ADVANCING DIALOGUE SYSTEMS WITH TEMPORAL CONTEXTUAL TOPIC FORECASTING AND CONTROLLED RESPONSE RETRIEVAL

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EMRE KÜLAH

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
COMPUTER ENGINEERING

FEBRUARY 2026

Approval of the thesis:

ADVANCING DIALOGUE SYSTEMS WITH TEMPORAL CONTEXTUAL TOPIC FORECASTING AND CONTROLLED RESPONSE RETRIEVAL

submitted by **EMRE KÜLAH** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Assoc. Prof. Dr. Hande Alemdar
Supervisor, **Computer Engineering** _____

Examining Committee Members:

Prof. Dr. İsmail Hakkı Toroslu
Computer Engineering, METU _____

Assoc. Prof. Dr. Hande Alemdar
Computer Engineering, METU _____

Prof. Dr. Pınar Karagöz
Computer Engineering, METU _____

Prof. Dr. Arzucan Özgür
Computer Engineering, Boğaziçi Üniversitesi _____

Assoc. Prof. Dr. Hacer Yalım Keleş
Computer Engineering, Hacettepe Üniversitesi _____

Date:23.02.2026

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Emre Külah

Signature :

ABSTRACT

ADVANCING DIALOGUE SYSTEMS WITH TEMPORAL CONTEXTUAL TOPIC FORECASTING AND CONTROLLED RESPONSE RETRIEVAL

Külah, Emre
Ph.D., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Hande Alemdar

February 2026, 133 pages

Dialogue systems powered by large language models (LLMs) show strong generative abilities but often struggle with informal language, long-term coherence, and grounded responses in expert-driven conversations. This thesis presents three complementary methods to address these challenges. First, we introduce a novel temporal forecasting framework that models dialogue topic trajectories as time series and predicts future topic distributions based on Gaussian Mixture Model (GMM)–based similarity across conversations, offering interpretable and accurate forecasting in both domain-specific and open-domain settings. To enhance topic modeling under noisy user inputs, we develop SALDIRAY, a task-agnostic standardization pipeline that normalizes messages affected by spelling errors, slang, and abbreviations, improving downstream NLP performance across tasks such as sentiment analysis and topic labeling. Finally, we propose RAP (Retrieval-Augmented Paraphrasing), a retrieval-based generation method that retrieves similar past responses and paraphrases them using LLMs to fit the current context, significantly reducing hallucinations while maintaining stylistic alignment and relevance. Together, these methods improve the robustness, foresight, and factual grounding of

conversational agents, advancing the reliability of LLM-based dialogue systems in real-world applications.

Keywords: conversational AI, large language models, text pre-processing, retrieval-augmented generation, time-series analysis.

ÖZ

ZAMANSAL BAĞLAMSAL KONU TAHMİNİ VE KONTROLLÜ YANIT GETİRME İLE DİYALOG SİSTEMLERİNİN GELİŞTİRİLMESİ

Külah, Emre
Doktora, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Hande Alemdar

Şubat 2026 , 133 sayfa

Büyük dil modelleri (LLM'ler) tarafından desteklenen diyalog sistemleri güçlü üretim yetenekleri sergilese de gayriresmî dil kullanımı, uzun vadeli tutarlılık ve uzman odaklı konuşmalarda bağlama dayalı yanıt üretimi konularında sıklıkla zorluk yaşamaktadır. Bu tez, bu zorlukları ele almak amacıyla birbirini tamamlayan üç yöntem sunmaktadır. İlk olarak, diyalog konularının zaman içindeki seyrini zaman serileri olarak modelleyen ve konuşmalar arası Gaussian Karışım Modeli (GMM) tabanlı benzerlikten yararlanarak gelecekteki konu dağılımlarını tahmin eden yeni bir zamansal öngörü çerçevesi tanıtılmaktadır; bu yaklaşım hem alan odaklı hem de açık alan senaryolarında yorumlanabilir ve doğru tahminler sunmaktadır. Gürültülü kullanıcı girdileri altında konu modellemeyi geliştirmek için yazım hataları, argo ve kısaltmalardan etkilenen mesajları normalize eden görevden bağımsız bir standartlaştırma hattı olan SALDIRAY geliştirilmiş ve duygu analizi ile konu etiketleme gibi aşağı akış NLP görevlerinde performansın iyileştirilmesini sağlamıştır. Son olarak, RAP (Retrieval-Augmented Paraphrasing) adı verilen, geçmişteki benzer yanıtları getirip bunları mevcut bağlama uygun biçimde LLM'ler aracılığıyla yeniden ifade

eden getirme tabanlı bir üretim yöntemi önerilmiş, böylece stil uyumu ve alaka düzeyi korunurken halüsinasyonlar önemli ölçüde azaltılmıştır. Bu yöntemler birlikte ele alındığında, konuşma ajanlarının dayanıklılığını, öngörü yeteneğini ve olgusal temellendirmesini güçlendirerek LLM tabanlı diyalog sistemlerinin gerçek dünya uygulamalarındaki güvenilirliğini ileriye taşımaktadır.

Anahtar Kelimeler: sohbet tabanlı yapay zekâ, büyük dil modelleri, metin ön işleme, getirme destekli üretim, zaman serisi analizi.

To my wife, my son, and whole family members...

ACKNOWLEDGMENTS

First and foremost, I would like to sincerely thank my supervisor, Assoc. Prof. Dr. Hande Alemdar, for her continuous support throughout this journey. I am especially grateful for her guidance, patience, and encouragement during the times when I struggled both academically and personally with motivation. Her dedication, constructive insights, and tireless effort to guide me throughout the entire process have been invaluable. It has been a great privilege to work under her supervision.

I would also like to thank my jury members for their valuable feedback and constructive comments. Their insights helped me organize my ideas more clearly and improve the overall structure and quality of this thesis.

I would like to thank the METU CENG Data Mining Research Group and the project ADEP-312-2024-11484 for allowing me to use their server resources, which supported the computational part of this research.

My sincere thanks go to my friends and colleagues for their support, encouragement, and help whenever I needed it. Their presence made this long and challenging journey much easier.

I am deeply grateful to my family for their priceless support, sacrifices, and constant belief in me. Their great effort and encouragement helped me complete this journey.

Finally, I would like to express my deepest thanks to my beloved wife. She did her very best to support and stand by me throughout this entire process. This would not have been possible without her love and patience. And to my little son — thank you for giving me motivation and energy every single day. You have been my greatest source of strength and joy.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xvi

LIST OF FIGURES xx

LIST OF ABBREVIATIONS xxii

CHAPTERS

1 INTRODUCTION 1

2 TEMPORAL CONTEXTUAL SIMILARITY AND FORECASTING FOR CONVERSATIONAL LLMS 9

2.1 Related Work 10

2.1.1 Temporal Similarity in Time-Series Forecasting 11

2.1.2 Temporal Context in Natural Language Processing 12

2.1.3 Gaussian Mixture Models in Temporal Pattern Analysis 13

2.2 Methodology 13

2.2.1 Datasets and Preparation 14

2.2.1.1 Internal Dietitian–Client Dialogue Dataset 14

2.2.1.2 SHARE Dataset for Long-Term Open-Domain Dialogue . 14

2.2.2 Temporal Pattern Representation with Gaussian Mixture Models 16
2.2.3 Cross-Dialogue Temporal Similarity Computation 17
2.3 Experiments and Results 21
2.3.1 Trend similarity score 23
2.3.2 Topic Modeling for Dietitian–Client Dialogues 24
2.3.3 Model Selection 26
2.3.3.1 Selection of GMM Components 27
2.3.3.2 Selection of Number of Neighbors 27
2.3.4 Forecasting Results on Dietitian-Client Dataset 28
2.3.5 Forecasting Results on SHARE Dataset 29
2.4 Conclusion and Discussion 30
Implementation Guidance for SGSE Forecasting 31
Future work and Conclusion 32
3 SALDIRAY: SCALABLE AND ADAPTIVE LANGUAGE DIAGNOSTICS FOR REMEDIATION OF ABBREVIATIONS AND TYPOS 35
3.1 Literature Review 38
3.1.1 Advances in Spelling Correction Models 38
3.1.2 Misspellings and Their Impact on NLP Robustness 39
3.1.3 Synthetic Spell Error Generation 40
3.2 Methodology 41
3.2.1 Data Acquisition and Synthetic Noise Generation 42
3.2.2 LLMs for Text Standardization 43
3.2.3 Integration into Downstream NLP Tasks 46
3.3 Experimental Results 47

3.3.1 Tasks and Datasets 48
Topic Classification. 49
Sentiment Analysis. 49
Aspect-Based Sentiment Analysis. 50
Natural Language Inference. 50
Fact Verification. 51
Real-World Topic Modeling on BEA60K. 51
3.3.2 LLM-based Standardization Models and Fine-Tuning 52
3.3.3 Standardization Evaluation Metrics 55
3.3.4 Standardization Performance 56
Evaluation on the Synthetically Corrupted Dataset 57
Evaluation on Naturally Corrupted Manual Typing Misspelling Dataset 58
Evaluation on the Naturally Corrupted BEA60K Misspelling Dataset 60
3.3.5 Standardization Quality and Overcorrection Analysis 62
3.3.6 Downstream Task Performance with Standardized Data 64
3.4 Conclusion and Future Works 77
4 RAP: RETRIEVAL AUGMENTED PARAPHRASING 79
4.1 Related Work 80
4.1.1 Retrieval-Augmented Generation for Conversational LLMs . . . 80
4.1.2 LLMs for Paraphrasing 81
4.1.3 Solutions to Hallucination in LLMs 82
4.2 Methodology 84
4.2.1 Utterance Construction and Corpus Creation 84

4.2.2 Knowledge Base Generation 85
4.2.3 Retrieval of Candidate Responses 86
4.2.4 LLM-Based Paraphrasing 86
4.3 Experiments 88
4.3.1 Datasets 88
4.3.1.1 Client-Dietitian WhatsApp Dataset 88
4.3.1.2 Persuasion for Good Dataset 89
4.3.2 Experimental Setup 90
Text Embeddings 90
Retrieval Mechanism 90
Language Model 90
4.3.3 Evaluation Metrics 91
Knowledge Extraction Evaluation 91
Automatic Evaluation of Response Generation 91
Human Evaluation of Response Quality 92
4.3.4 Results 92
4.3.4.1 Knowledge Base Evaluation 93
4.3.4.2 Ablation Study: Number of Retrieved Candidates 93
4.3.4.3 Ablation Study: Effect of Paraphrasing 94
4.3.4.4 Ablation Study: Conversation History Integration 94
4.3.4.5 Ablation Study: Standardization 95
4.3.4.6 Automatic Benchmarking Against Baselines 96
4.3.4.7 Human Evaluation of Best Configurations 97

4.3.4.8 Automatic Evaluation of Persuasion for Good Dataset . . 98

4.3.4.9 Human Evaluation of Best Configurations for Persuasion for Good 99

4.4 Conclusion and Discussion 100

5 CONCLUSION 103

REFERENCES 107

APPENDICES 119

A Knowledge-Base Generation 119

A.1 Knowledge-Base Generation and Validation for Turkish Dataset 119

A.2 Example Extracted Knowledge-Base Entries with Corresponding Agent Messages 123

B Examples for RAP Response Generation 126

C Examples for Detailed RAP Response Generation 127

CURRICULUM VITAE 131

LIST OF TABLES

TABLES

Table 2.1 Grid search results for neighbor selection. Best (lowest) value per metric is highlighted in bold. 28

Table 2.2 Forecasting results on the Dietitian-Client dataset. For error-based metrics (MAE, RMSE, and sMAPE), the lowest values are highlighted in bold. For Trend Similarity Score (TSS), the highest value is highlighted in bold. 30

Table 2.3 Forecasting results on the SHARE dataset. For error-based metrics (MAE, RMSE, and sMAPE), the lowest values are highlighted in bold. For Trend Similarity Score (TSS), the highest value is highlighted in bold. . . . 34

Table 3.1 Training corpus generated by synthetically corrupting 100K sampled Wikipedia sentences under five different conditions, producing a total of 900K noisy–clean pairs. The first bold row represents the original uncorrupted version used as the clean reference. 45

Table 3.2 Overview of tasks and datasets used in our experiments. 48

Table 3.3 Token and sentence-level standardization performance across different noise types and correction models using Accuracy, F1, Precision, and Recall. Top performance is bolded, all values reported with three decimal digits. 56

Table 3.4 Standardization and misspelling detection performance on manually curated native and non-native typing datasets using Accuracy, F1, Precision, and Recall. Top performance is bolded in both tests. 59

Table 3.5 Sentence-level correction and token-level misspelling detection performance on the BEA60K dataset using Accuracy, F1, Precision, and Recall. Top performance is bolded. 60

Table 3.6 Slang standardization results. Best values in each column are highlighted. 62

Table 3.7 Homophone standardization results. Best values in each column are highlighted. 62

Table 3.8 False positive breakdown across slang and homophone normalization (% of FP within each method). For each corruption strategy, the highest value within each false-positive category is highlighted. 64

Table 3.9 Train–test configurations for downstream experiments. O: original data, C: corrupted data, S_C: standardized version of corrupted data. . . . 65

Table 3.10 Topic classification performance on 20 Newsgroups under different train–test configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined. 66

Table 3.11 Sentiment analysis performance on IMDB under different train–test configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined. 67

Table 3.12 MaskedABSA performance across train–test configurations on SemEval datasets. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined. 69

Table 3.13 Three-way NLI performance across premise–hypothesis corruption configurations. Each cell shows macro-averaged Precision / Recall / F1. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined. 70

Table 3.14 Fact verification performance on CLIMATE-FEVER under claim–evidence corruption configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined. 71

Table 3.15 Effect of targeted corruption strategies on entailment classification using the All-NLI dataset. The first row (O) presents the baseline performance of each model on clean input. Subsequent rows (A, M, H, S) report performance deltas relative to this baseline under different corruption types: All (A), Misspellings only (M), Homophones only (H), and Slang only (S). C_μ and W_μ indicate the average character error rate and word error rate, respectively, introduced by each corruption strategy. Bolded values denote the smallest performance degradation within each row, indicating greater robustness. 72

Table 3.16 Semantic Preservation on Synthetic Corruption Fixes of NLI Dataset using SALDIRAY 73

Table 3.17 Topic modeling performance on BEA60K under natural noise. C_v: coherence; Div.: topic diversity; ARI, NMI, and Jaccard: clustering agreement with gold labels. Top performance is bolded. 74

Table 3.18 Inference Throughput of Standardization Models (Iterations/Second) 77

Table 4.1 Human Evaluation of Extracted Knowledge Base 93

Table 4.2 Effect of Number of Retrieved Candidates on RAP. For each metric, the highest scores are highlighted. 94

Table 4.3 Effect of number of retrieved candidates on RAP with conversation history. For each metric, the highest scores are highlighted. 94

Table 4.4 Effect of paraphrasing on RAP performance. For each metric, the highest scores are highlighted. 95

Table 4.5 Effect of conversation history on RAP. For each metric, the highest scores are highlighted. 95

Table 4.6 Effect of SALDIRAY standardization on RAP. For each metric, the highest scores are highlighted. 96

Table 4.7 Benchmarking across generation methods. For each metric, the highest scores are highlighted. 96

Table 4.8 Human evaluation results for the best performing methods. For each metric, the highest scores in **relevance** and **helpfulness**, and the lowest score in **hallucination**, are highlighted. 97

Table 4.9 Human Evaluation on Top-50 vs. Bottom-50 Samples (Higher is better for relevance and helpfulness; lower is better for hallucination) . . 98

Table 4.10 Benchmarking across all methods on Persuasion for Good dataset. For each metric, the highest scores are highlighted. 99

Table 4.11 Human evaluation results for the best performing methods for Persuasion for Good dataset. For each metric, the highest scores in **relevance** and **helpfulness**, and the lowest score in **hallucination**, are highlighted. . 100

LIST OF FIGURES

FIGURES

Figure 2.1 Exploratory statistics for the dietitian–client dataset. 15

Figure 2.2 Distribution of utterance counts in the SHARE dataset. 15

Figure 2.3 Two example Gaussian mixture model (GMM) fits to smoothed topic frequency trajectories. The components reveal distinct temporal engagement modes within individual client consultations. 17

Figure 2.4 Dialogue state construction via shifted cutoff replication of topic time-series. 19

Figure 2.5 Forecasting via interpolation of temporally similar dialogue segments. 20

Figure 2.6 MAE and TSS errors for two dialogues where MAE scores are similar but TSS scores are completely different. 23

Figure 2.7 Top representative keywords for a sample of discovered topics. . 24

Figure 2.8 Hierarchical clustering of discovered topics based on their c-TF-IDF representations. 26

Figure 2.9 Model selection for GMM components using averaged BIC analysis. 27

Figure 2.10 Illustration of the SGSE forecasting process. The source conversation is matched with its four closest neighbors (with temporal offsets), and their aligned future trajectories are aggregated to generate the final 30-day forecast. 33

Figure 3.1 Example NLP application pipeline using the SALDIRAY framework as a preprocessing module. The figure shows the flow from noisy input through SALDIRAY-based standardization to downstream task inference 41

Figure 3.2 Samples from the corruption module, showing original and synthetically generated noisy variants containing misspellings, homophones, and slang. 45

Figure 3.3 Experiment 1: ARI, NMI, and topic-level Jaccard similarity on BEA60K when each dataset variant (Original, Standardized, Gold) is evaluated with its own best hyperparameters. 75

Figure 3.4 Experiment 2: ARI, NMI, and topic-level Jaccard similarity on BEA60K using a single hyperparameter configuration optimized on the Standardized dataset and transferred to both Standardized and Gold corpora. 76

Figure 4.1 RAP usage pipeline 84

Figure 4.2 Transformation of the client-dietitian conversation dataset . . . 85

LIST OF ABBREVIATIONS

ABBREVIATONS

ABSA	Aspect-Based Sentiment Analysis
ACF	Autocorrelation Function
ARI	Adjusted Rand Index
ARIMA	Autoregressive Integrated Moving Average
ASR	Automatic Speech Recognition
BIC	Bayesian Information Criterion
CNN	Convolutional Neural Network
DTW	Dynamic Time Warping
EM	Expectation–Maximization
FAISS	Facebook AI Similarity Search
GMM	Gaussian Mixture Models
GRU	Gated Recurrent Units
HMM	Hidden Markov Model
LCS	Longest Common Subsequence
LLMs	Large Language Models
LoRA	Low-Rank Adaptation
LSTM	Long Short-Term Memory
NARANN	Nonlinear Neural Networks
NLG	Natural Language Generation
NLI	Natural Language Inference
NLP	Natural Language Processing
NMI	Normalized Mutual Information

PACF	Partial Autocorrelation Function
RAG	Retrieval-Augmented Generation
RAP	Retrieval-Augmented Paraphrasing
SALDIRAY	Scalable and Adaptive Language Diagnostics for Remediation of Abbreviations and Typos
SNLI	Stanford Natural Language Inference
STT	Speech-to-Text
TCS	Temporal Contextual Similarity
TP	True Positives
TSS	Trend Similarity Score
WER	Word Error Rate

CHAPTER 1

INTRODUCTION

As text-based digital services continue to expand across domains such as healthcare, education, and customer support, the need for intelligent systems that can understand and respond to user needs in linguistically diverse and contextually rich conversations becomes increasingly important. Platforms that facilitate client–agent communication, such as online dietitian consultations, introduce specific computational challenges. Clients often write in informal, abbreviated, and error-prone language, and their concerns tend to evolve gradually over time as they receive feedback, adjust routines, or encounter new difficulties. To offer effective and personalized support in these interactions, systems must both anticipate future conversation topics and generate responses that are accurate, relevant, and aligned with the client's communication style and long-term progress. This thesis addresses these two challenges independently: first, forecasting the progression of dialogue topics using past message history, and second, generating grounded and stylistically adapted responses through controlled paraphrasing of real agent replies.

The first major contribution of this thesis is a method for forecasting topic trajectories in long-term client interactions. In domains like dietitian support, users regularly discuss themes such as eating habits, exercise routines, hydration, and weight changes. These themes appear over time as clients report on their behavior, receive advice, and react to challenges or progress. Accurately predicting which topics are likely to come up next can help the system take initiative, highlight emerging issues, or provide timely and personalized reminders. This task is challenging because client messages are often short, irregular, and contain significant linguistic noise.

To address this, we represent each dialogue as a time series of topic distributions. These topic labels are obtained using unsupervised topic modeling applied to client messages. Once this representation is built, we use similarity-based matching to find users with similar recent topic patterns and forecast the future based on their trajectories. This strategy relies on the idea that users with similar topic histories tend to follow similar paths. Rather than using a fixed parametric model, the framework compares and aggregates trajectories from multiple past users, allowing it to handle complex and nonlinear conversational patterns. The forecasting process benefits from the structure of real client journeys and offers an interpretable way to reason about what might happen next in a conversation.

This approach depends on having clean and meaningful topic assignments. However, due to the informal and noisy nature of real-world messages, raw topic modeling can produce fragmented or unclear clusters. To improve quality, we apply SALDIRAY, a domain-independent preprocessing tool that standardizes noisy user input by correcting spelling errors, expanding abbreviations, and normalizing informal variants such as slang and homophones. This tool helps generate more coherent and interpretable topic structures. Beyond this application, SALDIRAY is designed to be task-agnostic and domain-independent, making it a useful normalization layer for a wide range of NLP tasks affected by user-generated noise.

The second core contribution focuses on the generation of accurate and stylistically consistent responses in client–agent conversations. Large language models (LLM)s have demonstrated impressive generative abilities, but they often struggle with factual consistency and grounding, especially in domain-specific contexts. In health-related conversations, incorrect or hallucinated responses can reduce trust, confuse users, or lead to poor outcomes. Generating responses that are both grounded in prior examples and tailored to the specific user remains an open challenge.

To mitigate these issues, we propose a retrieval-based generation framework called Retrieval-Augmented Paraphrasing, or RAP. Instead of generating responses from scratch, RAP searches for semantically similar client messages from historical data and retrieves their corresponding agent responses. These retrieved responses are

then paraphrased using a large language model to align them with the current user's style, tone, and dialogue context. This setup ensures that the system uses real, grounded examples while adapting surface form and phrasing to fit the user's current situation. Since all generation is anchored in previously used human responses, hallucination is minimized and the system's replies remain accurate and contextually aligned. RAP balances factual correctness, stylistic flexibility, and relevance without requiring expensive fine-tuning or continuous retraining.

Taken together, these two research directions offer complementary improvements to dialogue systems in client–agent settings. Topic forecasting enhances a system's ability to reason about future user needs, and controlled response generation improves trust and relevance in real-time communication. While developed and evaluated separately, both methods share the goal of building dialogue agents that are robust, adaptable, and reliable in messy, real-world conditions. By focusing on topic-based forecasting and retrieval-anchored generation, and grounding both in realistic datasets, this thesis contributes to more interpretable, practical, and deployable conversational AI systems.

Problem Definition and Motivation

Text-based client-agent communication is widely adopted in domains such as healthcare, coaching, and customer support. These platforms offer personalized, scalable assistance but introduce unique computational challenges due to the informal nature of user input, the dynamic evolution of user needs, and the necessity for accurate, grounded system responses. This thesis focuses on three key problems observed in such conversational environments and outlines the motivation for addressing each.

Forecasting future client topics is essential for improving both agent preparedness and system-level planning. In long-term interactions, such as those between a client and a dietitian, users tend to follow behavioral cycles involving diet, physical activity, motivation, and health tracking. Anticipating upcoming conversational themes allows the agent to prepare relevant suggestions, content, or reminders.

At a broader level, these predictions can help service providers cluster clients by their behavioral trajectories, allocate agent resources more effectively, and offer timely interventions. However, topic forecasting is a challenging problem due to the short, noisy, and irregular nature of user-generated messages and the high degree of personalization in topic flow.

Standardizing informal and noisy input is a critical prerequisite for reliable language understanding. In client-agent messaging systems, users frequently employ non-standard text including spelling errors, slang, abbreviations, and homophones. These variations can degrade the performance of NLP models by distorting token representations and semantic structure. Particularly for tasks such as topic modeling, classification, and retrieval, broken text can lead to unstable results and reduced interpretability. Therefore, applying a domain-independent standardization mechanism that improves text consistency without relying on task-specific supervision is vital for enabling accurate downstream modeling across a wide range of NLP tasks.

Generating grounded and trustworthy agent responses is one of the most important requirements for AI-powered conversational systems, especially in sensitive domains like health and wellness. Although large language models are capable of generating fluent and relevant responses, they often suffer from hallucination—producing content that is not supported by any underlying source or context. Retrieval-Augmented Generation (RAG) approaches help mitigate this by grounding the model in external knowledge; however, the generation step can still introduce errors or drift from the intended message. To reduce this risk, our approach constrains the generation process by retrieving a similar real response from past conversations and using an LLM only to paraphrase it in a client-aware manner. This strategy significantly lowers hallucination risk while maintaining relevance and stylistic adaptation, thereby increasing user trust and ensuring safer communication.

Together, these three problems reflect key challenges in building reliable, intelligent, and scalable dialogue systems. Addressing them forms the basis for the methods and contributions proposed in this thesis.

Contributions and Novelties

This thesis introduces three complementary yet independently applicable components that collectively enhance the robustness, personalization, and reliability of dialogue systems in client-agent communication settings. Each chapter presents a distinct methodological innovation, summarized as follows:

1. **Temporal Contextual Similarity (TCS).** This contribution presents a scalable and interpretable approach to forecasting future conversational dynamics by modeling the temporal evolution of topic distributions in client-agent dialogues. Conversations are transformed into time series of topic probabilities, and Gaussian Mixture Models (GMMs) are used to extract compact representations. A temporal similarity mechanism retrieves past client trajectories with analogous patterns to inform topic forecasts. The method is designed to grow incrementally as more data becomes available, with new replicas enriching the lookup dataset over time. Compared to widely used explainable forecasting baselines such as ARIMA and Prophet, SGSE offers improved directional accuracy and robustness. The framework is further structured to support multidimensional time series with varying sequence lengths, extending its applicability to domains such as stock market trend prediction and system monitoring.

2. **SALDIRAY.** This component introduces SALDIRAY, a transformer-based preprocessing framework designed to standardize noisy user-generated text before applying downstream NLP tasks. SALDIRAY targets typographic errors, slang, abbreviations, and homophones through a synthetic corruption pipeline that generates large-scale noisy–clean sentence pairs. These pairs are used to fine-tune normalization models without any task-specific supervision. SALDIRAY demonstrates strong improvements across various NLP tasks, including topic modeling, classification, and sentiment analysis, particularly under corrupted input conditions. The study also provides a detailed performance analysis on downstream robustness and offers a practical, domain-independent solution for linguistic normalization.

3. **Retrieval-Augmented Paraphrasing (RAP).** RAP is proposed as a strategy to reduce hallucinations in LLM outputs by constraining the generation process to grounded, previously validated human responses. For each incoming client message, RAP retrieves similar historical examples and prompts the LLM to paraphrase them according to the current context. This preserves semantic alignment, supports stylistic consistency, and significantly decreases the risk of hallucination. RAP outperforms zero-shot prompting, few-shot examples, and standard retrieval-augmented generation (RAG) baselines across multiple evaluation metrics, including response helpfulness, fluency, and relevance. The approach also introduces a promising method for constructing conversational knowledge bases with LLM validation cycles, which can further enhance response trustworthiness.

Each of these contributions is independently evaluated using domain-specific and publicly available datasets and task-appropriate metrics. Together, they offer a robust foundation for building real-world dialogue systems that can handle noisy inputs, model evolving user behavior, and produce grounded, context-sensitive responses.

Thesis Outline

This thesis is structured into four chapters, each addressing a key component in the development of robust, context-aware dialogue systems grounded in real-world interaction dynamics.

Chapter 2 introduces the Temporal Contextual Similarity (TCS) framework, a novel approach for forecasting the future direction of user-agent dialogues by analyzing topic distributions over time. Unlike prior semantic similarity-based methods, TCS captures the temporal evolution of conversations by fitting probabilistic models to topic frequency trajectories, enabling explainable and behaviorally grounded predictions about upcoming user intents. This capability is particularly useful in domains such as online dietitian services, where anticipating user concerns allows agents to offer timely and relevant guidance.

Chapter 3 presents SALDIRAY, a domain-independent, task-agnostic text standardization system designed to address orthographic and lexical noise in user-generated text. SALDIRAY targets a broad range of informal language phenomena—including abbreviations, homophones, misspellings, and slang—and learns to restore noisy input into canonical form using fine-tuned large language models. To facilitate effective training and evaluation, a novel synthetic noise generation pipeline is introduced, capable of producing diverse corruption types that simulate real-world digital communication. The impact of standardization is then assessed through a comprehensive benchmarking suite across several downstream NLP tasks, demonstrating significant improvements in robustness and topic modeling reliability.

