 99%.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

When a machine learning model achieves strong predictive performance, one might ask: why should we concern ourselves with the reasoning behind its decisions? The answer lies in accountability. Generating explanations for model predictions allows us to audit their behavior and verify critical properties such as scientific validity, safety, and ethical soundness (Doshi-Velez and Kim, 2017). An interpretable model, for instance, can expose the decision-making process behind a loan approval or rejection, making it possible for a human auditor to determine whether that decision stems from a relevant and generalizable pattern in the data or from spurious bias learned during training.

The literature has repeatedly shown that counterfactual explanations allow for intuitive formulations in domains such as computer vision or structured attribute data, where the mathematical form of an explanation is straightforward to define and interpret. In natural language processing, however, this task carries an additional layer of complexity: natural language does not follow an evident mathematical structure, and any such structure does not generalize readily across languages. As a consequence, the field of Explainable AI for NLP has seen limited success with counterfactual methods. Beyond the structural challenges, Baron (2023) argues that merely finding explanations that cross a model's decision boundary is insufficient; such explanations must also be grammatically correct, semantically coherent, and meaningful even when the original input is not. These constraints are further compounded by the lack of reliable fluency metrics capable of enforcing them during counterfactual generation.

Although no consensus has emerged, the most commonly used fluency proxies include ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), and perplexity-based measures (Jelinek et al., 1977; Salazar et al., 2020). The absence of a definitive standard is perhaps best illustrated by the Workshop on Machine Translation (Kocmi et al., 2023), which since its inception in 2006 has continuously identified fluency and translation quality metrics as an open and unsolved challenge. Further compounding this issue, little

work has examined how any of these metrics behave in domain-specific settings such as clinical reports or hate speech on social media.

In this thesis, we investigate these limitations and propose a framework to address them, with a particular focus on Spanish-language medical and hate speech domains.

1.1 Problem Statement

Despite the promise of contrastive explanation methods for NLP, existing approaches present three critical limitations that restrict their practical applicability. First, they have been developed predominantly for English, with no systematic evaluation in multilingual or low-resource settings. Second, they rely on domain-agnostic minimality metrics, primarily Levenshtein distance, that fail to capture the linguistic characteristics of specialized domains such as clinical text or informal social media. Third, existing implementations depend on deprecated software libraries, creating significant reproducibility barriers that have limited adoption and follow-up work. This thesis addresses all three limitations through the development of MMiCE (Multilingual Minimal Contrastive Editing).

1.2 Hypothesis

Our hypothesis is that by using fluency metrics adapted to the linguistic characteristics of each domain, we can significantly improve the quality of contrastive explanations generated by state-of-the-art methods.

1.3 Objectives

The specific objectives associated with this thesis are:

1. Analyze and implement the main fluency metrics from the state of the art.
2. Analyze and modify the MiCE method in order to relax its minimality and enforce its fluency constraints.
3. Formal specification of the proposed MMiCE framework, derived from the preceding analysis.
4. Benchmark the performance and impact of the proposed techniques over a medical corpus and a Hate Speech corpus.
    - Medical corpus of 42000 radiology reports belonging to the Chilean context.

- Hate Speech corpus of 4500 tweets belonging to the Chilean context.

5. Preparation of a journal article related to the previous work of the proposal (Under review as of March 24, 2026).
6. Write a conference article related to the work of this thesis (Accepted as a short paper on ICAART 2026).

1.4 Structure

The present work is organized as follows. Chapter 2 provides the theoretical background underlying this work, including a formal treatment of the counterfactual explanation problem in text, a systematic review of fluency and minimality metrics, and an introduction to gradient attribution methods.

Chapter 3 presents the State of the Art, describing the MiCE and Polyjuice methods that serve as baselines for this thesis, and reviewing existing work on explainability for hate speech detection and clinical NLP, with particular attention to the limitations that motivate our contributions.

Chapter 4 presents MMiCE, our proposed method for multilingual minimal contrastive editing. We detail the reproducibility challenges encountered when extending MiCE to multilingual settings, our novel inverse gradient attribution strategy for multilabel classifiers, and the integration of MAUVE as a domain-adapted fluency metric within the edit search framework.

Chapter 5 covers the experimental evaluation of MMiCE. We describe the training setup, LoRA fine-tuning details, and experimental design across three datasets spanning two languages and three domain-specific contexts: English movie reviews, Chilean hate speech, and radiology reports. We then present and analyze the results, including a comparison against MiCE and Polyjuice as baselines.

Finally, Chapter 6 presents the conclusions of this work. We review the degree to which the experimental results support our central hypothesis and the extent to which the stated objectives were achieved, outline the limitations of the proposed approach, and identify the most promising directions for future research.

Chapter 2

Theoretical Background

2.1 Contrastive and Counterfactual Explanations

"Explanations are contrastive" (Lipton, 1990): Humans do not simply ask why a certain prediction was made, but rather why that prediction was made and not another. This counterfactual mode of reasoning (Malmi et al., 2020) makes the generation of interpretable explanations for machine learning models considerably more complex than simple feature attribution: while feature importance methods identify which inputs influenced a prediction, a counterfactual explanation must search over an exponentially large space of possible input modifications to find the minimal change that would have produced a different outcome; a combinatorial search problem whose complexity grows rapidly with input dimensionality.

This challenge is particularly acute for black-box models such as neural networks, whose high-dimensional computational graphs are inherently difficult to interpret, especially in large language models with billions of parameters. The complexity of explaining these models gave rise to local explanation methods, which approximate the most relevant features driving a model's decision in the neighborhood of a given input. However, when applied to text, widely used methods such as LIME (Ribeiro et al., 2016) and SHAP (Lundberg and Lee, 2017) tend to produce diffuse and difficult-to-interpret results, failing to clearly indicate why certain tokens receive higher importance scores than others. Furthermore, recent work has shown that some of the tractability assumptions underlying these methods are theoretically unsound for certain classes of machine learning models (Arenas et al., 2023).

For these reasons, it has been argued that in NLP, contrastive explanations represent the most promising path toward meaningful interpretability. Unlike domains such as computer vision, where feature attribution produces visually interpretable saliency maps, text explanations benefit from explicitly identifying which words or tokens must be modified to change a model's decision. Formally, a counterfactual ex-

planation seeks the input x' that is as close as possible to the original input x while producing a different prediction (Wachter et al., 2018):

$$\hat{x} = \arg\min_{x' \in \mathcal{X}} d(x, x') \quad \text{s.t.} \quad \mathcal{M}(x') = y_{contrast} \tag{2.1}$$

where $\mathcal{M}$ denotes the model for which we are generating explanations, x is the original input, x' is the counterfactual candidate, $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ is a distance function over the input space, and $y_{contrast}$ denotes the contrast label towards which we seek to change the model's prediction. In practice, as proposed by Wachter et al. (2018), the hard constraint can be relaxed into a Lagrangian objective that jointly minimizes distance and prediction deviation. For natural language processing, fluency could be introduced as an additional constraint on the search space; nevertheless the absence of a universally accepted fluency metric makes this constraint difficult to enforce directly, which is precisely the gap this work seeks to address.

2.2 Fluency Metrics

Fluency denotes the degree to which text appears natural to human readers, representing a core objective for all NLG systems. Despite its importance, no universally accepted metric exists for fluency assessment, and it remains an open challenge in the NLP community. Current approaches can be broadly grouped into two categories: *reference-based* and *reference-free* metrics. This distinction is crucial, as reference-based methods assume the presence of a human-produced target that is presumed to be fluent, while reference-free methods aim to estimate fluency directly from the generated text without relying on gold references. The following summarizes the main strategies within each category:

- **Reference-based metrics**: These methods evaluate fluency by comparing a generated sentence to one or more human-written references. Classic lexical overlap metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), NIST (Doddington, 2002), and ROUGE (Lin, 2004) quantify similarity through n-gram or edit-distance-based comparisons. Although originally designed for tasks such as machine translation and summarization, they are often used as indirect fluency proxies under the assumption that high overlap correlates with well-formedness. Distribution-based metrics such as MAUVE (Pillutla et al., 2023) also fall into this category, as they compare global distributions of generated and human text.
- **Reference-free metrics**: These approaches estimate fluency directly from the generated text. Language-model-based perplexity measures (Salazar et al., 2020) provide an estimate of how surprising or well-formed a sequence is under a pretrained model. Readability formulas, such as the

Flesch Reading Ease score (Kincaid et al., 1975), offer interpretable heuristics for textual clarity and structural simplicity. More recently, *LLM-as-a-Judge* (Chiang and Lee, 2023) evaluations have gained traction, in which large language models are prompted to provide explicit fluency judgments or pairwise rankings, often yielding more stable and human-correlated assessments.

Despite this range of tools, the field still lacks a robust *multilingual-aware* fluency metric. Existing methods are predominantly designed for English and struggle with the syntactic and morphological diversity present across languages. Combined with the monolingual focus of prior counterfactual generation research, this gap underscores the need for more comprehensive multilingual fluency evaluation; a challenge that this work seeks to address.

2.3 Reference-based metrics

These approaches seek to estimate fluency by comparing automatically generated text to human-written references, this is often done by quantifying similarity through n-grams or simple edit-distance-based metrics, although the usage of divergence metrics has recently gained popularity (Pillutla et al., 2023).

2.3.1 BLEU

BLEU (Bilingual Evaluation Understudy) is a precision-based metric widely used for evaluating machine translation quality. It measures the overlap of n-grams between candidate translations and reference translations, with a brevity penalty (BP) to discourage overly short outputs. Scores are calculated via n-grams for sentences by comparing them to a set of good quality reference translations (Papineni et al., 2002), as follows:

$$p_n = \frac{\sum_{\mathcal{C} \in Candidates} \sum_{\text{n-gram} \in \mathcal{C}} \text{Count}_{\text{clip}}(\text{n-gram})}{\sum_{\mathcal{C}' \in Candidates} \sum_{\text{n-gram} \in \mathcal{C}'} \text{Count}(\text{n-gram}')} \tag{2.2}$$

$$BP = \begin{cases} 1, & \text{if } |c| > |r| \\ e^{\left(1 - \frac{|r|}{|c|}\right)} & \text{otherwise} \end{cases} \tag{2.3}$$

$$BLEU_N = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{2.4}$$

Where

- r: reference length.
- c: candidate length.

- w_n: n-gram precision weight, it usually follows $w_{0:n} = \frac{1}{N}$

Despite its widespread use, BLEU is a poor proxy for fluency in counterfactual generation: it measures lexical overlap against a reference, but counterfactual edits are by definition expected to differ from the original input. Furthermore, BLEU is insensitive to grammaticality and semantic coherence, and has been shown to correlate poorly with human fluency judgments in open-ended generation tasks (Kocmi et al., 2023).

2.3.2 METEOR

METEOR (Metric for Evaluation of Translation with Explicit Ordering) (Banerjee and Lavie, 2005) addresses limitations of BLEU by incorporating both precision and recall through an F-score formulation. It additionally penalizes fragmentation by considering the number of chunks (contiguous matched segments), making it sensitive to word ordering.

$$
\begin{aligned}
P &= \frac{\text{\# mapped unigrams}}{\text{\# mapped unigrams in candidate}} \\
R &= \frac{\text{\# mapped unigrams}}{\text{\# mapped unigrams in reference}} \\
F\text{-score} - &= \frac{10P \cdot R}{9P + R} \\
Penalty &= 0.5 \left[\frac{\#\text{chunks}}{\text{\# unigrams matched}} \right]^3 \\
METEOR &= F\text{-score} \cdot (1 - Penalty)
\end{aligned}
$$

While METEOR's recall orientation and stemming-based matching make it more flexible than BLEU, it shares the same fundamental limitation for counterfactual evaluation: it requires a human-written reference and does not capture whether an edit is fluent within the specific linguistic register of the target domain.

