GASTON: Graph-Aware Social Transformer for Online Networks

by

Olha Anastasia Wloch

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Data Science

Waterloo, Ontario, Canada, 2026

© Olha Anastasia Wloch 2026

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Online communities have become essential digital third places for socialization and support, yet they also possess toxicity, echo chambers, and misinformation. Mitigating these harms requires computational models that can understand the nuance of online interactions to accurately detect harmful content such as toxicity and norm violation. This is difficult because the meaning of an individual post is rarely self-contained; it is dynamically constructed through the interplay of what is written (textual content) and where it is posted (social structure). We require models that effectively fuse these two signals to generate representations for online entities such as posts, users, and communities.

Current approaches often treat these different signals in isolation: text-only models analyze content but miss the local social norms that define acceptable behavior, while structure-only models map relationships but ignore the semantic content of discussions. Recent hybrid approaches attempt to bridge this gap but some rely on simple text averaging mechanisms to represent a user and a community, and in so doing flatten the rich, norm-defining identity.

To address this limitation, this thesis proposes **GASTON** (Graph-Aware Social Transformer for Online Networks), a graph learning framework designed to capture the essence of online social networks. It does so by modeling connections between all online entities, such as users, communities, and text. This makes it possible to ground user and text representations in their local norms, providing the necessary context to accurately classify behaviour in downstream tasks. The heart of our solution is a contrastive initialization strategy which pre-trains community representations based on user membership patterns, effectively capturing the unique signature of a community's user base before the model processes any text. This allows GASTON to distinguish between communities (e.g., a support group vs. a hate group) based on who interacts there, even if they share similar vocabulary.

We evaluate GASTON across a diverse set of socially-aware downstream tasks, including mental health stress detection, toxicity scoring, and norm violation detection. Our experiments demonstrate that GASTON outperforms state-of-the-art baselines, particularly in tasks where social context is critical for classification, such as detecting norm violations (achieving a nearly 40% performance improvement over text-only baselines). Furthermore, we illustrate that these learned representations provide interpretable insights, offering a path toward user-empowered transparency in online spaces.

Acknowledgements

I would first like to thank my supervisors, Dr. Lukasz Golab and Dr. Robin Cohen, for their guidance, insightful feedback, and patience throughout my Master's. I am grateful for the freedom to explore my interests in the intersection of social science and data science, and for your support in my growth as a researcher.

I would also like to extend my appreciation to my committee members and readers, Dr. Jeff Orchard and Dr. Kate Larson, for contributing their time and expertise to the evaluation of this work.

Thank you Liam Hebert, my research mentor, whose mentorship, enthusiasm, and insights helped shape this thesis while also working on his own PhD. I also owe my gratitude to Zivojin for his technical wizardry and for constantly restarting the server when I needed it most—this work literally would not have run without you.

To my friends Mallory, Simone, and Amy: thank you for the camaraderie and for patiently listening to me stress over this thesis for the last two years. Your friendship has made the journey that much sweeter.

Finally, I owe my deepest gratitude to my family. Your unwavering love, encouragement, and belief in me has been the foundation of all my pursuits.

Dedication

To my mom, my sister, and my partner.
Your love and support mean everything to me.

Table of Contents

Author's Declaration ii

Abstract iii

Acknowledgements iv

Dedication v

List of Figures x

List of Tables xi

1 Introduction 1

1.1 Motivation 1

1.1.1 The Double-Edged Sword of Online Communities 1

1.1.2 The Technical Challenge: How to Add Context 2

1.2 A Data Science Perspective 4

1.3 Problem Statement 4

1.4 Contributions 5

1.5 Thesis Outline 6

2 Background **7**
2.1 Embeddings . . . 7
2.2 Transformers . . . 8
2.3 Language Models . . . 10
2.4 Graph Neural Networks . . . 12
2.5 Heterogeneous Graphs . . . 13
2.6 Heterogeneous Graph Transformers . . . 14
2.7 Graph Pre-Training and Fine Tuning . . . 15
2.8 Contrastive Learning and Ranking . . . 16
2.8.1 Bayesian Personalized Ranking (BPR) . . . 17

