Conversational Toxicity Detection and Real-Time Toxicity Assessment using BERT-based Models

Emanuele Fontana[1]

[1]*Università degli Studi di Bari Aldo Moro*

Abstract

This paper presents two studies on conversational toxicity. The first study investigates toxic conversation detection using simple Machine Learning models. The second study describes a novel BERT-based system for real-time toxic conversation tagging and recognition through fine-tuning, which also performs personality classification in dialogues, incorporating zero-shot learning and fine-tuning approaches. Both studies yielded excellent results, with the proposed methodologies achieving significant performance in their respective tasks and demonstrating the effectiveness of context-aware processing in conversational AI applications.

Keywords

Toxicity Detection, BERT, Personality Classification, Conversational AI, Real-time Detection, Italian Language Processing

1. Introduction and Motivations

Online conversation platforms and social media have become integral parts of modern communication, yet they often harbor toxic interactions that can cause psychological harm and create hostile environments. The automatic detection of toxic conversations and personality-driven behaviors has emerged as a critical challenge in maintaining healthy digital spaces.

Traditional approaches to toxicity detection often focus on individual messages or keywords, failing to capture the nuanced dynamics of conversational context and personality-driven behaviors. This limitation becomes particularly evident in intimate relationship conversations where toxicity manifests through subtle manipulation, emotional control, and psychological abuse patterns rather than explicit offensive language.

Our work addresses these challenges by developing a comprehensive system that combines personality classification with real-time toxicity detection, leveraging the contextual understanding capabilities of BERT-based models. The system is designed to identify toxic personality patterns in Italian conversational data, providing both immediate detection capabilities and detailed personality analysis.

2. Related Work

Automatic detection of toxicity in online conversations is an active and crucial research area for promoting healthier digital environments. Early approaches relied on traditional Machine Learning techniques, often focused on individual keywords or isolated messages [1]. However, with the advent of Deep Learning, and particularly Transformer-based models like BERT [2], there has been significant improvement in the ability to understand more complex linguistic nuances [3]. Despite this, many models still struggle to correctly interpret sarcasm, veiled insults, and extended conversational context [4].

Most research on toxicity has focused on the English language. However, there is a growing need to address this problem in other languages as well, such as Italian. To overcome the scarcity of annotated data, especially in specific contexts or underrepresented languages, synthetic data generation through LLMs has emerged as a promising strategy [5]. IDaToC is an example of a dataset generated in this way, containing toxic conversations in Italian.

Our work is positioned at the intersection of these areas, proposing a real-time toxicity detection system for Italian language conversations. This system is distinguished by the integration of fine-grained personality classification based on BERT (`BERT-base-italian-xxl-cased`), using both fine-tuning and zero-shot techniques. Furthermore, it addresses the challenge of data scarcity through the generation of non-toxic conversational data in Italian. The objective is to provide a more contextualized and sensitive toxicity assessment to interpersonal dynamics, with a focus on subtle forms of psychological abuse.

3. Dataset

In this section, we describe the dataset construction process, which integrates existing toxic conversation data with newly generated non-toxic conversations. The dataset is designed to support both binary toxicity classification and personality-based analysis in Italian conversational contexts.

NLP Course 2024-2025

e.fontana7@studenti.uniba.it (E. Fontana)

3.1. Dataset Generation and Integration

The toxic conversation dataset was already available from previous work, containing Italian conversational data with detailed toxicity annotations. To complement this, we developed an automated generation pipeline using Google's Gemini API specifically for creating high-quality non-toxic conversational scenarios.

3.1.1. Existing Toxic Dataset Integration

The foundation of our dataset construction is built upon a pre-existing collection of toxic conversations in Italian. This dataset contains conversational exchanges exhibiting various forms of toxicity including manipulation, emotional abuse, control patterns, and psychological violence. Each conversation in the toxic dataset is accompanied by detailed annotations explaining the specific toxic behaviors and relationship dynamics present.

3.1.2. Non-Toxic Dataset Generation

To create a balanced corpus, we developed a comprehensive generation pipeline for non-toxic conversations using Google's Gemini API. This automated approach ensures diversity while maintaining conversational authenticity and cultural appropriateness for Italian speakers.