2.3.3 NIST

NIST (Doddington, 2002) is a variant of BLEU that weights n-grams according to their informativeness. Rare n-grams receive higher weights based on the information gain, making the metric more sensitive to translation errors involving less common words. This addresses BLEU's tendency to treat all n-gram matches equally.

$$\text{Info}(n\text{-gram}) = \log_2 \frac{\text{\# of occurences of } w_1, \ldots, w_{n-1}}{\text{\# of occurences of } w_1, \ldots, w_n}$$

$$NIST = \sum_{n=1}^{N} \left\{ \frac{\sum_{\text{all n-grams that match}} \text{Info}(n\text{-gram})}{\sum_{n\text{-gram} \in Hypotheses}(1)\text{Info}(n\text{-gram})} \right\} \cdot \exp\left(\beta \log^2 \left[\min(\frac{|p|}{|r|}, 1)\right]\right)$$

Where

- w_i: i-th n-gram.
- $w_1, \ldots, w_n$: sentence split in n-gram.

NIST's informativeness weighting makes it more sensitive to rare and domain-specific terms than BLEU, which is relevant in specialized domains such as medical reports. However, it remains a reference-dependent metric and provides no direct signal about the naturalness or domain-appropriateness of generated text.

2.3.4 ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004) is primarily designed for summarization evaluation, though also applicable to translation. Unlike BLEU's precision focus, ROUGE emphasizes recall by measuring how much of the reference text appears in the candidate output.

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{References}} \sum_{\text{n-gram} \in S} Count_{match}(\text{n-gram})}{\sum_{S \in \text{References}} \sum_{\text{n-gram} \in S} Count(\text{n-gram})} \tag{2.5}$$

Where

- N: n-gram length

ROUGE's recall focus makes it particularly ill-suited for counterfactual evaluation, where edits are intentionally designed to diverge from the original input. Its use as a fluency proxy in this setting would penalize precisely the changes that make a counterfactual meaningful.

2.3.5 MAUVE

MAUVE measures the divergence between the statistical distributions of two collections of text. In our case, it compares the entire set of original inputs against the entire set of edited inputs. It works by first embedding all sentences from both sets into a vector space, then clustering (quantizing) these embeddings to create a discrete histogram for each set. MAUVE measures the similarity between these two histograms, quantifying whether the edited text, as a whole, has the same distributional properties as

the original text. We use the implementation made publicly available by the authors Pillutla et al. (2023) in GitHub.

$$
\begin{aligned}
\mathcal{F}_f(P,Q) = &\left\{(D_f(P\|R_\lambda), D_f(Q\|R_\lambda)) : \lambda \in (0,1)\right\} \\
&R_\lambda = \lambda P + (1-\lambda)Q; \quad \forall \lambda \in (0,1) \\
&\mathcal{F}_f\text{: divergence frontier} \\
&D_f\text{: f-divergence (e.g.Kullback-Leibler)} \\
&Q\text{: machine text distribution} \\
&P\text{: human text distribution} \qquad (2.6) \\
\text{MAUVE}_f(P,Q) = & \\
&\text{AUC}(\{(\exp(-x), \exp(-y)) : (x,y) \\
&\in \mathcal{F}_f(P,Q)\} \\
&\cup \{(1,0),(0,1)\}) \\
&\text{AUC: area under the curve}
\end{aligned}
$$

Crucially, due to MAUVE comparing the distribution of generated text against a *domain-specific* reference corpus, it implicitly captures the linguistic characteristics of that domain, its vocabulary, syntactic patterns, and stylistic conventions. This makes MAUVE a fluency metric that is naturally *domain-adaptive*: a high MAUVE score between edited and original texts indicates not only that the edit is statistically plausible, but that it is plausible *within the specific linguistic register of the domain*. This property distinguishes MAUVE from reference-free metrics such as perplexity, which are sensitive to a language model's general training distribution rather than to the target domain.

2.4 Reference-free metrics

Reference-free metrics evaluate generated text quality without requiring gold-standard reference outputs. These metrics assess intrinsic properties of the generated text, such as fluency, coherence, and readability, making them particularly useful when reference texts are unavailable or when evaluating open-ended generation tasks.

2.4.1 Perplexity

Perplexity measures how well a probability model predicts a sample, serving as an indicator of text fluency and naturalness. Lower perplexity values indicate that the model assigns higher probabilities to

the observed token sequences, suggesting more predictable and coherent text generation.

Given a sequence of tokens, perplexity is computed as the exponentiated average negative log-likelihood of the tokens in the sequence:

$$\begin{aligned} X &= (x_0, x_1, \ldots, x_t) \\ \text{PPL}(X) &= \exp\left\{-\frac{1}{t}\sum_{i}^{t}\log p_\theta(x_i|x_{<i})\right\} \end{aligned} \tag{2.7}$$

where X represents an array of tokens, t is the sequence length, p_θ is the language model with parameters θ, and $x_{<i}$ denotes all tokens before position i. Intuitively, perplexity can be interpreted as the weighted average branching factor of the language: the number of possible next tokens the model considers at each step.

2.4.2 Flesch Reading Ease Score

The Flesch Reading Ease (FRE) score (Kincaid et al., 1975) quantifies text readability based on average sentence length and average syllable count per word:

$$\text{FRE} = 206.835 - 1.015\left(\frac{\text{total words}}{\text{total sentences}}\right) - 84.6\left(\frac{\text{total syllables}}{\text{total words}}\right) \tag{2.8}$$

Scores range from 0 to 100, with higher values indicating easier readability. Scores above 60 are generally accessible to readers aged 13–15, while scores below 30 indicate college-level complexity. While FRE is a simple and interpretable heuristic, it is included here for completeness as it represents one of the earliest and most widely cited readability measures. It is worth noting that FRE was designed for English and relies on syllable counting, making it poorly suited for morphologically rich languages such as Spanish. Furthermore, it captures structural complexity rather than linguistic naturalness, and is therefore not used as a fluency metric in our experimental evaluation. We include it as a point of reference to illustrate the breadth of approaches that have been proposed for text quality assessment.

2.4.3 LLMs as a Judge

Recent work has explored the use of large language models as automated evaluators for text quality assessment (Chiang and Lee, 2023). In this paradigm, a powerful LLM is prompted to evaluate generated text according to specific criteria, such as fluency, coherence, or factual accuracy, and produces both a numerical score and a natural language justification. This approach has shown strong correlation with human judgments and can be adapted to task-specific criteria through prompt design.

However, LLM-as-a-Judge evaluations carry significant limitations that restrict their use in rigorous empirical evaluation. First, they exhibit strong positional and verbosity biases, tending to favor longer or more confidently written outputs regardless of quality (Chiang and Lee, 2023). Second, they lack the mathematical formality required for reproducible benchmarking, as results can vary substantially across model versions, prompt formulations, and temperature settings. Third, in domain-specific contexts such as clinical NLP or hate speech detection, general-purpose LLMs may lack the domain expertise required to make reliable quality judgments. For these reasons, while LLM-as-a-Judge represents a promising direction for fluency evaluation, we do not adopt it as a primary metric in this work, favoring instead metrics with well-defined mathematical formulations and established usage in the counterfactual generation literature.

2.5 Minimality Metrics

Effective counterfactual explanations must be both minimal (few edits) and plausible (fluent and meaningful). When generating our edits, we seek to determine whether syntactic minimality, as measured by Levenshtein distance, is the only valid metric for this goal, or if alternatives exist that better maintain fluency. Although Levenshtein is a common baseline, it only counts edit operations and ignores their semantic impact. We hypothesize that metrics focusing on semantic and distributional closeness may be better proxies for plausibility. Therefore, we supplement the Levenshtein baseline with two alternatives: cosine similarity and MAUVE.

- **Levenshtein distance:** Measured between the original and edited input, this is the minimum number of single-character deletions, insertions, or substitutions required to change one input into the other. We report a normalized score (distance divided by original word count) from 0 to 1.
- **Cosine similarity** (Eq. 2.9): Semantic distance between the embedding centroids of the original and edited inputs. Token embeddings are extracted from the **Editor**'s own embedding layer, meaning the semantic similarity is measured in the same representational space used during edit generation. Concretely, for each input we compute the mean of its token embedding vectors and measure the cosine similarity between the resulting sentence-level representations.
- **MAUVE** (Eq. 2.6): Although MAUVE is introduced in this work as a candidate minimality criterion, selecting among edits the one whose distribution diverges least from the original corpus, its underlying signal is fundamentally one of *domain-adapted fluency*. By measuring the distributional divergence between the set of edited texts and the set of original domain texts, MAUVE penalizes edits that deviate from the statistical regularities of the domain, effectively enforcing that

selected edits remain fluent within that specific linguistic context. Thus, incorporating MAUVE into the MMiCE search framework constitutes a direct operationalization of our hypothesis: it replaces a domain-agnostic minimality criteria (Levenshtein) with one that adapts to the linguistic characteristics of each domain.

$$S_C(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}; \quad \mathbf{A}, \mathbf{B} \in \mathbb{R}^n \tag{2.9}$$

2.6 Natural Language Processing and Generation

Natural Language Processing (NLP) is the subfield of artificial intelligence concerned with enabling machines to process, understand, and generate human language. Unlike structured data, natural language is inherently ambiguous, context-dependent, and shaped by cultural and domain-specific conventions, making it one of the most challenging frontiers in machine learning. Natural Language Generation (NLG) is the subdiscipline of NLP focused specifically on producing coherent and fluent text, encompassing tasks such as machine translation, summarization, dialogue generation, and, as in this work, contrastive text editing.

The development of modern NLP systems rests on three foundational building blocks: tokenization, which defines how raw text is decomposed into discrete units suitable for model input; embeddings, which provide dense vector representations that capture semantic and syntactic relationships between those units; and prompting, which enables pretrained models to be steered toward specific generative behaviors through structured input design. Together, these components underpin the architecture and training procedure of MMiCE, and a precise understanding of each is necessary to appreciate both the capabilities and the limitations of the proposed method. We additionally describe the span corruption pretraining objective, which is central to the generative mechanism of the T5 and mT5 **Editor** models used in this work.

2.6.1 Tokenization

Tokenization is the process of converting raw text into a sequence of discrete units, called tokens, that serve as the basic input representation for NLP models. Early approaches relied on word-level tokenization, where each word in the vocabulary is assigned a unique integer index. While intuitive, this approach suffers from two critical limitations: out-of-vocabulary words cannot be represented, and vocabulary sizes grow unboundedly with corpus size. Character-level tokenization addresses the out-of-vocabulary problem but produces excessively long sequences that are difficult for models to process.

Table 2.1: Comparison of tokenization strategies across key linguistic dimensions, illustrated by tokenizing the phrase `"playing games"`. ✓ indicates the property is satisfied, ✗ indicates it is not, and ~ indicates partial satisfaction.

Strategy	**OOV-free**	**Seq. Length**	**Morphology**	**Multilingual**	**Tokenization of `"playing games"`**
Word-level	✗	Short	✗	✗	`[playing] [games]`
Character-level	✓	Very long	✓	✓	`[p][l][a][y][i][n][g] [g][a][m][e][s]`
BPE	✓	Medium	~	~	`[play][ing] [game][s]`
WordPiece	✓	Medium	~	~	`[play][##ing] [game][##s]`
SentencePiece*	✓	Medium	✓	✓	`[_playing] [_games]`

* SentencePiece tokenization shown assumes `playing` and `games` are present as full units in the vocabulary. In practice, further splitting may occur depending on vocabulary size and training corpus.

Modern NLP systems instead rely on subword tokenization algorithms, which decompose words into reusable subword units and strike a balance between vocabulary size and sequence length. The most widely used algorithms are Byte Pair Encoding (BPE) (Sennrich et al., 2016), WordPiece (Wu et al., 2016), and SentencePiece (Kudo and Richardson, 2018). BPE iteratively merges the most frequent pair of adjacent tokens in the training corpus until a target vocabulary size is reached. WordPiece follows a similar approach but selects merges that maximize the likelihood of the training data under a language model. SentencePiece operates directly on raw Unicode characters without requiring language-specific pre-tokenization, making it particularly suitable for multilingual models such as mT5 (Xue et al., 2021). The tokenizer used by a model directly affects how special tokens, domain-specific terminology, and informal text are represented, which has direct implications for the quality of contrastive edits generated by MMiCE, as discussed in Chapter 4.