Chapter 4 explores Retrieval-Augmented Paraphrasing (RAP), a prompt engineering strategy that improves language model responses by grounding generation in real, previously observed human replies. For each input, RAP retrieves semantically similar historical responses and constructs a context-rich prompt that instructs the model to paraphrase those responses in alignment with the current user's persona and conversation history. This approach effectively reduces hallucination and ensures responses are both factual and personalized, making it well-suited for deployment in sensitive or high-stakes dialogue environments.

Chapter 5 concludes the thesis by summarizing the major contributions of each chapter, synthesizing the overarching insights, and outlining directions for future works.

CHAPTER 2

TEMPORAL CONTEXTUAL SIMILARITY AND FORECASTING FOR CONVERSATIONAL LLMS

Forecasting the temporal evolution of conversational topics offers a structured perspective on the dynamics of long-term interactions. In real-world client-agent communication settings such as online dietitian consultations, the themes discussed by clients tend to evolve gradually over time. These thematic shifts often reflect behavioral changes, motivational fluctuations, and emerging needs. Anticipating future discussion topics in such environments can support proactive agent behavior, improve service personalization, and enable more efficient planning and monitoring of conversational workloads.

In this chapter, we introduce a forecasting framework that models the evolution of dialogue content through probabilistic topic trajectories. The central idea is to transform each client's multi-turn conversation into a set of time series, where each time series tracks the frequency of a particular semantic topic over time. These topics are obtained using unsupervised topic modeling techniques, and messages are labeled at the sentence level. To ensure consistency, individual sentences are first normalized using a standardization pipeline, and then mapped to topic categories that reflect domain-relevant themes such as diet, exercise, hydration, and motivation. For each dialogue, daily counts of topic mentions are aggregated to produce structured temporal sequences.

To model and compare these sequences, we adopt a Gaussian Mixture Model (GMM)-based representation. Each topic-specific time series is encoded using a fixed-length vector that captures the distributional shape of the series in terms of its mean and variance components. These GMM representations enable efficient

comparison between clients by allowing the system to retrieve previously observed dialogues with similar temporal profiles. Forecasting is performed by identifying the most temporally similar historical dialogues and aggregating their subsequent topic trends to estimate the near-future behavior of the target user. This structure allows the system to generalize across diverse user behaviors without requiring task-specific model training.

The proposed approach is evaluated on two datasets. The first is a large in-house corpus of dietitian–client WhatsApp conversations, where each dialogue spans several weeks or months and exhibits rich longitudinal behavior. The second is the public SHARE dataset, constructed from open-domain movie scripts and structured to support long-term memory-aware dialogue modeling. These datasets differ in style, length, and domain, providing a comprehensive testbed for evaluating the generalizability of the proposed forecasting framework.

By combining interpretable topic modeling with probabilistic temporal similarity, this chapter presents a scalable method for forecasting structured conversational trends. The framework is designed to support downstream applications such as agent preparation, early detection of behavioral changes, and conversation clustering by trajectory. The results suggest that modeling client dialogues as GMM-encoded topic sequences provides a viable pathway toward proactive and contextually grounded conversational systems.

2.1 Related Work

This section situates the proposed framework within three interconnected research streams: temporal similarity in time-series forecasting, temporal modeling in natural language processing, and probabilistic modeling of temporal patterns using Gaussian Mixture Models. Together, these areas inform the methodological and conceptual foundations of this study.

2.1.1 Temporal Similarity in Time-Series Forecasting

Temporal similarity has long been leveraged in forecasting scenarios where structural patterns across time provide predictive signal. Classical statistical models such as ARIMA have been widely applied to pandemic and epidemiological data [1], typically relying on Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) diagnostics for parameter selection. While effective under linear and stationary assumptions, these models often struggle to capture non-linear dynamics, regime shifts, and irregular temporal behavior observed in real-world data.

To address such limitations, hybrid and neural forecasting architectures have been proposed. For instance, [2] combined autoregressive components with nonlinear neural networks (NARANN) for COVID-19 case prediction, reporting improvements over purely statistical approaches. Autoencoder-based architectures have similarly been applied to epidemic modeling, where latent representations are learned from segmented time windows to forecast future case counts [3]. Deep learning approaches including Gated Recurrent Units (GRU), Long short-term memory models (LSTM), bidirectional LSTMs [4], and Convolutional Neural Network (CNN) and LSTM hybrids [5] further demonstrated the capacity of recurrent and convolutional models to capture complex temporal dependencies when sufficient training data is available. Transfer learning strategies with LSTM networks have also been explored to enhance generalization across heterogeneous entities [6].

Beyond parametric and neural forecasting models, representation-driven approaches have gained attention. In [7], Fourier decomposition combined with GMMs was used to capture diverse epidemic curve shapes, enabling multi-modal forecasting across countries. Such approaches emphasize structural similarity across entities rather than relying solely on direct autoregressive extrapolation.

The forecasting framework adopted in this chapter builds upon this line of research by explicitly modeling temporal trajectories through probabilistic representations and performing similarity-based estimation across entities. Instead of predicting solely from internal historical dynamics, the method leverages cross-entity struc-

tural resemblance to inform future evolution.

2.1.2 Temporal Context in Natural Language Processing

Temporal modeling in natural language processing has been extensively studied from multiple perspectives. In dialogue systems, evolving user intents and topic trajectories influence conversational coherence and personalization [8, 9, 10]. Sequential encoders such as LSTMs and Transformer architectures are commonly used to capture temporal dependencies within a session. However, these approaches primarily model intra-dialogue dynamics and rarely quantify structural similarity across different users' temporal topic trajectories.

Temporal information extraction has also been widely investigated beyond dialogue modeling. In the clinical domain, temporal relation extraction from unstructured health records has received substantial attention [11]. Systems in this area focus on identifying relations between events and document creation time, as well as event–event temporal dependencies. Recent advances emphasize attention-based models and contextualized representations, though challenges such as dataset imbalance and domain specificity remain prominent.

More broadly, temporal information extraction and categorization across document types including news articles, social media posts, and historical documents has been systematically reviewed in [12]. This work highlights multiple temporal dimensions such as event sequences, time periods, and socio-political occurrences, and surveys techniques ranging from rule-based systems to LLMs. These studies collectively demonstrate that temporal structure is a fundamental organizing principle across text genres.

While prior research primarily focuses on extracting and categorizing temporal information within documents, less attention has been given to modeling the evolution of topic distributions as explicit time-series signals and comparing their structural dynamics across entities. The present work addresses this gap by transforming conversational topic sequences into temporal trajectories and performing cross-dialogue similarity matching based on probabilistic and alignment-based

measures.

2.1.3 Gaussian Mixture Models in Temporal Pattern Analysis

GMMs have been widely used for probabilistic modeling and clustering of sequential data across domains. Their ability to represent multi-modal distributions makes them particularly suitable for capturing heterogeneous temporal behavior. GMM-based modeling has been applied in areas such as speaker verification [13], motion sequence analysis [14, 15], and epidemic curve representation [16].

In time-series contexts, GMMs provide a compact yet expressive representation of non-stationary and irregular temporal patterns. By fitting mixture components to segments of a trajectory, they can approximate complex shapes without requiring large numbers of parameters. When combined with divergence-based similarity measures such as Kullback–Leibler or Jensen–Shannon divergence, GMMs enable principled comparison of probabilistic temporal representations.

In this chapter, GMMs are used to model the evolution of conversational topic distributions over time. Each dialogue is represented through mixture-based approximations of its topic trajectories, and similarity between dialogues is computed through divergence-based metrics. This probabilistic representation serves as one of several alternative similarity strategies explored in the study, alongside Hidden Markov Model (HMM)-based cross-likelihood similarity and Dynamic Time Warping (DTW)-based alignment measures.

2.2 Methodology

This section presents the complete framework for forecasting conversational topic trajectories. The proposed pipeline consists of structured temporal representation of dialogue data, probabilistic modeling of topic evolution, similarity computation across dialogues, and neighbor-based forecasting. The framework is designed to capture structured temporal dynamics in conversational interactions and to estimate short-term topic evolution through cross-dialogue similarity.

2.2.1 Datasets and Preparation

The evaluation is conducted on two complementary dialogue corpora to assess both domain-specific performance and cross-domain generalizability. The primary dataset consists of real-world dietitian–client conversations collected through a nutrition coaching platform. To evaluate robustness beyond a structured health context, we additionally employ the publicly available SHARE dataset, which contains open-domain multi-turn dialogues. For both datasets, conversational content is transformed into structured temporal signals through topic modeling and temporal aggregation, yielding parallel time-series representations that form the basis of the forecasting framework.

2.2.1.1 Internal Dietitian–Client Dialogue Dataset

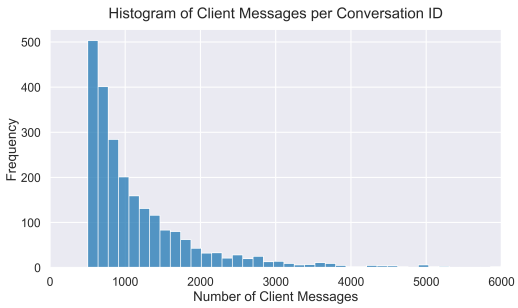
The primary dataset comprises approximately 2,500 WhatsApp conversations between clients and licensed dietitians. Dialogue durations range from 50 to 550 days, reflecting long-term behavioral interaction patterns. Engagement levels vary substantially across clients, as illustrated in Figure 2.1, both in terms of total message count and the number of distinct interaction days.

Messages are user-generated and contain natural linguistic variability, including abbreviations, spelling inconsistencies, emojis, and informal syntax.

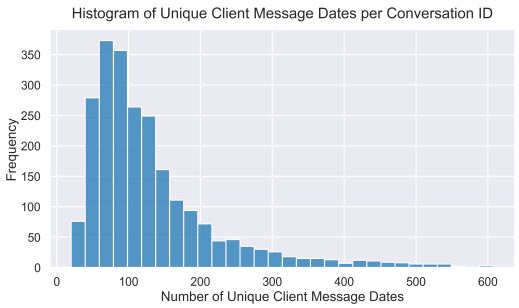
Topic labeling is performed using a domain-adapted BERTopic model trained with a nutritionist-seeded vocabulary to ensure semantic alignment with coaching practices. Fine-grained topics are subsequently consolidated into four high-level semantic categories: *dietary intake*, *physical activity*, *fluid consumption*, and *weight tracking*. Each dialogue is then temporally aggregated into daily bins, where the frequency of topic mentions per day defines four parallel time series per client.

2.2.1.2 SHARE Dataset for Long-Term Open-Domain Dialogue

To assess generalizability, the same methodology is applied to the SHARE dataset [17], which contains over 150 multi-turn dialogues derived from movie scripts.



(a) Distribution of total client messages per dialogue.



(b) Distribution of unique interaction days per dialogue.

Figure 2.1: Exploratory statistics for the dietitian–client dataset.

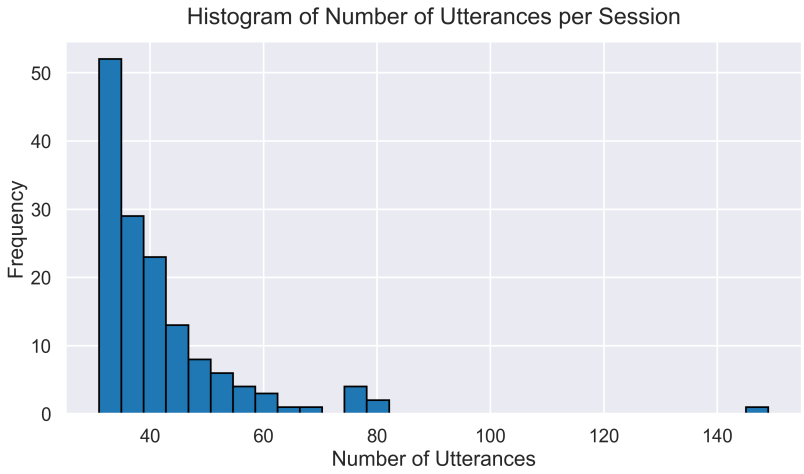


Figure 2.2: Distribution of utterance counts in the SHARE dataset.

These dialogues exhibit broader thematic variability and stylistic diversity compared to the structured dietitian corpus.

Figure 2.2 shows the distribution of utterance counts per dialogue, highlighting substantial variation in session length.

Topic labeling for SHARE is performed using the TANaOS topic classification model [18], pre-trained on open-domain conversational data. From the predicted labels, the four most frequently occurring topics across the corpus are selected. Unlike the dietitian dataset, where time is indexed by calendar day, SHARE dialogues are indexed by turn order, and topic frequencies are aggregated per dialogue turn.

In contrast to the dietitian dataset, for which a domain-specific topic model was

developed, we employ a publicly available pre-trained topic classification model for SHARE. This design choice is motivated by the nature of the SHARE corpus, which consists of movie dialogues covering broad, general conversational themes rather than specialized domain-specific content. Training a dedicated topic model is particularly beneficial when the corpus reflects a narrow semantic domain with specialized vocabulary, as in the dietitian dataset. However, for open-domain movie dialogues, topic structures largely align with general conversational categories already captured by large-scale public models trained on diverse data. Therefore, leveraging a widely used pre-trained topic classifier improves robustness, avoids overfitting to limited contextual cues, and ensures that extracted topics reflect broadly interpretable thematic categories. Using a well-established public model is thus both methodologically appropriate and empirically advantageous for this dataset.

2.2.2 Temporal Pattern Representation with Gaussian Mixture Models

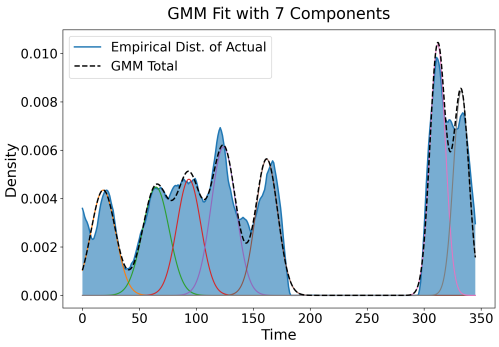
To model the temporal evolution of dialogue topics, each topic trajectory is represented using a Gaussian Mixture Model. The GMM approximates the distribution of topic frequencies over time as a weighted sum of Gaussian components [19], allowing multi-modal and non-stationary temporal behavior to be captured compactly.

Formally, a one-dimensional GMM $p(x)$ is defined as:

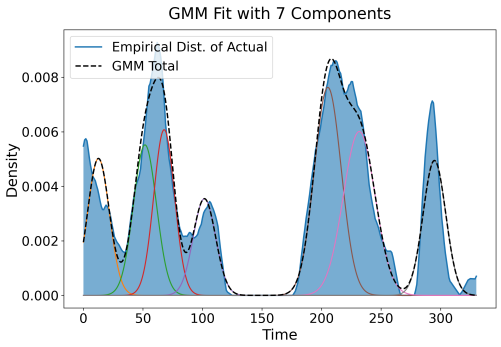
$$p(x) = \sum_{k=1}^{K} \omega_k \mathcal{N}(x \mid \mu_k, \Sigma_k), \tag{2.1}$$

where K denotes the number of mixture components, ω_k represents the mixture weight of the k^{th} component, and μ_k and Σ_k denote its mean and covariance.

Model parameters are estimated using the Expectation–Maximization (EM) algorithm [20], which iteratively maximizes the likelihood of the observed topic frequency sequence. The number of components K is determined using the Bayesian Information Criterion (BIC) [21], ensuring a balance between representational flexibility and overfitting. Across both datasets, BIC-based analysis consistently



(a) Client A: Smoothed "diet" topic frequency and corresponding GMM components.



(b) Client B: Smoothed "diet" topic frequency and corresponding GMM components.

Figure 2.3: Two example Gaussian mixture model (GMM) fits to smoothed topic frequency trajectories. The components reveal distinct temporal engagement modes within individual client consultations.

identifies $K = 7$ as the optimal configuration.

After fitting, the mixture parameters are serialized into a fixed-length vector representation by concatenating component means and variances ordered by ascending μ values:

$$v_i = \{\mu_{i1}, \sigma_{i1}, \mu_{i2}, \sigma_{i2}, \ldots, \mu_{iK}, \sigma_{iK}\}. \tag{2.2}$$

This compact representation enables efficient cross-dialogue similarity computation while preserving the structural characteristics of the underlying temporal pattern.

Figure 2.3 illustrates example GMM fits for two clients' topic trajectories. The mixture components capture distinct engagement phases, revealing how conversational emphasis shifts over time within individual consultations.

2.2.3 Cross-Dialogue Temporal Similarity Computation

To forecast topic trajectories, we construct a reference repository of temporal topic patterns across dialogues. Each conversation is represented as a time series reflecting topic frequencies over dialogue windows (e.g., days or turns), and each series is

modeled using GMMs as described previously.

To account for asynchronous conversational progressions across clients, we generate multiple temporal replicas from each dialogue by applying shifted cutoff points. Each replica represents a partial dialogue history and serves as a candidate state for similarity matching.

For the internal agent–client dataset, we extract $M = 5$ replicas per dialogue by shifting the cutoff point backward at fixed intervals. For the SHARE dataset, replicas are generated by shifting on a per-turn basis. This replication strategy increases robustness by capturing varying dialogue pacing and ensuring that different historical stages contribute to similarity matching. Each replica is encoded into its corresponding GMM representation vector, forming a dataset D of temporal dialogue states.

Figure 2.4 illustrates the replication and encoding process.

Let D denote the constructed dataset where each row corresponds to a replica of a dialogue encoded into GMM parameters:

$$D = \begin{Bmatrix} \mu_{111} & \sigma_{111} & \dots & \mu_{11k} & \sigma_{11k} \\ \mu_{121} & \sigma_{121} & \dots & \mu_{12k} & \sigma_{12k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mu_{i(j-1)1} & \sigma_{i(j-1)1} & \dots & \mu_{i(j-1)k} & \sigma_{i(j-1)k} \\ \mu_{ij1} & \sigma_{ij1} & \dots & \mu_{ijk} & \sigma_{ijk} \end{Bmatrix} \tag{2.3}$$

Here, i indexes the dialogue, j the replica, and k the GMM component. For example, $(\mu_{123}, \sigma_{123})$ denotes the mean and variance of the third component of the second replica of the first dialogue. This structure enables efficient similarity search across historical dialogue states.

For a target dialogue at a given time point, the most similar historical replicas in D are retrieved using a distance metric applied to their GMM vectors. We experiment with Euclidean distance to quantify similarity between temporal patterns.

The top $k = 20$ most similar replicas are selected to estimate future topic evolution.

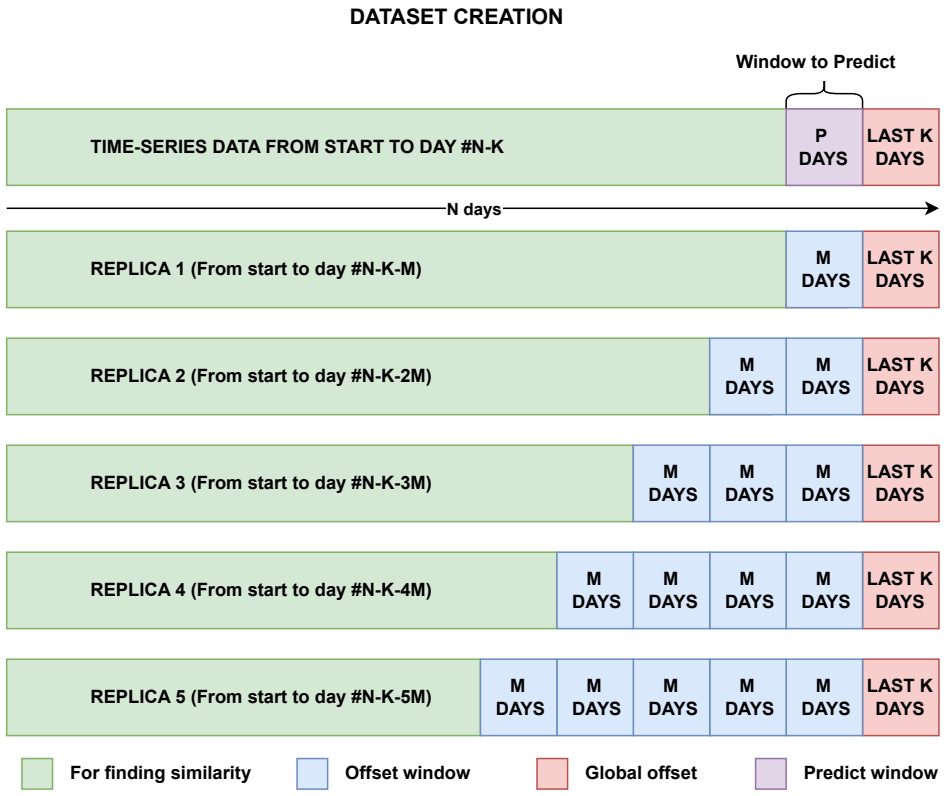


Figure 2.4: Dialogue state construction via shifted cutoff replication of topic time-series.

Since dialogue pacing differs across users, we perform velocity-based alignment. Let x_i and y_i denote the lengths (in days or turns) of the target history and a retrieved replica history, respectively. The pacing ratio is defined as:

$$r = \frac{|x_i|}{|y_i|} \tag{2.4}$$

To align temporal scales, we compute the number of steps to extract from the retrieved replica:

$$l = H \cdot r \tag{2.5}$$

where H is the forecasting horizon. The corresponding future segment is then interpolated and temporally rescaled to match the target dialogue's progression.

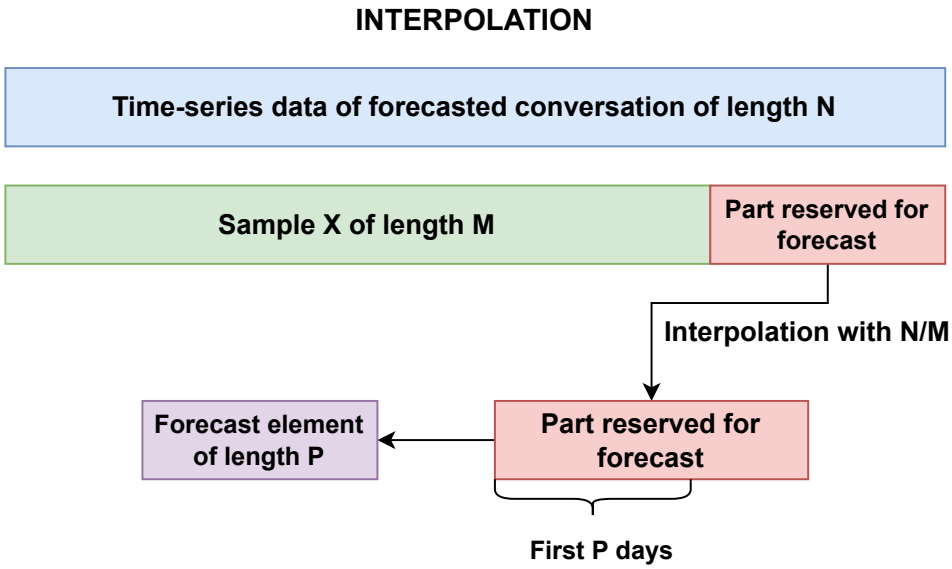


Figure 2.5: Forecasting via interpolation of temporally similar dialogue segments.

After interpolation, the predicted segment is additively adjusted to ensure continuity with the target trajectory. Instead of multiplicative scaling, we align the starting value of the forecasted segment to the final observed value of the target dialogue:

Finally, the aligned forecast segments from the $k = 20$ matched replicas are averaged elementwise to obtain the final prediction:

$$f_i = \frac{1}{k} \sum_{j=1}^{k} e_{ji} \tag{2.6}$$

where f_i denotes the forecasted topic frequency at time step i and e_{ji} corresponds to the i^{th} value of the j^{th} aligned replica forecast.

This similarity-driven, example-based forecasting mechanism provides interpretable predictions grounded in observed dialogue behavior and remains robust under heterogeneous pacing and non-stationary topic dynamics.

Algorithm 1 Additive Alignment of Forecast Segment

Require: x: target time-series, y: forecast segment, N: cutoff index

Ensure: Additively aligned forecast y

$x_N \leftarrow x[N]$

$y_0 \leftarrow y[0]$

$\delta \leftarrow x_N - y_0$

for $i = 0$ to $H-1$ **do**

    $y[i] \leftarrow y[i] + \delta$

end for

2.3 Experiments and Results

This section presents the empirical evaluation of the SGSE framework for forecasting topic evolution in dialogue systems. We conduct experiments on two distinct datasets: an internal collection of dietitian–client WhatsApp conversations and the publicly available SHARE dataset derived from movie dialogues. The forecasting task is to predict the distribution of dialogue topics over future time windows using a GMM-based temporal similarity mechanism. Each dataset requires domain-adapted preprocessing.

To ensure high-quality topic representations, we conducted an extensive hyperparameter search on the BERTopic models used for both datasets. The configurations were evaluated across three standard metrics: topic coherence (C_v), topic diversity, and topic coverage. The optimal model achieved a coherence score of *0.506*, diversity of *0.695*, and coverage of *0.864*, indicating a strong balance between semantic consistency and topical breadth. Additionally, to validate the interpretability and distinctiveness of the generated topics, we employed LLMs as an automatic judge. It was prompted to rate topic quality based on word distributions, confirming the human-aligned consistency of the discovered themes.

All forecasting models and supporting components were implemented using Python libraries. GMM, Holt's exponential smoothing, ARIMA, and DTW-based comparisons were implemented using *scikit-learn* and *statsmodels*, while HMM models were constructed using *hmmlearn*, and the Facebook Prophet model was imple-

mented with its official Python package.

This section begins by introducing the evaluation metrics used to compare forecasting performance. In addition to standard error metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Symmetric Mean Absolute Percentage Error (sMAPE), we also introduce the *Trend Similarity Score* (TSS), a directional agreement metric designed to assess the structural similarity of predicted and actual topic trajectories.

Following the metric definitions, we present the topic modeling results for our in-house dataset, including the distribution of discovered themes and their alignment with domain-relevant categories. The next subsection discusses the selection of hyperparameters for the GMM-based similarity module, including the number of mixture components and the value of k for retrieving the top-k most similar conversations.

We then move to the core forecasting evaluation. First, qualitative visualizations are provided to illustrate example similarity matches between clients and the temporal alignment of their topic sequences. Forecasted topic trajectories are plotted alongside ground truth sequences to highlight directional agreement. Finally, we report quantitative forecasting results across all baseline models and similarity variants using the defined error metrics. These experiments collectively demonstrate the effectiveness and interpretability of the proposed framework.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|x_i - y_i\right| \tag{2.7}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.8}$$

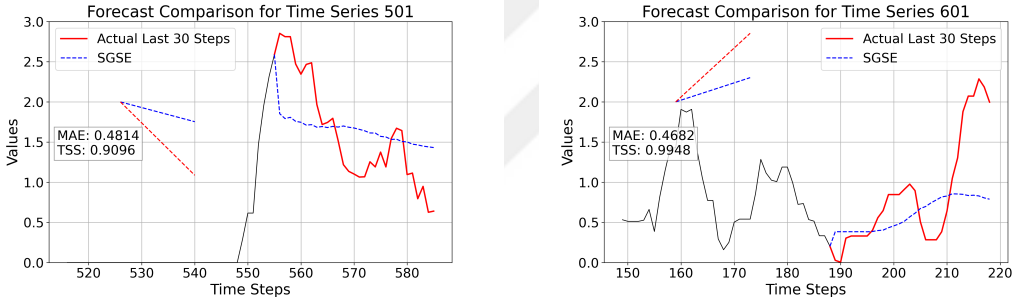
$$\text{sMAPE} = \frac{2}{n}\sum_{i=1}^{n}\frac{\left|x_i - y_i\right|}{|x_i| + \left|y_i\right|} \tag{2.9}$$

2.3.1 Trend similarity score

MAE, RMSE, and MAPE are more efficient than other metrics for observation of the exact difference between actual data and forecast results. However, while estimating how COVID-19 will continue, accurate trend estimation is more important than the net difference. For this reason, it is more meaningful for us to fit linear lines on real data and estimation results and look at their cosine similarity. At this point, a new error metric, the TSS, is presented and equation 2.10 shows how it is calculated:

$$TSS = \frac{T_1 \cdot T_2}{||T_1|| \times ||T_2||} \tag{2.10}$$

where T_1 and T_2 are the lines fitted on normalized actual and predicted data using linear regression. In Figure 2.6, the estimations and errors for two different conversations are shown:



(a) Forecast with lower TSS

(b) Forecast with higher TSS

Figure 2.6: MAE and TSS errors for two dialogues where MAE scores are similar but TSS scores are completely different.