The generation system employs Gemini-2.0-flash-lite as the base model with carefully tuned parameters to ensure conversational diversity and authenticity: temperature 1.8 to encourage creative and varied responses, top-p 0.95 for nucleus sampling to balance creativity with coherence, top-k 40 to maintain reasonable vocabulary constraints, and max output tokens 2048 to accommodate detailed conversations and explanations.

Safety settings are configured to block medium and above content for harassment, hate speech, sexually explicit material, and dangerous content, ensuring the generated conversations remain within appropriate boundaries while still capturing toxic behavioral patterns.

The generation process utilizes carefully crafted prompts designed to elicit realistic conversational patterns. The prompts focus on healthy relationship dynamics such as "Entusiasta e Sostenitore", "Preoccupato e Rassicurante", "Affettuoso e Rispettoso", and "Tranquillo e Confortante".

The generation pipeline implements multiple validation layers: format validation through regex patterns to ensure proper conversation structure with numbered, quoted messages; content completeness verification that all required fields (couple type, names, conversation, explanation) are present; conversation length checks to ensure minimum message requirements for meaningful interactions; and retry mechanism with up to 3 attempts per conversation with exponential backoff for failed generations.

3.1.3. Dataset Integration and Final Processing

The final dataset creation process involves merging the existing toxic conversations with the newly generated non-toxic conversations through format standardization where both datasets are processed to ensure consistent conversation format and metadata structure, label assignment where toxic conversations are labeled with value 1 while generated non-toxic conversations receive label 0, conversation format verification using regex-based validation to ensure proper message structure with quoted numbered alternating dialogue, dataset concatenation by merging both datasets while preserving conversation integrity and metadata consistency, and final cleaning through removal of conversations that fail validation criteria ensuring dataset quality and consistency.

The integration process produces a balanced corpus containing both authentic toxic relationship patterns from the original dataset and diverse healthy relationship dynamics from the generated conversations, providing a comprehensive foundation for both binary classification and personality-based analysis.

4. Proposed Approach

Our comprehensive methodology encompasses three main components: (1) binary classification using traditional machine learning approaches, (2) a dual-approach personality classification system using both zero-shot and fine-tuned BERT models, and (3) a real-time toxicity detection system based on personality patterns.

4.1. Binary Classification Approach

To establish a baseline understanding of the dataset's separability, we implemented a traditional machine learning approach for binary toxicity classification. This study compares the effectiveness of text preprocessing on Italian conversational data.

4.1.1. Feature Extraction and Vectorization

We employ TF-IDF (Term Frequency-Inverse Document Frequency) vectorization as our primary feature extraction method. Two distinct approaches are evaluated:

Approach 1: Raw Text Processing involves direct application of TF-IDF to unprocessed conversation text with minimal computational overhead and preservation of original linguistic patterns and colloquialisms.

Approach 2: Italian Language Preprocessing utilizes spaCy's Italian language model (it_core_news_sm) with tokenization using Italian-specific rules, lemmatization to reduce words to their base forms, stop word removal using Italian stop word lists, and punctuation and numeric token filtering.

Since stopwords and lemmatization can remove important contextual information like negation of a word, we wanted to test the impact of these preprocessing steps on the classification performance.

4.1.2. Model Architecture and Hyperparameter Optimization

The classification employs Logistic Regression with comprehensive hyperparameter tuning through nested cross-validation. **Hyperparameter Grid** includes regularization strength (C) values [0.01, 0.1, 1, 10], penalty types ['l1', 'l2'] for feature selection and ridge regularization, maximum iterations [100, 200, 500] to ensure convergence, and 'liblinear' solver for compatibility with both L1 and L2 penalties. **Cross-Validation Strategy** employs inner loop 5-fold cross-validation for hyperparameter selection, outer loop 5-fold cross-validation for unbiased

performance estimation, grid search optimization using accuracy as the primary metric, and final model evaluation on held-out test set (30% of data).