Table 2.1 summarizes the key trade-offs between tokenization strategies across five linguistic dimensions. Note how word-level tokenization preserves the full surface form of each word but cannot handle unseen vocabulary, while character-level tokenization guarantees full coverage at the cost of very long sequences. Subword methods such as BPE, WordPiece, and SentencePiece strike a middle ground: BPE and WordPiece split morphological suffixes (e.g., `##ing`, `##s`) while retaining the stem as a separate token, whereas SentencePiece treats the full subword unit as atomic and uses the _ prefix to mark word boundaries, making it language-agnostic and particularly well suited for multilingual models such as mT5 (Xue et al., 2021).

2.6.2 Word Embeddings

A word embedding is a dense, low-dimensional vector representation of a token that encodes semantic and syntactic properties of the corresponding word. The key insight underlying word embeddings is the distributional hypothesis (Harris, 1954): words that appear in similar contexts tend to have similar meanings, and this similarity can be captured by training a model to predict contextual co-occurrence patterns.

The seminal Word2Vec model (Mikolov et al., 2013) introduced two training objectives for learning word embeddings: the Continuous Bag of Words (CBOW) model, which predicts a target word from its surrounding context, and the Skip-gram model, which predicts the surrounding context from a target word. These static embeddings assign a single fixed vector to each word regardless of context, which limits their ability to capture polysemy. GloVe (Pennington et al., 2014) extended this approach by training on global word co-occurrence statistics rather than local context windows. Figure 2.1 illustrates how GloVe embeddings organize semantically related words into coherent clusters in the vector space, with structured offsets encoding relational properties such as gender, number, and verb tense.

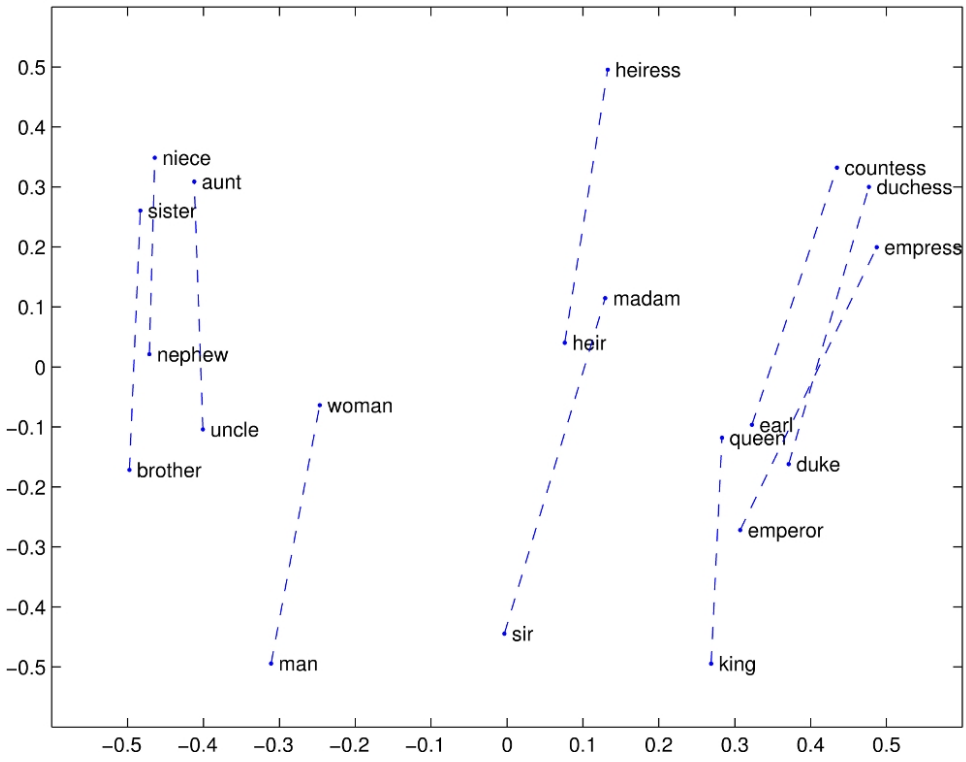

Figure 2.1: Two-dimensional t-SNE projection of GloVe word embeddings, illustrating how semantically related words cluster together in the embedding space. Vector offsets between related pairs (e.g., `king − man + woman ≈ queen`) reflect structured semantic relationships encoded during training (Pennington et al., 2014).

The limitations of static embeddings motivated the development of *contextual embeddings*, where the representation of a token depends on its surrounding context. ELMo (Peters et al., 2018) introduced deep contextualized embeddings by extracting representations from all layers of a bidirectional LSTM. The subsequent introduction of the Transformer architecture (Vaswani et al., 2017) and BERT (Devlin et al., 2019) established the current paradigm of large-scale pretraining followed by task-specific fine-tuning, producing contextual embeddings of significantly higher quality. In MMiCE, embeddings play a role both in gradient attribution, where gradients are computed with respect to the input embedding matrix, and in cosine similarity as a minimality metric, where the centroid of token embeddings serves as a sentence-level semantic representation.

2.6.3 Prompting

Prompting refers to the practice of conditioning a pretrained language model's output by prepending a structured natural language instruction or context to the input, without modifying the model's parameters. The concept emerged from the observation that large language models, trained on diverse text corpora, implicitly encode a broad range of tasks and can be steered toward specific behaviors through input design alone.

The earliest systematic exploration of prompting was conducted by Brown et al. (2020), who demonstrated that GPT-3 could perform few-shot learning by including input-output examples directly in the prompt, a paradigm known as *in-context learning*. This was formalized into three settings: *zero-shot* prompting, where the model receives only a task description; *one-shot* prompting, where a single example is provided; and *few-shot* prompting, where several examples are included. These findings established that model behavior could be substantially controlled through prompt design without any gradient updates.

For encoder-decoder models such as T5 (Raffel et al., 2020), prompting takes the form of a task prefix prepended to the input sequence, such as `summarize:` or `translate English to German:`, which conditions the decoder on the desired output type. This task-prefix paradigm directly informs MMiCE's prompting scheme, where a target label is prepended to the masked input to condition the **Editor** on the desired contrastive output, as described in Chapter 4.

More recent work has explored *prompt tuning* (Lester et al., 2021) and *prefix tuning* (Li and Liang, 2021), which learn continuous prompt vectors in the embedding space rather than discrete token sequences, offering a parameter-efficient alternative to full fine-tuning. While MMiCE does not adopt these approaches, they represent a natural direction for future work in which the prompting strategy itself could be optimized for each target domain.

2.6.4 Span Corruption in Language Modeling

Span corruption is a self-supervised pretraining objective introduced by Raffel et al. (2020) for the T5 (Text-to-Text Transfer Transformer) model. It generalizes the masked language modeling objective of BERT (Devlin et al., 2019) from single token prediction to the prediction of contiguous spans of tokens, making it particularly well suited for sequence-to-sequence architectures.

Formally, given an input sequence $X = (x_1, x_2, \ldots, x_n)$, a random subset of contiguous token spans is selected and replaced with unique sentinel tokens ⟨extra_id_0⟩, ⟨extra_id_1⟩, ... The corrupted sequence $\tilde{X}$ is passed to the encoder, and the decoder is trained to reconstruct the original spans in order, each preceded by its corresponding sentinel token. For example:

- **Original:** The patient presented with chest pain and shortness of breath.
- **Corrupted input:** The patient ⟨extra_id_0⟩ chest ⟨extra_id_1⟩ of breath.
- **Target output:** ⟨extra_id_0⟩ presented with ⟨extra_id_1⟩ and shortness

This formulation has two key advantages over token-level masking. First, by corrupting spans rather than individual tokens, the model is forced to learn richer contextual dependencies and produce coherent multi-token completions. Second, the text-to-text format is naturally compatible with a wide range of downstream tasks, as any NLP problem can be cast as a sequence-to-sequence mapping by appropriate input formatting (Raffel et al., 2020).

The span corruption objective is directly exploited by MMiCE's **Editor** component. During fine-tuning (Stage 1), gradient attribution scores are used to *non-randomly* select the most task-relevant spans for masking, replacing the uniform random span selection used during pretraining with an attribution-guided strategy. This allows the **Editor** to focus its generative capacity on the tokens most responsible for the **Predictor**'s original decision, rather than reconstructing arbitrary spans. At inference time (Stage 2), the masked input is prepended with a target contrast label and passed to the **Editor**, which infills the masked spans conditioned on that label, producing a candidate contrastive edit. The sentinel token format used during pretraining is thus preserved throughout both stages of MMiCE, ensuring compatibility between the pretrained model and the fine-tuning objective.

The multilingual variant of this objective, used by mT5 (Xue et al., 2021), applies the same span corruption formulation across 101 languages drawn from the mC4 corpus, with language sampling controlled by a temperature parameter to balance coverage across high- and low-resource languages. As noted in

Chapter 4, the language sampling imbalance in mC4 has measurable consequences for MMiCE's edit quality across languages, with English-dominant pretraining affecting the model's generative fluency in Spanish-language domains.

Table 2.2: Comparison of self-supervised language modeling objectives. Each row shows how a given objective transforms an original input sequence into a corrupted input and a reconstruction target. Tokens marked with ⟨M⟩ are masked, sentinel tokens ⟨X⟩,⟨Y⟩ replace corrupted spans, and ∅ indicates dropped tokens. Adapted from Raffel et al. (2020).

Objective	**Architecture**	**Original Input**	**Corrupted Input**	**Target Output**	**Models**
Causal LM	Decoder-only	`The cat sat on the mat`	`The cat sat on the`	`mat`	GPT (Brown et al., 2020)
Masked LM (MLM)	Encoder-only	`The cat sat on the mat`	`The ⟨M⟩ sat on ⟨M⟩ mat`	`cat, the`	BERT (Devlin et al., 2019)
Prefix LM	Encoder-decoder	`The cat sat on the mat`	`The cat sat`	`on the mat`	UniLM (Dong et al., 2019)
Permutation LM	Encoder-only	`The cat sat on the mat`	`mat The on cat the sat`	`The cat sat on the mat`	XLNet (Yang et al., 2019)
Token corruption	Encoder-decoder	`The cat sat on the mat`	`The ⟨M⟩ ⟨M⟩ on the mat`	`cat sat`	MASS (Song et al., 2019)
Span corruption	Encoder-decoder	`The cat sat on the mat`	`The ⟨X⟩ on ⟨Y⟩`	`⟨X⟩ cat sat ⟨Y⟩ the mat`	T5 (Raffel et al., 2020), mT5 (Xue et al., 2021)
Deshuffling	Encoder-decoder	`The cat sat on the mat`	`mat on cat The sat the`	`The cat sat on the mat`	(Raffel et al., 2020)

Table 2.2 summarizes the key self-supervised pretraining objectives used in modern language models, illustrating how each transforms an original input sequence into a corrupted input and a reconstruction

target. Span corruption, used by T5 and mT5, is distinguished from token-level masking approaches by its use of sentinel tokens to mark and reconstruct multiple non-contiguous spans jointly, making it particularly well suited for the contrastive editing task performed by MMiCE's **Editor** component.

2.7 Gradient Attribution

Gradient attribution or Saliency Maps first proposed by Simonyan et al. (2014), showcase a method of easily obtaining feature importance from a model through an input vector or a set of input vectors. In particular the authors provide the following motivational example considering a linear class model for the class c with its class score function $S_c(\cdot)$:

$$S_c(I) = w_c^T I + b_c \tag{2.10}$$

where the input vector I can represent and image, text, audio, etc., in vectorized form, w_c and b_c are respectively the weight and the bias vectors of the model. In this case is easy to see how the magnitude of w_c clearly shows the importance of the corresponding features for class c.