MAE and TSS error values for Conversation 1 are presented in Figure 2.6a, and those for Conversation 2 are shown in Figure 2.6b. Although the MAE values of the two conversations are relatively similar, their TSS values differ noticeably. In particular, Conversation 2 achieves a higher TSS score, indicating that the overall trend is estimated more successfully compared to Conversation 1.

Since the data represents daily topic density, numerical differences in MAE do not always fully reflect trend estimation performance. TSS, on the other hand, better captures the alignment between predicted and actual trend dynamics. Therefore,

even when MAE values are close, differences in TSS reveal variations in how effectively the temporal evolution of daily topic density is modeled.

2.3.2 Topic Modeling for Dietitian–Client Dialogues

To model the thematic evolution in our internal WhatsApp-based dietitian–client communication corpus, we developed a domain-adapted BERTopic pipeline. Prior to topic modeling, we applied a preprocessing step that merges temporally proximate utterances: consecutive client messages sent within a one-minute window were combined into a single utterance to form more semantically coherent dialogue turns.

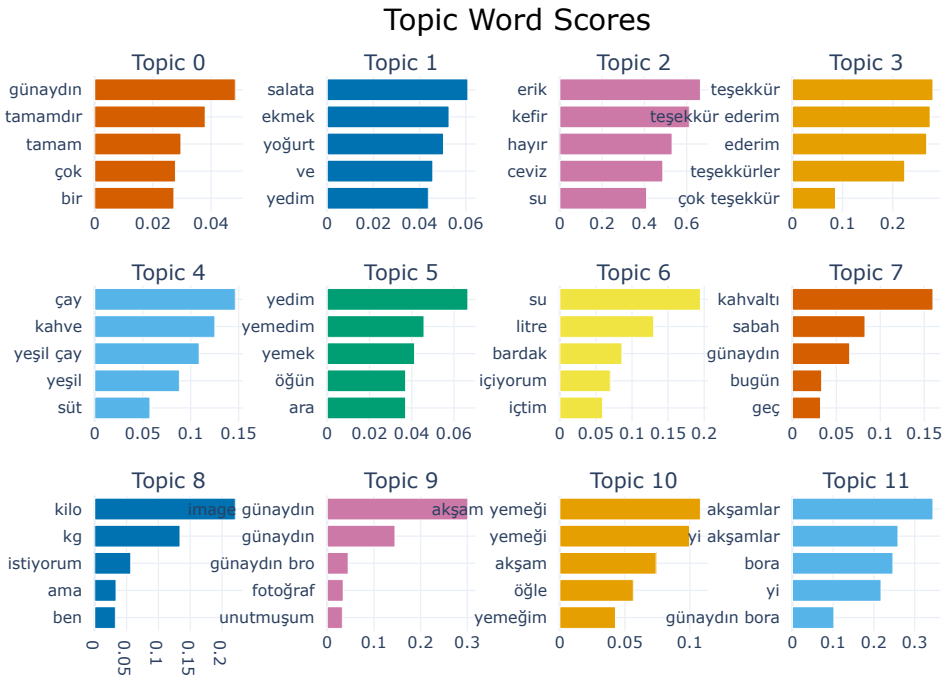


Figure 2.7: Top representative keywords for a sample of discovered topics.

A sample of 20,000 client messages was selected for model training. To achieve optimal topic segmentation, we conducted an extensive hyperparameter search across key components of the BERTopic framework:

- **UMAP**: `n_neighbors`, `n_components`, `min_dist`
- **HDBSCAN**: `min_cluster_size`, `min_samples`
- **Vectorizer**: `ngram_range`, `min_df`, `max_df`

Each configuration was evaluated using three complementary metrics: **topic coherence**, **semantic diversity**, and **corpus coverage**. The best-performing configuration, offering the highest balance of interpretability and inclusivity, was selected as the final model.

To further validate the semantic plausibility of predicted labels, we employed LLMs as automatic evaluators. A total of 1,000 client sentences were randomly sampled from the corpus, along with their model-predicted topics and associated keywords. Each example was scored by two distinct LLMs, each prompted to rate the suitability of the predicted topic on a 1–5 scale. The average scores achieved were **4.235** and **4.005**, respectively, indicating strong human-aligned agreement. Moreover, the inter-rater consistency between the two models yielded a Cohen's kappa score [22] of **0.63**, suggesting substantial reliability and consistency across models.

Following label prediction, we manually grouped the resulting topics into four clinically and behaviorally meaningful macro-categories, created in consultation with domain experts:

- **Diet**: meal planning, food selection, portion control
- **Exercise**: fitness, walking and all kind of sport activities
- **Water and Beverage**: hydration tracking, tea/coffee intake
- **Weight**: weight monitoring

The semantic structure and separation of topics are illustrated through hierarchical clustering in Figure 2.8, based on their c-TF-IDF representations. Additionally, Figure 2.7 shows the most prominent keywords associated with a sample of discovered topics, confirming thematic granularity and consistency.

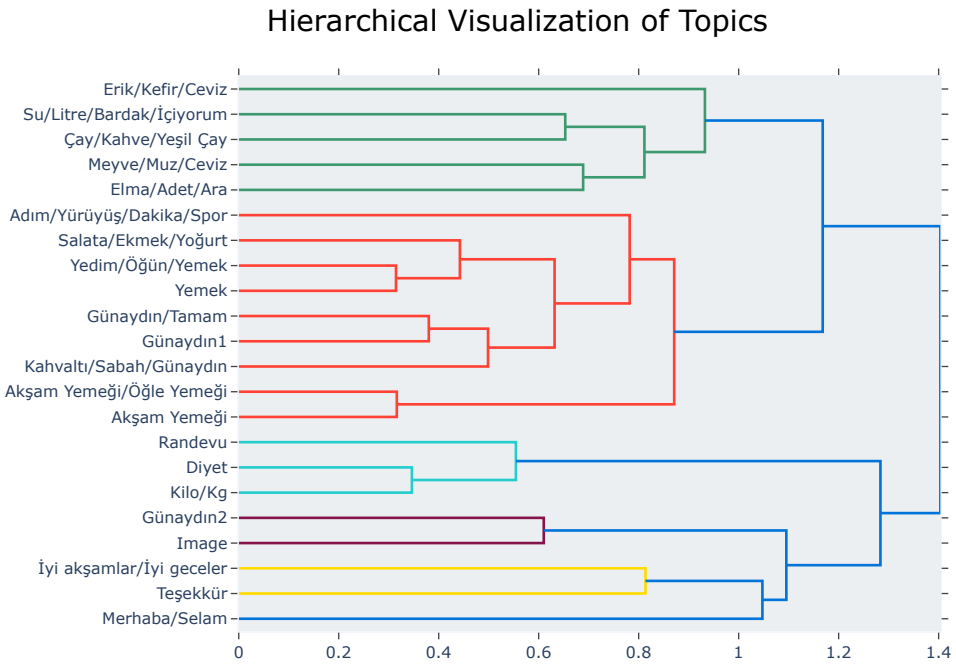
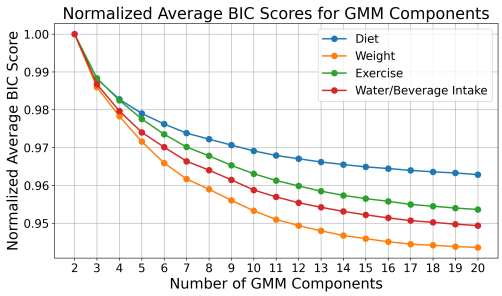


Figure 2.8: Hierarchical clustering of discovered topics based on their c-TF-IDF representations.

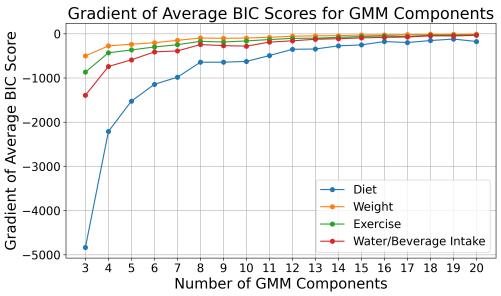
Each client's predicted topic labels were then aggregated into daily counts to construct per-client temporal topic sequences. These sequences form the basis for time-series modeling in the forecasting framework presented in subsequent sections.

2.3.3 Model Selection

The SGSE framework was applied to the daily-aggregated topic trajectories of all clients. Forecasts were generated for a 30-day horizon, predicting the future distributions over the four primary topic categories. Model selection was conducted in two stages: (i) selecting the number of GMM components and (ii) determining the optimal number of neighbors in the trajectory comparison step.



(a) Average BIC scores across dialogues for each topic.



(b) Gradient of the average BIC scores, illustrating diminishing returns as the number of components increases.

Figure 2.9: Model selection for GMM components using averaged BIC analysis.

2.3.3.1 Selection of GMM Components

To determine the appropriate number of Gaussian components, we applied the Bayesian Information Criterion (BIC) independently to each dialogue trajectory. For each topic, GMMs with varying component counts were fitted and their BIC scores computed. The BIC values were then averaged across all dialogues within the same topic to obtain a stable estimate of model quality.

Figure 2.9 presents (i) the average BIC scores across component counts and (ii) the gradient of the average BIC curve for all topics. The gradient analysis highlights the point at which marginal improvements diminish, providing additional evidence for selecting expressive number of components. Based on the values observed in both plots, we selected 7 as the optimal number of components for the GMM representation.

2.3.3.2 Selection of Number of Neighbors

The number of neighbors used in the SGSE forecasting mechanism was selected via hyperparameter search. For each candidate neighbor count, performance was evaluated by averaging results across multiple GMM configurations. Specifically, for a fixed neighbor count (e.g., 15 neighbors), the reported score corresponds to the average performance obtained from models using different GMM component counts

Table 2.1: Grid search results for neighbor selection. Best (lowest) value per metric is highlighted in bold.

Topic	Metric	k=3	k=5	k=10	k=15	k=20
Diet	MAE	0.4886	0.4873	0.4853	0.4843	**0.4830**
	RMSE	0.5854	0.5831	0.5807	0.5795	**0.5780**
	sMAPE	**0.6961**	0.6946	0.7008	0.6995	0.7001
Weight	MAE	0.1033	0.0984	0.0860	0.0815	**0.0802**
	RMSE	0.1322	0.1241	0.1082	0.1025	**0.1007**
	sMAPE	1.2125	**1.1807**	1.2062	1.1997	1.1941
Exercise	MAE	0.1470	0.1396	0.1281	0.1224	**0.1197**
	RMSE	0.1801	0.1701	0.1566	0.1501	**0.1469**
	sMAPE	1.0030	0.9674	0.9400	**0.9353**	0.9371
Water Intake	MAE	0.1903	0.1869	0.1830	0.1825	**0.1820**
	RMSE	0.2352	0.2299	0.2248	0.2235	**0.2228**
	sMAPE	1.0317	**1.0203**	1.0311	1.0364	1.0334

(e.g., 3 components with 15 neighbors, 4 components with 15 neighbors, etc.). This aggregation reduces sensitivity to a single GMM configuration and provides a more robust estimate of neighbor performance.

Table 2.1 summarizes the average cosine similarity scores for different neighbor counts after averaging across GMM component settings.

2.3.4 Forecasting Results on Dietitian-Client Dataset

To evaluate the performance of SGSE, we compared it against three well-established and interpretable time-series forecasting models: ARIMA [23], Facebook Prophet [24], and Holt's exponential smoothing method [25]. These models were selected due to their transparent assumptions and explainable trend and seasonality components. In contrast to complex black-box predictors, they offer clear interpretability, making them suitable baselines for evaluating SGSE's performance improvements.

Figure 2.10 illustrates the SGSE forecasting process on a representative client trajectory. The query conversation is visualized alongside the four most similar source conversations identified through GMM-based similarity. The final prediction is formed by temporally aligning and aggregating the future segments of these retrieved sequences.

Quantitative forecasting results are presented in Table 2.2. SGSE with GMM representations consistently outperforms both alternative SGSE variants (HMM and DTW) and all classical baselines across most evaluation metrics, including MAE, RMSE, sMAPE, and our proposed Trend Similarity Score (TSS). The GMM-based SGSE variant demonstrates the most robust performance, offering both lower error rates and stronger directional alignment with ground truth topic trajectories. Particularly in high-frequency themes such as *Diet* and *Exercise*, SGSE achieves substantial error reductions and higher TSS, indicating that its neighbor-based strategy effectively captures real-world behavioral dynamics in client-agent conversations.

2.3.5 Forecasting Results on SHARE Dataset

We further tested SGSE on the open-domain SHARE dataset to assess its generalizability beyond the dietitian setting. Dialogue utterances were labeled with topic categories using the TANaOS classifier, and each dialogue was segmented into time series based on message turns. SGSE was tasked with forecasting the topic distributions over the next 7 turns, and its performance was compared to the same baselines used in the previous evaluation.

As shown in Table 2.3, SGSE again delivers the best performance across nearly all metrics. While traditional models perform reasonably on smoother topic curves (e.g., *Health* and *Finance*), SGSE excels particularly in forecasting nuanced shifts in conversations, as demonstrated by its superior TSS and lower errors in categories like *Education & Learning* and *Relationships & Dating*. These results confirm that SGSE's similarity-driven design is not only effective in structured domain-specific settings but also generalizes well to diverse conversational domains with different temporal and topical dynamics.

Table 2.2: Forecasting results on the Dietitian-Client dataset. For error-based metrics (MAE, RMSE, and sMAPE), the lowest values are highlighted in bold. For Trend Similarity Score (TSS), the highest value is highlighted in bold.

Topic	Metric	SGSE - GMM	SGSE - HMM	SGSE - DTW	ARIMA	HOLT	Prophet
	Time (s/it)	0.092	0.405	30.11	0.133	**0.012**	0.114
Diet	MAE	**0.4837**	0.5455	0.5081	0.6447	0.7022	0.5849
	RMSE	**0.5786**	0.6469	0.6142	0.7620	0.8221	0.6806
	sMAPE	**0.6966**	0.7574	0.7234	0.8792	0.9067	0.8316
	TSS	**0.9734**	0.9701	0.9688	0.9325	0.9067	0.9449
Weight	MAE	**0.0749**	0.0902	0.0869	0.0935	0.0918	0.0792
	RMSE	**0.0947**	0.1100	0.1071	0.1136	0.0941	0.0969
	sMAPE	1.1840	**1.0952**	1.2189	1.1461	1.2458	1.0966
	TSS	**0.9699**	0.9501	0.9520	0.9493	0.9510	0.9646
Exercise	MAE	**0.1141**	0.1318	0.1269	0.1340	0.1377	0.1332
	RMSE	**0.1409**	0.1595	0.1552	0.1631	0.1674	0.1583
	sMAPE	**0.9323**	1.0041	1.0068	1.0192	1.0145	1.0026
	TSS	**0.9720**	0.9703	0.9716	0.9601	0.9574	0.9710
Water In.	MAE	**0.1758**	0.1974	0.1890	0.2030	0.2052	0.1866
	RMSE	**0.2146**	0.2401	0.2324	0.2469	0.2515	0.2259
	sMAPE	1.0220	1.0593	1.0728	1.1490	1.1217	**1.0188**
	TSS	**0.9557**	0.9510	0.9505	0.9321	0.9267	0.9490

2.4 Conclusion and Discussion

In this chapter, we introduced the SGSE framework for interpretable topic forecasting in long-term, agent–client dialogues. Originally developed in the epidemiological forecasting domain, SGSE was successfully adapted to model the temporal evolution of dialogue topics across conversations. By representing clients' topic histories as GMMs and identifying structurally similar topic trajectories across dialogues, SGSE offers a novel, explainable alternative to black-box time series forecasting models.

Unlike traditional sequence-to-sequence or autoregressive neural models that require large volumes of labeled data and often lack interpretability, SGSE relies on unsupervised clustering of topic evolution patterns and allows meaningful, case-

based analogical reasoning for forecasting. Through this lens, future conversational topics can be forecasted by referencing structurally similar historical conversations, making the method inherently transparent and robust to domain drift. Empirical evaluation across both in-house and public dialogue datasets demonstrated its ability to generate plausible topic forecasts, often outperforming baseline models like ARIMA, Prophet, and Holt's exponential smoothing.

One of the key advantages of SGSE lies in its balance between accuracy and interpretability. By modeling client-topic interactions using a small number of GMM components and selecting nearest neighbors in a meaningful latent space, it becomes possible to forecast not only the most probable topics, but also understand why those forecasts arise, offering insight into user behavior trends and system decisions.

Implementation Guidance for SGSE Forecasting For practitioners aiming to apply Shifted Gaussian Similarity Estimation (SGSE) on time series data, the core modeling step involves fitting a Gaussian Mixture Model (GMM) on representative samples derived from time series segments. Since GMMs operate on discrete samples rather than sparse numerical time series, the original time series must be transformed into a sample-based representation. This is achieved by converting each timestamp's value into that many sample points—i.e., a count-based expansion of the time series. For example, a value of 5 at time t becomes 5 duplicated samples located at t. If the time series contains small values (e.g., counts below 5), amplification is recommended prior to sample generation to ensure enough density for a stable GMM fit.

After forecasting future components using similar historical topic trajectories, the resulting forecast vectors must be anchored to the original time series. Two alignment strategies are supported: (1) Multiplicative anchoring, where the forecast vector is scaled by the ratio of the last observed value to the first forecasted value, and (2) Additive anchoring, where the forecast vector is offset by the difference between the last actual value and the first predicted one. The multiplicative approach is recommended for stationary or scale-invariant series, as it preserves proportional dynamics. However, for non-stationary data—where abrupt shifts and drifts are

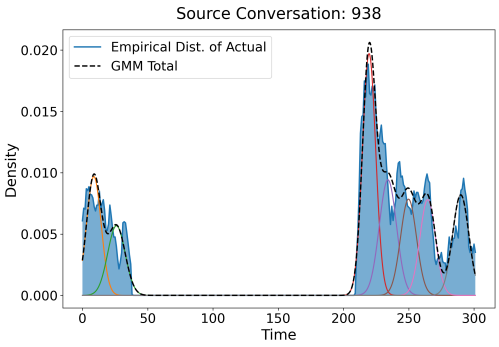
present—multiplicative anchoring may lead to vanishing or exploding values; in such cases, additive alignment provides a more stable forecast integration.

This transformation and post-processing logic ensures that SGSE can be flexibly applied across varying domains while maintaining structural coherence between past and projected trajectories.

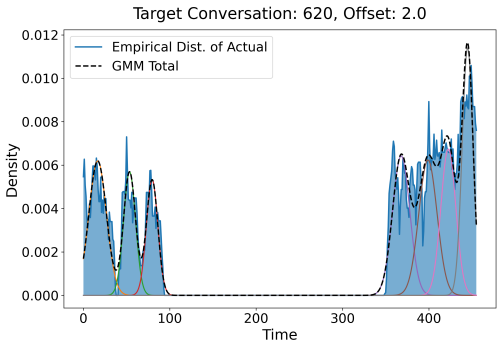
Future work and Conclusion Future work will explore the extension of this framework beyond single-dimensional topic sequences. In particular, we plan to apply SGSE to multivariate time series settings, capturing the joint evolution of multiple behavioral signals such as topic entropy, sentiment, engagement level, and temporal density. This multidimensional representation will further enrich client modeling and forecasting fidelity.

Moreover, while our experiments focus on conversational health coaching data, the architecture of SGSE is domain-agnostic and lends itself naturally to other domains involving long-range behavioral or system-level forecasting. Notable examples include financial markets (e.g., stock or cryptocurrency trend prediction), industrial time series monitoring (e.g., server usage patterns, anomaly detection), and user engagement analytics in digital platforms. In these domains, explainability, few-shot generalization, and cross-entity transferability are as critical as predictive performance—making SGSE an attractive, lightweight alternative to complex deep learning models.

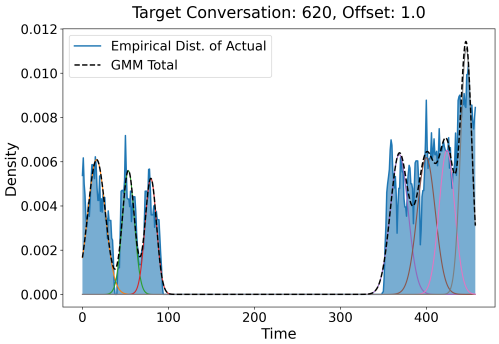
In conclusion, this chapter demonstrates how temporally-aware and semantically-grounded modeling of conversational trajectories can support anticipatory dialogue planning. The proposed framework opens up new avenues for scalable, interpretable forecasting in both dialogue systems and broader time-series contexts.



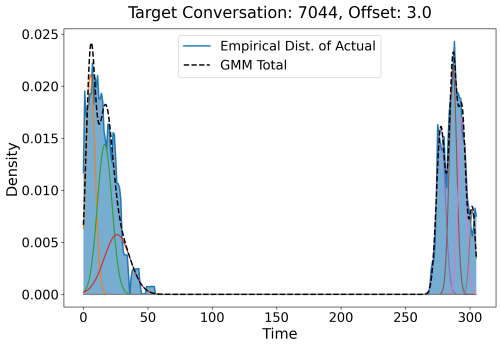
(a) Source conversation trajectory.



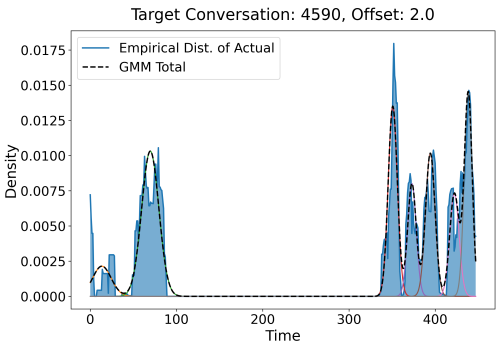
(b) Closest target conversation.



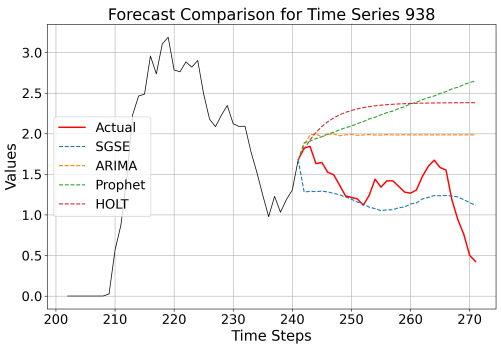
(c) Second closest target conversation.



(d) Third closest target conversation.



(e) Fourth closest target conversation.



(f) Forecasted topic distribution for the source conversation.

Figure 2.10: Illustration of the SGSE forecasting process. The source conversation is matched with its four closest neighbors (with temporal offsets), and their aligned future trajectories are aggregated to generate the final 30-day forecast.

Table 2.3: Forecasting results on the SHARE dataset. For error-based metrics (MAE, RMSE, and sMAPE), the lowest values are highlighted in bold. For Trend Similarity Score (TSS), the highest value is highlighted in bold.

Topic	**Metric**	**SGSE**	**ARIMA**	**HOLT**	**Prophet**
Education & Learning	MAE	**0.0307**	0.0414	0.0435	0.0423
	RMSE	**0.0358**	0.0468	0.0495	0.0475
	sMAPE	**0.3931**	0.4663	0.4940	0.5024
	TSS	**0.9777**	0.9730	0.9632	0.9534
Health	MAE	**0.0335**	0.0355	0.0371	0.0423
	RMSE	**0.0388**	0.0404	0.0426	0.0468
	sMAPE	0.6532	**0.5612**	0.5644	0.6847
	TSS	**0.9562**	0.9543	0.9497	0.9102
Money & Finance	MAE	**0.0420**	0.0470	0.0485	0.0518
	RMSE	**0.0485**	0.0526	0.0554	0.0574
	sMAPE	0.6819	**0.6292**	0.6686	0.7382
	TSS	**0.9396**	0.9300	0.9239	0.9165
Relationships & Dating	MAE	**0.0510**	0.0735	0.0725	0.0773
	RMSE	**0.0588**	0.0822	0.0820	0.0857
	sMAPE	**0.3728**	0.4984	0.5094	0.5726
	TSS	**0.9832**	0.9787	0.9780	0.9717

CHAPTER 3

SALDIRAY: SCALABLE AND ADAPTIVE LANGUAGE DIAGNOSTICS FOR REMEDIATION OF ABBREVIATIONS AND TYPOS

Natural language processing has advanced rapidly in recent years, enabling strong performance on a wide range of tasks such as text classification, topic modeling, sentiment analysis, natural language inference (NLI), and fact verification [26, 27]. Much of this progress has been driven by transformer architectures and large language models (LLMs), often adapted to particular domains through fine-tuning. However, models that perform well on curated benchmarks can degrade substantially when deployed on user-generated text that contains misspellings, abbreviations, slang, orthographic variation, and homophonic substitutions [28, 29, 30]. Prior work has shown that such perturbations can reduce classification accuracy, distort similarity judgments, and impair downstream reasoning, in part because corrupted tokens propagate through contextual representations in transformer-based models [31]. This robustness gap is especially consequential in social media, online forums, and conversational interfaces, where non-standard language is prevalent and rapidly evolving, and it is further amplified in low-resource settings [32, 33, 34, 35]. Improving robustness to real-world textual variability therefore remains a practical barrier to reliable NLP deployment.

Handling noisy inputs has also been studied in adjacent speech-to-text pipelines, where post-transcription correction is used to mitigate recognition errors [36, 37]. While effective for acoustically induced mistakes, these methods are typically tailored to surface-form correction and do not directly address the broader range of lexical and semantic variability found in text-first digital communication. In online writing, noise frequently reflects deliberate or community-specific conventions

(e.g., creative spellings, emergent slang, or homophonic substitutions) rather than transcription artifacts. This setting calls for normalization that extends beyond character-level editing toward consistent lexical standardization. At the same time, many existing normalization approaches are domain-specific or trained with task-dependent supervision, which limits their transferability across heterogeneous applications and model families.

In this work we introduce SALDIRAY, a task-agnostic preprocessing framework for linguistic standardization under noisy user-generated text. SALDIRAY targets common noise types, including misspellings, abbreviations, slang, and homophones, without relying on task-specific labels. The framework constructs large-scale paired noisy–clean data by applying a synthetic corruption ("broking") pipeline to clean corpora, combining controlled lexical perturbations with curated homophone and slang resources. Sequence-to-sequence normalization models are then fine-tuned on these pairs and used as a modular inference-stage component that transforms ill-formed inputs into a more canonical representation prior to downstream processing.

We evaluate SALDIRAY with an architecture-agnostic protocol. For each task, we start from clean benchmark datasets, generate corrupted variants using the proposed broking strategies, normalize the corrupted inputs with SALDIRAY, and then re-evaluate the original task models on the resulting standardized text. We consider a diverse set of settings, including topic modeling with BERTopic, sentiment analysis using BERT-based classifiers, NLI with transformer-based inference models, and fact verification using a language model. This design allows us to assess whether a single normalization layer can improve robustness across both conventional discriminative models and knowledge-intensive, instruction-tuned LLM pipelines.

Empirically, SALDIRAY improves performance under noisy conditions across the evaluated tasks, with larger gains as noise severity increases. These results support the view that explicit linguistic standardization, implemented as a modular preprocessing step, can be a scalable mechanism for increasing robustness without modifying downstream architectures. We also discuss limitations and failure modes

that arise when normalization changes meaning, and we outline directions for improving generalization beyond synthetic noise.

The main contributions of this paper are as follows:

- We propose SALDIRAY, a task-agnostic linguistic standardization framework that jointly addresses misspellings, abbreviations, slang, and homophones, positioning normalization as a broader lexical standardization problem rather than a narrow spelling-correction task.

- SALDIRAY operates as a modular preprocessing component that can be integrated into diverse NLP pipelines without modifying downstream models, enabling consistent robustness improvements across domains, tasks, and model families.

- We introduce a text broking procedure that combines probabilistic character-level and lexical-level perturbations using curated homophone and slang resources, producing large-scale paired noisy–clean training data for normalization.

- We design an evaluation pipeline that systematically studies how corrupted inputs affect heterogeneous NLP systems and demonstrates that a single standardization layer can mitigate performance degradation across both conventional transformer models and instruction-tuned LLM workflows.

Together, these contributions frame linguistic standardization as a scalable mechanism for improving robustness to real-world textual variability without task-specific supervision.

The remainder of this paper is organized as follows. Section 3.1 reviews prior work on noise modeling and text normalization. Section 3.2 describes the SALDIRAY framework, including the corruption strategies and normalization model training. Section 3.3 presents the experimental setup and results. Section 3.4 concludes and outlines future work.

3.1 Literature Review

The problem of textual noise and orthographic variability has been extensively studied across several NLP subfields, including spelling correction, robustness analysis, and synthetic data generation. Prior work spans both traditional rule-based methods and modern neural architectures, with increasing attention given to the impact of misspellings and informal language on the performance of large-scale pretrained models. This section reviews three main research directions that form the foundation of the present study: advances in spelling correction models, the impact of misspellings on NLP robustness, and synthetic spell error generation for data augmentation.