4.2. BERT-based Personality Classification

For the BERT-based personality classification task, the dataset consists of conversational exchanges between individuals exhibiting various personality types, categorized into toxic and non-toxic relationships. We developed a robust preprocessing pipeline that validates conversation format using regex patterns to ensure proper message structure, extracts and cleans individual messages from conversational threads, maps personality couple types to individual personality classifications, creates personality tokens in the format [PERSONALITY] for model training, and applies context-aware tagging to maintain conversational coherence.

The personality mapping includes distinct personality types, ranging from toxic patterns (e.g., PSICOPATICO, MANIPOLATORE, NARCISISTA) to healthy relationship dynamics (e.g., AFFETTUOSO, RISPETTOSO, RASSICURANTE).

We implemented two complementary approaches for personality classification:

4.2.1. Zero-Shot Learning Approach

The zero-shot method leverages pre-trained BERT embeddings to classify personalities without task-specific training. The process involves creating detailed personality descriptions for each of the personality types, computing contextual embeddings for both conversation messages and personality descriptions, using cosine similarity to match messages with the most appropriate personality type, and building conversational context incrementally to improve prediction accuracy.

4.2.2. Fine-Tuned Classification Model

The fine-tuning approach adapts BERT specifically for personality classification by implementing a custom PersonalityClassifier with dropout regularization, and training with early stopping based on validation loss.

The fine-tuned model architecture includes BERT-base-italian-xxl-cased as the base model, custom classification head with dropout (rate: 0.3), AdamW optimizer with linear learning rate scheduling, and maximum sequence length of 512 tokens.

Training Configuration: The model training employs Cross-Entropy Loss as the primary loss function, formally defined as:

$$\mathscr{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(p_{i,c}) \tag{1}$$

where N is the batch size, C is the number of personality classes (27), $y_{i,c}$ is the binary indicator (1 if class c is the correct classification for sample i, 0 otherwise), and $p_{i,c}$ is the predicted probability for class c for sample i.

The training process includes:

- Early stopping with patience of 10 epochs to prevent overfitting
- Gradient clipping with max norm 1.0 for training stability
- Learning rate scheduling with linear warmup (100 steps) and decay
- Batch size of 16 with maximum sequence length of 512 tokens
- Validation split of 20% from training data for monitoring

The model achieved convergence after 15 epochs with a final validation loss of 0.2504.

4.3. Real-Time Toxicity Detection

The real-time detection system analyzes conversations message-by-message, using a weighted scoring mechanism that predicts personality type for each incoming message using conversational context, calculates toxicity scores based on personality classification confidence, applies weighted scoring where toxic personalities increase the score and healthy personalities decrease it, and triggers toxicity alerts when the average weighted score exceeds a threshold (0.3).

Prediction Mechanism: For each incoming message, the system uses the fine-tuned BERT model to predict personality type through the following process:

1. **Context Construction**: The message is concatenated with previous conversation context in the format: "previous_message [PERSONALITY] current_message"
2. **BERT Encoding**: The contextualized message is tokenized and fed through the fine-tuned BERT model
3. **Classification**: The model outputs logits for each of the personality classes
4. **Confidence Extraction**: Softmax is applied to obtain probability distribution, and the maximum probability serves as the confidence score:

$$p_i = \frac{e^{z_i}}{\sum_{j=1} e^{z_j}} \tag{2}$$

where z_i are the logits for class i, and confidence = $\max(p_i)$.

4.3.1. Weighted Scoring Formula

The real-time toxicity detection employs a weighted scoring mechanism based on the following formula:

$$\text{Toxic Score} = \sum_{i=1}^{n} w_i \cdot c_i \tag{3}$$

where:

- n is the number of messages processed so far
- w_i is the weight for message i:

$$w_i = \begin{cases} +1 & \text{if personality}_i \in \text{Toxic Personalities} \\ -0.5 & \text{if personality}_i \in \text{Healthy Personalities} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

- c_i is the confidence score for the personality prediction of message i

The average weighted toxicity score is calculated as:

$$\text{Average Toxic Score} = \frac{\text{Toxic Score}}{n} \tag{5}$$

A conversation is flagged as toxic when:

$$\text{Average Toxic Score} > \theta \tag{6}$$

where $\theta = 0.3$ is the toxicity threshold. The idea behind this value is that we preferred to classify non-toxic conversations as toxic rather than the opposite, so we set a low threshold to avoid false negatives.