Nevertheless in the case for any other Deep Neural Network the class score function is a highly non-linear function of I, meaning that the reasoning of 2.10 does no longer hold. However as shown by Simonyan et al. (2014), given an input vector I_o we can now approximate $S_c(\cdot)$ with a linear function in the neighborhood of I_o by computing the first-order Taylor expansion:

$$S_c(I) \approx w_c^T I + b_c \tag{2.11}$$

where w_c is the derivative of S_c with respect to the vector I at the point of the vector I_o:

$$w_c = \frac{\delta S_c}{\delta I}\bigg|_{I_o} \tag{2.12}$$

This gradient can be computed efficiently via backpropagation and extends naturally to any differentiable neural network architecture. Intuitively, the magnitude of each component of w_c indicates how sensitive the model's prediction is to small changes in the corresponding input dimension, providing a local measure of feature importance. This work and its extensions by Ancona et al. (2019); Sikdar et al. (2021); Sundararajan et al. (2016) have shown strong results across multiple domains, and gradient attribution has proven particularly effective for guiding token masking in text counterfactual generation (Ross et al., 2021).

Chapter 3

State of the Art

3.0.1 Polyjuice

Polyjuice (Wu et al., 2021) is a counterfactual text generation method that produces diverse perturbations of an input sentence without requiring access to a target classifier. Given an input sentence, Polyjuice fine-tunes GPT-2 on a set of human-written counterfactual pairs to learn general-purpose perturbation patterns, which are then applied at inference time using control codes that specify the desired perturbation type (e.g., negation, lexical substitution, or insertion). Unlike MiCE, Polyjuice operates independently of any downstream classifier, making it classifier-agnostic but also unable to guarantee that generated perturbations cross a model's decision boundary. This fundamental limitation motivates its inclusion as a baseline in our experimental evaluation, where we assess whether classifier-guided edit generation produces more effective contrastive explanations than classifier-agnostic perturbation.

3.0.2 GYC: Generate Your Counterfactuals

Madaan et al. (2021) propose GYC (Generate Your Counterfactuals), a framework for controlled counterfactual text generation designed to stress-test NLP and ML systems. Unlike methods that generate arbitrary perturbations, GYC conditions generation on specific linguistic properties such as named-entity tags, semantic role labels, or sentiment polarity, producing counterfactuals that are simultaneously plausible, diverse, goal-oriented, and effective. The framework is evaluated across multiple domains and demonstrates that generated counterfactuals can serve both as test cases for model evaluation and as inputs for debiasing algorithms. A key limitation of GYC, however, is that it does not incorporate feedback from a target classifier during generation, making it unable to guarantee that produced samples cross the model's decision boundary, a limitation that classifier-guided methods such as MiCE address through

beam search.

3.0.3 LIT: Linguistically-Informed Transformations

Li et al. (2020) introduce LIT (Linguistically-Informed Transformations), a method for automatically generating contrast sets by applying rule-based linguistic transformations to existing NLP datasets. LIT enables practitioners to target specific linguistic phenomena of interest, such as negation, quantifier substitution, or syntactic restructuring, and compose multiple transformations to generate complex contrast sets at scale. Experiments on SNLI and MNLI demonstrate that state-of-the-art pretrained language models, despite claiming broad linguistic knowledge, struggle significantly on LIT-generated contrast sets. Furthermore, using LIT for data augmentation improves model robustness on contrast sets without degrading performance on the original data. LIT differs from MMiCE in that it operates through predefined linguistic rules rather than learned generative models, which makes it highly interpretable but limits its applicability to domains where such rules can be explicitly defined, excluding informal or domain-specific text such as clinical reports or Chilean Spanish tweets.

3.0.4 Counterfactual Explanations in Financial NLP

Yang et al. (2020) propose a methodology for generating plausible counterfactual explanations for Transformer-based classifiers in the domain of financial text classification, specifically targeting corporate mergers and acquisitions (M&A) analysis. The authors combine adversarial training with counterfactual generation to simultaneously improve model robustness and produce explanations that are more plausible according to human evaluators, outperforming prior state-of-the-art methods on both accuracy and plausibility. This work is particularly relevant to MMiCE as it demonstrates the value of domain-specific counterfactual explanation methods in high-stakes settings, and highlights the inadequacy of domain-agnostic approaches when applied to specialized text. It also represents one of the earliest examples of counterfactual explanation applied to a domain-specific NLP task, prefiguring the multilingual and domain-adapted focus of this thesis.

3.1 Hate Speech Detection and Explainability

Automated hate speech detection has seen significant progress through the application of transformer-based classifiers, with models such as BERT and its multilingual variants achieving strong performance across several benchmark datasets. However, the opacity of these models poses a critical challenge for deployment in practice: without insight into which linguistic features drive a classification decision, it

is difficult to audit models for bias, identify failure modes, or provide actionable feedback to content moderators.

Existing work on explainability for hate speech classification has relied predominantly on local attribution methods such as LIME (Mittal and Singh, 2023) and SHAP (Lundberg and Lee, 2017), which highlight tokens contributing to a prediction but do not provide actionable counterfactual explanations, that is, they do not indicate what would need to change in the text for the model to reach a different decision. To the best of our knowledge, counterfactual explanation methods have not previously been applied to hate speech detection in Spanish, and no prior work has addressed the specific linguistic characteristics of Chilean informal Spanish in this context. This work addresses both gaps by applying MMiCE to the `ChileanHate` dataset (Benoit Cea et al., 2025), a corpus of approximately 4500 human-annotated tweets collected from the Chilean social media context.

3.2 Natural Language Processing in Medicine

Clinical NLP has emerged as a critical area of research, with applications ranging from automated diagnosis coding to information extraction from radiology reports. Despite the high stakes of medical decision-making, the application of Explainable AI methods to clinical NLP models remains largely unexplored. The few existing works in this space rely on feature attribution methods (Lundberg and Lee, 2017) that highlight relevant terms in a report but do not provide clinicians with a clear understanding of what would need to change in the text for a model to reach a different diagnosis.

Counterfactual explanations are particularly valuable in this context: a clinician presented with a radiology report classified as indicating a certain pathology would benefit not only from knowing which terms drove the prediction, but from a minimally edited version of the report that would have led to a different classification, providing a concrete and interpretable audit of the model's decision boundary. This work is, to the best of our knowledge, the first to apply counterfactual explanation methods to Spanish-language clinical NLP, using the `42K_HCUCH` dataset (De Ferrari et al., 2025), a corpus of 42000 radiology reports with labels corresponding to the presence or absence of three radiological findings: *nódulos* (pulmonary nodules), *condensación* (condensation), and *quiste* (cyst).

3.3 Minimal Contrastive Editing (MiCE)

Despite its promising results, MiCE presents several limitations that motivate the contributions of this work. First, MiCE was designed exclusively for English, with no support for multilingual settings or

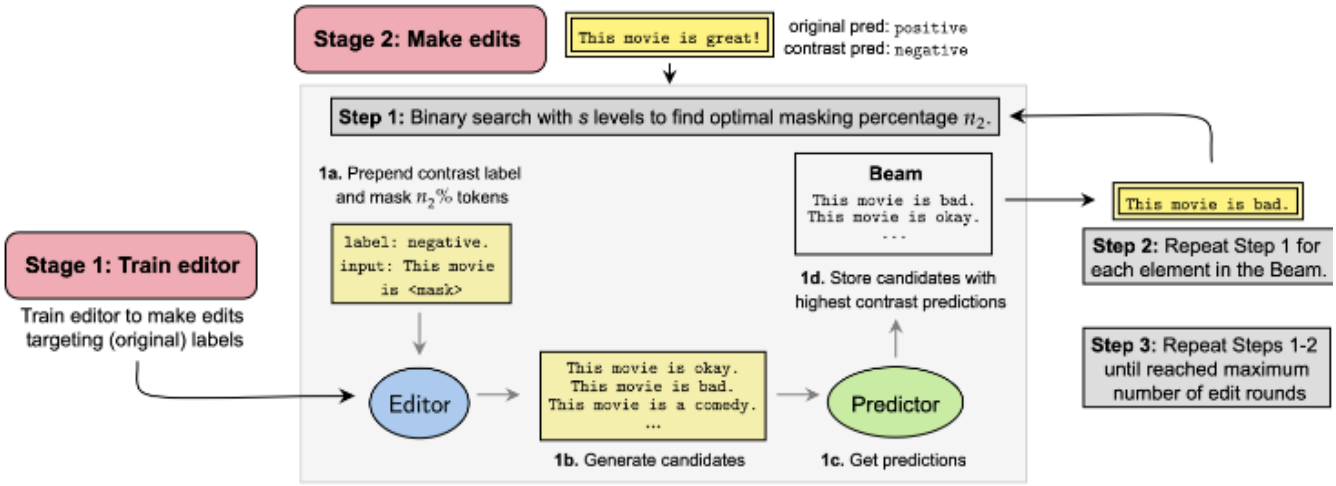

Figure 3.1: MiCE method description.Ross et al. (2021).

domain-specific non-English corpora. Second, it does not support multilabel classification tasks, limiting its applicability to binary or multiclass problems. Third, its implementation relies on the AllenNLP library, which was deprecated in the same year the paper was published, creating significant reproducibility barriers. Finally, MiCE uses Levenshtein distance as its sole minimality metric, which is domain-agnostic and insensitive to the linguistic characteristics of the target domain; precisely the limitation that motivates our central hypothesis. MMiCE, introduced in Chapter 3, addresses each of these limitations directly.

Chapter 4

Methodology

4.1 MMiCE: Multilingual Minimal Contrastive Editing

This section details Multilingual Minimal Contrastive Editing (MMiCE) (see Figure 4.1), our proposed method for generating contrastive and counterfactual explanations for NLP classifiers. MMiCE extends the MiCE framework by introducing several key improvements: it accelerates training, extends support to multiple languages and multilabel classifiers, and incorporates novel metrics for evaluating minimality and fluency.

4.1.1 Formal Definition

Let $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ be a differentiable **Predictor** model, where $\mathcal{X}$ denotes the input text space and $\mathcal{Y}$ the label space. Given an input $x \in \mathcal{X}$ with original prediction $y_p = \mathcal{M}(x)$ and a target contrast label $y_c \in \mathcal{Y} \setminus \{y_p\}$, MMiCE seeks to find a minimal contrastive edit $\hat{x}$ such that:

$$\hat{x} = \arg\min_{x' \in \mathcal{X}} d(x, x') \quad \text{s.t.} \quad \mathcal{M}(x') = y_c \tag{4.1}$$

where $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ is a minimality metric instantiated as Levenshtein distance, cosine similarity, or MAUVE. Fluency is implicitly enforced through the **Editor**'s fine-tuning on domain-specific data, which conditions the model to generate text consistent with the linguistic register of the target domain. When d is instantiated as MAUVE, the minimality criterion additionally acts as an explicit domain-adapted fluency signal, directly operationalizing the central hypothesis of this work.