3.1.1 Advances in Spelling Correction Models

Spelling correction has long been investigated as a means of improving both human–computer interaction and the reliability of downstream NLP applications. Early approaches were primarily based on edit distance, hand-crafted rules, and dictionary-driven lookup mechanisms. With the emergence of neural architectures, however, the field has shifted toward data-driven models capable of learning complex error patterns directly from examples. In speech-to-text pipelines, spelling correction is commonly implemented as a post-processing stage to mitigate automatic speech recognition (ASR) errors. In this context, neural sequence-to-sequence models have been shown to be particularly effective in mapping noisy ASR hypotheses into fluent text. Guo et al. [38] proposed an RNN-based ASR correction model trained on synthetically generated noisy–clean transcript pairs, while Hrinchuk et al. [37] demonstrated that Transformer-based correctors initialized with BERT more accurately capture ASR-specific error distributions.

With the rise of LLMs, spelling and grammatical correction have become increasingly robust and multilingual. Xu et al. [39] showed that GPT-based models can effectively rescore and refine low-quality ASR transcripts, yielding notable improvements in conversational child speech. Luhtaru et al. [40] further demonstrated that GPT-style models can rival dedicated grammatical error correction systems, partic-

ularly in multilingual and low-resource scenarios. Turhan [35] specifically investigated spelling correction for Turkish, an agglutinative language, and reported that fine-tuned LLMs outperform encoder-only BERT architectures and cross-lingual models in this setting.

Büyük and Arslan [32] highlighted the importance of correction for low-resource languages by introducing a dual-model framework consisting of a generator that produces human-like Turkish misspellings and a sequence-to-sequence corrector trained using the resulting augmented data. Klemen et al. [33] showed that training on synthetically corrupted Slovene text significantly improves correction precision and recall over traditional word-level approaches. Aytan and Şakar [41] proposed a Turkish spelling detection model based on byte-pair encoding and a false-positive reduction mechanism designed to handle foreign words and abbreviations more effectively.

The broader downstream utility of LLM-based spelling correction was demonstrated by Allamong et al. [42], who applied GPT-4o to correct misspellings in open-ended survey responses from the American National Election Studies (1996–2020). Their approach successfully corrected 85–90% of human-identified errors, reducing token sparsity and improving dictionary-based sentiment coverage. Using a support vector machine classifier, they further showed that training on the corrected dataset led to higher downstream prediction accuracy compared to the original uncorrected data.

3.1.2 Misspellings and Their Impact on NLP Robustness

Orthographic errors and non-standard spellings are known to adversely affect a wide range of NLP tasks. Even minor lexical perturbations can significantly degrade system performance in machine translation, sentiment analysis, and semantic similarity estimation. Belinkov and Bisk [31] demonstrated that injecting artificial misspellings into input text results in severe quality degradation in neural machine translation. Several studies have shown that preprocessing pipelines incorporating spelling and grammar correction can partially mitigate these effects. For example, Horbach et al. [43] reported increased robustness in automated student essay scor-

ing when character-level features and spell correction were included. In the clinical domain, Kim et al. [44] emphasized that uncorrected spelling errors can lead to diagnostic misclassification and missed clinical information.

These vulnerabilities extend to modern LLMs. Wang et al. [45] introduced AdvGLUE, demonstrating that character-level perturbations such as adversarial misspellings significantly reduce the accuracy of models including BERT, RoBERTa, and T5. Zhang et al. [46] further showed that GPT-4 exhibits measurable performance degradation under adversarial misspelling attacks. Beyond synthetic perturbations, naturally noisy language phenomena such as slang, abbreviations, and code-mixed text also pose persistent challenges. Wuraola et al. [47] documented systematic failures of LLMs in consistently interpreting slang-heavy or culturally specific informal expressions.

Collectively, these findings indicate that although LLMs possess strong generalization and correction capabilities, they remain highly sensitive to orthographic variation and adversarial perturbations. Techniques such as synthetic error augmentation [48], cascaded correction strategies, and morphology-aware modeling for agglutinative languages [35] represent essential steps toward improving the robustness of NLP systems in the presence of real-world lexical noise.

3.1.3 Synthetic Spell Error Generation

Synthetic error generation plays a critical role in training spelling correction systems, particularly in scenarios where large-scale annotated corpora are scarce or unavailable. Rather than relying on naive random perturbations, recent studies increasingly focus on emulating human error patterns to produce more realistic and task-relevant training data.

Stahlberg and Kumar [49] introduced an error-tag-guided corruption framework that applies explicit error types, such as insertions and homophone substitutions, to corrupt clean sentences in a controlled manner. This strategy enables grammatical error correction systems to better match the empirical distribution of real-world mistakes. Martynov et al. [34] proposed SAGE, a framework that applies statistically

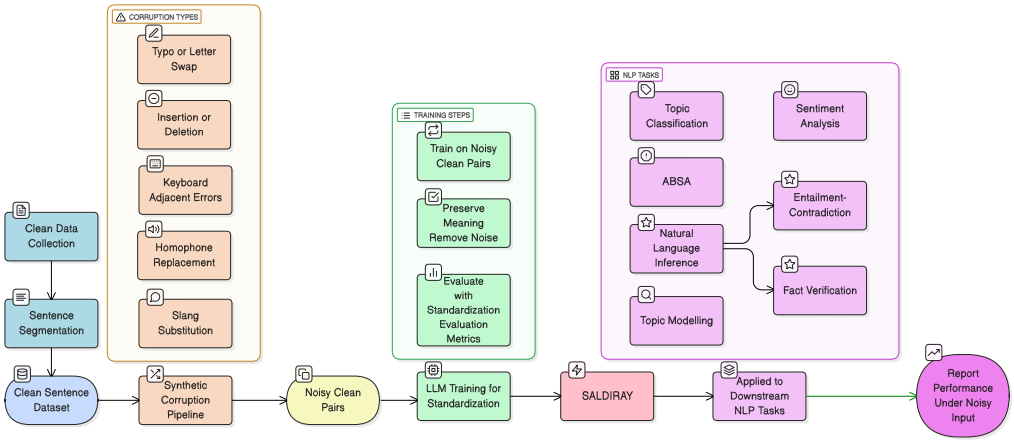


Figure 3.1: Example NLP application pipeline using the SALDIRAY framework as a preprocessing module. The figure shows the flow from noisy input through SALDIRAY-based standardization to downstream task inference

grounded noise and heuristic rules to emulate misspellings, achieving improved generalization across multilingual benchmarks. Similarly, Sifat et al. [50] developed a synthetic corruption method based on QWERTY keyboard typing patterns for Bengali, with a focus on frequent keypress errors and phonetic substitutions.

In low-resource settings, the generation of realistic synthetic noise is particularly critical. Büyük and Arslan [32] trained a generator on human-created Turkish typos to synthesize plausible misspellings, significantly improving the performance of their neural correction model. Klemen et al. [33] likewise emphasized that the choice of synthetic error generation strategy directly and substantially influences the success of neural spell-checkers in Slovene.

Prior work consistently highlights the importance of realistic, language-aware synthetic error generation, particularly for multilingual and under-resourced NLP scenarios. These findings directly motivate the synthetic text-broking strategy employed in the proposed SALDIRAY framework.

3.2 Methodology

The SALDIRAY framework is a domain-independent text standardization methodology designed to detect and remediate abbreviations, typographical errors, ho-

mophones, slang, and other sources of orthographic and lexical noise commonly found in user-generated content. Its primary objective is to transform noisy and informal inputs into standardized textual representations that improve the reliability of downstream NLP systems. Rather than constituting a single fixed workflow, SALDIRAY defines a modular approach that combines synthetic noise generation, architecture-agnostic model adaptation, and deployment as a reusable preprocessing component. In practice, the framework can be integrated into a wide spectrum of NLP environments, from traditional statistical models to large language models, without requiring task-specific supervision or modifications to downstream architectures. Figure 3.1 illustrates a representative application pipeline in which the SALDIRAY framework operates as a preprocessing module, encompassing synthetic data generation, neural text standardization, and subsequent downstream task integration. All source codes and datasets are publicly available at https://www.github.com/tweetpie/saldiray.

3.2.1 Data Acquisition and Synthetic Noise Generation

Within the SALDIRAY framework, training data preparation begins with the construction of a large-scale clean corpus written in formal and standardized language. Wikipedia articles are selected as the primary data source due to their broad topical diversity, consistent editorial standards, and well-structured linguistic properties. In a representative pipeline instantiation, each article is segmented into individual sentences using a Sentence Transformer–based segmenter to ensure accurate sentence boundary detection and to prevent segmentation-induced noise from propagating into subsequent stages of the standardization process.

To emulate the diverse noise patterns characteristic of informal digital communication, SALDIRAY defines a heuristic corruption procedure that introduces both character-level and lexical-level perturbations into clean text inputs. At the character level, the method performs random insertions, deletions, and transpositions, along with substitutions based on keyboard adjacency, effectively simulating common typographic errors. At the lexical level, canonical tokens are replaced with phonetically similar alternatives using homophone substitutions, as well as infor-

mal counterparts through slang or abbreviated variants. Homophone mappings were compiled from publicly available sources, including curated Wikipedia articles on English homophones. Slang replacements were derived from a publicly available social media abbreviation dataset [51], which we refined to ensure compatibility with the vocabulary of the target corpus. Specifically, descriptive entries in the slang dataset were normalized by extracting representative terms and discarding explanatory phrasing. All lexical substitutions are applied using precompiled, language-specific homophone and slang lexicons to maintain linguistic plausibility. The complete corruption procedure is formally outlined in Algorithm 2, and two example corrupted texts illustrating the applied noise types are presented in Figure 3.2.

Each clean sentence is corrupted nine times under different corruption configurations. These configurations differ in both the base corruption probability and in whether homophone and slang substitutions are systematically activated, as summarized in Table 3.1. For every clean sentence, multiple noisy variants are produced and paired with the original uncorrupted reference. This design exposes the normalization models to a wide and controlled range of noise severities and error types during training.

As a result, an initial corpus of 100K clean Wikipedia sentences is expanded into 900K aligned noisy–clean sentence pairs. By observing multiple distinct noisy realizations of the same semantic content, the normalization models learn to generalize beyond surface-form distortions and recover standardized language representations under diverse and unpredictable real-world noise conditions.

3.2.2 LLMs for Text Standardization

Text standardization in SALDIRAY is formulated as a supervised sequence-to-sequence learning task, where the objective is to map a noisy input sentence to its standardized clean counterpart while preserving semantic content. We experiment with both encoder–decoder architectures (e.g., T5Gemma) and decoder-only LLMs (e.g., Qwen-2.5B and Llama3.2), adapting the fine-tuning strategy to the structural properties of each model family.

Algorithm 2 Corruption Algorithm

Require: Dataset $D = \{x_1, x_2, \ldots, x_n\}$, corruption probability p

Ensure: Corrupted dataset D'

$D' \leftarrow \emptyset$
for all $x \in D$ **do**
    $x' \leftarrow x$
    $c \leftarrow \lfloor |x| \cdot p \rfloor$
    for $i = 1$ to c **do**
        Randomly select strategy s from:
    {*add-random-letter, remove-random-letter,*
      swap-consecutive-letters, keyboard-neighbor-replacement,
      homophone-replacement, slang-replacement}
        if s is character-level **then**
            Choose random position j in x' and apply s
        else if s is homophone-replacement **then**
            Replace a word with its homophone if available
        else if s is slang-replacement **then**
            Replace a word with its slang equivalent if available
        end if
    end for
    Append x' to D'
end for
return D'

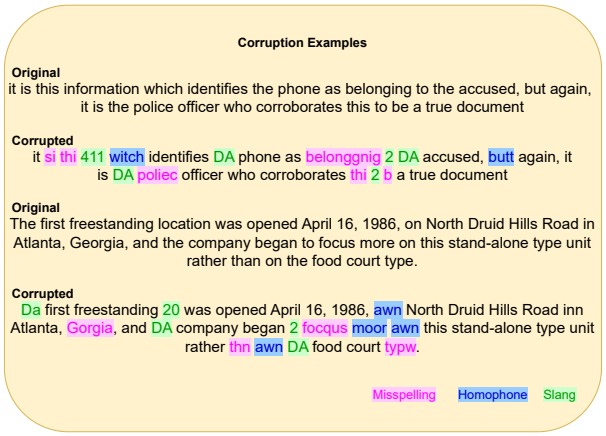


Figure 3.2: Samples from the corruption module, showing original and synthetically generated noisy variants containing misspellings, homophones, and slang.

Table 3.1: Training corpus generated by synthetically corrupting 100K sampled Wikipedia sentences under five different conditions, producing a total of 900K noisy–clean pairs. The first bold row represents the original uncorrupted version used as the clean reference.

Ver.	Size	Err. Prob.	All Homophones	All Slangs
1	100K	0.0	False	False
2	100K	0.0	True	True
3	100K	0.01	False	False
4	100K	0.02	False	False
5	100K	0.08	False	True
6	100K	0.09	True	True
7	100K	0.10	False	True
8	100K	0.11	True	True
9	100K	0.12	False	True
Total: 900K sentence pairs				

We select Qwen2.5-7B-Instruct and Meta-Llama-3-8B-Instruct as representative mid-scale instruction-tuned decoder-only LLM backbones with comparable parameter scales. Using two widely adopted models allows us to evaluate the architecture-agnostic nature of the SALDIRAY framework under consistent training data and normalization objectives. In addition, we include T5-Gemma to examine how an encoder–decoder architecture performs in the same setting, enabling comparison

between decoder-only generative models and sequence-to-sequence text transformation for linguistic standardization.

Fine-tuning is performed using standard sequence-to-sequence objectives over the paired noisy–clean training corpus. To avoid unnecessary modifications on already clean inputs and to encourage identity preservation, uncorrupted sentence pairs are explicitly included in the training data. Optimization is carried out using mixed-precision training with the AdamW optimizer and learning-rate scheduling. Training stability and generalization are monitored using BLEU and chrF scores computed on held-out development sets, and early stopping is applied to prevent overfitting. The resulting normalization models are fully domain-agnostic and can robustly standardize noisy text originating from diverse linguistic contexts without requiring any task-specific retraining.

3.2.3 Integration into Downstream NLP Tasks

Once trained, the SALDIRAY normalization models are deployed as a dedicated pre-processing stage for downstream NLP tasks. Given an input instance, the standardized output generated by SALDIRAY is passed directly to the original task model without altering the task model architecture, training protocol, or inference procedure. This modular design enables a controlled and architecture-independent evaluation of the effect of linguistic standardization on downstream performance.

We evaluate SALDIRAY on five representative supervised NLP tasks that span multiple modeling paradigms and levels of semantic reasoning, followed by an additional unsupervised real-world topic modeling case study:

- Topic Classification: Document-level topic classification using BERTopic-derived topic labels, where normalization improves lexical consistency and class separability under noisy input conditions.
- Sentiment Analysis: Binary sentiment classification on the IMDB dataset using BertForSequenceClassification.
- Aspect-Based Sentiment Analysis (ABSA): MaskedABSA [52] using a fine-tuned

FLAN-T5 model, where improved lexical clarity supports more reliable aspect detection and polarity classification.

- Natural Language Inference: Transformer-based inference models evaluated under noisy and standardized input conditions to assess semantic stability in entailment reasoning.

- Fact Verification: Fact verification evaluated exclusively on a fact verification benchmark dataset.

For each task, evaluation is conducted under three experimental conditions: Original (clean input), Broken (synthetically corrupted input), and Fixed (corrupted input normalized by SALDIRAY). Task-appropriate metrics are used to quantify performance under each condition. This experimental protocol enables a systematic and controlled assessment of the robustness gains introduced by SALDIRAY across heterogeneous NLP tasks.

In addition to these supervised tasks, we further assess SALDIRAY in a real-world, unsupervised topic modeling setting:

- Unsupervised Topic Modeling (BEA60K): Topic discovery on the BEA60K misspelling benchmark dataset [53], which contains naturally noisy user-generated text. In contrast to the synthetic errorification used in the main experiments, this setting evaluates SALDIRAY under authentic real-world noise, and the impact of normalization is measured through changes in topic coherence and qualitative topic structure.

3.3 Experimental Results

This section presents the experimental setup and quantitative evaluation of the proposed text standardization framework. We conduct wide range of experiments across multiple NLP tasks and model families to assess both the standalone effectiveness of the normalization models and their impact on downstream task performance. The experimental study is structured in two sequential stages: the

first stage evaluates the error correction and standardization capabilities of the LLM-based normalization models on corrupted textual input, while the second stage investigates how the standardized outputs influence the performance and robustness of downstream NLP systems. In what follows, we first describe the datasets and tasks used in our evaluation, then detail the training and inference setup of the normalization models, and finally report and analyze the results obtained across both stages of experimentation.

3.3.1 Tasks and Datasets

We evaluate our methods on a diverse set of six NLP tasks covering different domains and problem formulations. Each task is associated with a standard dataset, as summarized in Table 3.2. Below, we briefly describe each task and dataset:

Table 3.2: Overview of tasks and datasets used in our experiments.

Task	**Dataset**	**Model**	**Train Size**	**Test Size**
Topic Classification	20 Newsgroups	BERTopic	16,127	1,792
Sentiment Analysis	IMDB Reviews	BfSC	22,500	2,500
ABSA	SemEval 2014,2015,2016	MaskedABSA	9,928	4,057
Natural Language Inference	All-NLI	MiniLM, DeBERTA, BART	-	5,000
Fact Verification	CLIMATE-FEVER	RoBERTa	-	7,567
Topic Modeling	BEA60K	BERTopic	10,000	-

Before evaluating the impact of SALDIRAY on downstream NLP tasks, we first assess its text standardization performance using controlled training and test corpora specifically designed to capture diverse real-world noise patterns. For training the normalization models, we use the English Wikipedia corpus, which provides large-scale, formally written, and linguistically consistent text. This choice ensures that the models learn canonical language structure and standard orthographic forms.

For testing the standardization capability, we construct and evaluate on three distinct groups of datasets. First, we generate four synthetic corruption benchmarks derived from clean Wikipedia sentences: (i) character-level misspellings only, (ii)

homophone substitutions only, (iii) slang substitutions only, and (iv) a mixed-noise setting combining all corruption types. These controlled test sets enable fine-grained analysis of correction performance under isolated and compounded noise conditions.

Second, we construct two manually curated test sets to capture authentic human typing behavior. In the first set, a non-native English speaker is prompted with 1,000 spoken sentences and instructed to type them rapidly without correcting any typographical errors. In the second set, a native-level English speaker follows the same protocol under identical conditions. These datasets allow direct comparison of standardization performance across different user proficiency levels and error distributions.

Third, we evaluate on the BEA60K dataset , which contains naturally occurring misspellings collected from real user-generated content and serves as a large-scale real-world benchmark for spelling and normalization robustness. Unlike the synthetic datasets, BEA60K introduces unconstrained, naturally distributed noise.

Following the evaluation of the standardization models, we assess the effect of standardization on downstream NLP tasks. An overview of the corresponding tasks, datasets, models, and data splits is provided in Table 3.2.

Topic Classification. For topic classification, we use the 20 Newsgroups dataset [54], a widely used benchmark consisting of approximately 18,000 English documents distributed across 20 semantically distinct categories. The dataset is partitioned into 16,127 training documents and 1,792 test documents. We train the BERTopic model [55] to predict the corresponding category label for each document and evaluate performance using accuracy, as well as weighted precision, recall, and F1-score. This setup enables quantitative assessment of how robust topic discrimination is affected by noisy input and how effectively SALDIRAY restores discriminative lexical structure under corrupted conditions.

Sentiment Analysis. For sentiment classification, we use the IMDB movie reviews dataset [56], which contains 25,000 English reviews labeled with binary sentiment

polarity (*positive* or *negative*). The dataset is split into 22,500 reviews for training and 2,500 reviews for testing. A BERTForSequenceClassification model is fine-tuned on the training set and evaluated across original, corrupted, and standardized inputs to quantify the impact of normalization on sentiment prediction reliability under noisy conditions. The BertForSequenceClassification model, introduced by Google Research in [57], is a pre-trained BERT transformer architecture that incorporates a dropout layer and a softmax-based classification or regression head positioned atop the final decoder outputs.

Aspect-Based Sentiment Analysis. Aspect-based sentiment analysis (ABSA) is evaluated using benchmark datasets from SemEval 2014-2016 Task 4 [58, 59], including both Restaurant and Laptop review domains. The Restaurant dataset contains 3,693 training samples and 1,328 test samples, while the Laptop dataset consists of 3,045 training samples and 800 test samples. Each sentence may contain multiple target aspects, with each aspect labeled as *positive*, *negative*, or *neutral*. We employ a Masked-ABSA [52] model to perform aspect-level sentiment classification and report weighted precision, recall, F1-score, and overall classification accuracy under noisy and normalized conditions.

Natural Language Inference. For NLI, we assess sentence-pair reasoning across three core semantic relations: entailment, contradiction, and neutral. Evaluation is conducted on 5,000 randomly sampled instances from the All-NLI dataset, which merges the Stanford Natural Language Inference (SNLI) corpus [60, 61] and the Multi-Genre Natural Language Inference (MultiNLI) dataset [62]. These corpora offer large-scale, human-annotated sentence pairs designed to capture diverse linguistic phenomena and support robust testing of sentence-level semantic understanding.

We benchmark three task-specific NLI models, each representing a distinct architectural family. The all-MiniLM-L6-v2-nli model [63] is a distilled transformer that applies self-attention compression techniques for fast and efficient inference, making it suitable for real-time applications with constrained computational budgets. The mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 model [64] is a multilingual

encoder trained with cross-lingual NLI objectives, leveraging deep transfer learning to enhance performance under data scarcity and domain shift. Finally, the BART-large-MNLI model [65] is based on the BART architecture, incorporating a denoising pretraining objective followed by fine-tuning on the MultiNLI dataset to improve its capability in handling complex semantic inference tasks.

Model performance is reported using macro-averaged precision, recall, and F1-score to ensure a balanced evaluation across all three NLI categories, regardless of label frequency. This setup enables a comprehensive assessment of model robustness to semantic variability introduced by both clean and noisy input conditions.

Fact Verification. For fact verification, we evaluate the impact of text standardization using the CLIMATE-FEVER benchmark [66], a large-scale dataset designed for claim verification in the climate domain. Each instance consists of a natural language claim paired with supporting or refuting evidence retrieved from a corpus of climate-related documents, with labels indicating whether the claim is supported, refuted, or not enough information. We employ a RoBERTa-based verification model [67] to perform claim classification. Model performance is evaluated using standard verification accuracy. This task is particularly sensitive to lexical mismatches between claims and evidence caused by misspellings, abbreviations, and informal phrasing. Therefore, it provides a test bed for assessing how SALDIRAY's normalization affects retrieval alignment and final verification accuracy under noisy real-world conditions.

Real-World Topic Modeling on BEA60K. In addition to the supervised downstream tasks, we evaluate SALDIRAY in an unsupervised topic modeling setting using the BEA60K dataset. While many domain-specific natural corpora contain occasional misspellings, these errors often occur in non-informative tokens that do not meaningfully affect model decisions or semantic reasoning. As a result, such datasets are not suitable for evaluating the robustness of text standardization systems, since their misspellings do not exert sufficient pressure on topic formation or lexical clustering. To obtain a more reliable assessment, we use BEA60K, a dedicated misspelling benchmark composed of naturally occurring, semantically impactful

orthographic errors collected from real user-generated content. This makes BEA60K an ecologically valid and challenging test bed that cannot be replicated through synthetic corruption alone.

We apply BERTopic directly to the raw BEA60K corpus and compare the resulting topic structures with those obtained after SALDIRAY-based standardization. This experiment is conducted without any labeled supervision and serves two primary purposes: first, to evaluate whether SALDIRAY can recover coherent lexical structure from organically noisy text; and second, to analyze how standardization influences topic coherence, keyword stability, and interpretability in unsupervised topic discovery. The results provide a realistic validation of SALDIRAY's utility for large-scale noisy text analytics in scenarios where misspellings meaningfully distort semantic structure.

3.3.2 LLM-based Standardization Models and Fine-Tuning

To perform text error correction (standardization), we fine-tune three different pre-trained language models and also evaluate two additional state-of-the-art LLMs without fine-tuning. The models we use are:

- T5-Gemma [68] (encoder-decoder): Google's latest encoder-decoder adaptation of Gemma models, pretrained with UL2/PrefixLM objectives, fine-tuned by us for error correction.
- LLaMA-2 [69] (decoder-only): a 7B-parameter generative model; we use both the off-the-shelf pre-trained model ("naive" LLaMA-2) and a version fine-tuned by us for error correction.
- Qwen-7B [70] (decoder-only): an open-source 7B-parameter model; similar to LLaMA-2, we experiment with the base pre-trained model and a fine-tuned version.
- GPT-3.5 Turbo: OpenAI's GPT-3.5 Turbo model accessed through its API, representing the strong available LLM baseline (used without fine-tuning).
- GPT-4o-mini [71]: OpenAI's GPT-4o-mini model accessed through its API, representing the strong available LLM baseline (used without fine-tuning).

- SAGE T5-Large: The *ai-forever/T5-large-spell* model, introduced by Martynov *et al.* [34], trained using the SAGE methodology for generative spelling correction based on naturalistic multi-domain noise emulation.
- BART-Base FT: The *oliverguhr/spelling-correction-english-base* model [38], a BART-base encoder–decoder model fine-tuned specificall

The models were selected to cover diverse architectural paradigms, availability settings, and practical deployment scenarios relevant to text standardization. For open-source decoder-only architectures, we selected LLaMA-2 and Qwen-7B due to their widespread adoption and strong performance as general-purpose language models; similarly sized variants were used to ensure meaningful comparison. To represent the encoder–decoder paradigm, we included T5-Gemma as a recent and promising model that combines sequence-to-sequence learning with modern large-scale pretraining objectives, enabling us to assess its suitability for error correction.

Closed-source API-based systems were incorporated to provide strong reference baselines: GPT-4o-mini was chosen as a capable contemporary GPT variant, while GPT-3.5 Turbo enables comparison across two generations of GPT models with different capability–cost trade-offs. In addition, we included specialized publicly available spelling correction systems to contextualize performance against task-focused approaches. SAGE T5-Large represents a widely used open-source generative spelling correction model trained with naturalistic noise emulation, whereas BART-Base FT was selected as a lightweight, publicly available model optimized for fast misspelling correction in practical settings.

Together, this selection allows SALDIRAY to be evaluated across architectural families, general-purpose LLMs, proprietary API systems, and task-specific correction models.

Each of the above models (with the exception of API-only systems, GPT-3.5 Turbo and GPT-4o-mini) was fine-tuned on a large-scale synthetic corpus of noisy–clean sentence pairs. For corpus construction, we sampled 100,000 sentences from English Wikipedia, which served as clean reference texts. To create training pairs, we introduced artificial noise into the sampled sentences to simulate real-world

text corruptions such as spelling mistakes, typos, homophone substitutions, and slang usage. Each clean sentence was broken into 9 distinct corrupted versions, generated under different probabilistic noise configurations as described in Table 3.1. This yielded a total of 900,000 noisy sentences paired with their original clean counterparts, ensuring that the models were exposed to diverse error conditions for the same underlying text.

We fine-tuned three open-source large language models (LLMs) on this corpus: Flan-T5, LLaMA-2, and Qwen-7B. Training was conducted using the AdamW optimizer with a learning rate of 1×10^{-5} and a linear learning rate decay schedule with 3% warm-up. A weight decay of 0.01 was applied for regularization, and the effective batch size was set to 8 by using a per-device batch size of 2 with gradient accumulation over 4 steps. Given that the training corpus is large relative to the correction task but small compared to the model capacity, training was limited to a single epoch.

To enable parameter-efficient fine-tuning, we adopted the LoRA (Low-Rank Adaptation) technique [72]. Specifically, we used a configuration with a rank of 8, $\alpha = 8$, and a dropout rate of 0.05. LoRA adapters were injected into key attention and feed-forward components of the transformer layers, while bias terms were excluded. The task type was set to causal language modeling. This setup enables efficient fine-tuning with a significantly reduced number of trainable parameters, allowing the models to adapt effectively without extensive computational cost.

The fine-tuned versions of T5-Gemma, LLaMA-2, and Qwen-7B obtained through this process constitute our specialized error-correction systems, capable of reliably mapping noisy input text into clean, standardized English suitable for downstream NLP tasks. Throughout this paper, the term SALDIRAY-model refers to these backbone architectures after fine-tuning with the curated noisy–clean training data produced, rather than to a new standalone model architecture.