5. Experimental Setup and Results

5.1. Binary Classification Baseline

Initial experiments compared TF-IDF vectorization with and without Italian text preprocessing using Logistic Regression. Both approaches achieved perfect classification performance, indicating that the classes are clearly distinguishable. However, preprocessing required 20 times more computational time without performance benefits, making the raw text approach optimal for efficiency.

Table 1
Binary Classification Results

Approach	Accuracy	F1-Score	Precision	Recall
Without Preprocessing	1.0000	1.0000	1.0000	1.0000
With Preprocessing	1.0000	1.0000	1.0000	1.0000

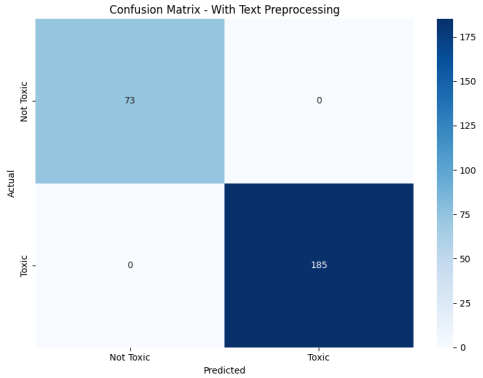

Figure 1: Confusion Matrix for Binary Classification with Preprocessing

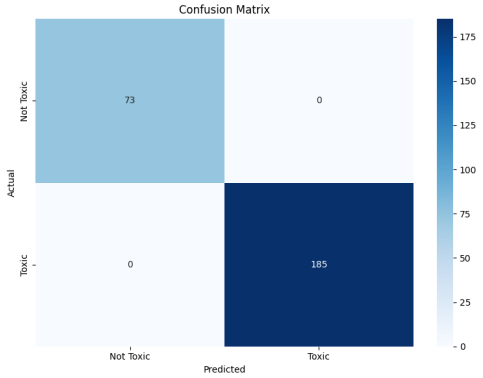

Figure 2: Confusion Matrix for Binary Classification without Preprocessing

Table 2
Zero-Shot Personality Classification Results

Metric	Score
Accuracy	0.0268
Macro Precision	0.0010
Macro Recall	0.0364
Macro F1-Score	0.0020

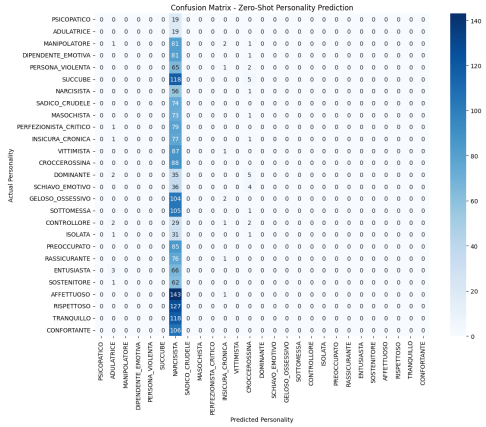

Figure 3: Confusion Matrix for Zero-Shot Personality Classification

5.2. Personality Classification Results

5.2.1. Zero-Shot Performance

From the results, it is evident that the zero-shot approach struggled to accurately classify personalities, achieving an accuracy of only 2.68%. The model's inability to generalize effectively to the personality classification task highlights the limitations of zero-shot learning in this context. We can notice that the model predominantly predicted the "NARCISISTA" personality type. Since recall is slightly higher than precision, it indicates that the model was more likely to identify instances of "NARCISISTA" correctly, but it also produced many false positives for this class.

5.2.2. Fine-Tuned Model Performance

The fine-tuned BERT model significantly outperformed the zero-shot approach, achieving an accuracy of 56.28%. Training converged after 15 epochs with early stopping, achieving a validation loss of 0.2504.

The model demonstrated improved precision and recall across multiple personality types, indicating its ability to learn and generalize from the training data effectively.