To identify which tokens to mask, MMiCE uses gradient attribution. For a standard classification task, the attribution score a_i for embedded token $x_i \in \mathbb{R}^d$ with respect to label y_c is:

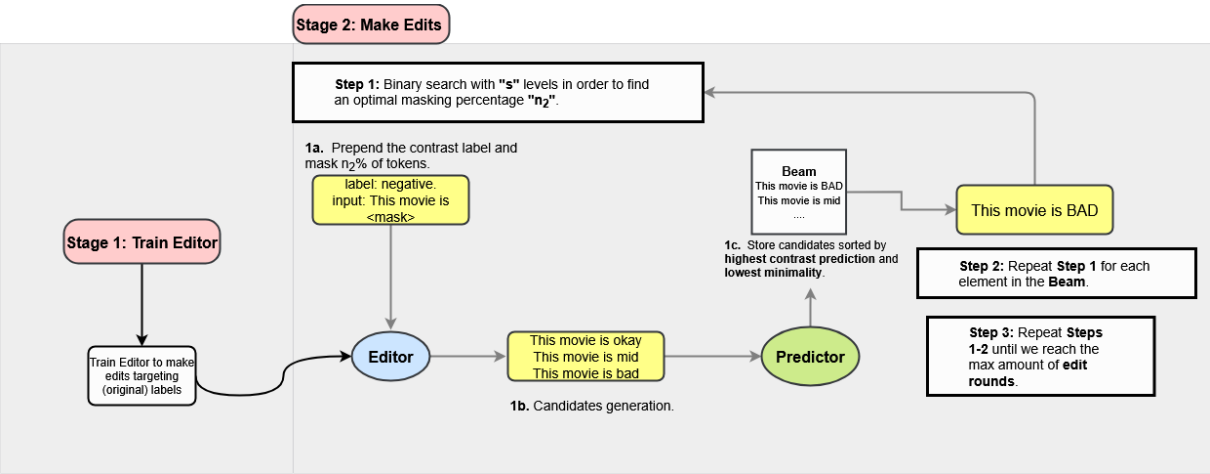

Figure 4.1: MMiCE generation procedure consisting of two fully separate stages: (i) a **training stage** (performed once offline) where the **Editor** learns to infill masked spans conditioned on a prepended label; and (ii) an **edit stage** (performed at inference time, with no parameter updates) which receives a masked input, uses the frozen **Editor** to infill spans conditioned on a contrast label, and selects the most minimal edit according to the chosen metric. The **Editor**'s weights are fixed during the edit stage.

$$a_i = \sqrt{\sum_{j=1}^{d} \left(\frac{\partial \mathcal{M}_Y(x)}{\partial \mathbf{x_{ij}}} \right)^2} \tag{4.2}$$

For multilabel classification, given a target label Y with high predicted probability $P = \mathcal{M}_Y(x)$, MMiCE seeks to increase the probability of the complement $\neg Y$, e.g., $1 - P$. This is achieved through *inverse gradient attribution*, which reverses the token ranking by computing signed gradients and inverting their order:

$$a_i^{\text{inv}} = -\sum_{j=1}^{d} \frac{\partial \mathcal{M}_Y(x)}{\partial \mathbf{x_{ij}}} \tag{4.3}$$

such that tokens most responsible for positively predicting Y receive the lowest attribution scores and are masked last, pushing the **Editor** to replace the evidence for Y with evidence for $\neg Y$.

The beam $\mathcal{B}$ of candidate edits is maintained and updated by ranking candidates first by the predicted probability of the contrast label $P(\mathcal{M}(x') = y_c)$, and then by minimality $d(x, x')$ among candidates with equal contrast probability:

$$\mathcal{B} \leftarrow \text{top-}b\left(\{x'\} \cup \mathcal{B},\ \text{key} = \left(P(\mathcal{M}(x') = y_c),\ -d(x, x')\right)\right) \tag{4.4}$$

4.1.2 MMiCE Algorithm

Algorithm 1 summarizes the full MMiCE pipeline, covering both the training stage and the edit stage, including the multilabel inverse gradient variant.

It is important to note that Stage 1 (fine-tuning) is performed once prior to inference. During Stage 2 (edit generation), the **Editor**'s parameters are frozen and no gradient updates are performed; the beam search operates solely through forward passes through the **Predictor** and **Editor**.

Algorithm 1 MMiCE: Multilingual Minimal Contrastive Editing

Require: **Predictor** $\mathcal{M}$, **Editor** $\mathcal{E}$, input x, contrast label y_c, minimality metric d, beam width b, search levels s, edit rounds R, samples per mask m

Ensure: Minimal contrastive edit $\hat{x}$

— Stage 1: Editor Fine-tuning —

for each training instance (x, y) **do**
  Sample mask rate $n_1\% \sim \mathcal{U}[20, 55]$
  Compute attribution scores $\{a_i\}$ via Eq. 4.2 or Eq. 4.3 if multilabel
  Mask top-n_1 tokens by $\{a_i\}$ to obtain $\tilde{x}$
  Prepend contrast label: $\tilde{x}^+ \leftarrow [y \oplus \tilde{x}]$
  Update $\mathcal{E}$ to minimize reconstruction loss on $\tilde{x}^+$
end for

— Stage 2: Edit Generation —

Initialize beam $\mathcal{B} \leftarrow \varnothing$
$n_2^{\min} \leftarrow 0$, $n_2^{\max} \leftarrow 0.51$
if multilabel task **then**
  Compute inverse attribution $\{a_i^{\text{inv}}\}$ via Eq. 4.3
else
  Compute standard attribution $\{a_i\}$ via Eq. 4.2
end if
for round $r = 1, \ldots, R$ **do**
  for each candidate $x_b \in \mathcal{B}$ (or x if $\mathcal{B} = \varnothing$) **do**
    // Binary search over mask rate n_2
    for search level $l = 1, \ldots, s$ **do**
      $n_2 \leftarrow (n_2^{\min} + n_2^{\max})/2$
      Mask top-n_2 tokens of x_b to obtain $\tilde{x}_b$
      $\tilde{x}_b^+ \leftarrow [y_c \oplus \tilde{x}_b]$
      Sample m candidates: $\{x'_j\}_{j=1}^m \sim \mathcal{E}(\tilde{x}_b^+)$
      if any x'_j satisfies $\mathcal{M}(x'_j) = y_c$ **then**
        $n_2^{\max} \leftarrow n_2$ ▷ reduce mask
      else
        $n_2^{\min} \leftarrow n_2$ ▷ increase mask
      end if
    end for
    for each valid x'_j **do**
      Compute $d(x, x'_j)$ using minimality metric
      Compute $P(\mathcal{M}(x'_j) = y_c)$
    end for
    Update $\mathcal{B}$ via Eq. 4.4 with valid candidates
  end for
end for
return $\hat{x} = \arg\min_{x' \in \mathcal{B}^*} d(x, x')$, where $\mathcal{B}^* = \{x' \in \mathcal{B} \mid \mathcal{M}(x') = y_c\}$

4.1.3 Reproducibility Challenges in MiCE

While adapting the MiCE framework (Ross et al., 2021) to multilingual settings, we encountered two major reproducibility challenges that may partially explain the lack of related work in this area despite its promising results.

Dependency on Deprecated Frameworks The official implementation was built on the AllenNLP library, which was deprecated in the same year the MiCE paper was published. This dependency rendered the code incompatible with modern environments and hindered its direct integration into current NLP pipelines. We resolved this issue by re-implementing all AllenNLP-dependent components to be self-contained, relying solely on widely used and actively maintained libraries such as `PyTorch` and HuggingFace's `Transformers`.

Mismatched mT5 tokenizer Configuration The mT5 model tokenizer configuration files released were found to be mismatched, causing improper text tokenization of special tokens and consequently degraded model performance. While some issues were partially addressed via community forums, the core problem was traced to a mismatch in the SentencePiece tokenizer configuration. We resolved this by manually correcting the sentinel tokens in the binary file and adjusting related code to ensure full compatibility with recent library versions; the working tokenizer has been made publicly available in HuggingFace under Apache License 2.0.

These fixes were crucial to obtaining reproducible results and enabling the application of MiCE in multilingual contexts. The resulting codebase is fully self-contained and has been made publicly available in GitHub under Apache License 2.0.

4.1.4 Multilabel Explanation Strategy

For multilabel classification, a different approach is needed. In multilabel settings, where labels are not mutually exclusive, the traditional contrastive paradigm of changing a prediction to a single alternative class breaks down. Instead, we generate a counterfactual explanation by focusing on a specific label Y that we want to change. We identify the most probable label Y in the original prediction and aim to generate a minimal edit that flips the prediction for this label to its complement, $\neg Y$.

Inverse Gradient Attribution for multilabel Tasks To achieve this, we introduce an inverse gradient attribution strategy. While standard gradient attribution ranks tokens based on their influence on a

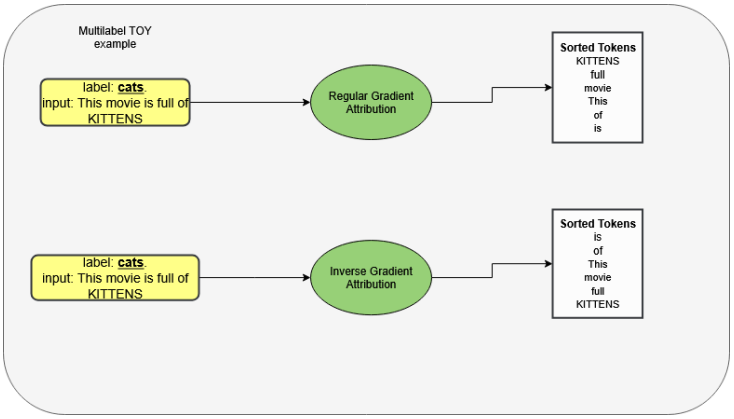

Figure 4.2: **Inverse Gradient Toy example**: for better clarity we showcase how our Inverse Gradient Attribution scheme would work in changing relevant tokens sorting order.

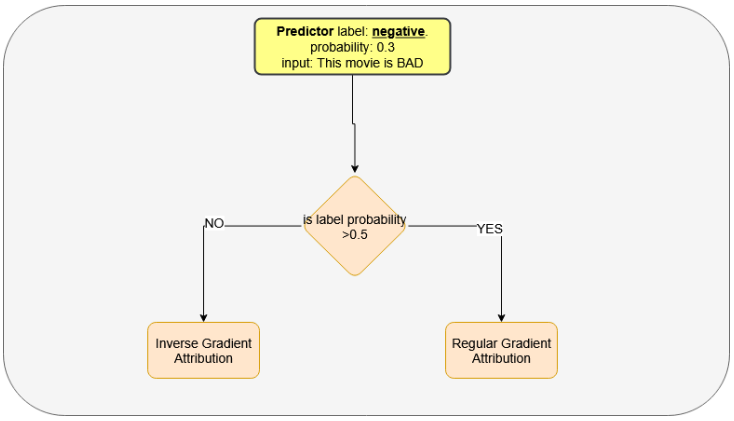

Figure 4.3: **Inverse Gradient Masking algorithm**: We showcase when does the inverse gradient attribution masking triggers depending on a label's **Predictor** predicted probability or gold label value.

given label, our inverse method identifies and masks tokens that are most crucial for the presence of the complement label $\neg Y$. This is achieved by inverting the ranking of the gradient-attributed tokens (see Fig. 4.2). The goal is to generate an edit that maximally decreases the probability of the original label Y, effectively pushing the model's prediction towards its complement, $\neg Y$. In short, our gradient masking direction changes depending on the masked label probability (see Fig. 4.3). To maintain consistency in masked tokens, we need to distinguish between tokens that exert a negative or positive influence on a specific label. This can be achieved through signed gradient attribution; therefore, our multilabel strategy is only available for the regular and integrated variants of this algorithm.

Chapter 5

Experimental Evaluation

This chapter presents the experimental evaluation of MMiCE. We describe the training setup, experimental design, and evaluation metrics used across three datasets spanning two languages and three domain-specific contexts. We then present empirical evidence that MMiCE produces minimal and fluent contrastive and counterfactual edits in multilingual settings, outperforming both the original MiCE framework and Polyjuice as baselines.

5.1 Training Details

For LoRA we use a rank of 8, α of 16 following the $\alpha = 2 \cdot r$ heuristic and dropout of 0.1 on all **Editors**, these hyperparameter choices come from several recent works that have shown positive results (Li et al., 2025; Zhang et al., 2025).

Following Ross et al. (2021), for T5 **Editor**, we use Adam with a learning rate of $1e-4$, we use batch size 6 for fine-tuning with gold labels. For mT5 **Editor**, we use Adam with a learning rate of $1e-3$ as indicated by the authors Xue et al. (2021), we use batch size 4 for fine-tuning with gold labels.

5.2 Language Prompting Scheme

For `IMDB` and `ChileanHate`, we simply prepend target labels to the masked original inputs. For `42K_HCUCH`, we generate N masked inputs where N is the number of labels in the **Predictor**, each with a prepended target label corresponding to either the positive or negated version of that label. For all datasets, we generate two prompt variations: one in Spanish and one in English in order to determine whether prompting in the same language as the input data yields a positive transfer effect on edit quality.

See Table 5.1 for concrete examples.