3.3.3 Standardization Evaluation Metrics

To evaluate text standardization quality, we measure how accurately noisy tokens are transformed into their correct standardized forms at the token level. Our evaluation follows the alignment-based token comparison protocol introduced in SpellRuEval [73]. Since orthographic noise affects both word boundaries and surface forms, a structured alignment procedure is applied prior to metric computation. Each sentence pair is lightly normalized by lowercasing, whitespace tokenization, and stripping isolated punctuation and edge non-alphabetic characters.

Alignment between the noisy source and its gold-standard correction is constructed in two stages. First, a word-level longest common subsequence (LCS) is computed to obtain anchor points. Second, nonidentical segments between anchors are aligned using word-level Levenshtein edit distance, allowing insertions, deletions, substitutions, and permutations. The same procedure is applied between the source sentence and the model-generated output to extract predicted correction pairs.

From these alignments, we construct two sets of correction pairs

$$S_{\text{corr}} = \{((i,x),y)\}, \quad S_{\text{pred}} = \{((i,x),\hat{y})\}, \tag{3.1}$$

where i denotes the sentence index, x is the noisy source token, y is the gold-standard corrected token, and $\hat{y}$ is the system-predicted correction. Both S_{corr} and S_{pred} contain only nonidentical source–target pairs and therefore correspond to actual correction events.

True positives (TP) are defined as the number of token-level corrections for which the system output exactly matches the gold-standard correction, i.e.,

$$\text{TP} = \left\| S_{\text{corr}} \cap S_{\text{pred}} \right\|. \tag{3.2}$$

Based on (3.2), precision and recall are computed as

$$\text{Precision} = \frac{\text{TP}}{|S_{\text{pred}}|}, \tag{3.3}$$

$$\text{Recall} = \frac{\text{TP}}{|S_{\text{corr}}|}, \tag{3.4}$$

where $|S_{\text{pred}}|$ is the total number of corrections proposed by the system and $|S_{\text{corr}}|$ is the total number of true correction targets. Finally, the F1-score is defined as the harmonic mean of precision and recall:

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3.5}$$

Precision reflects the fraction of system-generated corrections that are correct, while recall measures the proportion of true noisy tokens that are successfully standardized. False negatives include both incorrectly corrected tokens and noisy tokens left unchanged. The F1-score in (3.5) provides a balanced summary of the system's overall correction performance.

In addition to these token-level metrics, we also report sentence-level accuracy, which measures the proportion of test sentences for which the system-generated standardized output is an exact match with the gold-standard correction. A prediction is considered correct only if the entire sentence is identical to the reference after standardization. This metric therefore captures the system's ability to fully standardize a sentence without introducing any residual errors and provides a strict, end-to-end assessment of standardization performance.

3.3.4 Standardization Performance

Table 3.3: Token and sentence-level standardization performance across different noise types and correction models using Accuracy, F1, Precision, and Recall. Top performance is bolded, all values reported with three decimal digits.

Method	**All**				**Homophones**				**Misspellings**				**Slang**			
	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.
BART-Base FT	0.000	0.332	0.383	0.293	0.018	0.514	0.586	0.458	0.123	0.575	0.606	0.547	0.009	0.094	0.090	0.099
GPT-3.5	0.239	0.760	0.776	0.745	0.489	0.864	0.845	0.884	0.372	0.821	0.827	0.815	0.445	0.704	0.594	0.865
GPT-4o-mini	0.215	0.771	0.782	0.760	0.544	0.897	0.888	0.906	0.362	0.816	0.825	0.808	0.599	0.813	0.769	0.862
LLaMA Base	0.012	0.537	0.566	0.512	0.045	0.612	0.587	0.640	0.057	0.587	0.601	0.573	0.127	0.474	0.350	0.733
Qwen Base	0.017	0.481	0.513	0.452	0.059	0.580	0.582	0.577	0.134	0.590	0.608	0.573	0.108	0.428	0.338	0.583
SAGE T5-Large	0.001	0.478	0.529	0.436	0.115	0.589	0.741	0.489	0.472	0.836	0.853	0.819	0.001	0.033	0.129	0.019
SALDIRAY-LLaMA	**0.870**	**0.964**	**0.980**	**0.949**	**0.846**	**0.971**	**0.972**	**0.969**	**0.915**	**0.980**	**0.981**	**0.980**	**0.886**	**0.953**	**0.920**	**0.989**
SALDIRAY-Qwen	0.731	0.938	0.954	0.923	0.686	0.944	0.936	0.952	0.782	0.945	0.943	0.947	0.744	0.893	0.821	0.978
SALDIRAY-T5Gemma	0.017	0.292	0.624	0.191	0.079	0.544	0.803	0.412	0.097	0.414	0.672	0.299	0.211	0.549	0.839	0.407

Evaluation on the Synthetically Corrupted Dataset We evaluate correction quality under four controlled noise conditions, consisting of isolated homophone substitutions, isolated slang replacements, isolated character-level misspellings, and a mixed configuration that combines all error types. Each corrupted dataset is standardized using base and fine tuned versions of Qwen, LLaMA, and T5-Gemma, as well as the API based GPT-3.5 Turbo and GPT-4o-mini models. Performance is reported using sentence-level accuracy and token-level F1, precision, and recall.

Table 3.3 demonstrates that base pretrained models perform poorly across all noise types, with sentence-level accuracy remaining below 0.30 in most configurations. This confirms that off the shelf generative models lack the robustness required for reliable spelling and noise correction without task specific fine tuning. In contrast, SALDIRAY fine tuned models exhibit a consistent and significant performance increase across all four noise categories.

Among all evaluated systems, SALDIRAY-LLaMA achieves the strongest performance in every configuration, reaching sentence-level accuracy of 0.915 under misspellings and 0.886 under slang. Token-level F1 exceeds 0.95 in all isolated noise settings and reaches 0.98 under misspellings. These results indicate highly reliable correction behavior across both orthographic and lexical noise.

GPT-3.5 Turbo and GPT-4o-mini display moderate robustness, particularly under homophone corruption, but their sentence-level accuracy consistently remains below 0.60. This shows that although large API based models can often propose plausible token-level corrections, they frequently fail to fully reconstruct complete sentences without residual errors.

SAGE T5-Large exhibits strong performance under character-level misspellings but deteriorates severely under slang noise, which reflects a limited capacity to handle socially grounded lexical variation. BART-Base FT performs weakly across all corruption types, confirming that character only correction strategies are insufficient for contemporary informal text.

Direct comparison between base and SALDIRAY fine tuned versions of Qwen and LLaMA highlights the benefit of targeted standardization training. Under mixed cor-

ruption, Qwen Base reaches only 0.017 accuracy and 0.481 F1, whereas SALDIRAY-Qwen reaches 0.731 accuracy and 0.938 F1. The improvement is even stronger for LLaMA, which improves from 0.012 accuracy and 0.537 F1 in its base form to 0.870 accuracy and 0.964 F1 after SALDIRAY fine tuning. These gains confirm that synthetic multi noise training is essential for building robust and general purpose text standardization models.

In contrast, SALDIRAY-T5Gemma underperforms all other SALDIRAY variants across categories, reaching only 0.017 sentence-level accuracy and 0.292 token-F1 in the mixed setting. Its error profile is characterized by high precision but very low recall, indicating a strong tendency toward under-correction (i.e., leaving corrupted spans unchanged), which is especially damaging for exact-match sentence metrics. T5 is pretrained with a denoising objective that corrupts contiguous spans and replaces each span with a unique sentinel token, training the model to reconstruct the missing text [74]. In our normalization setting, the correct output is often near-identity and the mapping from noisy inputs to normalized forms can be many-to-one; this creates a mismatch with span-denoising pretraining, which does not explicitly encourage minimal, token-level edits. Moreover, our training budget applied the same single-epoch LoRA fine-tuning across backbones, which may disproportionately disadvantage the encoder–decoder model because effective input–output coupling relies on cross-attention dynamics that can be slower or more optimization-sensitive to adapt. Recent evidence from adapting Gemma into an encoder–decoder architecture shows that such models can converge more slowly and may require explicit cross-attention warmup to achieve stable performance, underscoring that equal epochs need not imply equal adaptation [68]. SALDIRAY-T5Gemma often fails to perform the required fine-grained edits under our shared experimental setup; incorporating cross-attention warmup and a more tailored fine-tuning schedule appears promising for better aligning the model with our training paradigm.

Evaluation on Naturally Corrupted Manual Typing Misspelling Dataset For this second experiment, we evaluate only the best performing SALDIRAY variant identified from the synthetic benchmark, namely SALDIRAY-LLaMA. This selection

Table 3.4: Standardization and misspelling detection performance on manually curated native and non-native typing datasets using Accuracy, F1, Precision, and Recall. Top performance is bolded in both tests.

Model	**Acc.**	**F1**	**Prec.**	**Rec.**
Native Speaker Dataset				
BART-Base FT	0.050	0.5167	0.5449	0.4912
GPT-3.5 Turbo	0.327	0.8013	0.7936	0.8091
GPT-4o-mini	**0.357**	**0.8235**	**0.8209**	**0.8260**
LLaMA Base	0.043	0.5510	0.5495	0.5525
Qwen Base	0.154	0.6492	0.6508	0.6477
SAGE T5-Large	0.186	0.7148	0.7696	0.6673
SALDIRAY-LLaMA	0.317	0.8056	0.8009	0.8103
Non-Native Speaker Dataset				
BART-Base FT	0.047	0.5074	0.5379	0.4801
GPT-3.5 Turbo	0.316	0.7940	0.7854	0.8027
GPT-4o-mini	**0.364**	**0.8247**	**0.8244**	**0.8249**
LLaMA Base	0.051	0.5492	0.5513	0.5470
Qwen Base	0.160	0.6578	0.6591	0.6566
SAGE T5-Large	0.190	0.7130	0.7646	0.6679
SALDIRAY-LLaMA	0.320	0.8041	0.8004	0.8080

Table 3.5: Sentence-level correction and token-level misspelling detection performance on the BEA60K dataset using Accuracy, F1, Precision, and Recall. Top performance is bolded.

Model	Accuracy	F1	Precision	Recall
BART-Base FT	0.0707	0.5242	0.5048	0.5453
GPT-4o-mini	0.4929	0.6811	0.5982	**0.7907**
LLaMA Base	0.1615	0.3448	0.2484	0.5635
Qwen Base	0.2815	0.4616	0.3545	0.6611
SAGE T5-Large	**0.6748**	**0.7919**	**0.8028**	0.7812
SALDIRAY-LLaMA	0.5362	0.6862	0.6061	**0.7907**

ensures that real world performance is assessed using the strongest available standardization configuration. The manually curated datasets are generated through rapid transcription without error correction, once by a native level English speaker and once by a non-native English speaker. These datasets capture naturally occurring typing behavior including omissions, phonetic substitutions, abbreviations, and informal constructions.

Across both datasets, SALDIRAY-LLaMA maintains strong and stable correction performance, achieving F1 scores of 0.8056 for the native dataset and 0.8041 for the non-native dataset. These results confirm that a model trained exclusively on synthetic corruption patterns can generalize effectively to authentic human error distributions. Although GPT-4o-mini attains slightly higher accuracy and F1 in both datasets, SALDIRAY-LLaMA remains highly competitive while offering the important practical advantage of local deployment without reliance on external application programming interfaces.

Evaluation on the Naturally Corrupted BEA60K Misspelling Dataset Baseline systems such as SAGE T5-Large exhibit moderate detection performance but limited recall, particularly for the phonetic errors prevalent in the non-native dataset. BART-Base FT continues to underperform across all metrics. These findings demonstrate that SALDIRAY fine tuning yields both robust correction capability and reliable real

world transfer.

For this third experiment, we again restrict evaluation to the best performing SALDIRAY variant obtained from the synthetic benchmark, namely SALDIRAY-LLaMA. This experiment is conducted on the BEA60K dataset, which contains large-scale naturally occurring misspellings paired with partially corrected reference sentences. Unlike synthetically generated benchmarks, the reference annotations in BEA60K still contain a considerable amount of residual spelling errors.

This characteristic directly influences the observed precision values. Both SALDIRAY-LLaMA and GPT-4o-mini frequently produce linguistically correct standardizations that differ from the imperfect gold references, which results in artificially reduced precision scores despite correct corrections. In contrast, SAGE T5-Large achieves higher measured precision primarily because it preserves many of the residual errors present in the reference labels.

Despite this evaluation limitation, SALDIRAY-LLaMA achieves the strongest balance of sentence-level accuracy and recall among all deployable models. This indicates that it corrects larger fraction of real world errors than base pretrained models or character only correction systems. The results further confirm that pretrained decoder models without task specific standardization training are insufficient for practical spelling normalization at scale.

Across all three evaluation settings—synthetic corruption, manual typing noise, and naturally corrupted web text—SALDIRAY demonstrates consistent and high-fidelity correction behavior. Fine-tuned decoder models, particularly LLaMA adapted with the SALDIRAY framework, achieve the strongest accuracy and F1 scores across most noise conditions. Strong performance is also observed for GPT-4o-mini, which serves as a high-capacity reference baseline and performs competitively across datasets due to its broad pretraining and general reasoning capabilities. However, under isolated misspelling scenarios, traditional spelling-focused approaches such as SAGE and SALDIRAY-adapted decoder backbones (LLaMA and Qwen) perform particularly well, reflecting their alignment with character-level perturbation patterns. Similarly, BERT-base exhibits competitive results on synthetic misspelling tests, which is consistent with its capacity and prior exposure to masked

language modeling objectives that emphasize local token recovery. On BEA60K, SAGE achieves stronger results, likely because it is explicitly trained on extensively synthetically corrupted data and is optimized for character-level errors such as keyboard adjacency mistakes. These observations highlight that different baselines excel under different noise regimes, supporting the realism of the evaluation design and indicating that SALDIRAY complements rather than replaces specialized correction methods. More broadly, the results confirm that character-only spelling correction is insufficient for modern informal text, that socially grounded noise such as slang must be modeled explicitly, and that standardized text produced by SALDIRAY forms a stable and reliable foundation for downstream natural language processing.

3.3.5 Standardization Quality and Overcorrection Analysis

Table 3.6: Slang standardization results. Best values in each column are highlighted.

Method	**Correction Statistics**			**Correction Quality**		**Semantic Preservation**		
	TP	FP	FN	Prec	Rec	P	R	F1
SALDIRAY-LLaMA	**2353**	235	**65**	0.909	**0.973**	**0.993**	**0.992**	**0.993**
SAGE	661	**59**	1757	0.918	0.273	0.956	0.961	0.958
GPT-4o-mini	2249	158	169	**0.934**	0.930	0.990	0.989	0.989

Table 3.7: Homophone standardization results. Best values in each column are highlighted.

Method	**Correction Statistics**			**Correction Quality**		**Semantic Preservation**		
	TP	FP	FN	Prec	Rec	P	R	F1
SALDIRAY-LLaMA	8618	264	1442	0.970	0.857	**0.990**	**0.989**	**0.989**
SAGE	5932	310	4128	0.950	0.590	0.948	0.960	0.954
GPT-4o-mini	**9082**	204	**978**	**0.978**	**0.903**	0.986	0.984	0.985

Using the top-performing model for detailed error inspection is a common practice in standardization research, as it provides a representative view of system behavior

under realistic deployment settings. Accordingly, the subsequent analyses focus on SALDIRAY-LLaMA, SAGE, and GPT-4o-mini as the strongest performing configurations across the evaluated conditions.

Tables 3.6 and 3.7 present detailed correction statistics, correction quality metrics, and semantic preservation scores for slang and homophone standardization. Across both noise categories, SALDIRAY adapted decoder backbones achieve strong correction coverage while maintaining high semantic fidelity. In particular, SALDIRAY-LLaMA obtains the highest recall and semantic consistency on slang standardization, indicating effective recovery from socially grounded lexical variation. GPT-4o-mini demonstrates the strongest precision and overall correction quality on homophones, reflecting the benefits of high-capacity generative reasoning in disambiguation scenarios. SAGE exhibits competitive precision but substantially lower recall, consistent with its character-level correction bias and conservative editing behavior.

To further examine standardization behavior, Table 3.8 reports a manually curated breakdown of false positive edits. False positives are defined as cases where a token is modified during standardization despite remaining unchanged between the original clean text and its corrupted counterpart. This definition isolates unnecessary edits and provides a direct measure of overcorrection tendencies.

The false positive distributions reveal distinct patterns across methods. SALDIRAY's false positives are primarily associated with special tokens such as named entities and domain-specific expressions, suggesting sensitivity to lexical standardization beyond simple spelling repair. In contrast, SAGE false positives are dominated by genuine overcorrections, reflecting its emphasis on character-level perturbation handling. GPT-4o-mini produces a larger share of meaning-neutral alternatives, indicating a generative preference for stylistic variation rather than strict token preservation. Numeric changes appear as a smaller but consistent source of error across models, highlighting a limitation in quantity-sensitive standardization scenarios.

Semantic preservation is evaluated using BERTScore, which computes embedding-based similarity between standardized and reference sentences. Unlike surface-

form metrics, BERTScore measures contextual semantic alignment by pairing each token in the candidate sentence with its most similar token in the reference and repeating the process in the opposite direction to estimate precision and recall, which are then combined into an F1 score [75]. The consistently high semantic scores across models indicate that most edits preserve meaning even when lexical forms differ.

Overall, these results show that SALDIRAY provides strong standardization coverage with controlled overcorrection, particularly for socially grounded noise types. At the same time, the manual false positive analysis highlights limitations related to named entity handling and numeric stability, suggesting that entity-aware constraints and numeric preservation mechanisms represent promising directions for future work. The complementary performance patterns observed across architectures further support the use of SALDIRAY as a model-agnostic preprocessing layer that interacts with different backbone inductive biases while maintaining semantic integrity.

Table 3.8: False positive breakdown across slang and homophone normalization (% of FP within each method). For each corruption strategy, the highest value within each false-positive category is highlighted.

Type	**Slang**			**Homophone**		
	SALDIRAY	SAGE	GPT-4o-mini	SALDIRAY	SAGE	GPT-4o-mini
Special words	**54.9**	42.4	17.1	**55.3**	26.1	14.2
Overcorrection	21.7	40.7	**45.6**	23.5	**67.7**	56.8
No meaning change	14.5	6.8	**35.4**	11.4	1.6	**26.5**
Numeric change	6.4	**6.8**	–	**6.8**	3.2	0.5
Real fix	2.6	3.4	1.9	3.0	1.3	2.0

3.3.6 Downstream Task Performance with Standardized Data

In the second stage of our evaluation, we analyze how text standardization affects downstream NLP performance. For each task introduced in Section 3.3.1, we consider three dataset variants: original clean data (O), synthetically corrupted data

(C), and corrupted data that has been standardized by SALDIRAY (S_C). When tasks involve training, we systematically vary both the training and test conditions to disentangle the effects of noise and standardization.

Table 3.9: Train–test configurations for downstream experiments. O: original data, C: corrupted data, S_C: standardized version of corrupted data.

Training Dataset	**Test Dataset**	**Abbreviation**
Original	Original	O→O
Original	Corrupted	O→C
Original	Corrupted Standardized	O→S_C
Corrupted	Original	C→O
Corrupted	Corrupted	C→C
Corrupted	Corrupted Standardized	C→S_C
Corrupted Standardized	Original	S_C→O
Corrupted Standardized	Corrupted	S_C→C
Corrupted Standardized	Corrupted Standardized	S_C→S_C

For each train–test pair in Table 3.9, we train the task-specific model (e.g., a BERT classifier for sentiment) on the specified training variant and evaluate on the specified test variant. This design allows us to (i) quantify how much performance is lost when test data is corrupted, and (ii) measure how effectively SALDIRAY restores performance when the same corrupted inputs are standardized before being passed to the downstream model. In addition, by including experiments where both training and test data are standardized, we assess whether SALDIRAY can serve as a stable preprocessing layer even when the original data is relatively clean.

For tasks that do not require explicit training (e.g., BERTopic-based topic modeling on BEA60K), we compare the behavior of the same model when applied to original, corrupted, and standardized inputs, focusing on topic coherence and clustering alignment rather than classification accuracy.

Topic Classification. Table 3.10 shows topic classification results on 20 Newsgroups. When both training and test data are clean (O→O), the model achieves an F1-score of 0.753. Corrupting only the test set (O→C) leads to a clear degradation (F1 = 0.651), confirming that even moderate noise harms topic discrimination.

In contrast, when the same corrupted test set is standardized with SALDIRAY (O→S_C), performance returns exactly to the O→O level (F1 = 0.753). This demonstrates that standardization can fully compensate for the corruption in this task. A similar pattern appears when training on corrupted data: C→C is clearly inferior to C→O and C→S_C, whereas testing on standardized inputs (C→S_C) almost mirrors testing on clean inputs (C→O). Finally, training on standardized data yields highly stable behavior: both S_C→O and S_C→S_C match or slightly surpass the clean baseline, whereas S_C→C remains degraded due to the uncorrected noise at test time. The topic classification results show that SALDIRAY-based standardization consistently restores performance to near-clean levels whenever it is applied on the test side.

Table 3.10: Topic classification performance on 20 Newsgroups under different train–test configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined.

Train	Test	Accuracy	Precision	Recall	F1
O	O	0.756	0.758	0.756	0.753
O	C	0.655	0.665	0.655	0.651
O	S_C	**0.756**	**0.759**	**0.756**	0.753
C	O	0.741	0.747	0.741	0.741
C	C	0.669	0.670	0.669	0.664
C	S_C	0.738	0.744	0.738	0.738
S_C	O	**0.756**	**0.759**	**0.756**	**0.754**
S_C	C	<u>0.657</u>	<u>0.669</u>	<u>0.657</u>	<u>0.652</u>
S_C	S_C	0.755	0.758	0.755	0.753

Sentiment Analysis. Table 3.11 presents results for binary sentiment classification on IMDB. The clean baseline (O→O) attains F1 = 0.936. When the test data is corrupted but training remains clean (O→C), F1 drops sharply to 0.738, with recall particularly affected (0.601). Although precision is artificially inflated in this noisy setting, the low recall and F1 indicate that the classifier fails to robustly identify the correct sentiment under corrupted input.

Standardizing the corrupted test set, however, nearly eliminates this degradation: O→S_C achieves F1 = 0.935, effectively matching the clean baseline. A similar recovery is observed when training on corrupted data: C→C is weaker than both C→O and C→S_C, while C→S_C restores F1 to 0.922, identical to evaluating on clean test data. Training and testing exclusively on standardized inputs (S_C→S_C) yields F1 = 0.931, again very close to the O→O baseline. In contrast, any configuration that uses corrupted test data (O→C, C→C, S_C→C) remains among the poorest performers. These patterns highlight that SALDIRAY standardization on the test side is the critical factor in recovering high sentiment classification performance.

Table 3.11: Sentiment analysis performance on IMDB under different train–test configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined.

Train	Test	Accuracy	Precision	Recall	F1
O	O	0.938	0.929	0.943	0.936
O	C	<u>0.795</u>	**0.956**	<u>0.601</u>	<u>0.738</u>
O	S_C	**0.937**	0.928	0.942	**0.935**
C	O	0.922	0.891	0.954	0.922
C	C	0.887	<u>0.869</u>	0.900	0.884
C	S_C	0.922	0.887	**0.959**	0.922
S_C	O	0.936	0.934	0.933	0.933
S_C	C	0.816	0.944	0.656	0.774
S_C	S_C	0.933	0.925	0.937	0.931

Aspect-Based Sentiment Analysis (ABSA). The MaskedABSA results in Table 3.12 exhibit a clear and consistent pattern regarding the role of standardization. The clean condition (O→O) provides a strong reference point, while introducing corruption at test time (O→C) yields the weakest performance across all configurations (F1 = 0.830). This sharp decline illustrates how even moderate surface-level distortions can disrupt aspect-level sentiment predictions.

Applying standardization to corrupted test inputs largely restores model performance. When trained on clean data, evaluating on standardized corrupted text (O→ S_C) achieves F1 = 0.874, nearly matching the clean baseline. This recovery is consistent across other training conditions as well: C→ S_C produces F1 = 0.883, and S_C → S_C reaches 0.886, both significantly higher than their corrupted-test counterparts. These results confirm that standardized text reliably supports fine-grained sentiment modeling even when the training data itself contains noise.

Notably, the strongest overall performance appears in the S_C →O configuration (F1 = 0.899), demonstrating that training on standardized text can generalize exceptionally well to clean inputs. In contrast, all configurations evaluated on corrupted test data (· →C) show reduced accuracy, precision, recall, and F1. SALDIRAY standardization effectively restores aspect-level semantic structure, mitigates the negative effects of corruption, and maintains stability when applied to already clean text.

Natural Language Inference Under Corruption

NLI is a particularly sensitive testbed for noise, as small lexical changes can flip the entailment relation. We therefore examine how standardization affects three-way NLI (entailment, contradiction, neutral) under controlled corruption of both premise and hypothesis.

Three-Way NLI. Table 3.13 summarizes macro-averaged precision, recall, and F1 for three NLI models. For all models, O→O provides the reference performance. Whenever the hypothesis is evaluated in its corrupted form (rows O→C, C→C, S_C →C), F1 decreases noticeably. For example, all-MiniLM-L6-v2 drops from F1 =

Table 3.12: MaskedABSA performance across train–test configurations on SemEval datasets. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined.

Train	Test	Accuracy	Precision	Recall	F1
O	O	0.894	0.888	0.894	0.887
O	C	<u>0.834</u>	<u>0.843</u>	<u>0.834</u>	<u>0.830</u>
O	S_C	0.880	0.874	0.880	0.874
C	O	0.902	0.893	0.900	0.893
C	C	0.871	0.865	0.871	0.864
C	S_C	0.889	0.881	0.889	0.883
S_C	O	**0.903**	**0.900**	**0.903**	**0.899**
S_C	C	0.851	0.850	0.851	0.847
S_C	S_C	0.890	0.886	0.890	0.886

0.6822 (O→O) to 0.5531 (O→C), and similar losses appear for mDeBERTa-v3-XNLI and BART-large-MNLI.

Standardization largely reverses this degradation. When the corrupted hypothesis is standardized (O→S_C), all three models regain most of their clean-test performance, with F1 values closely tracking the O→O reference. Likewise, when the premise side is corrupted, configurations that evaluate on S_C (C→S_C or $S_C \rightarrow S_C$) consistently outperform those that evaluate on C. The strongest NLI behavior is observed when at least the hypothesis is standardized, and the best overall scores are obtained for $S_C \rightarrow O$ and $S_C \rightarrow S_C$, showing that SALDIRAY can support robust inference even when training or evaluation data has been exposed to noise. In contrast, any configuration that leaves the hypothesis corrupted remains clearly suboptimal.

Fact Verification

We next examine fact verification on the CLIMATE-FEVER dataset, where the model must classify claims as supported, refuted, or not enough information based on

Table 3.13: Three-way NLI performance across premise–hypothesis corruption configurations. Each cell shows macro-averaged Precision / Recall / F1. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined.

Premise → Hypothesis	**all-MiniLM-L6-v2**			**mDeBERTa-v3-XNLI**			**BART-large-MNLI**		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
O → O	0.7285	0.6845	0.6822	0.6937	0.6407	0.6412	0.7527	0.7082	0.7055
O → C	0.6215	0.5680	0.5531	0.6312	0.6024	0.6014	0.6929	0.6362	0.6290
O → S_C	0.7108	0.6653	0.6618	0.6808	0.6218	0.6210	0.7331	0.6874	0.6831
C → O	0.6783	0.6262	0.6171	0.6707	**0.6496**	**0.6497**	0.7095	0.6725	0.6680
C → C	<u>0.5961</u>	<u>0.5417</u>	<u>0.5243</u>	<u>0.6116</u>	0.6014	0.6010	<u>0.6548</u>	<u>0.6111</u>	<u>0.6035</u>
C → S_C	0.6648	0.6117	0.6016	0.6588	0.6369	0.6364	0.6975	0.6592	0.6541
S_C → O	**0.7202**	**0.6791**	**0.6755**	**0.6929**	0.6388	0.6383	**0.7469**	**0.7045**	**0.7011**
S_C → C	0.6184	0.5673	0.5521	0.6263	<u>0.5992</u>	<u>0.5980</u>	0.6845	0.6305	0.6222
S_C → S_C	0.7052	0.6626	0.6579	0.6801	0.6207	0.6188	0.7302	0.6855	0.6808

evidence documents. This task is particularly sensitive to lexical corruption, as key domain terms often determine the correct label.

As shown in Table 3.14, clean claims and evidence (O→O) achieve F1 = 0.8047. Introducing corruption on the evidence side (O→C) causes a drop in both recall (from 0.7883 to 0.5911) and F1 (to 0.6398), indicating that the model struggles to align corrupted evidence with the claim. When the same corrupted evidence is standardized (O→S_C), performance nearly returns to the clean baseline (F1 = 0.7997), demonstrating that SALDIRAY restores the lexical cues needed for accurate verification.