5.3. Real-Time Detection Performance

The real-time toxicity detection system using weighted scoring achieved exceptional performance:

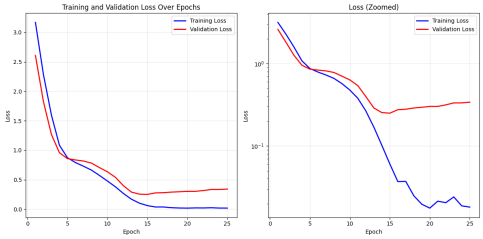

Figure 4: Training and Validation Loss Over Epochs

Table 3
Fine-Tuned Personality Classification Results

Metric	Score
Accuracy	0.5628
Macro Precision	0.5093
Macro Recall	0.5043
Macro F1-Score	0.5015

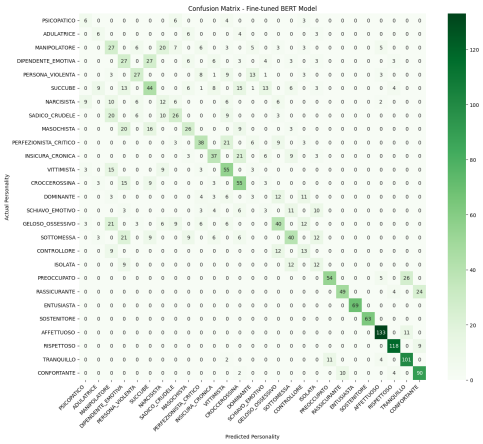

Figure 5: Confusion Matrix for Fine-Tuned Personality Classification

Table 4
Real-Time Toxicity Detection Results

Metric	Score
Accuracy	0.9884
Precision	0.9943
Recall	0.8889
F1-Score	0.9915

The system achieved high accuracy (98.84%) and precision (99.43%), indicating its effectiveness in identifying toxic messages in real-time conversations. The recall of 88.89% suggests that while the system is highly precise, it may miss some toxic instances.

6. Conclusions and Limitations

In this work, we presented a comprehensive study on conversational toxicity detection and personality classification using traditional machine learning approaches and advanced BERT-based models. The key findings are summarized below:

6.1. Binary Classification Analysis

The perfect accuracy achieved with both raw text and preprocessed approaches indicates that the classes are clearly distinguishable. However, given that preprocessing incurred 20 times higher computational cost while yielding identical performance results, the preprocessing step is

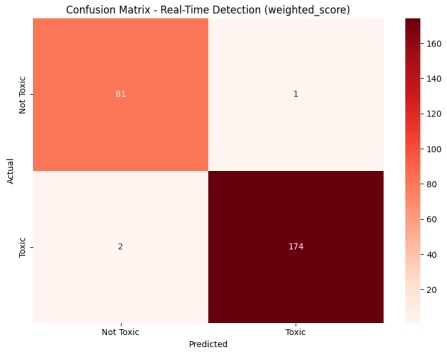

Figure 6: Confusion Matrix for Real-Time Toxicity Detection

not recommended for this binary classification task. The raw text approach provides optimal efficiency without sacrificing accuracy.

6.2. Model Comparison for Personality Classification

The fine-tuned BERT model demonstrated substantial improvements over zero-shot learning, with accuracy increasing from 2.6% to 56.28%. This highlights the importance of task-specific training in achieving meaningful performance in personality classification tasks. The fine-tuned model's ability to learn from the dataset and generalize across multiple personality types underscores the effectiveness of fine-tuning pre-trained models for specialized tasks.

6.3. Real-Time Detection Effectiveness

Since the fine-tuned model was able to classify personalities with an accuracy of 56.28%, the real-time detection system effectively utilized these predictions to achieve impressive results in real-time toxicity detection. The system's high accuracy (98.84%) indicates that it is highly effective at identifying toxic messages in real-time conversations. The weighted scoring mechanism allows the system to adaptively respond to the conversational context.

6.4. Current Limitations

Current limitations include: (1) language specificity as the system is currently optimized for Italian conversations, (2) personality framework dependency on a specific personality classification scheme, and (3) context window limitations imposed by BERT's maximum sequence length (512 tokens).

6.5. Future Directions

Future research directions encompass: (1) multilingual extension by adapting the system for other languages, (2) larger datasets expansion for improved generalization, (3) advanced architectures exploration including GPT-based models and longer context windows, and (4) real-world deployment integration with chat platforms and social media monitoring.