Table 5.1: Examples of input formats to our **Editor** in different languages.

Language	Original Input	Input to **Editor**
English	Michael, you sent your inquiry to the bmw mailing list, but the sw replaces your return addr with the list addr so I can't reply or manually add you. please see my post re the list or contact me directly	***label:*** misc. ***input:*** <extra_id_0>, you sent your <extra_id_1> to the <extra_id_2>, but the <extra_id_3> your return <extra_id_4> with the list <extra_id_5> so I can't <extra_id_6> or <extra_id_7> add you. please see my post re the list or contact me directly.
Spanish	Michael, you sent your inquiry to the bmw mailing list, but the sw replaces your return addr with the list addr so I can't reply or manually add you. please see my post re the list or contact me directly	***etiqueta:*** misc. ***entrada:*** <extra_id_0>, you sent your <extra_id_1> to the <extra_id_2>, but the <extra_id_3> your return <extra_id_4> with the list <extra_id_5> so I can't <extra_id_6> or <extra_id_7> add you. please see my post re the list or contact me directly.

5.3 Experimental Setup

5.3.1 Tasks

We evaluate MMiCE on one English and two Spanish language datasets, chosen to cover a range of linguistic registers, label complexities, and domain-specific characteristics: `IMDB` (Maas et al., 2011) a binary sentiment classification task on movie reviews; `ChileanHate` (Benoit Cea et al., 2025) a binary hate speech detection task on informal Chilean Spanish tweets; and `42K_HCUCH` (De Ferrari et al., 2025) a multilabel (3-label) classification task on Spanish radiology reports. This selection allows us to evaluate MMiCE across binary and multilabel settings, with a range of different linguistic registers, and domain-specific complexities.

ChileanHate	
Original pred y_p = <u>ODIO</u>	**Contrast pred y_c = NO ODIO**
@maiteorsini lo q pasa rodrigo es q ser ~~feminista~~ **victima** es patrimonio de izqda, para el resto de mujeres da lo mismo q ellas "las ~~feministas~~ **victimas**"~~ofendan~~ **sedan**, incluso también se sienten con el derecho de ofender o agredir a hombres caso contrario son victimas. la igualdad es de verdad o no sirve	
Original pred y_p = <u>NO ODIO</u>	**Contrast pred y_c = ODIO**
~~@wistohor milicia~~ **@Johor y ignorancia** bolivariana dónde el pueblo puede y la patria se crece.	

Table 5.2: Examples of edits produced by MMiCE for inputs from the ChileanHate dataset. Insertions are bolded in red. Deletions are struck through. y_p is the **predictor's** original prediction, and y_c the contrast prediction. True labels for original inputs are <u>underlined</u>.

5.3.2 Predictors

In the same way as MiCE, MMiCE can be used to make contrastive edits for any differentiable **Predictor** model. In this work, we trained a BERT model for the IMDB task, a BETO model for the ChileanHate task, and an XLM-RoBERTa for the 42K_HCUCH dataset. The models reached at least an 81% of test set accuracy on IMDB and ChileanHate, which are both binary classification tasks. For the 42K_HCUCH multilabel classification task, the **Predictor** reached a 0.87 of test set micro-F1.

5.3.3 Editors

We limit our study to the small variants of the T5 model family to maintain a moderate computational footprint. This makes MMiCE accessible to smaller research labs and deployments where large-scale computational resources are unavailable. Specifically, we use T5-Small (≈77M parameters) (Raffel et al., 2020) and mT5-Small (≈300M parameters) (Xue et al., 2021), combined with LoRA (Hu et al., 2021) to further reduce the number of trainable parameters. T5 is evaluated only on the IMDB task, while mT5 is evaluated on all three datasets.

For fine-tuning, we use a $75\%/25\%$ train/validation split for IMDB and 42K_HCUCH, and an $85\%/15\%$ split for ChileanHate given its smaller size (≈4500 instances). Each **Editor** is trained until validation loss stops decreasing, yielding 10 epochs for IMDB, 60 for ChileanHate, and 20 for 42K_HCUCH. Predictions are made only over the best-performing epoch.

In Stage 1, $n_1\%$ of most relevant input tokens are masked, with n_1 drawn uniformly from $[20, 55]$. In

42K_HCUCH	
Original pred y_p = NO nodulos	**Contrast pred y_c = nodulos**
~~árbol~~ **Elemento** traqueo-~~bronquial~~ **arquial** principal permeable. No se observan imágenes de condensación. ~~Banda~~ **Imagen** atelectásica cicatricial en el segmento superior del lóbulo inferior derecho. Finas opacidades lineales basales de aspecto atelectásico. Regresión de las opacidades nodulares centrolobulillares bilaterales en los lóbulos inferiores, persistiendo solo leve engrosamiento de las paredes bronquiales y opacidades lineales irregulares en la base derecha, de significativa menor cuantía con respecto estudio previo. El control actual se observa estabilidad de tamao y calcificación parcial de los nódulos previamente descritos en el lóbulo superior derecho (imagen 121, 138 y 156). Nódulo calcificado en segmento anterolateral del lóbulo inferior izquierdo (imagen 223), benigno. Nódulo adosado paracisural derecho (imagen 199), con características de linfonodo, sin cambios. No han aparecido nódulos pulmonares sospechosos. No se observa derrame pleural. Pericardio sin alteraciones. Corazón de tamaño normal. Calcificaciones en vasos ~~coronarios~~ **costarios**. Grandes vasos mediastínicos de trayecto y calibre conservados. Placas ~~calcificadas aórticas~~ **bandas a las malasticas** No se observa adenopatías mediastínicas. Esqueleto torácico lesiones infiltrativas evidentes. Lesiones en cuerpos vertebrales de T11 y L1, con el aspecto de hemangiomas. Nódulos tiroideos bilaterales.	

Table 5.3: Examples of edits produced by MMiCE for inputs from the 42K_HCUCH dataset. Insertions are bolded in red. Deletions are struck through. y_p is the **predictor's** original prediction, and y_c the contrast prediction. True labels for original inputs are underlined.

Stage 2, for multiclass tasks (IMDB, ChileanHate), the contrast label is set to the class with the second highest predicted probability. For the multilabel task (42K_HCUCH), we target the complement of the label with the highest predicted probability P, seeking to increase the probability of $\neg Y$, e.g., $1 - P$ (see Section 4.1.4). Following (Ross et al., 2021), we set beam width $b = 3$, $s = 4$ binary search levels per edit round, and a maximum of 3 edit rounds. For each mask rate n_2, we sample $m = 15$ generations using top-p nucleus sampling ($p = 0.95$) and top-k sampling ($k = 30$).

5.3.4 Metrics

We evaluate MMiCE on a random sample of 1000 instances from the test set of each dataset. For each configuration, we report the following metrics:

- **Flip-rate**: The proportion of instances for which an edit successfully produces the contrast or

Table 5.4: Class distribution across train and test splits for each dataset. For `42K_HCUCH`, percentages reflect label prevalence in the multilabel setting.

Dataset	Label	Train (%)	Test (%)
`IMDB`	Positive	50.0	50.0
	Negative	50.0	50.0
`ChileanHate`	Hate	45.7	43.2
	No Hate	54.3	56.8
`42K_HCUCH`	Nódulos	42.7%	42.4%
	Condensación	21.0%	21.1%
	Quiste	5.5%	5.7%

counterfactual label.

- **Edit Fluency** (Eq. 2.7): The average perplexity of the edited texts as measured by the mGPT-OSS-20b model (Shliazhko et al., 2023), a multilingual autoregressive language model. Lower perplexity indicates more fluent and natural text. We additionally report the average perplexity of the original (unedited) inputs as a reference point. We select mGPT-OSS-20b as our perplexity model due to its broad multilingual coverage, open weights, and its training on a diverse multilingual corpus, making it suitable for evaluating fluency across both English and Spanish domains without introducing a monolingual bias.
- **Minimality**: We evaluate three alternative minimality metrics, each instantiating the distance function d in Eq. 4.1 differently, and report results separately for each:
  - **Levenshtein distance**: normalized character-level edit distance between original and edited input (see Section 2.5).
  - **Cosine similarity**: semantic distance between word embedding centroids of original and edited inputs (Eq. 2.9).
  - **MAUVE**: distributional divergence between the set of original and edited texts, acting as a domain-adapted fluency metric (Eq. 2.6).

When MMiCE finds multiple valid edits, we report metrics for the edit with the lowest minimality score under the metric being evaluated. Results are reported separately per minimality metric to allow direct comparison of their effect on edit quality.

5.4 Results

Table 5.5: Methods Baseline comparison for `IMDB` using only LoRA models with Levenshtein distance and the english language prompting scheme. * marks what was reported in MiCE (Ross et al., 2021) as **GOLD + GRAD**.

Method	Model	Minimality ↓	Flip-Score ↑	Mean Time [s] ↓
MiCE	*T5	0.173	100.00	1253.84*
MMiCE	T5	0.146	100.00	43.56
	mT5	0.133	100.00	45.32
PolyJuice	-	0.990	0.03	-

* Obtained by running the original code with a sample size of 2500.

The experimental results are detailed in Tables 5.5, 5.6, 5.7, and 5.8. All reported configurations achieve high flip-rates, demonstrating that MMiCE reliably finds contrastive edits across all datasets, languages, and minimality metrics. The following subsections analyze the effect of each experimental dimension on edit quality.

5.4.1 Baseline Comparison

We study the effects of using LoRA in conjunction with the small versions of the **Editor** models by comparing with two baselines: MiCE (Ross et al., 2021) and Polyjuice (Wu et al., 2021) (Table 5.5).

Against MiCE, whose baseline consists of a T5-Base model trained on `IMDB`, all of our **Small + LoRA Editors** outperform it in minimality while maintaining a perfect flip-score of 100%, demonstrating that LoRA fine-tuning improves edit minimality and accelerates training without sacrificing counterfactual effectiveness.

Against Polyjuice, the difference is more pronounced: MMiCE achieves a flip-score of 100% versus Polyjuice's 0.03%, while also producing considerably more minimal edits (0.133 vs. 0.990 Levenshtein). This gap stems from a fundamental design difference: Polyjuice generates perturbations without any feedback from the target **Predictor** $\mathcal{M}$, making it unlikely to produce edits that consistently flip its decision. MMiCE explicitly optimizes for contrast prediction through beam search guided by $\mathcal{M}$'s gradients, which explains its consistently superior performance across both metrics.

Efficiency Analysis Table 5.5 additionally reports mean inference time per explanation in seconds. MMiCE with T5-Small and LoRA achieves a mean inference time of 43.56s per explanation, compared to 1253.84s for the original MiCE implementation, representing a speedup of approximately 28×. This reduction stems from two sources: the smaller parameter count of T5-Small relative to T5-Base used in the original MiCE, and the efficiency that come from further optimizing the pytorch code implementation. However, it is important to note that MMiCE's algorithmic complexity remains non-trivial: each explanation requires up to $b \times s \times m = 3 \times 4 \times 15 = 180$ forward passes through the **Predictor** and **Editor** per edit round, with up to 3 rounds, yielding a worst-case of 540 forward passes per explanation. The MAUVE minimality metric additionally incurs significant overhead due to its distributional computation, as reflected in the consistently higher inference times observed in the MAUVE rows of Tables 5.6, 5.7, and 5.8. These trade-offs between model size, algorithmic complexity, and edit quality should be carefully considered when deploying MMiCE in latency-sensitive applications.

5.4.2 T5 vs. mT5

We investigate the impact of monolingual versus multilingual **Editor** models on edit generation for `IMDB` and `ChileanHate`. On `IMDB` (Table 5.6), T5 consistently achieves better fluency, though mT5 follows closely with a maximum perplexity increase of 22%. We attribute this gap to T5's English specialization and exposure to downstream task training, while mT5 must distribute its capacity across many languages and has not been fine-tuned on downstream tasks, requiring more adaptation to match T5's performance on English text.