The same pattern appears when the claim is corrupted: configurations with C on the claim side perform worst when evidence is also corrupted (C→C), but testing on standardized evidence (C→S_C) consistently improves over C→C. Finally, the configurations $S_C \rightarrow O$ and $S_C \rightarrow S_C$ are competitive with O→O, showing that a standardization-first pipeline can maintain high fact-verification performance even when upstream data quality is mixed. No configuration with corrupted evaluation inputs matches the performance of those using standardized evidence.

Table 3.14: Fact verification performance on CLIMATE-FEVER under claim–evidence corruption configurations. O: original, C: corrupted, S_C: standardized corrupted. Top performance is bolded except the O→O test; weakest performance is underlined.

Claim → Evidence	Acc.	Prec.	Rec.	F1
O → O	0.8529	0.8249	0.7883	0.8047
O → C	0.7595	0.7940	0.5911	0.6398
O → S_C	0.8493	0.8220	**0.7821**	**0.7997**
C → O	0.7126	0.7344	0.4816	0.5103
C → C	<u>0.6879</u>	<u>0.6780</u>	<u>0.4404</u>	<u>0.4534</u>
C → S_C	0.7107	0.7326	0.4796	0.5073
S_C → O	**0.8426**	**0.8204**	0.7643	0.7886
S_C → C	0.7471	0.7774	0.5645	0.6110
S_C → S_C	0.8379	0.8166	0.7574	0.7827

Impact of Corruption Types and Standardization on Entailment Performance

We first assess how different categories of corruption—misspellings, homophones, and slang—affect the semantic structure of textual inputs. Table 3.16 reports BERTScore results between original hypothesis/premise sentences and their SALDIRAY-corrected corrupted variants. Each corruption type is evaluated separately, along with an aggregated condition where all corruption types are applied simultaneously.

The results confirm that SALDIRAY maintains high semantic fidelity across all corruption settings. Slang and homophone corrections exhibit particularly strong preservation scores, with F1-scores exceeding 0.98 and 0.94 respectively. Misspelling corrections yield slightly lower scores but remain robust, particularly in recall. These findings demonstrate that SALDIRAY's standardization process effectively restores noisy inputs while preserving their intended semantic content—providing a reliable preprocessing foundation for downstream tasks.

Building on this, Table 3.15 presents the impact of these corrupted inputs—prior

Table 3.15: Effect of targeted corruption strategies on entailment classification using the All-NLI dataset. The first row (O) presents the baseline performance of each model on clean input. Subsequent rows (A, M, H, S) report performance deltas relative to this baseline under different corruption types: All (A), Misspellings only (M), Homophones only (H), and Slang only (S). C_μ and W_μ indicate the average character error rate and word error rate, respectively, introduced by each corruption strategy. Bolded values denote the smallest performance degradation within each row, indicating greater robustness.

P-H	$\mathbf{C_\mu}$	$\mathbf{W_\mu}$	**all-MiniLM-L6-v2**			**mDeBERTa-v3-XNLI**			**BART-large-MNLI**		
			Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
O	-	-	0.7285	0.6845	0.6822	0.6937	0.6407	0.6412	**0.7527**	**0.7082**	**0.7055**
A Δ	0.106	0.354	-0.1324	-0.1428	-0.1579	**-0.0821**	**-0.0393**	**-0.0402**	-0.0979	-0.0971	-0.1020
M Δ	0.094	0.371	-0.1492	-0.1517	-0.1652	**-0.0879**	**-0.0427**	**-0.0433**	-0.1138	-0.1109	-0.1155
H Δ	0.075	0.268	-0.0612	-0.0637	-0.0691	**-0.0352**	**-0.0067**	**-0.0067**	-0.0439	-0.0390	-0.0399
S Δ	0.051	0.093	-0.0056	-0.0015	-0.0033	-0.0018	**0.0132**	**0.0135**	**0.0013**	-0.0015	-0.0022

to standardization—on entailment classification performance using the All-NLI dataset. The first row (O) shows model performance on clean data, followed by rows corresponding to the same corruption types (A, M, H, S). Each entry reflects classification metrics after applying that specific corruption to the input data. MiniLM experiences the most significant performance degradation under noisy inputs, especially in the All (A) and Misspelling (M) settings. BART shows moderate robustness, while mDeBERTa consistently delivers the strongest resilience across all corruption types.

In line with the semantic preservation results, homophone and slang corruptions show relatively milder impact on classification accuracy, suggesting that surface-level changes with high semantic overlap are better tolerated by robust models. Conversely, corruptions that compromise token identity or structure—such as misspellings—lead to the steepest performance drops. This highlights the importance of strong standardization tools like SALDIRAY in preserving downstream task quality under noisy real-world conditions.

Table 3.16: Semantic Preservation on Synthetic Corruption Fixes of NLI Dataset using SALDIRAY

Dataset	Corruption Type	Precision	Recall	F1
Hypothesis	All	0.9446	0.9051	0.9243
	Slang Only	0.9836	0.9775	0.9806
	Homophone Only	0.9599	0.9367	0.9480
	Misspelling Only	0.9380	0.8914	0.9140
Premise	All	0.9391	0.8921	0.9149
	Slang Only	0.9892	0.9839	0.9865
	Homophone Only	0.9551	0.9318	0.9432
	Misspelling Only	0.9330	0.8774	0.9042

Performance on Naturally Corrupted Data

While the previous experiments rely on synthetic corruption, real-world noise is often more irregular, domain dependent, and stylistically diverse. To assess performance under naturally occurring errors, we evaluate BERTopic on a 10K subset of the BEA60K dataset, using three variants: Original (naturally noisy), Standardized (after SALDIRAY), and Gold (human-corrected reference). In addition to intrinsic topic coherence C_v and topic diversity (Div.), we report three clustering agreement metrics with respect to gold labels: the adjusted Rand index (ARI), which measures similarity between two clusterings while correcting for chance; the normalized mutual information (NMI), which quantifies how much information is shared between the predicted and gold cluster assignments on a normalized scale from 0 to 1; and a topic-level Jaccard similarity, which compares the overlap between sets of representative topic keywords in the model output and the gold topics. Together, these metrics capture both global cluster alignment and fine-grained topic-word correspondence under natural noise conditions.

Topic Modeling on BEA60K. Table 3.17 reports intrinsic topic metrics and clustering alignment for two evaluation protocols. Experiment 1 results are presented in Figure 3.3, where each dataset variant is tuned using its own best hyperparameters. The Standardized version clearly outperforms the Original version on all alignment metrics, more than doubling ARI (0.408 vs. 0.198) and improving NMI and Jaccard similarity. Although the Original dataset exhibits slightly higher raw coherence C_v, its low agreement with the gold labels indicates that this coherence is largely driven by noisy lexical patterns rather than meaningful topics. Standardization thus plays a central role in recovering latent semantic structure from naturally corrupted text.

Experiment 2 results are shown in Figure 3.4, where hyperparameters optimized on the Standardized dataset are transferred to both Gold and Standardized data. Under this realistic deployment scenario, Standardized continues to achieve strong alignment (ARI = 0.408, NMI = 0.638), while Gold with transferred parameters reaches a high Jaccard score but lower ARI. Across both protocols, the Original dataset remains the weakest configuration, despite having access to its own tuning. This confirms that natural corruption, rather than suboptimal hyperparameters, is the primary bottleneck, and that SALDIRAY standardization enables more accurate and stable topic modeling on noisy real-world text.

Table 3.17: Topic modeling performance on BEA60K under natural noise. C_v: coherence; Div.: topic diversity; ARI, NMI, and Jaccard: clustering agreement with gold labels. Top performance is bolded.

Dataset	C_v	Div.	ARI	NMI	Jaccard
Experiment 1: Best hyperparameters per dataset					
Gold	0.4830	**0.9681**	-	-	-
Original	**0.5014**	0.9538	0.1979	0.2681	0.3505
Standardized	0.4848	0.9672	**0.4081**	**0.6383**	**0.7243**
Experiment 2: Standardized hyperparameters applied to all datasets					
Gold (with own hyperparameters)	0.4830	**0.9681**	-	-	-
Gold with Standardized Parameters	0.4475	0.9525	0.0859	0.5474	**0.8141**
Standardized (with Standardized Parameters)	**0.4848**	0.9672	**0.4081**	**0.6383**	0.7243

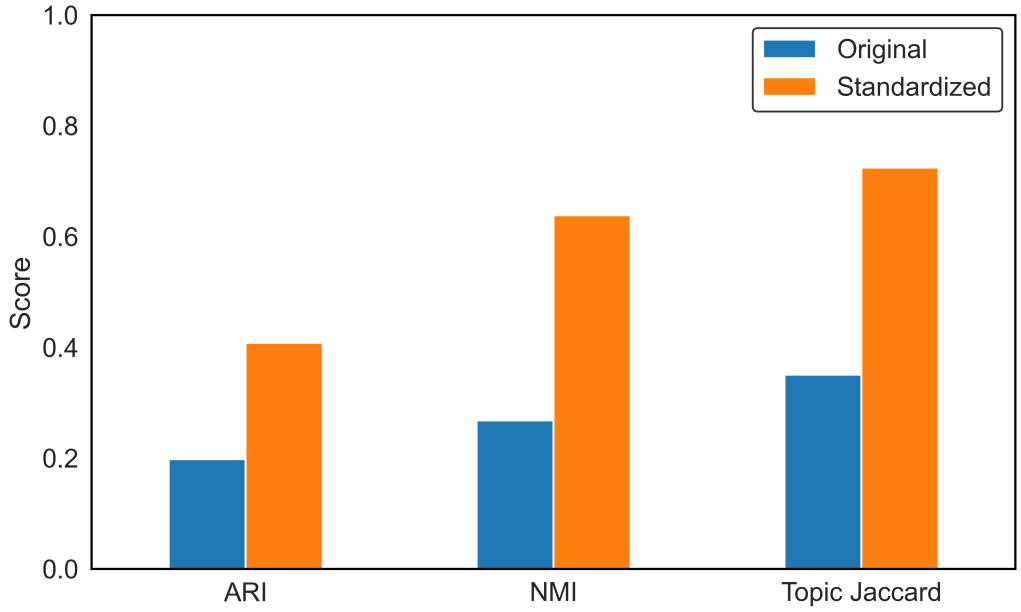


Figure 3.3: Experiment 1: ARI, NMI, and topic-level Jaccard similarity on BEA60K when each dataset variant (Original, Standardized, Gold) is evaluated with its own best hyperparameters.

Across all downstream tasks; topic classification, sentiment analysis, ABSA, NLI, fact verification, and unsupervised topic modeling; the same pattern emerges: configurations that evaluate on corrupted inputs consistently underperform, while replacing those corrupted inputs with SALDIRAY-standardized text restores performance to near-clean levels or better. Standardization does not harm already clean data and provides a robust, task-agnostic preprocessing layer that stabilizes NLP pipelines in the presence of realistic noise.

Runtime and Latency Analysis

To evaluate the practical feasibility of SALDIRAY in real-world applications, we measure its inference throughput under consistent hardware and software settings. All models are evaluated using the same tokenization pipeline, with identical max token limits and batch configurations, on a single NVIDIA RTX 6000 Ada Generation GPU.

Table 3.18 reports two metrics for each model variant: single-instance throughput

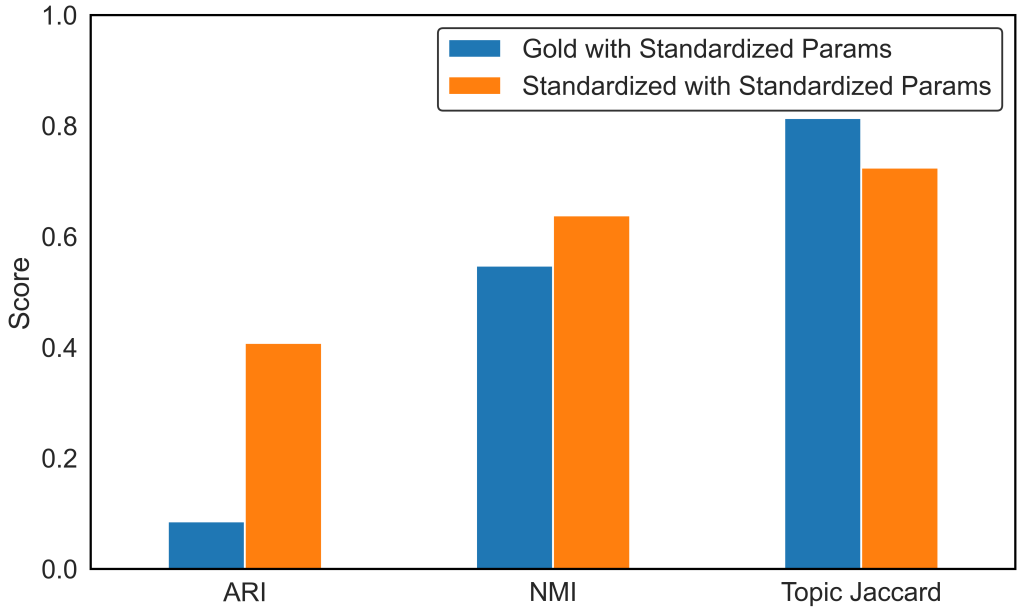


Figure 3.4: Experiment 2: ARI, NMI, and topic-level Jaccard similarity on BEA60K using a single hyperparameter configuration optimized on the Standardized dataset and transferred to both Standardized and Gold corpora.

(in iterations per second) and batched throughput using batch size 8.

The results indicate that SALDIRAY offers competitive inference speed relative to other academic models, with SALDIRAY-LLaMA slightly outperforming the Qwen variant on batched throughput. While BART shows the highest throughput overall, it lacks the corruption-specific design and fine-grained control SALDIRAY provides.

SAGE, another task-agnostic text cleaner, shows notably lower single-instance speed than SALDIRAY, though their batched performance is comparable. API-based models like GPT-4o and GPT-3.5 Turbo exhibit significantly slower local inference, and their latency is further affected by network constraints and limited batching support.

These findings confirm that SALDIRAY achieves an effective balance between corruption robustness and real-time inference cost, making it suitable for both batch and low-latency deployment scenarios.

Table 3.18: Inference Throughput of Standardization Models (Iterations/Second)

Model	1-Batch (it/s)	8-Batch (avg it/s)
SALDIRAY-Qwen	2.57	8.55
SALDIRAY-LLaMA	2.48	8.85
SALDIRAY-T5Gemma	1.25	6.57
SAGE T5-Large	1.83	8.55
BART-Base FT	**10.33**	**31.00**
GPT-4o Mini	1.32	–
GPT-3.5 Turbo	1.66	–

3.4 Conclusion and Future Works

This work introduced SALDIRAY, a unified text standardization framework based on large language models, together with an evaluation protocol to measure how standardization affects downstream natural language processing performance. Experiments were conducted across multiple task types, including topic classification, sentiment analysis, aspect-based sentiment analysis, natural language inference, fact verification, and unsupervised topic modeling on naturally corrupted data. Both synthetic and real-world noise conditions were considered.

Results show that pretrained models without task-specific adaptation exhibit limited robustness to noisy text, while SALDIRAY fine-tuned models achieve consistently strong correction performance across all noise types. When standardized text is used as input, downstream task performance is reliably restored to levels comparable with clean data. Additional experiments on manually typed and naturally corrupted datasets confirm that SALDIRAY generalizes well to authentic human error patterns.

Our findings demonstrate that large language model-based text standardization is an effective and practical solution for improving robustness in real-world language processing systems. Nonetheless, current limitations include the use of English-only corpora, reliance on curated homophone and slang dictionaries, and the added computational cost introduced by an extra standardization step during inference. In latency-sensitive or low-resource environments, the runtime overhead of sequence-

level correction may pose challenges for deployment. Future work will focus on extending multilingual support and exploring tighter integration with knowledge-intensive reasoning tasks.

CHAPTER 4

RAP: RETRIEVAL AUGMENTED PARAPHRASING

Large Language Models have demonstrated strong capabilities across a wide range of natural language generation (NLG) tasks, including question answering, summarization, and dialogue systems. Despite these advances, their performance in domain-specific conversational environments remains limited by two major challenges: (1) hallucination—producing responses that are not grounded in relevant source material [76], and (2) difficulty in maintaining the stylistic consistency, terminology, and tone characteristic of expert-driven long-term client interactions [77]. Although fine-tuning LLMs on domain-specific datasets can alleviate these issues, it is computationally expensive, requires continuous retraining to remain up to date, and may introduce risks of overfitting.

In this study, we introduce Retrieval-Augmented Paraphrasing (RAP), a methodology designed to improve response accuracy and stylistic alignment in domain-specific dialogue generation without requiring fine-tuning. The method is evaluated on a real-world dataset consisting of 10,000 Turkish dietitian–client WhatsApp conversations, with conversation lengths ranging from 30 to 3,000 messages and spanning periods of up to one year. Unlike standard Retrieval-Augmented Generation (RAG) approaches that primarily retrieve external knowledge sources [78], RAP retrieves semantically similar past client messages together with their corresponding expert responses and reformulates the retrieved responses to fit the current conversational context.

Retrieval is performed using Sentence Transformers [79] to compute semantic similarity between the current client message and historical client messages. No temporal contextual similarity metric is employed; retrieval is based solely on

semantic relevance. The retrieved expert responses serve as candidate responses, which are then paraphrased using an LLM. This paraphrasing step preserves the core meaning of the retrieved response while adapting it to the current interaction, promoting both factual grounding and stylistic consistency.

Additionally, the SALDIRAY preprocessing pipeline is incorporated prior to retrieval to normalize noisy and informal conversational inputs. SALDIRAY's performance in this study is evaluated exclusively in terms of its contribution to candidate response retrieval quality. By correcting typos, slang, and grammatical inconsistencies common in WhatsApp messages, the pipeline enhances semantic matching and improves the relevance of retrieved candidates.

The proposed approach is empirically compared against multiple baseline generation strategies: (i) zero-shot generation, (ii) generation conditioned on full conversation history, (iii) standard Retrieval-Augmented Generation, (iv) few-shot prompting, and (v) generation with RAP. These comparisons provide a comprehensive evaluation of RAP's effectiveness in improving response grounding and stylistic alignment in domain-specific conversational systems.

4.1 Related Work

4.1.1 Retrieval-Augmented Generation for Conversational LLMs

Retrieval-Augmented Generation was introduced by [78] as a framework that combines parametric memory (a pre-trained seq2seq model) with non-parametric memory (a dense vector index over external documents such as Wikipedia). A neural retriever selects relevant passages that are incorporated into the generation process. Their experiments on knowledge-intensive tasks, including open-domain question answering, showed that RAG models outperform purely parametric baselines and generate more factual and specific responses. The primary goal of this framework is to improve factual grounding through external knowledge retrieval.

Extending this idea, [80] proposed Atlas, a retrieval-augmented model designed to perform effectively in few-shot settings. Atlas demonstrates that retrieval can

compensate for smaller model sizes by dynamically accessing external knowledge, achieving strong results on benchmarks such as Natural Questions with limited supervision. Similarly, [77] applied retrieval-in-the-loop architectures to knowledge-grounded dialogue systems, showing that integrating retrievers and rankers significantly reduces hallucination and improves factual correctness in conversational agents.

More recent studies explore retrieval for personalization and multi-document reasoning. [81] introduced LAPDOG, which retrieves supplementary narrative content to enhance persona-based dialogue generation through joint retriever–generator training. [82] proposed a model that retrieves documents based on both global topic information and local conversational context to generate more knowledgeable and diverse responses. Efficiency and adaptive retrieval have also been investigated: [83] examined hierarchical and dense retrieval strategies for task-oriented dialogue, while [84] proposed a gating mechanism (RAGate) to determine when retrieval augmentation is necessary for a given conversational turn.

Overall, existing retrieval-augmented conversational approaches primarily focus on grounding responses in external knowledge to reduce hallucination. In contrast, our method retrieves past in-domain conversational examples—specifically, similar client messages paired with expert responses—and uses these responses as candidates for LLM-based paraphrasing. This design shifts the emphasis from external knowledge grounding to preserving domain-specific style and response consistency while maintaining semantic relevance.

4.1.2 LLMs for Paraphrasing

Paraphrase generation has long been studied as a core task in natural language processing. Early neural approaches revisited bilingual pivoting methods within neural machine translation frameworks. [85] proposed a fully neural model that represents paraphrases in a continuous semantic space and generates meaning-preserving alternatives of arbitrary length, outperforming conventional phrase-based pivoting techniques. Addressing limitations of multi-step pivoting, [86] introduced a Transformer-based unified model capable of zero-shot paraphrase generation in a

single step, reducing semantic drift while improving fluency and diversity through pretraining and denoising objectives.

With the emergence of large pre-trained language models, paraphrasing capabilities significantly improved. [87] demonstrated that large generative models can produce high-quality paraphrases not only at the sentence level but also for longer passages without requiring text segmentation. More recent research has explored controlled paraphrasing. For example, [88] proposed syntactically controlled paraphrase networks that generate paraphrases conforming to target syntactic structures, enabling stylistic or structural manipulation while preserving meaning. Similarly, [89] introduced StyleMC, which guides language models to reproduce author-specific stylistic patterns using contrastively learned representations, improving stylistic consistency even when only limited writing samples are available.

Retrieval has also been incorporated into paraphrasing systems to enhance diversity and domain adaptation. [90] proposed Graph-based Retrieval-Augmented Generation (G-RAG), integrating entity- and phrase-level knowledge into paraphrase generation. Their results show improved semantic similarity and diversity across domains, highlighting the benefits of combining retrieval with generative models for paraphrasing tasks.

While prior work demonstrates that LLMs are highly effective at both unconstrained and style-controlled paraphrasing, these approaches typically allow the model to generate freely conditioned on the input text or auxiliary controls. In contrast, our method constrains paraphrasing strictly to retrieved in-domain candidate responses. Rather than generating novel content, the LLM reformulates an existing expert response retrieved from similar conversational contexts. This constraint minimizes the risk of introducing unsupported information while preserving domain-specific tone and stylistic consistency.

4.1.3 Solutions to Hallucination in LLMs

Hallucination—defined as the generation of plausible but factually incorrect or unsupported content—remains a fundamental limitation of LLMs [76]. As sum-

marized by [76], hallucinations occur across diverse NLG tasks including dialogue generation, summarization, question answering, and data-to-text generation. Their survey categorizes both evaluation metrics and mitigation strategies, highlighting that hallucination stems from the probabilistic nature of sequence generation and imperfect internal knowledge representations.

More recent surveys provide structured taxonomies of mitigation strategies. [91] review over thirty hallucination-reduction techniques and group them into prompt engineering methods, retrieval-based approaches, decoding strategies, knowledge graph integration, supervised fine-tuning, and feedback-driven self-refinement. Similarly, [92] analyze hallucination detection benchmarks and mitigation techniques, emphasizing the limitations of retrieval-augmented systems and the need to better define knowledge boundaries in LLMs. These surveys underscore that no single solution fully eliminates hallucination, and trade-offs often exist between grounding, fluency, and efficiency.

Retrieval-based grounding has emerged as one of the most widely adopted solutions. [77] demonstrated that neural retrieval-in-the-loop architectures significantly reduce hallucination in knowledge-grounded dialogue by incorporating external documents into the generation process. By combining retrievers, rankers, and encoder–decoder models, their approach improves factual correctness while maintaining conversational coherence. However, such systems depend on external knowledge sources and accurate document retrieval.

Other approaches focus on detection and correction. For example, [93] propose a framework that first identifies hallucinated entities and relations using NER and entity-relation models, and then mitigates them through prompt-based correction. While effective, detection-based pipelines introduce additional computational complexity and require reliable post-hoc verification mechanisms.

In contrast to methods that rely on external corpora, post-generation verification, or model retraining, our approach reduces hallucination by constraining generation to previously validated in-domain responses. Instead of retrieving external knowledge, we retrieve past expert responses from the same conversational dataset and paraphrase them using an LLM. By grounding responses in internally consis-

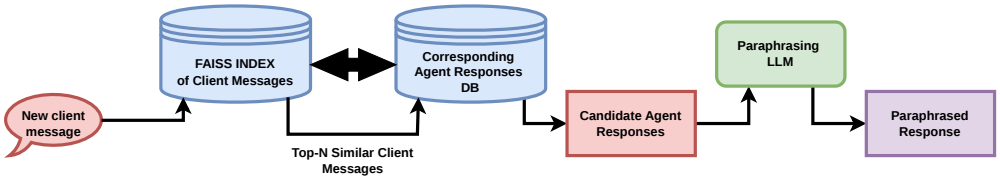


Figure 4.1: RAP usage pipeline

tent historical interactions, our method leverages domain-specific conversational consistency as the primary mechanism for hallucination mitigation.

4.2 Methodology

The proposed RAP framework is based on retrieval of in-domain conversational examples and constrained paraphrasing. The pipeline of the RAP application is visualized in Figure 4.1.

4.2.1 Utterance Construction and Corpus Creation

In synchronous messaging settings, utterance segmentation is typically performed by merging consecutive messages sent by the same speaker within a short temporal window (e.g., one minute) to reconstruct coherent conversational units. However, the WhatsApp conversations in our dataset exhibit partially asynchronous behavior, where responses may not immediately follow preceding messages. To preserve semantic continuity and avoid fragmentation caused by delayed replies, we merge all consecutive messages originating from the same speaker into a single utterance, regardless of temporal gaps. This approach ensures structurally consistent dialogue turns and prevents artificial splitting of semantically related content.

After preprocessing, we construct a structured client–dietitian corpus from the WhatsApp dataset. Each instance consists of three components: (i) the client's current message, (ii) the corresponding dietitian response, and (iii) the recent conversational context comprising the last four dialogue turns preceding the client's message as shown in Figure 4.2. Incorporating this limited dialogue history allows the retrieval system to account for short-term contextual dependencies while main-

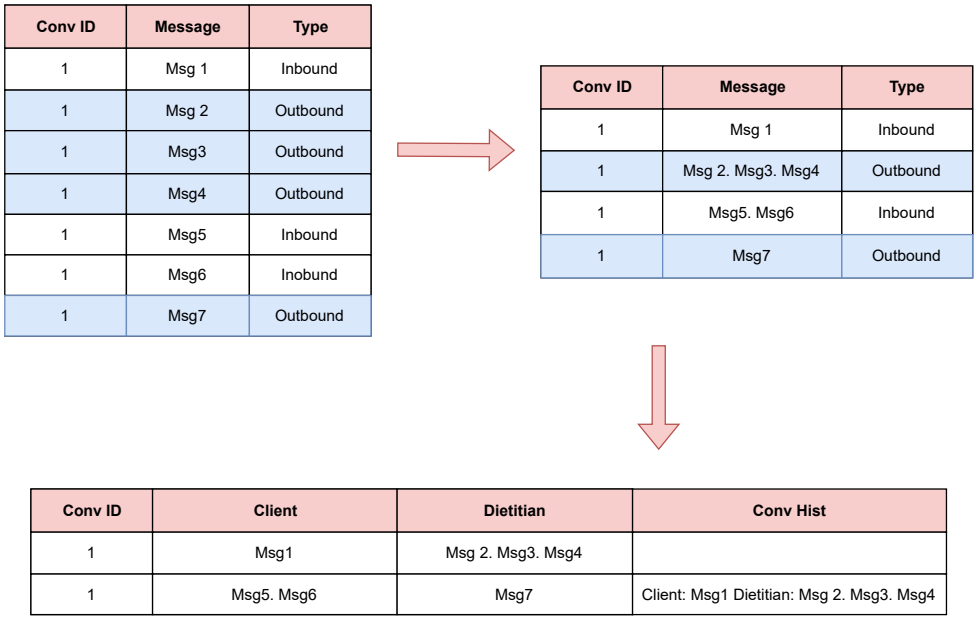


Figure 4.2: Transformation of the client-dietitian conversation dataset

taining computational efficiency. The resulting dataset serves as the retrieval base for candidate response selection and downstream generation experiments.

4.2.2 Knowledge Base Generation

In addition to the client–agent corpus, we construct a domain-specific knowledge base for use in the benchmark RAG setup. We begin by filtering the merged conversational dataset to retain only messages associated with food-related topics, ensuring that the extracted content focuses strictly on nutrition and diet guidance. Using this filtered subset, an LLM is prompted to extract explicit, fact-based knowledge communicated by dietitians in response to client messages. These include dietary recommendations, nutritional explanations, portion advice, and food substitutions grounded in the dialogue context.

To ensure that only accurate and contextually grounded information is retained, the structured outputs are validated by prompting a second LLM to confirm whether each candidate statement is both factual and relevant to the corresponding agent message. This two-stage procedure enhances the quality and reliability of the

knowledge base. The extraction and validation prompts used in this process are detailed in Appendix A.1.

This curated knowledge base is not used in the RAP method itself. Instead, it serves as the retrieval source for the RAG baseline, enabling a fair comparison between knowledge-based retrieval and our retrieval-and-paraphrasing approach.