The code and dataset are available on GitHub at: https://github.com/Fonty02/NLP/tree/main/Exam

References

[1] P. Fortuna, S. Nunes, A survey on automatic detection of hate speech in text, ACM Computing Surveys 51 (2018) 1–30.
[2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (2019).
[3] E. Martinez, C. Rodriguez, R. Singh, Large language models for cyberbullying and online abuse detection: A comprehensive survey, Computer Communications and Security 89 (2024) 123–145.
[4] S. Thompson, M. Kim, R. Anderson, Benchmarking large language models for cyberbullying detection, in: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 2024, pp. 2156–2168.
[5] X. Wang, J. Liu, W. Chen, Synthetic data generation using large language models: A survey, AI and Data Science Review 12 (2024) 45–78.

7. Appendix

7.1. Gemini Prompt

The following is the prompt used to generate non-toxic conversations with Gemini: The following is the prompt used to generate non-toxic conversations with Gemini:

```
Genera una conversazione non tossica tra due persone in una coppia, seguendo questi tipi di
    relazioni non tossiche:
- Entusiasta e Sostenitore
- Preoccupato e Rassicurante
```

```
- Affettuoso e Rispettoso
- Tranquillo e Confortante

La conversazione deve contenere battute numerate alternate tra le due persone.
Ogni battuta deve essere COMPLETA e ben formattata. Non lasciare frasi a meta.
La conversazione non deve essere tossica e non deve contenere comportamenti manipolativi,
    controllo, umiliazione, ricatto emotivo, violenza psicologica.

Fornisci:
1. Il tipo di coppia (uno dei tipi sopra elencati), SOLO UNO
2. Due nomi italiani per i partecipanti
3. Una conversazione non tossica con 8 battute numerate complete
4. Una spiegazione dettagliata di perche la conversazione non e tossica

Formato richiesto:
TIPO_COPPIA: [tipo di relazione], SOLO UN TIPO
NOME1: [nome italiano]
NOME2: [nome italiano]
CONVERSAZIONE:
1. Nome1: "frase completa"
2. Nome2: "frase completa"
3. Nome1: "frase completa"
4. Nome2: "frase completa"
5. Nome1: "frase completa"
6. Nome2: "frase completa"
SPIEGAZIONE: [spiegazione dettagliata di perche non e tossica]

IMPORTANTE: Assicurati che ogni battuta sia completa e finisca con le virgolette. Non lasciare
    frasi incomplete.

Rispondi SOLO in italiano
```

7.2. Hyperparameters for Binary Classification

7.2.1. Cross-Validation Configuration

Both approaches employed identical cross-validation strategies:

Table 5
Cross-Validation Configuration

Parameter	Value
Inner CV Folds	5
Outer CV Folds	5
Scoring Metric (Grid Search)	Accuracy
Scoring Metric (Final Evaluation)	F1-Score
Test Set Split	30%
Stratification	Applied

The nested cross-validation approach with 5×5 fold configuration provided robust hyperparameter selection while maintaining unbiased performance estimates for model comparison.

7.2.2. Hyperparameter Search Space

The complete grid search explored the following parameter space:

Table 6
Hyperparameter Search Grid

Hyperparameter	Search Values
Regularization Strength (C)	[0.01, 0.1, 1, 10]
Penalty Type	['l1', 'l2']
Maximum Iterations	[100, 200, 500]
Solver	['liblinear']

The liblinear solver was selected for its compatibility with both L1 and L2 penalties and efficiency with sparse feature matrices typical of TF-IDF vectorization.

7.2.3. Approach 1: With Text Preprocessing

Table 7
Optimal Hyperparameters - With Preprocessing

Hyperparameter	Optimal Value
Regularization Strength (C)	10
Penalty Type	L2 (Ridge)
Maximum Iterations	100
Solver	liblinear
Random State	42

7.2.4. Approach 2: Without Italian Text Preprocessing

Table 8
Optimal Hyperparameters - Without Preprocessing

Hyperparameter	Optimal Value
Regularization Strength (C)	1
Penalty Type	L2 (Ridge)
Maximum Iterations	100
Solver	liblinear
Random State	42