For `ChileanHate` (Table 5.7), the high original fluency score (1072.25) reflects the inherently noisy and informal nature of social media text, where conventional grammar is frequently disregarded. Notably, mT5 produces edits considerably closer in perplexity to the original data distribution than T5, suggesting that mT5's multilingual training better equips it to handle the informal and code-switching patterns characteristic of Chilean Spanish Twitter data. This is consistent with our hypothesis that domain-adapted metrics help ensure edits fluency remains close to the original data fluency.

5.4.3 Same Language vs. Foreign Language Prompting

We investigate whether prepending labels in the same language as the input data (same-language prompting) yields better edit quality than foreign-language prompting (Tables 5.6, 5.7, 5.8; see Table 5.1 for prompt format examples).

On `IMDB`, the difference is marginal, with average perplexity decreasing by approximately 2% under

Table 5.6: **MMiCE results on `IMDB`.** Original fluency: 255.56. Best results per model are highlighted.

Lang	Editor Model	Minimality Metric	Flip-Score ↑	Edit Fluency ↓	Mean Time [s] ↓
en	mT5	Levenshtein	100.0	**327.47**	44.03
		Cosine	100.0	331.92	18.73
		Mauve	100.0	338.26	153.40
	T5	Levenshtein	100.0	320.51	51.13
		Cosine	100.0	320.18	15.69
		Mauve	100.0	**318.85**	169.04
es	mT5	Levenshtein	100.0	329.53	44.83
		Cosine	100.0	328.91	19.14
		Mauve	100.0	**327.18**	150.74
	T5	Levenshtein	100.0	**311.24**	46.16
		Cosine	100.0	315.60	13.5
		Mauve	100.0	314.67	157.44

Table 5.7: **MMiCE results on `ChileanHate`.** Original fluency: 1072.25. Best results per model are highlighted.

Lang	Editor Model	Minimality Metric	Flip-Score ↑	Edit Fluency ↓	Mean Time [s] ↓
en	mT5	Levenshtein	99.6	**1778.44**	5.13
		Cosine	99.2	2065.34	4.95
		Mauve	99.8	2463.64	49.59
es	mT5	Levenshtein	99.4	2574.78	5.39
		Cosine	99.2	1579.64	5.00
		Mauve	99.6	**1443.34**	51.15

Table 5.8: **MMiCE results on `42K_HCUCH`.** Original fluency: 166.06. Best results per model are highlighted. Note that the Attribution column (Normal vs. Inverted) applies only to this multilabel dataset, where our inverse gradient strategy is evaluated. IMDB and ChileanHate are binary classification tasks for which the standard gradient attribution is used.

Attribution	Lang	Editor Model	Minimality Metric	Flip-Score ↑	Edit Fluency ↓	Mean Time [s] ↓
Normal	en	mT5	Levenshtein	85.6	347.03	25.19
			Cosine	98.2	**339.68**	15.65
			Mauve	99.7	352.75	95.77
	es	mT5	Levenshtein	97.3	364.08	31.56
			Cosine	98.5	347.14	22.98
			Mauve	96.8	**347.09**	100.49
Inverted	en	mT5	Levenshtein	99.7	**334.01**	30.54
			Cosine	99.7	334.85	19.37
			Mauve	99.0	336.35	95.31
	es	mT5	Levenshtein	98.5	**310.11**	29.05
			Cosine	99.3	347.01	20.53
			Mauve	98.9	353.13	112.22

English prompting for both T5 and mT5, consistent with English being the native language of both the dataset and the prompting scheme. On `ChileanHate`, foreign-language (English) prompting yields higher perplexity than same-language (Spanish) prompting in most configurations, though the absolute differences are modest relative to the dataset's inherently high baseline perplexity. The best result on `ChileanHate` is achieved by mT5 with Spanish prompting and MAUVE as the minimality metric (1443.34), supporting the intuition that same-language prompting better preserves the informal register of the target domain.

For `42K_HCUCH`, English prompting consistently improves edit fluency over Spanish prompting despite the dataset being in Spanish. We attribute this to the significant imbalance in language representation within the mC4 pretraining corpus (Xue et al., 2021), where English accounts for 5.67% of tokens versus 3.09% for Spanish, nearly half the representation resulting in stronger language modeling capacity for English prompts even on Spanish inputs.

5.4.4 Levenshtein vs. Cosine vs. Mauve

These results must be interpreted in light of our central hypothesis: that fluency metrics adapted to the linguistic characteristics of each domain can improve counterfactual generation. MAUVE operationalizes this hypothesis within the MMiCE search framework. Unlike Levenshtein, which measures syntactic distance, or Cosine, which captures generic semantic similarity, MAUVE guides edit selection by rewarding distributional similarity to the target domain corpus, effectively acting as a domain-aware fluency metric. The results below therefore constitute an empirical evaluation of this hypothesis across three distinct linguistic domains.

We compared minimality metrics effects on edit fluency to determine optimal consistency (Tables 5.6, 5.7, 5.8). For `IMDB`, Mauve achieves best results in 2 of 4 experiments with average fluency of 322.57; Levenshtein achieves the best result in the remaining 2 experiments with average 322.18 (best overall); Cosine performs worst on average. On `ChileanHate`, Levenshtein achieves best fluency in 1 of 2 experiments with worst average (2176.61); Mauve wins the other experimental setting with second best average (1953.49), though achieving best average flip-score.

For `42K_HCUCH`, Levenshtein wins 2 of 4 experiments with best fluency average (338.80); cosine wins 1 with a second best average of 342.17, Mauve wins the last setting with the worst average (348.58). Considering flip-scores: cosine leads ($\approx$ 98.9%), Mauve second ($\approx$ 98.6%) and Levenshtein last ($\approx$ 95.2%). The underperformance of MAUVE in this domain constitutes the clearest counterevidence against our hypothesis in its strong form.

Table 5.9: Stability comparison between evaluation runs with n = 100 and n = 1000 instances. We report Edit Fluency for the best configuration per dataset.

Dataset	**Best Config**	n = 100	n = 1000
`IMDB`	T5, en, Mauve	70.82	318.85
`ChileanHate`	mT5, es, Mauve	659.25	1443.34
`42K_HCUCH`	mT5, en, Lev. (Inv.)	122.61	334.01
Flip-score (`IMDB`)		1.000	1.000
Flip-score (`ChileanHate`)		1.000	0.998
Flip-score (`42K_HCUCH`, Inv.)		0.990	0.997

5.4.5 Multilabel: Normal vs. Inverse Gradient Attribution

We evaluate our proposed inverse gradient attribution strategy against standard gradient attribution for the multilabel `42K_HCUCH` task (Table 5.8, Section 4.1.4). Standard attribution achieves an average edit fluency of 349.63 across all configurations, while inverse attribution achieves 335.91, outperforming standard attribution in all configurations except Spanish prompting with MAUVE. Notably, the improvement in flip-score is even more pronounced: inverse attribution achieves an average flip-score of $\approx 98.7\%$ compared to $\approx 95.1\%$ for standard attribution, with the most dramatic difference observed under Levenshtein with English prompting (99.7% vs. 85.6%).

These results validate our inverse gradient attribution strategy: by masking tokens most responsible for the presence of the original label Y rather than those most relevant to the contrast label, the **Editor** is guided to replace evidence for Y with evidence for $\neg Y$, producing edits that are simultaneously more fluent and more effective at flipping the **Predictor**'s decision in complex multilabel settings.

Chapter 6

Conclusions & Future Work

Our results demonstrate that MMiCE is a robust and extensible framework for generating contrastive and counterfactual explanations across multilingual NLP classifiers. By combining LoRA-based fine-tuning with small language models, we significantly reduce training costs while maintaining or exceeding the performance of larger baselines. This validates our design choice of prioritizing minimal and fluent edits while expanding compatibility to multilabel classification settings, maintaining or exceeding the minimality and flip-rate of larger baselines across all three datasets.

6.0.1 Model and Language Variant Performance

In our comparison between T5 and mT5 **Editors**, we observe that T5 consistently achieves better fluency scores on English-language tasks. However, mT5, while slightly behind in fluency, remains competitive and is capable of generalizing across languages even when these multilingual models are not trained on downstream tasks. The larger gap in performance observed in the `ChileanHate` dataset reveals the impact of data quality and informal register on edit generation, particularly when the model must handle informal register and code-switching patterns.

6.0.2 Effect of Prompt Language

These findings suggest that the optimal prompting language is task-dependent: for `ChileanHate` and `42K_HCUCH`, same-language (Spanish) prompting achieves the best results, while `IMDB` benefits from Spanish prompting despite being an English dataset, likely due to T5's exposure to translation as a downstream task during pretraining, which may introduce a cross-lingual transfer effect. Future MMiCE prompting strategies should therefore be tuned per domain rather than defaulting to a fixed language.

6.0.3 Minimality Metrics and Edit Fluency

Our comparison of Levenshtein distance, cosine similarity, and MAUVE as minimality metrics constitutes a direct empirical evaluation of our central hypothesis: that fluency metrics adapted to the linguistic characteristics of each domain can significantly improve state-of-the-art counterfactual generation methods.

The results provide partial support for this hypothesis. The most consistent evidence comes from the `ChileanHate` dataset, where MAUVE with Spanish prompting achieves the best fluency across both evaluation runs (n = 100 and n = 1000), precisely in the domain with the most distinctive and informal linguistic register. This aligns directly with the hypothesis: MAUVE's distributional comparison against the domain corpus acts as a domain-aware fluency metric, rewarding edits that conform to the statistical regularities of Chilean informal Spanish. On `IMDB`, MAUVE is competitive, winning in 2 of 4 experimental settings, though the differences with Levenshtein are less pronounced, likely because English movie reviews represent a more homogeneous and well-represented linguistic register in the pretraining corpora of the embedding models used by MAUVE.

However, on `42K_HCUCH`, MAUVE does not consistently outperform Levenshtein, suggesting that in highly specialized medical domains, either **Editor** model embeddings or the embedding models underlying MAUVE may not adequately capture domain-specific terminology, limiting its effectiveness as a domain-adapted fluency metric. This constitutes the clearest counter-evidence against the hypothesis in its strong form.

When taken together, these results suggest that the hypothesis holds in domains with marked and informal linguistic characteristics, while its benefits are less pronounced in highly specialized technical domains where the underlying embedding models lack sufficient domain coverage. A stronger validation of the hypothesis would require domain-specific embedding models for MAUVE computation and **Editor** embeddings; for instance, embeddings trained on medical corpora for the `42K_HCUCH` task, which we identify as the most promising direction for future work.

6.0.4 Gradient Attribution in Multilabel Tasks

We found that our inverse gradient attribution strategy consistently improves both edit fluency and flip-rate over standard attribution in the multilabel setting of `42K_HCUCH`. In terms of fluency, inverse attribution achieves an average edit perplexity of 335.91 compared to 349.63 for standard attribution. More notably, the improvement in flip-rate is substantial: inverse attribution achieves an average flip-score of $\approx 98.7\%$ versus $\approx 95.1\%$ for standard attribution, with the most dramatic difference observed under

Levenshtein distance with English prompting (99.7% vs. 85.6%). This gap has direct practical implications: in a clinical deployment context, a flip-rate gap of over 14 percentage points means that standard attribution would fail to generate a valid contrastive explanation for roughly one in seven inputs, significantly undermining the reliability of the explanation system. The concept of complement labels provides a useful and principled framework for extending contrastive explanation methods to complex multilabel outputs, where the traditional one-vs-rest paradigm breaks down. By masking tokens most responsible for the presence of the original label Y rather than those most relevant to the contrast label, the **Editor** is guided to replace evidence for Y with evidence for $\neg Y$, producing edits that are simultaneously more fluent and more effective. These results validate inverse gradient attribution as a necessary component of MMiCE for multilabel settings.

6.0.5 Review of Objectives

The specific objectives stated in Chapter 1 were addressed as follows:

1. **Analyze and implement the main fluency metrics from the state of the art.** Achieved. Chapter 2 provides a systematic review of reference-based and reference-free fluency metrics, and three of these (Levenshtein distance, cosine similarity, and MAUVE) were implemented and evaluated within the MMiCE framework.