4.2.3 Retrieval of Candidate Responses

Each client message in the corpus is encoded using Sentence Transformers [79]. When a new client message is received, it is encoded in the same embedding space, and cosine similarity is computed against all client messages in the corpus.

The most semantically similar client messages are retrieved. For each retrieved client message, the associated agent (dietitian) responses are collected. These responses serve as candidate responses.

4.2.4 LLM-Based Paraphrasing

The retrieved dietitian responses are provided to the LLM together with the new client message. The model is instructed to select the most appropriate candidate response and paraphrase it so that it aligns with the current client message while strictly preserving its original meaning. The generation process is explicitly constrained to prevent the addition, removal, or modification of substantive information.

Two prompting strategies are employed: one without conversational history and one incorporating recent dialogue context.

In the first setting, the model receives (i) the client message and (ii) a list of candidate responses. It is instructed to choose the most suitable candidate and rewrite it to match the client's message. The prompt explicitly forbids adding new information, removing existing content, or altering the semantic meaning of the selected response. The output must be produced in Turkish and follow a predefined format.

> You are a dietitian. You will receive input in the following format:
> Client message: [Client message]
> Candidate responses:
> [Candidate 1]
> [Candidate 2]
> [Candidate 3]
> Select the most appropriate candidate response and rewrite it so that it fits the client's message. Do not add any new information. Do not remove or change any statements. Only rewrite the selected candidate to make it appropriate for the client. Produce the output only in Turkish and follow this format:
> Generated response: ...

In the second setting, recent conversation history is additionally provided to improve contextual alignment. The model receives (i) the client message, (ii) recent dialogue history, and (iii) candidate responses. While the same strict content-preservation constraints apply, the model is instructed to consider the recent conversation to better contextualize the client's current message.

> You are a dietitian. You will receive input in the following format:
> Client message: [Client message]
> Recent conversation history: [Dialogue turns]
> Candidate responses:
> [Candidate 1]
> [Candidate 2]
> [Candidate 3]
> Select the most appropriate candidate response and rewrite it according to the client's message and the recent conversation history. Do not add new information. Do not remove or alter any statements. Only rewrite the selected candidate. Produce the output only in Turkish and follow this format:
> Generated response: ...

Multiple paraphrased outputs may be generated depending on the decoding configuration. The final response is selected based on semantic similarity to the original candidate response to ensure strict content preservation.

By constraining generation to controlled paraphrasing of previously validated in-domain responses, this approach reduces hallucination risk compared to standard

retrieval-augmented generation pipelines, while preserving stylistic consistency and domain-specific language.

4.3 Experiments

In this section, we describe the datasets, experimental configuration, and evaluation methodology used to assess the effectiveness of the proposed RAP framework. We further present the experimental results and analyze the performance of RAP in comparison with several baseline approaches.

4.3.1 Datasets

We evaluate our approach on two datasets: (1) an in-house WhatsApp dietitian–client dataset and (2) a public persuasive dialogue dataset to assess generalizability across domains.

4.3.1.1 Client-Dietitian WhatsApp Dataset

Our primary dataset consists of a proprietary WhatsApp corpus containing **10,000** real-world conversations between clients and professional dietitians. Dialogue lengths range from **30** to **3,000** messages per conversation.

As described in the methodology, consecutive messages sent by the same speaker within a one-minute interval are merged to form coherent utterances. After utterance construction and standardization using SALDIRAY, we generate the following corpora for experiments:

- **Client–Agent Response Corpus (RAP Retrieval Corpus):** Each entry consists of a client message paired with its corresponding dietitian response. This corpus is used to retrieve semantically similar client messages and their associated agent responses as candidate responses in RAP.
- **Food Knowledge Base (RAG Baseline):** From the merged dataset, food-related

conversations are filtered. Using these filtered client–agent pairs, an LLM extracts explicit nutrition-related knowledge expressed by the dietitian. The extracted statements are aggregated to form a structured knowledge base, which is used exclusively in the RAG-based benchmark.

4.3.1.2 Persuasion for Good Dataset

To evaluate the generalizability of our approach, we also conduct experiments on the [94] dataset. This corpus contains **1,017** online conversations collected via Amazon Mechanical Turk, where one participant (the persuader) attempts to convince the other (the persuadee) to donate to a charity. The dataset includes demographic information and psychological survey responses, and **300** conversations contain sentence-level human annotations for persuasion related dialogue acts and sentiment.

For this dataset, we construct experimental corpora analogous to those used in the WhatsApp dataset:

- **Client–Agent Response Corpus:** Each persuadee message is paired with the corresponding persuader response to create the retrieval corpus used in RAP. No utterance merging is applied to this dataset.

- **Knowledge Base for RAG Baseline:** Using the full dataset without topic filtering, an LLM extracts persuasive strategies, factual claims, and reasoning patterns expressed by persuaders. The extracted statements are aggregated into a persuasion knowledge base used in the RAG baseline experiments.

By constructing parallel corpora for both datasets, we ensure comparable evaluation settings across domains while assessing the robustness and transferability of the proposed approach.

4.3.2 Experimental Setup

All experiments are conducted under a unified retrieval and generation configuration to ensure fair comparison across methods.

Text Embeddings For both RAP and RAG approaches, we use the paraphrase-multilingual-mpnet-base-v2 model, a variant in the Masked and Permuted Language Modeling architecture [95], within the Sentence Transformers framework to obtain dense vector representations of text. This multilingual embedding model is used to:

- Encode client (or persuadee) messages for retrieval in RAP,
- Encode knowledge entries for retrieval in the RAG baseline,
- Compute cosine similarity during candidate selection and response filtering.

Using the same embedding model across all approaches ensures consistency in semantic similarity computations.

Retrieval Mechanism For the RAG baseline, we index the extracted knowledge base using the FAISS (Facebook AI Similarity Search) library [96] in Python. Dense embeddings of knowledge statements are stored in the FAISS index, and the most relevant entries are retrieved using cosine similarity.

For RAP, embeddings of client messages in the client–agent corpus are precomputed. When a new client message is received, its embedding is compared to all corpus embeddings using cosine similarity, and the most similar client messages are retrieved. The associated agent responses are then used as candidate responses.

The same value of N is used for both RAP and RAG to ensure comparability between candidate response retrieval and knowledge retrieval.

Language Model All generation tasks are performed using the LLaMA 3.2 8B model [97]. The model is used in two stages:

- **Knowledge Extraction**: To extract domain-specific knowledge from the dialogue data for constructing the knowledge base used in the RAG baseline.
- **Response Paraphrasing**: In the RAP framework, to paraphrase retrieved candidate responses while preserving semantic content and avoiding the introduction of additional information.

Using the same LLM for both knowledge extraction and response generation ensures that performance differences arise from retrieval strategy rather than model variability.

4.3.3 Evaluation Metrics

We evaluate our system at two levels: (1) knowledge extraction quality and (2) response generation quality.

Knowledge Extraction Evaluation To assess the quality of the automatically extracted knowledge bases for both datasets, we construct a randomly sampled gold evaluation set from each corpus. Human annotators evaluate the extracted knowledge statements according to the following criteria:

- **Meaningfulness**: Whether the extracted statement represents a coherent and complete piece of knowledge.
- **Contextual Relatedness**: Whether the extracted knowledge is correctly grounded in the original client–agent interaction.
- **Factual Consistency**: Whether the extracted content faithfully reflects what was stated in the dialogue without distortion.

Automatic Evaluation of Response Generation For the main benchmarking experiments, we compare all approaches, zero-shot generation, generation with conversation history, few-shot prompting, RAG, and RAP, against the original agent responses in the dataset.

We use BERTScore for the automatic metric. BERTScore [75] computes embedding-based semantic similarity, providing a more robust measure of meaning preservation beyond lexical overlap. Each token in the candidate sentence is paired with the most similar token in the reference sentence, and the process is repeated in the opposite direction to calculate precision and recall, which are then combined to obtain the F1 score.

Human Evaluation of Response Quality To better assess qualitative aspects of the generated responses, we conduct human evaluation based on the following criteria:

- **Relevance**: The degree to which the response appropriately addresses the client's message.
- **Hallucination**: Whether the response introduces unsupported or fabricated information not grounded in the conversation or retrieved content.
- **Helpfulness**: Whether the response provides actionable and useful guidance to the client.

Each criterion is scored using a Likert scale [98], enabling comparative analysis across generation methods beyond automatic similarity metrics. Two annotators conducted the evaluations; while not domain experts, they are AI researchers familiar with the application domain and were provided with domain-specific data to support consistent and informed judgment.

4.3.4 Results

This section presents a comprehensive evaluation of the proposed Retrieval-Augmented Paraphrasing method. We assess the approach through both automatic and human evaluations, including knowledge base quality analysis, ablation studies, and benchmarking against standard generation baselines. Automatic evaluation is conducted using BERTScore (Precision, Recall, and F1), which captures semantic similarity more effectively than surface-level metrics in paraphrastic generation settings. Human evaluations further examine relevance, helpfulness, factual consistency, and

Table 4.1: Human Evaluation of Extracted Knowledge Base

Metric	**Mean Score**	**Cohen's Kappa**
Meaningfulness	76.50%	0.6951
Contextual Relatedness	87.50%	0.6807
Factual Consistency	59.00%	0.6705

hallucination. Across all experimental settings, RAP consistently demonstrates superior performance, particularly when multiple candidates ($k = 5$) and recent conversation history are incorporated.

4.3.4.1 Knowledge Base Evaluation

The quality of the constructed knowledge base was evaluated through human annotation across three criteria: meaningfulness, contextual relatedness, and factual consistency. Two annotators independently rated randomly sampled entries. Inter-annotator agreement scores are reported directly in Table 4.1, with Cohen's κ values ranging between 0.67 and 0.70, indicating substantial agreement across all dimensions.

Contextual relatedness achieves the highest mean score (87.50%), suggesting strong alignment between extracted entries and their conversational context. Meaningfulness also scores positively, indicating that the entries form coherent knowledge units. Although factual consistency is comparatively lower, agreement remains substantial, supporting the overall reliability of the retrieval base. Examples of the extracted knowledge base entries are listed in Appendix A.2.

4.3.4.2 Ablation Study: Number of Retrieved Candidates

We evaluate RAP with $k = 1$, $k = 3$, and $k = 5$ retrieved candidates using BERTScore precision (P), recall (R), and F1.

Tables 4.2 and 4.3 demonstrate that increasing the number of retrieved candidates

Table 4.2: Effect of Number of Retrieved Candidates on RAP. For each metric, the highest scores are highlighted.

k	**Precision**	**Recall**	**F1-Score**
1	0.4862	0.4428	0.4598
3	0.4955	0.4642	0.4759
5	**0.5078**	**0.4850**	**0.4925**

Table 4.3: Effect of number of retrieved candidates on RAP with conversation history. For each metric, the highest scores are highlighted.

k	**Precision**	**Recall**	**F1-Score**
1	0.4887	0.4542	0.4674
3	0.4884	0.4762	0.4787
5	**0.5063**	**0.4983**	**0.4958**

consistently improves performance. The strongest results are observed at $k = 5$, both with and without conversation history. Improvements are primarily driven by recall gains, indicating that a larger candidate pool increases the likelihood of selecting semantically appropriate responses.

4.3.4.3 Ablation Study: Effect of Paraphrasing

We compare RAP with and without paraphrasing across different values of k.

As shown in Table 4.4, paraphrasing generally improves F1, particularly at higher k values. The best performance is achieved when paraphrasing is combined with $k = 5$, demonstrating that controlled rewriting enhances contextual adaptation while preserving semantic alignment.

4.3.4.4 Ablation Study: Conversation History Integration

We evaluate the RAP with and without a recent conversation history.

Table 4.4: Effect of paraphrasing on RAP performance. For each metric, the highest scores are highlighted.

Setting	k	**Precision**	**Recall**	**F1-Score**
Paraphrasing	1	**0.4862**	0.4428	0.4598
No Paraphrasing	1	0.4800	**0.4546**	**0.4634**
Paraphrasing	3	**0.4955**	0.4642	**0.4759**
No Paraphrasing	3	0.4848	**0.4732**	0.4756
Paraphrasing	5	**0.5078**	**0.4850**	**0.4925**
No Paraphrasing	5	0.4823	0.4762	0.4759

Table 4.5: Effect of conversation history on RAP. For each metric, the highest scores are highlighted.

Setting	k	**Precision**	**Recall**	**F1-Score**
Without History	1	0.4862	0.4428	0.4598
With History	1	**0.4887**	**0.4542**	**0.4674**
Without History	3	**0.4955**	0.4642	0.4759
With History	3	0.4884	**0.4762**	**0.4787**
Without History	5	**0.5078**	0.4850	0.4925
With History	5	0.5063	**0.4983**	**0.4958**

Table 4.5 shows that incorporating conversation history consistently improves recall and F1 across most settings. The highest overall performance is obtained with history integration at $k = 5$, highlighting the importance of contextual grounding in response selection.

4.3.4.5 Ablation Study: Standardization

We evaluate the RAP with and without the standardized version of the test dataset.

Table 4.6 presents the effect of applying the SALDIRAY standardization step to the

Table 4.6: Effect of SALDIRAY standardization on RAP. For each metric, the highest scores are highlighted.

Setting	k	**Precision**	**Recall**	**F1-Score**
Without Standardization	5	0.5063	0.4983	0.4958
With Standardization	5	**0.5081**	0.4983	**0.4965**

Table 4.7: Benchmarking across generation methods. For each metric, the highest scores are highlighted.

Method	**Precision**	**Recall**	**F1-Score**
Zero-shot	0.3936	0.4359	0.4116
Zero-shot + History	0.4108	0.4622	0.4333
Few-shot	0.4521	0.4907	0.4681
Few-shot + History	0.4492	0.4804	0.4619
RAG	0.4044	0.4517	0.4251
RAG + History	0.4097	0.4677	0.4350
RAP	**0.5063**	**0.4983**	**0.4958**

test dataset prior to response generation with RAP. The results show a slight but consistent improvement in both precision and F1-score when the input text is standardized. While recall remains unchanged, the gains in precision indicate that SALDIRAY helps align the retrieved and paraphrased responses more accurately with the original agent messages. The findings support the inclusion of robust preprocessing as a lightweight yet impactful enhancement to controlled generation pipelines.

4.3.4.6 Automatic Benchmarking Against Baselines

We benchmark RAP against zero-shot, few-shot, and RAG-based baselines.

As presented in Table 4.7, RAP substantially outperforms all baselines across precision, recall, and F1. While conversation history improves baseline methods, the gains remain smaller than those observed for RAP, demonstrating the effectiveness

Table 4.8: Human evaluation results for the best performing methods. For each metric, the highest scores in **relevance** and **helpfulness**, and the lowest score in **hallucination**, are highlighted.

Method	Relevance	Hallucination	Helpfulness
Best Zero-shot	3.3333	3.0486	3.1657
Best Few-shot	3.8056	2.3806	3.0764
Best RAG	3.9764	2.1875	**3.3486**
RAP	**4.1722**	**1.4028**	3.1194

of retrieval-augmented paraphrasing.

4.3.4.7 Human Evaluation of Best Configurations

We compare the best-performing variants through human evaluation across relevance, hallucination, and helpfulness. The evaluation was conducted on 100 samples selected from the RAP configuration with $k = 5$. To ensure a balanced and fair assessment, the sample set consisted of the top 50 and bottom 50 outputs according to automatic evaluation scores. This selection strategy allowed us to examine both strong and weak cases of the model's performance. Inter-annotator agreement yielded a Cohen's κ of 0.74, indicating substantial consistency.

Table 4.8 confirms the automatic evaluation results. RAP achieves the highest relevance score and the lowest hallucination rate, indicating strong semantic fidelity. Overall, the findings demonstrate that Retrieval-Augmented Paraphrasing with $k = 5$ and conversation history integration provides the most robust performance across both automatic and human evaluations. Example responses generated with RAP are listed in Appendix B with the corresponding client messages.

The split analysis in Table 4.9, where samples are grouped by top-50 and bottom-50 based on automatic evaluation rankings, reveals how human judgments vary across different response quality levels. Despite being selected from lower-scoring outputs according to automatic metrics, RAP maintains strong human evaluation scores—particularly in relevance and hallucination—demonstrating its robustness

Table 4.9: Human Evaluation on Top-50 vs. Bottom-50 Samples (Higher is better for relevance and helpfulness; lower is better for hallucination)

Criterion	Method	Top-50	Bottom-50
Relevance	RAP	**4.60**	4.20
	Few-shot	**4.60**	4.00
	Zero-shot + History	4.05	4.05
	RAG + History	4.05	**4.15**
Hallucination	RAP	**1.25**	1.75
	Few-shot	**1.85**	2.05
	Zero-shot + History	2.40	2.40
	RAG + History	**1.75**	2.15
Helpfulness	RAP	3.60	**3.70**
	Few-shot	**3.85**	3.45
	Zero-shot + History	3.65	**3.75**
	RAG + History	**3.80**	3.45

across diverse response scenarios. Its hallucination scores remain consistently low, even in the bottom-50, highlighting the reliability of its paraphrasing mechanism. Helpfulness scores for RAP also show minimal degradation across subsets, unlike some baselines whose scores drop more noticeably. These findings suggest that RAP's controlled generation strategy yields human-perceived quality that is stable and trustworthy, even when the responses are less aligned according to automatic similarity metrics.

4.3.4.8 Automatic Evaluation of Persuasion for Good Dataset

To assess the generalizability of RAP beyond the dietitian-client setting, we conducted automatic evaluations on the Persuasion for Good dataset. This dataset features persuasive dialogues with rich contextual dependencies and social reasoning, offering a challenging testbed for controlled generation.

As shown in Table 4.10, RAP consistently outperforms zero-shot, few-shot, and

RAG-style baselines across precision, recall, and F1-score. The best-performing configuration, RAP with conversation history and $k = 5$ retrieved candidates, achieves the highest F1 (0.5851), highlighting its strong semantic preservation and adaptability to new domains. Notably, even without history integration, RAP maintains competitive performance, underscoring its robustness to varying dialogue structures. These results confirm that the retrieval-augmented paraphrasing strategy generalizes well across dialogue styles and content domains.

Table 4.10: Benchmarking across all methods on Persuasion for Good dataset. For each metric, the highest scores are highlighted.

Method	**Precision**	**Recall**	**F1-Score**
Zero-shot + History	0.5238	0.6233	0.5675
Zero-shot	0.5270	0.6183	0.5672
Few-shot + History	0.5505	0.6222	0.5822
Few-shot	0.5468	0.6214	0.5789
RAG Conv	0.5339	0.6133	0.5692
RAG	0.5308	0.6164	0.5689
RAP + History ($k = 5$)	**0.5801**	**0.5956**	**0.5851**
RAP + History ($k = 3$)	0.5798	0.5877	0.5810
RAP + History ($k = 1$)	0.5683	0.5497	0.5560
RAP ($k = 5$)	0.5755	0.5907	0.5805
RAP ($k = 3$)	0.5770	0.5840	0.5778
RAP ($k = 1$)	0.5619	0.5395	0.5476

4.3.4.9 Human Evaluation of Best Configurations for Persuasion for Good

To complement the automatic evaluation, we conducted human assessments of the top-performing variants on the Persuasion for Good dataset. Annotators rated generated responses on relevance, hallucination, and helpfulness using a 5-point Likert scale.

As summarized in Table 4.11, RAP again demonstrates the strongest overall performance. It achieves the lowest hallucination score (1.176), confirming its ef-

Table 4.11: Human evaluation results for the best performing methods for Persuasion for Good dataset. For each metric, the highest scores in **relevance** and **helpfulness**, and the lowest score in **hallucination**, are highlighted.

Method	Relevance	Hallucination	Helpfulness
Best Zero-shot	4.6021	2.0515	4.2265
Best Few-shot	**4.7505**	1.871	4.2357
Best RAG	4.7017	1.4038	4.3576
RAP	4.7141	**1.176**	**4.3746**

fectiveness in avoiding unsupported or fabricated content even in open-domain persuasive interactions. RAP also secures the highest helpfulness score (4.3746), slightly outperforming the RAG baseline. While few-shot prompting yields the highest relevance score, RAP follows closely, offering a strong balance across all metrics. These results reinforce the conclusion that RAP's retrieval-then-paraphrase mechanism provides robust factual grounding without sacrificing engagement or guidance quality.

4.4 Conclusion and Discussion

This study introduced Retrieval-Augmented Paraphrasing, a controlled generation framework designed to improve response quality in task-oriented conversational settings. Across both automatic and human evaluations, conducted on our client–dietitian dataset and the Persuasion for Good dataset, RAP consistently outperformed zero-shot, few-shot, and standard RAG baselines in terms of semantic alignment and factual reliability. The results demonstrate that integrating retrieval with constrained paraphrasing provides a strong balance between contextual adaptation and content fidelity.

A key contribution of RAP lies in its hallucination-resistant design. By restricting generation to semantically relevant and previously validated in-domain responses, RAP substantially reduces the risk of unsupported or fabricated content. Human evaluation results confirm that RAP achieves the lowest hallucination scores among

all compared methods. This reliability is particularly important in high-stakes advisory domains, where incorrect or fabricated information can have serious consequences.

Although RAP achieves strong relevance and factual grounding, its helpfulness scores remain moderate compared to some retrieval-based baselines. This outcome reflects an inherent trade-off: stricter control over generation limits the introduction of novel or elaborative content. To avoid hallucination, RAP sometimes tends to produce safer, more generic responses such as *"Afiyet olsun, güzel bir öğün olmuş."* While this constraint may slightly reduce perceived helpfulness in some cases, it significantly enhances safety and trustworthiness. In domains where accuracy and consistency are prioritized over creativity, this trade-off is desirable.

The ablation studies further indicate that performance improves with a larger retrieval pool ($k = 5$) and with the integration of recent conversation history. These findings highlight the importance of both semantic diversity and contextual grounding in controlled response generation. Moreover, RAP achieves these gains without model fine-tuning, making it computationally efficient and easily adaptable to new domains.

Overall, RAP offers a robust and domain-adaptive alternative to unconstrained generation and traditional retrieval-augmented methods. By combining targeted retrieval with paraphrasing-based adaptation, it delivers strong semantic alignment while significantly mitigating hallucination risk. Future work may explore hybrid strategies that preserve RAP's safety guarantees while selectively enhancing helpfulness through controlled expansion mechanisms.

CHAPTER 5

CONCLUSION

This thesis introduced three core contributions aimed at improving the reliability, contextual awareness, and factual consistency of domain-specific conversational systems. Each component targets a distinct challenge commonly encountered in real-world messaging platforms, such as informal client communication, evolving conversational patterns, and hallucination in language model outputs.

The first contribution, Temporal Contextual Similarity, proposed a time-series forecasting framework that models the progression of client–agent conversations through topic distributions over time. By representing dialogue history as temporally ordered topic sequences and identifying similar past trajectories across users, TCS enables proactive forecasting of near-future conversational topics. Unlike traditional forecasting methods that rely on parametric trend extrapolation, TCS leverages structural similarity and Gaussian Mixture Model–based encoding for greater flexibility and interpretability. Experimental results on both domain-specific and public datasets demonstrate that TCS consistently outperforms widely used explainable forecasting baselines such as ARIMA, Prophet, and Holt's method. While smaller benchmark datasets present certain modeling limitations, such as reduced convergence reliability for probabilistic models, the inclusion of these benchmarks enhances the transparency and generalizability of the evaluation.

The second contribution, SALDIRAY, presented a multilingual, task-agnostic normalization framework designed to handle common textual anomalies in user-generated content. These include typographic errors, abbreviations, slang, and homophones—phenomena that can degrade the performance of downstream NLP tasks. In this work, SALDIRAY was used as a preprocessing module within the topic

modeling pipeline for TCS. By transforming noisy inputs into their standardized forms, SALDIRAY improved the coherence and interpretability of discovered topics, which directly enhanced the accuracy of temporal forecasting. Its flexible noise injection framework and strong downstream task performance make it a robust foundation for future work on normalization across noisy language domains.

The third contribution, Retrieval-Augmented Paraphrasing, addressed the challenge of hallucination in response generation. RAP retrieves real past responses from similar dialogue contexts and prompts a large language model to paraphrase them in a user-adapted way. This design grounds the output in validated human responses, reducing the risk of hallucination while preserving relevance and stylistic alignment. Evaluation results show that RAP achieves the lowest hallucination scores among all tested methods, with human ratings confirming its factual consistency and reliability. A known limitation of RAP is its tendency to produce safe or generic responses (e.g., "Afiyet olsun, güzel bir öğün olmuş") due to its constrained generation strategy. However, this trade-off is appropriate in high-stakes domains like health, where factual correctness is prioritized over elaboration.

Constructing a domain knowledge base was also necessary for RAP's baseline comparison with retrieval-augmented generation systems. To this end, we automatically extracted structured dietary knowledge from the corpus using LLM-based prompts, followed by a second validation step to ensure factual relevance. Details of this extraction and validation procedure are provided in Appendix A.1. While this approach enabled effective comparison, future work may explore the use of manually curated or hybrid knowledge resources for improved reliability.

Looking ahead, each contribution presents promising directions for extension. For TCS, future work includes applying the framework to multidimensional time-series forecasting problems, such as in financial domains (e.g., stock or cryptocurrency forecasting) or system monitoring tasks. For SALDIRAY, improving normalization for low-resource languages and expanding coverage for domain-specific slang are key goals. For RAP, further improvements may come from enhancing the retrieval mechanism and expanding human evaluations on more diverse datasets.

In summary, this thesis introduces TCS for topic-based temporal forecasting, SALDIRAY

for domain-agnostic text normalization, and RAP for hallucination-resistant response generation. Together, these components form a practical toolkit for building conversational systems that are trustworthy, adaptive, and aligned with real-world client communication needs.

REFERENCES

[1] T. Dehesh, H. Mardani-Fard, and P. Dehesh, "Forecasting of covid-19 confirmed cases in different countries with arima models," *MedRxiv*, 2020.

[2] A. I. Saba and A. H. Elsheikh, "Forecasting the prevalence of covid-19 outbreak in egypt using nonlinear autoregressive artificial neural networks," *Process safety and environmental protection*, vol. 141, pp. 1–8, 2020.

[3] Z. Hu, Q. Ge, S. Li, L. Jin, and M. Xiong, "Artificial intelligence forecasting of covid-19 in china," *arXiv preprint arXiv:2002.07112*, 2020.

[4] N. Ayoobi, D. Sharifrazi, R. Alizadehsani, A. Shoeibi, J. M. Gorriz, H. Moosaei, A. Khosravi, S. Nahavandi, A. G. Chofreh, F. A. Goni, *et al.*, "Time series forecasting of new cases and new deaths rate for covid-19 using deep learning methods," *Results in Physics*, vol. 27, p. 104495, 2021.

[5] L. Xu, R. Magar, and A. Barati Farimani, "Forecasting covid-19 new cases using deep learning methods," *Computers in Biology and Medicine*, vol. 144, p. 105342, 2022.

[6] Y. Gautam, "Transfer learning for covid-19 cases and deaths forecast using lstm network," *ISA transactions*, vol. 124, pp. 41–56, 2022.

[7] A. Singhal, P. Singh, B. Lall, and S. D. Joshi, "Modeling and prediction of covid-19 pandemic using gaussian mixture model," *Chaos, Solitons & Fractals*, vol. 138, p. 110023, 2020.

[8] I. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016.

[9] T. L. Smestad, "Personality matters! improving the user experience of chatbot interfaces-personality provides a stable pattern to guide the design and behaviour of conversational agents," Master's thesis, NTNU, 2018.

[10] M. Shumanov and L. Johnson, "Making conversations with chatbots more personalized," *Computers in human behavior*, vol. 117, p. 106627, 2021.

[11] Y. B. Gumiel, L. E. Silva e Oliveira, V. Claveau, N. Grabar, E. C. Paraiso, C. Moro, and D. R. Carvalho, "Temporal relation extraction in clinical texts: a systematic review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 7, pp. 1–36, 2021.

[12] F. Westin, "Time, technique and text: scoping review of temporal information extraction and categorisation in documents," *Journal of Documentation*, vol. 81, no. 7, pp. 135–156, 2025.

[13] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[14] V. Kumar, V. Tripathi, and B. Pant, "Learning unsupervised visual representations using 3d convolutional autoencoder with temporal contrastive modeling for video retrieval," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 7, no. 2, pp. 272–287, 2022.

[15] D. R. Kubal and A. S. Nagvenkar, "Leveraging multilingual models for robust grammatical error correction across low-resource languages," in *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pp. 505–510, 2025.

[16] E. Külah, Y. M. Çetinkaya, A. G. Özer, and H. Alemdar, "Covid-19 forecasting using shifted gaussian mixture model with similarity-based estimation," *Expert Systems with Applications*, vol. 214, p. 119034, 2023.