2. **Analyze and modify the MiCE method to relax its minimality and enforce its fluency constraints.** Achieved. Chapter 4 details the reproducibility challenges encountered, the re-implementation using modern libraries, and the integration of alternative minimality metrics including MAUVE as a domain-adapted fluency signal.

3. **Formal formulation of the proposed method.** Achieved. Chapter 4 provides a complete formal definition of MMiCE, including the optimization objective, gradient attribution strategy, and beam search procedure.

4. **Benchmark the performance of the proposed techniques over a medical corpus and a hate speech corpus.** Achieved. Chapter 5 presents a full experimental evaluation across `ChileanHate` and `42K_HCUCH`, as well as `IMDB` as an English-language baseline.

5. **Preparation of a journal article.** Partially achieved. A journal article related to this work has been submitted and is currently under review.

6. **Write a conference article.** Achieved. A short paper based on this work was accepted and presented at ICAART 2026.

6.0.6 Limitations

While MMiCE shows strong empirical performance, it inherits certain limitations from the underlying **Editor** models. The fine-tuning process, although accelerated by LoRA, remains computationally non-negligible, particularly for low-resource languages or highly domain-specific datasets. Furthermore, while our new metrics offer valuable alternatives to Levenshtein distance, a more formal human expert evaluation is needed to definitively assess the quality and plausibility of the generated edits, especially in complex domains such as medical reports.

A limitation of this work is the absence of formal statistical testing over the experimental results. The MMiCE pipeline contains several sources of stochasticity: the random mask rate $n_1 \sim \mathcal{U}[0.20, 0.55]$ in Stage 1, nucleus sampling (p = 0.95) and top-k sampling (k = 30) in Stage 2, and the random sampling of the evaluation subset. Re-running the full pipeline under multiple random seeds was not feasible within the computational budget of this work.

However, we provide empirical evidence of result stability by comparing runs conducted on evaluation subsets of n = 100 and n = 1000 instances (Table 5.9). Across all three datasets, the qualitative rankings between minimality metrics, **Editor** models, and language prompting schemes are preserved between both sample sizes: on `IMDB` all configurations maintain a perfect flip-score of 100%; on `ChileanHate`, mT5 with Levenshtein (English prompting) and MAUVE (Spanish prompting) achieve best fluency in both runs; on `42K_HCUCH`, the Inverted gradient strategy with Levenshtein outperforms Normal attribution in both runs. These consistent rankings suggest that the conclusions are not sensitive to the particular sample chosen for evaluation. Furthermore, the large magnitude of performance gaps, particularly against Polyjuice, where the flip-score difference exceeds 99%, makes it unlikely that stochastic variation would reverse the qualitative findings. Future work should nonetheless report confidence intervals over multiple runs to formally validate these results.

6.0.7 Conclusion

We introduced MMiCE, a multilingual extension of the MiCE method, enabling both contrastive and counterfactual explanation generation across binary and multilabel classifiers. By leveraging lightweight LoRA fine-tuning, we show that small-scale models can achieve state-of-the-art performance in minimal edit generation, maintaining high flip-rates and fluency across a variety of languages and domains. MMiCE significantly outperforms Polyjuice as a baseline with a flip-score difference exceeding 99%.

Regarding our central hypothesis, the results provide partial empirical support: domain-adapted fluency metrics such as MAUVE demonstrably improve edit quality in domains with distinctive linguistic char-

acteristics, most notably in informal Chilean Spanish. However, the hypothesis does not hold uniformly across all domains; in particular, the highly specialized medical domain of `42K_HCUCH` suggests that the effectiveness of MAUVE as a domain-adapted metric is bounded by the domain coverage of its underlying embedding models. We therefore conclude that the hypothesis is *partially validated*: domain-adapted fluency metrics improve counterfactual generation when the domain's linguistic characteristics are well-represented in the metric's embedding space. We identify domain-specific embedding models as the key open challenge for the full validation of this hypothesis and the most promising direction for future work.

Our experiments highlight the importance of minimality metrics, language prompt design, and attribution strategies in producing effective, interpretable edits. MMiCE's contributions pave the way for more practical and scalable cross-lingual explanation systems, with immediate applicability to high-stakes domains such as hate speech detection and clinical NLP. We identify two primary directions for future work. First, domain-specific embedding models represent the key open challenge for improving MMiCE in highly specialized domains such as clinical text, where general-purpose embeddings lack sufficient terminology coverage. Second, extending MMiCE to multimodal settings—where explanations must account for inputs combining text with other modalities such as images or audio—would represent a significant step toward more faithful and complete explanations; in clinical NLP, for instance, radiology reports are inherently paired with the imaging data they describe, and a contrastive explanation operating solely on text ignores the visual evidence that drove the original clinical observation. Together, these directions outline a path toward explanation systems that are simultaneously more domain-aware and more perceptually complete.

Bibliography

Ancona, M., Ceolini, E., Öztireli, C., and Gross, M. (2019). Gradient-Based Attribution Methods. In Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., and Müller, K.-R., editors, *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, volume 11700, pages 169–191. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

Arenas, M., Barcelo, P., Bertossi, L., and Monet, M. (2023). On the Complexity of SHAP-Score-Based Explanations: Tractability via Knowledge Compilation and Non-Approximability Results. *Journal of Machine Learning Research*, 24(63):1–58.

Banerjee, S. and Lavie, A. (2005). METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Goldstein, J., Lavie, A., Lin, C.-Y., and Voss, C., editors, *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Baron, S. (2023). Explainable AI and Causal Understanding: Counterfactual Approaches Considered. *Minds and Machines*, 33(2):347–377.

Benoit Cea, D., Ñanculef, R., and Mendoza, M. (2025). Chilean Twitter Hate Speech Dataset: CL2.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv:2005.14165 [cs].

Chiang, C.-H. and Lee, H.-y. (2023). Can Large Language Models Be an Alternative to Human Evaluations? In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

De Ferrari, J., Ñanculef, R., Benoit, D., Araya, M., and Solar, M. (2025). Assessing GPT as a Weak Oracle for Annotating Radiological Studies. In *Artificial Intelligence in Medicine: 23rd International Conference, AIME 2025, Pavia, Italy, June 23–26, 2025, Proceedings, Part I*, pages 98–109, Berlin, Heidelberg. Springer-Verlag.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, HLT '02, pages 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., and Hon, H.-W. (2019). Unified Language Model Pre-training for Natural Language Understanding and Generation. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Doshi-Velez, F. and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning. arXiv:1702.08608 [stat].

Harris, Z. S. (1954). Distributional Structure. *WORD*, 10(2-3):146–162.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685.

Jelinek, F., Mercer, R. L., Bahl, L. R., and Baker, J. K. (1977). Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Kincaid, J. P., Fishburne, J., Rogers, R. L., and Chissom, B. S. (1975). Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. Number: RBR875.

Kocmi, T., Avramidis, E., Bawden, R., Bojar, O., Dvorkovich, A., Federmann, C., Fishel, M., Freitag, M., Gowda, T., Grundkiewicz, R., Haddow, B., Koehn, P., Marie, B., Monz, C., Morishita, M., Murray, K., Nagata, M., Nakazawa, T., Popel, M., Popović, M., and Shmatova, M. (2023). Findings of the 2023 Conference on Machine Translation (WMT23): LLMs Are Here but Not Quite There Yet. In Koehn, P., Haddow, B., Kocmi, T., and Monz, C., editors, *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42, Singapore. Association for Computational Linguistics.

Kudo, T. and Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In Blanco, E. and Lu, W., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Lester, B., Al-Rfou, R., and Constant, N. (2021). The Power of Scale for Parameter-Efficient Prompt Tuning. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t., editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Li, C., Shengshuo, L., Liu, Z., Wu, X., Zhou, X., and Steinert-Threlkeld, S. (2020). Linguistically-Informed Transformations (LIT): A Method for Automatically Generating Contrast Sets. In Alishahi, A., Belinkov, Y., Chrupała, G., Hupkes, D., Pinter, Y., and Sajjad, H., editors, *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 126–135, Online. Association for Computational Linguistics.

Li, T., He, Z., Li, Y., Wang, Y., Shang, L., and Huang, X. (2025). Flat-LoRA: Low-Rank Adaptation over a Flat Loss Landscape. In *Forty-second International Conference on Machine Learning*.

Li, X. L. and Liang, P. (2021). Prefix-Tuning: Optimizing Continuous Prompts for Generation. In Zong, C., Xia, F., Li, W., and Navigli, R., editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Lipton, P. (1990). Contrastive Explanation. *Royal Institute of Philosophy Supplements*, 27:247–266.

Lundberg, S. M. and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In Lin, D., Matsumoto, Y., and Mihalcea, R., editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Madaan, N., Padhi, I., Panwar, N., and Saha, D. (2021). Generate Your Counterfactuals: Towards Controlled Counterfactual Generation for Text. arXiv:2012.04698 [cs].

Malmi, E., Severyn, A., and Rothe, S. (2020). Unsupervised Text Style Transfer with Padded Masked Language Models. In Webber, B., Cohn, T., He, Y., and Liu, Y., editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8671–8680, Online. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs].

Mittal, D. and Singh, H. (2023). Enhancing Hate Speech Detection through Explainable AI. In *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, pages 118–123.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a Method for Automatic Evaluation of Machine Translation. In Isabelle, P., Charniak, E., and Lin, D., editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global Vectors for Word Representation. In Moschitti, A., Pang, B., and Daelemans, W., editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In Walker, M., Ji, H., and Stent, A., editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Pillutla, K., Liu, L., Thickstun, J., Welleck, S., Swayamdipta, S., Zellers, R., Oh, S., Choi, Y., and Harchaoui, Z. (2023). MAUVE Scores for Generative Models: Theory and Practice. *Journal of Machine Learning Research*, 24(356):1–92.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ribeiro, M., Singh, S., and Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In DeNero, J., Finlayson, M., and Reddy, S., editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 97–101, San Diego, California. Association for Computational Linguistics.

Ross, A., Marasović, A., and Peters, M. (2021). Explaining NLP Models via Minimal Contrastive Editing

(MiCE). In Zong, C., Xia, F., Li, W., and Navigli, R., editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3840–3852, Online. Association for Computational Linguistics.

Salazar, J., Liang, D., Nguyen, T. Q., and Kirchhoff, K. (2020). Masked Language Model Scoring. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In Erk, K. and Smith, N. A., editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Shliazhko, O., Fenogenova, A., Tikhonova, M., Mikhailov, V., Kozlova, A., and Shavrina, T. (2023). mGPT: Few-Shot Learners Go Multilingual. arXiv:2204.07580.

Sikdar, S., Bhattacharya, P., and Heese, K. (2021). Integrated Directional Gradients: Feature Interaction Attribution for Neural NLP Models. In Zong, C., Xia, F., Li, W., and Navigli, R., editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 865–878, Online. Association for Computational Linguistics.

Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv:1312.6034 [cs].

Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. (2019). MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *Proceedings of the 36th International Conference on Machine Learning*, pages 5926–5936. PMLR.

Sundararajan, M., Taly, A., and Yan, Q. (2016). Gradients of Counterfactuals. arXiv:1611.02639 [cs].

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, u., and Polosukhin, I. (2017). Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wachter, S., Mittelstadt, B., and Russell, C. (2018). Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR. arXiv:1711.00399.

Wu, T., Ribeiro, M. T., Heer, J., and Weld, D. S. (2021). Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models. arXiv:2101.00288 [cs].

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, , Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv:1609.08144 [cs].

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2021). mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Yang, L., Kenny, E. M., Ng, T. L. J., Yang, Y., Smyth, B., and Dong, R. (2020). Generating Plausible Counterfactual Explanations for Deep Transformers in Financial Text Classification. arXiv:2010.12512 [cs].

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Zhang, L., Lou, Z., Ying, Y., Yang, C., and Zhou, H. (2025). Efficient Fine-Tuning of Large Language Models via a Low-Rank Gradient Estimator. *Applied Sciences*, 15(1):82.