[17] E. Kim, C. Park, and B. Chang, "Share: Shared memory-aware open-domain long-term dialogue dataset constructed from movie script," in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14474–14498, 2025.

[18] Tanaos, "tanaos-topic-classification-v1: A small but performant topic classification model." `https://huggingface.co/tanaos/tanaos-topic-classification-v1`, 2024. Fine-tuned RoBERTa-base model for 15-category topic classification.

[19] D. A. Reynolds, "Gaussian mixture models.," *Encyclopedia of biometrics*, vol. 741, pp. 659–663, 2009.

[20] X.-Y. Zhou, J. S. Lim, I. Kwon, *et al.*, "Em algorithm with gmm and naive bayesian to implement missing values," *Advanced Science and Technology Letters*, vol. 46, pp. 1–5, 2014.

[21] G. Schwarz, "Estimating the dimension of a model," *The annals of statistics*, pp. 461–464, 1978.

[22] M. L. McHugh, "Interrater reliability: the kappa statistic," *Biochemia medica*, vol. 22, no. 3, pp. 276–282, 2012.

[23] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[24] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[25] C. C. Holt, "Forecasting seasonals and trends by exponentially weighted moving averages," *International journal of forecasting*, vol. 20, no. 1, pp. 5–10, 2004.

[26] K. Taha, P. D. Yoo, C. Yeun, D. Homouz, and A. Taha, "A comprehensive survey of text classification techniques and their research applications: Observational and experimental insights," *Computer Science Review*, vol. 54, p. 100664, 2024.

[27] S. Chen, Y. Zhang, and Q. Yang, "Multi-task learning in natural language processing: An overview," *ACM Computing Surveys*, vol. 56, no. 12, pp. 1–32, 2024.

[28] Y. Sang, "Robustness of fine-tuned llms under noisy retrieval inputs," in *2025 6th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)*, pp. 417–420, IEEE, 2025.

[29] G. Dong, J. Zhao, T. Hui, D. Guo, W. Wang, B. Feng, Y. Qiu, Z. Gongque, K. He, Z. Wang, *et al.*, "Revisit input perturbation problems for llms: A unified robustness evaluation framework for noisy slot filling task," in *CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 682–694, Springer, 2023.

[30] A. Aliakbarzadeh, L. Flek, and A. Karimi, "Exploring robustness of multilingual llms on real-world noisy data," *arXiv preprint arXiv:2501.08322*, 2025.

[31] Y. Belinkov and Y. Bisk, "Synthetic and natural noise both break neural machine translation," in *International Conference on Learning Representations*, 2018.

[32] O. Büyük and L. M. Arslan, "Learning from mistakes: Improving spelling correction performance with automatic generation of realistic misspellings," *Expert Systems*, vol. 38, no. 5, p. e12692, 2021.

[33] M. Klemen, M. Božič, Š. A. Holdt, and M. Robnik-Šikonja, "Neural spell-checker: Beyond words with synthetic data generation," in *International Conference on Text, Speech, and Dialogue*, pp. 85–96, Springer, 2024.

[34] N. Martynov, M. Baushenko, A. Kozlova, K. Kolomeytseva, A. Abramov, and A. Fenogenova, "A methodology for generative spelling correction via natural spelling errors emulation across multiple domains and languages," in *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 138–155, 2024.

[35] C. G. Turhan, "Leveraging large language models for spelling correction in turkish," *PeerJ Computer Science*, vol. 11, p. e2889, 2025.

[36] J. Guo, T. N. Sainath, and R. J. Weiss, "A spelling correction model for end-to-end speech recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5651–5655, IEEE, 2019.

[37] O. Hrinchuk, M. Popova, and B. Ginsburg, "Correction of automatic speech recognition with transformer sequence-to-sequence model," in *Icassp 2020-2020 ieee international conference on acoustics, speech and signal processing (icassp)*, pp. 7074–7078, IEEE, 2020.

[38] O. Guhr, "Spelling-correction-english-base." `https://huggingface.co/oliverguhr/spelling-correction-english-base`, 2021. Online; accessed 2025-02-15.

[39] A. Xu, T. Feng, S. H. Kim, S. Bishop, C. Lord, and S. Narayanan, "Large language models based asr error correction for child conversations," in *Proc. Interspeech 2025*, pp. 2840–2844, 2025.

[40] A. Luhtaru, E. Korotkova, and M. Fishel, "No error left behind: Multilingual grammatical error correction with pre-trained translation models," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1209–1222, 2024.

[41] B. Aytan and C. O. Şakar, "Deep learning-based turkish spelling error detection with a multi-class false positive reduction model," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 31, no. 3, pp. 581–595, 2023.

[42] M. B. Allamong, J. Jeong, and P. M. Kellstedt, "Spelling correction with large language models to reduce measurement error in open-ended survey responses," *Research & Politics*, vol. 12, no. 1, p. 20531680241311510, 2025.

[43] A. Horbach, Y. Ding, and T. Zesch, "The influence of spelling errors on content scoring performance," in *Proceedings of the 4th workshop on natural language processing techniques for educational applications (nlptea 2017)*, pp. 45–53, 2017.

[44] J. Kim, J. C. Weiss, and P. Ravikumar, "Context-sensitive spelling correction of clinical text via conditional independence," in *Conference on Health, Inference, and Learning*, pp. 234–247, PMLR, 2022.

[45] B. Wang, C. Xu, S. Wang, Z. Gan, Y. Cheng, J. Gao, A. H. Awadallah, and B. Li, "Adversarial glue: A multi-task benchmark for robustness evaluation of language models," *Advances in Neural Information Processing Systems*, 2021.

[46] Z. Zhang, B. Hao, J. Li, Z. Zhang, and D. Zhao, "E-bench: Towards evaluating the ease-of-use of large language models," in *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 2329–2339, 2025.

[47] I. Wuraola, N. Dethlefs, and D. Marciniak, "Understanding slang with llms: Modelling cross-cultural nuances through paraphrasing," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15525–15531, 2024.

[48] A. Girón, J. Huertas-Tato, and D. Camacho, "Llm synthetic generation to enhance online content moderation generalization in hate speech scenarios," *Computing*, vol. 107, no. 8, p. 164, 2025.

[49] F. Stahlberg and S. Kumar, "Synthetic data generation for grammatical error correction with tagged corruption models," in *Proceedings of the 16th workshop on innovative use of NLP for building educational applications*, pp. 37–47, 2021.

[50] M. H. R. Sifat, C. R. Rahman, M. Rafsan, and H. Rahman, "Synthetic error dataset generation mimicking bengali writing pattern," in *2020 IEEE Region 10 Symposium (TENSYMP)*, pp. 1363–1366, IEEE, 2020.

[51] M. Rizwan, "Social media abbreviations dataset." `https://www.kaggle.com/datasets/rizdelhi/socialmediaabbrevations`, 2021. Accessed: 2025-12-16.

[52] Y. Lee, Y. M. Çetinkaya, E. Külah, H. Toroslu, and H. Davulcu, "Masking the bias: From echo chambers to large scale aspect-based sentiment analysis," in *International Conference on Advances in Social Networks Analysis and Mining*, pp. 214–225, Springer, 2024.

[53] S. M. Jayanthi, D. Pruthi, and G. Neubig, "Neuspell: A neural spelling correction toolkit," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 158–164, 2020.

[54] K. Lang, "Newsweeder: Learning to filter netnews," in *Machine Learning Proceedings 1995* (A. Prieditis and S. Russell, eds.), pp. 331–339, San Francisco (CA): Morgan Kaufmann, 1995.

[55] M. Grootendorst, "Bertopic: Neural topic modeling with a class-based tf-idf procedure," *arXiv preprint arXiv:2203.05794*, 2022.

[56] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (Portland, Oregon, USA), pp. 142–150, Association for Computational Linguistics, June 2011.

[57] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for compu-*

tational linguistics: human language technologies, volume 1 (long and short papers), pp. 4171–4186, 2019.

[58] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "SemEval-2014 task 4: Aspect based sentiment analysis," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)* (P. Nakov and T. Zesch, eds.), (Dublin, Ireland), pp. 27–35, Association for Computational Linguistics, Aug. 2014.

[59] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "SemEval-2015 task 12: Aspect based sentiment analysis," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (P. Nakov, T. Zesch, D. Cer, and D. Jurgens, eds.), (Denver, Colorado), pp. 486–495, Association for Computational Linguistics, June 2015.

[60] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (L. Màrquez, C. Callison-Burch, and J. Su, eds.), (Lisbon, Portugal), pp. 632–642, Association for Computational Linguistics, Sept. 2015.

[61] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.

[62] A. Williams, N. Nangia, and S. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, Association for Computational Linguistics, 2018.

[63] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," *Advances in neural information processing systems*, vol. 33, pp. 5776–5788, 2020.

[64] M. Laurer, W. Van Atteveldt, A. Casas, and K. Welbers, "Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli," *Political Analysis*, vol. 32, no. 1, pp. 84–100, 2024.

[65] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 7871–7880, 2020.

[66] J. Bulian, J. Boyd-Graber, M. Leippold, M. Ciaramita, and T. Diggelmann, "Climate-fever: A dataset for verification of real-world climate claims," in *NeurIPS 2020 Workshop on Tackling Climate Change with Machine Learning*, 2020.

[67] T. H. Tsang, "Fine-tuning pre-trained transformers for climate claim verification," 12 2023.

[68] B. Zhang, F. Moiseev, J. Ainslie, P. Suganthan, M. Ma, S. Bhupatiraju, F. Lebron, O. Firat, A. Joulin, and Z. Dong, "Encoder-decoder gemma: Improving the quality-efficiency trade-off via adaptation," *arXiv preprint arXiv:2504.06225*, 2025.

[69] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[70] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, *et al.*, "Qwen technical report," *arXiv preprint arXiv:2309.16609*, 2023.

[71] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[72] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *et al.*, "Lora: Low-rank adaptation of large language models.," *ICLR*, vol. 1, no. 2, p. 3, 2022.

[73] A. A. Sorokin, T. O. Shavrina, A. V. Baytin, I. E. Galinskaya, and E. D. Rykunova, "Spellrueval: the first competition on automatic spelling correction for russian," in *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2016"*, (Moscow, Russia), June 2016.

[74] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.

[75] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," *arXiv preprint arXiv:1904.09675*, 2019.

[76] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM computing surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[77] K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston, "Retrieval augmentation reduces hallucination in conversation," *arXiv preprint arXiv:2104.07567*, 2021.

[78] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in neural information processing systems*, vol. 33, pp. 9459–9474, 2020.

[79] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[80] G. Izacard, P. Lewis, M. Lomeli, L. Hosseini, F. Petroni, T. Schick, J. Dwivedi-Yu, A. Joulin, S. Riedel, and E. Grave, "Atlas: Few-shot learning with retrieval augmented language models," *Journal of Machine Learning Research*, vol. 24, no. 251, pp. 1–43, 2023.

[81] Q. Huang, S. Fu, X. Liu, W. Wang, T. Ko, Y. Zhang, and L. Tang, "Learning retrieval augmentation for personalized dialogue generation," *arXiv preprint arXiv:2406.18847*, 2024.

[82] Y. Ahn, S.-G. Lee, J. Shim, and J. Park, "Retrieval-augmented response generation for knowledge-grounded conversation in the wild," *IEEE Access*, vol. 10, pp. 131374–131385, 2022.

[83] D. Thulke, N. Daheim, C. Dugast, and H. Ney, "Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog," *arXiv preprint arXiv:2102.04643*, 2021.

[84] X. Wang, P. Sen, R. Li, and E. Yilmaz, "Adaptive retrieval-augmented generation for conversational systems," in *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 491–503, 2025.

[85] J. Mallinson, R. Sennrich, and M. Lapata, "Paraphrasing revisited with neural machine translation," in *The 15th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 881–893, Association for Computational Linguistics (ACL), 2017.

[86] Y. Guo, Y. Liao, X. Jiang, Q. Zhang, Y. Zhang, and Q. Liu, "Zero-shot paraphrase generation with multilingual language models," *arXiv preprint arXiv:1911.03597*, 2019.

[87] S. Witteveen and M. Andrews, "Paraphrasing with large language models," *arXiv preprint arXiv:1911.09661*, 2019.

[88] M. Iyyer, J. Wieting, K. Gimpel, and L. Zettlemoyer, "Adversarial example generation with syntactically controlled paraphrase networks," *arXiv preprint arXiv:1804.06059*, 2018.

[89] A. Khan, A. Wang, S. Hager, and N. Andrews, "Learning to generate text in arbitrary writing styles," *arXiv preprint arXiv:2312.17242*, 2023.

[90] L. Jayawardena and P. Yapa, "Improving quality and domain-relevancy of paraphrase generation with graph-based retrieval augmented generation," in *Proceedings of the 2024 10th International Conference on Computing and Artificial Intelligence*, pp. 196–208, 2024.

[91] S. Tonmoy, S. Zaman, V. Jain, A. Rani, V. Rawte, A. Chadha, and A. Das, "A comprehensive survey of hallucination mitigation techniques in large language models," *arXiv preprint arXiv:2401.01313*, vol. 6, 2024.

[92] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, *et al.*, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–55, 2025.

[93] Z. Wei, D. Guo, D. Huang, Q. Zhang, S. Zhang, K. Jiang, and R. Li, "Detecting and mitigating the ungrounded hallucinations in text generation by llms," in *Proceedings of the 2023 International Conference on Artificial Intelligence, Systems and Network Security*, pp. 77–81, 2023.

[94] X. Wang, W. Shi, R. Kim, Y. Oh, S. Yang, J. Zhang, and Z. Yu, "Persuasion for good: Towards a personalized persuasive dialogue system for social good," in *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 5635–5649, 2019.

[95] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "Mpnet: Masked and permuted pre-training for language understanding," *Advances in neural information processing systems*, vol. 33, pp. 16857–16867, 2020.

[96] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The faiss library," *IEEE Transactions on Big Data*, 2025.

[97] AI@Meta, "Llama 3 model card," 2024.

[98] A. Joshi, S. Kale, S. Chandel, and D. K. Pal, "Likert scale: Explored and explained," *British journal of applied science & technology*, vol. 7, no. 4, pp. 396–403, 2015.

APPENDICES

A Knowledge-Base Generation

A.1 Knowledge-Base Generation and Validation for Turkish Dataset

Below is the prompt used as system instruction for knowledge base generation.

```
ROL:
Sen diyetisyen mesajlarından beslenme üzerine GENEL ve
GENELLENEBİLİR bilgileri çıkaran bir bilgi çıkarım
sistemisin.
AMAÇ:
Sadece diyetisyenin verdiği cevaplardan beslenme ile
ilgili:
- öneri
- kural
- sağlık bilgisi
- dikkat edilmesi gereken durum
- miktar bilgisi
- bilimsel veya genel beslenme bilgisi
çıkar ve listele.

GİRDİ FORMATI:
Danışan: danışan mesajı.
Diyetisyen: diyetisyen mesajı.

ÖNEMLİ KURALLAR:
1) SADECE "Diyetisyen:" kısmındaki metni analiz et.
2) "Danışan:" kısmındaki hiçbir bilgiyi yazma.
```

```
3) Danışanın yaptığı eylemleri ASLA bilgi olarak yazma.
   Yasak örnekler:
   - tavuk yendi
   - su içildi
   - nohut getirildi
   - pilav yedim
   - yürüyüş yaptım
   Bu tür ifadeler kesinlikle yazılmayacak.
4) Eğer diyetisyen danışanın yaptığı şeyi tekrar
ediyorsa
(örn: "Tavuk yemen iyi olmuş") bunu bilgi olarak yazma.
5) Sadece genellenebilir beslenme bilgilerini çıkar.
6) Sohbet, motivasyon, tebrik, onay, hal hatır, küçük
konuşmalar
   bilgi değildir.
7) Diyetisyen mesajında kayda değer beslenme bilgisi
yoksa
SADECE şu çıktıyı ver:

##Bilgi yok.

8) Eğer bilgi varsa HER bilgi ayrı satırda yazılmalı.
9) Çıktı formatı kesinlikle şu şekilde olmalı:

##Anahtar1, Anahtar2: Bilgi metni

10) Anahtar kelimeler:
   - Sadece o bilginin geçtiği cümledeki önemli
     besin veya kavramları yaz.
   - Genel kelimeler kullanma (yemek, öğün, yedim,
     yaptım, vardı gibi yasak).
```

- Danışanın yaptığı aktivite anahtar kelime olarak yazılmaz.
- En fazla 3 anahtar kelime yaz.
- Küçük harfle yaz.

11) Bilgi metni:
- Kısa ve net olmalı.
- Metnin dışına çıkma.
- Yorum ekleme.
- Yeni bilgi üretme.
- Metni anlam değiştirmeden sadeleştirebilirsin.

12) Çıktıda açıklama, giriş, sonuç, ek cümle, boşluk veya başka metin OLMAYACAK.
Sadece istenen format üretilecek.

POZİTİF ÖRNEK:

Girdi:

Danışan: Bu öğlen tavuk sote ve yoğurt vardı.

Diyetisyen: Yoğurt ile proteini aynı anda tüketmemek daha iyi olabilir. Günde en az 2 litre su tüketelim.

Çıktı:

##yoğurt, protein: yoğurt ile protein aynı anda tüketilmemeli

##su: günde en az 2 litre su tüketilmeli

NEGATİF ÖRNEK 1:

Girdi:

Danışan: Sabah yumurta yedim.

Diyetisyen: Harika, çok iyi yapmışsın.

Çıktı:

##Bilgi yok.

```
-------------------------------------------------
NEGATİF ÖRNEK 2:
Girdi:
Danışan: Akşam pilav yedim.
Diyetisyen: Pilav yemen sorun değil.
Çıktı:
##Bilgi yok.
-------------------------------------------------
NEGATİF ÖRNEK 3:
Girdi:
Danışan: Su içtim.
Diyetisyen: Süper.
Çıktı:
##Bilgi yok.
-------------------------------------------------
ŞİMDİ YENİ GİRDİYİ İŞLE:
```

Then the user instruction format is as follows.

```
Danışan: Bu öğlen tavuk sote ve yoğurt vardı.
Diyetisyen: Yoğurt ile proteini aynı anda tüketmemek
            daha iyi olabilir. Günde en az 2 litre su
            tüketelim.
```

Created knowledge-base candidates validated using LLM with following system instruction.

```
Sana gelen metin beslenme anlamında kullanılabilir bir
bilgi metni ise 1 değilse 0 yaz.
0 ya da 1 dışında başka hiçbir şey yazma.

Pozitif örnek:
```

```
Yumurta: Yumurta protein içerir.
Çıktı:
1

Pozitif örnek:
Su: Günde en az 2.5 lt su içilmeli.
Çıktı:
1

Negatif örnek:
Su: Tavuk pilav yendi.
Çıktı:
0

Negatif örnek:
Su: Meyve konuldu.
Çıktı:
0
```

A.2 Example Extracted Knowledge-Base Entries with Corresponding Agent Messages

```
Client: ananas benim bagırsaklarımı cok çalıştırıyor ve
tuvalete cıkarıyor ödem de atıyorum. ama ilk defa
duydum muz gibi oldugunuz kalori açısındanmı agır
Dietitian: Glisemik indeks açısından biraz yüksek incir
ve muz gibi.... Lifli çok haklısın ama haftada 2-3 den
sık olmazsa daha iyi olur

Information: Ananas: Ananasın glisemik indeksi yüksek

-------------------------------------------------------
```

Client: bugün kızımın yine hastahane işlemleri var aç olması lazım o yzden biz de kahvaltı yapamıyoruz. herhalde 11:30 u bulacak ilk öğünümüz. bana bugün için ona göre bir planlama yaparmısın lütfen. teşekkürler
Dietitian: Teşekkür ediyorum bilgilendirme için, o zaman kahvaltımızı mutlaka proteinli yani yumurta-peynirli yapalım yanına fırsat olursa bol yeşillik ve sebze ile. Hareketlere haftada 2 gün olan yürüyüşü 4 güne çıkarmayı hedefledik ve başaracağız Özellikle maden suyunu da egzersiz sonrasına ekleyelim

Information: Maden Suyu: Egzersiz sonrasına maden suyu tüketilmeli

---

Client: evet peynirsiz yiyemem zaten:). domates zeytin biber hepsi vardı sadece yumurta yok
Dietitian: Süper hatta minik bir bilgi -> 1 adet yumurta = 1 dilim (30g) peynir olduğu için yumurta yerine de eklemeler yapabilirsin

Information: Yumurta: 1 adet yumurta = 1 dilim (30g) peynir olduğu için yumurta yerine de eklemeler yapabilirsin

---

Client: Kuruyemisim yok. Elma getirmiştim yqbimda bugun araya onu yerim. Muz dursun yarin kuruyemis getiririm onla ara olur

Dietitian: Tamamdır aslında tüm meyveler için bu gerekli listende de bu yüzden meyve yanı mutlaka ya kuruyemiş ya süt grubu var tek başına meyve insülin direncimizi tetikler bu yüzden biz yarın tedariklerimizi sağlarız. Akşam yemeğinde neler bizimle, öğünün fotoğrafını bekliyorum mutlaka

Information: Insülin Direnci: Meyve tek başına yenilirse insülin direncimizi tetikler

---

Client: Yanında da bi bardak portakal suyu içtim
Dietitian: Afiyet olsun Kahvaltı güzel ama porkala suyumuz fazla olmuş... 1 barsask portakal suyu = 2-3 portakal demektir :)

Information: Portakal suyu: 1 bardak = 2-3 portakal demektir.

Client: Sabah haşlanmış yumurta, salatalık, yeşil zeytin ve peynir. Bi de nesquik tozlu laktozsuz süt. Laktozsuz süt alıyorum artık, normali şişiriyor. Hem bunun tadı daha tatlı. Öğle yemeği kurufasulye ve yoğurt
Dietitian: laktoza özel bir intolerans veya alerjimiz yoksa tüketmeni önermem normal süt tüketimine devam edelim derim yada ikisinide tüketiriz ama tamammen birini tüketerek devam etmeyelim ara ara normal süt tüketimi de olsun

```
Information: Laktozsuz Süt: Laktoza özel bir intolerans
veya alerjimiz yoksa normal süt tüketimine devam edelim
```

B Examples for RAP Response Generation

Below, several examples are shown including client message, original dietitian response, and the response generated with the RAP approach.

```
Client: Aksam yemekten sonra anlik bir agri oldu ama
cabuk gecti yogun bir mide bulantisi vardi simdilik yok
denecek kadar az
Original Response: Çok da aç bırakmayalım kendimizi bu
süreçte.
RAP: Geçmiş olsun, öğünümüzü hızlı tükettiysek
hazımsızlık, şişkinlik, mide ağrısı yapabilir. Dikkatli
olalım.

--------------------------------------------------

Client: Öğleni de salata ile geçireceğim. Biraz
dengeler.
Original Response: Salatamız protein eklemeli olsun
veya yoğurt/ayranla takviye edelim.
RAP: Afiyet olsun. Salatamızın içerisinde peynir
ekleyerek protein ihtiyacımıza destek olabiliriz.
yanında yoğurt/ayran tüketimimizi de unutmayalım.

--------------------------------------------------
```

Client: 5-6 tane çilek ve bir tane muz ara öğünüm.
Original Response: Afiyet olsun tekrar. Meyve ile birlikte kuruyemiş de tüketebilirsin dengeli bir ara öğün oluşturmak için.
RAP: Afiyet olsun tekrar. Yanında 2 adet ceviz içi veya 9-10 adet çiğ badem/ çiğ fındık tüketebilirsin, yarım adet muz ile birlikte 6 adet çilek toplam 1 porsiyon meyve olacaktır.

Client: Fırında köfte yapıcam yanına ne olsun diye düşünüyorum.
Dietitian: Yanında yoğurt + salata + tambuğday ekmek veya bulgur pilavı olabilir.
RAP: çok iyi yanında bol salata veya pişmiş sebze iyi olur.

Client: Yulaf bar yedim. meyve ve kuruyemiş alışverişi yapmam lazım
Original Response: Yulaf bar da iyi bir tercih. Sağlıklı market alışverişi ile ilgili bu bilgimizi inceleyebilirsin.
RAP: Anlaştık.. Evet sebze meyve stoklarımızı bir güzel yenileyelim.

C Examples for Detailed RAP Response Generation

Below, two detailed examples are presented. Each example includes the original client message, the actual response written by the dietitian, and the RAP-generated

response. Additionally, it displays the candidate dietitian message retrieved from a similar client interaction and selected for paraphrasing, along with the corresponding client message that led to its retrieval. This structure illustrates the end-to-end RAP process: from retrieval based on client similarity to controlled paraphrasing for producing the final response.

```
Client: Akşam yemekten sonra anlık bir ağrı oldu ama
çabuk geçti, yoğun mide bulantısı vardı.
Original Response: Çok da aç bırakmayalım kendimizi bu
süreçte.
Similar Client Message #2: Yemekten sonra inanılmaz bir
mide ağrısı başladı. Acaba öncesinde çok açtım birden
yedim diye mi.
Candidate Response #2: Afiyet olsun, öğünümüzü hızlı
tükettiysek hazımsızlık, şişkinlik, mide ağrısı
yapabilir. Dikkatli olalım. Leblebiyi kepekli
bisküviyle değiştirelim ki daha uzun süre tokluk
sağlayalım olur mu? Erken acıkmana neden olabilir
çünkü.
RAP: Geçmiş olsun, öğünümüzü hızlı tükettiysek
hazımsızlık ve mide ağrısı yapabilir. Dikkatli olalım.

------------------------------------------------------

Client: Öğleni de salata ile geçireceğim. Biraz
dengeler.

Original Response: Salatamız protein eklemeli olsun
veya yoğurt/ayranla takviye edelim

Similar Client Message #2: Öğlen biraz salata yiyeceğim
```

Candidate Response #2: Afiyet olsun. Salatamızın içerisinde peynir ekleyerek protein ihtiyacımıza destek olabiliriz. yanında yoğurt/ayran tüketimimizi de unutmayalım

RAP: Afiyet olsun. Salatamıza peynir ekleyerek protein desteği sağlayabiliriz, yanında yoğurt/ayran tüketimini unutmayalım.

CURRICULUM VITAE

PERSONAL

NAME : Emre Külah

PLACE : Ankara, Türkiye

WORK EXPERIENCE

2018-2026	**Research Assistant** METU, Ankara, Turkiye
2017-2018	**Software Engineer** DİA Software, Ankara, Türkiye
2016-2017	**Software Engineer** Forte Arge, Ankara, Türkiye

EDUCATION

2026 | Ph.D. in COMPUTER ENGINEERING
Graduate School of Natural and Applied Sciences
Middle East Technical University (METU), Türkiye
Thesis: "Advancing Dialogue Systems with Temporal Contextual Topic Forecasting and Controlled Response Retrieval"
Advisor: Hande ALEMDAR

2019 | MSc. Degree in COMPUTER ENGINEERING
Graduate School of Natural and Applied Sciences
Middle East Technical University (METU), Türkiye
Thesis: "Spatio-temporal cohesive networks for evaluating team behavior in soccer"
Advisor: Hande ALEMDAR

2016 | Bs Degree in COMPUTER ENGINEERING
Faculty of Engineering
Middle East Technical University (METU), Türkiye

PUBLICATIONS

Journal and conference publications are annotated with the "**J**" and "**C**" prefixes.

C Lee, Y., Çetinkaya, Y. M., **Külah, E.**, Toroslu, H., & Davulcu, H. (2024, September). Masking the bias: From echo chambers to large scale aspect-based sentiment analysis. In International Conference on Advances in Social Networks Analysis and Mining (pp. 214-225). Cham: Springer Nature Switzerland.

J Çetinkaya, Y. M., **Külah, E.**, Toroslu, İ. H., & Davulcu, H. (2024). Targeted marketing on social media: utilizing text analysis to create personalized landing pages. Social Network Analysis and Mining, 14(1), 77.

J Çetinkaya, Y. M., Lee, Y., **Külah, E.**, Toroslu, İ. H., Cowan, M. A., & Davulcu, H. (2024). Towards a programmable humanizing ai through scalable stance-directed architecture. IEEE internet computing.

J **Külah, E.**, Çetinkaya, Y. M., Özer, A. G., & Alemdar, H. (2023). COVID-19 forecasting using shifted Gaussian Mixture Model with similarity-based esti-

mation. Expert Systems with Applications, 214, 119034.

J **Külah, E.**, & Alemdar, H. (2020). Quantifying the value of sprints in elite football using spatial cohesive networks. Chaos, Solitons & Fractals, 139, 110306.

C Seddati, O., **Külah, E.**, Pironkov, G., Dupont, S., Mahmoudi, S., & Dutoit, T. (2015, September). UMons at MediaEval 2015 Affective Impact of Movies Task including Violent Scenes Detection. In MediaEval (Vol. 1436).