3 Related Work **19**
3.1 Representing Online Interactions . . . 19
3.1.1 Structure-Only Representations: community2vec . . . 19
3.1.2 Text-only Representations for Social Media . . . 20
3.1.3 Unified Hybrid Models . . . 20
3.1.4 Applications Outside of Social Media . . . 21
3.2 Computational Social Science and Relevant Applications . . . 22
3.2.1 Macro-Level Analysis: Quantifying Polarization . . . 22
3.2.2 Micro-Level Analysis: Modeling Local Discussion Context . . . 22
3.3 Qualitative Evidence: Why Text-Only Models Fail . . . 23
3.4 Limitations of Static Graphs and Temporal Dynamics . . . 24

4 Proposed Solution **25**
4.1 Preliminaries . . . 25
4.2 Graph Construction . . . 27
4.3 Node Embedding Methods . . . 28
4.3.1 Text Node Embeddings . . . 28

4.3.2 Community Node Embeddings . . . 31
4.3.3 Model Architecture . . . 32
4.3.4 Dynamic User Embeddings . . . 33
4.4 Pretraining Setup and Tasks . . . 34
4.4.1 Pretraining Metrics . . . 35

5 Experiments and Discussion **38**
5.1 Data . . . 38
5.1.1 Pretraining Data . . . 38
5.1.2 Downstream Tasks . . . 39
5.1.3 Evaluation Metrics . . . 42
5.2 Baselines . . . 44
5.2.1 Naive Baselines . . . 44
5.2.2 OMCA . . . 44
5.3 Experimental Environment . . . 45
5.3.1 Hardware and Data Split . . . 45
5.3.2 Model Configuration and Training Details . . . 45
5.4 GASTON Performance . . . 46
5.4.1 Qualitative Analysis and Error Analysis . . . 47
5.4.2 Embedding Visualization . . . 51
5.5 Ablations . . . 53
5.5.1 Impact of Language Model Architecture: XLNet vs. EmbeddingGemma 53
5.5.2 Community Initialization Strategy: Trainable vs. Frozen Weights . 54
5.6 Societal Implications: From Algorithmic Censorship to Informed Engagement for Mental Health . . . 55
5.6.1 The Double-Edged Sword of Online Support . . . 55
5.6.2 Emotional Contagion and User Agency . . . 55

6 Conclusion **57**

6.1 Conclusion 57

6.1.1 Limitations and Future Work 58

References **62**

List of Figures

1.1 The Interplay of Context on Reddit. 3

2.1 The Scaled Dot-Product Self-Attention Mechanism (Adapted from Vaswani et al. [43]) 9

2.2 The Message Passing Paradigm in Graph Neural Networks. Information is aggregated from the local neighborhood to update the central node's representation. 13

2.3 A heterogeneous social graph, highlighting the distinction between node types. 14

2.4 Contrastive Learning Intuition. The model optimizes the embedding space by pulling positive pairs closer while pushing negative pairs apart. 18

4.1 The GASTON Heterogeneous Graph Schema. We model three distinct node types and the directed relationships between them. 26

4.2 GASTON Graph Construction 29

4.3 High-Level Architecture Comparison. 30

4.4 The GASTON Pre-training Architecture showing Text Reconstruction and Edge Prediction tasks. The loss weighting parameter α is set to 0.5 to balance the semantic and structural objectives equally. 36

5.1 GASTON Trained Community Embeddings 52

List of Tables

1.1 Example of Context-Dependent Labeling: Reclaimed Language 3

5.1 Summary of Downstream Evaluation Tasks 40

5.2 Main results comparing GASTON against baselines across all downstream tasks. 47

5.3 Ablation study for GASTON. 53

Chapter 1

Introduction

1.1 Motivation

1.1.1 The Double-Edged Sword of Online Communities

The rise of internet socialization, popularized by platforms such as Reddit and X (formerly Twitter), has fundamentally altered human interaction. These platforms have created vast digital ecosystems where individuals form diverse communities around shared interests, identities, and goals. As a result, many of these online communities have become de-facto 'digital third places' [40]: social environments distinct from home and work where community is created and socialization occurs.

However, this shift to interaction occurring in digital spaces acts as a double-edged sword. On the positive side, these spaces provide invaluable peer support [8], fostering connection for everything from niche hobbies to sensitive health discussions. On the negative side, pseudo-anonymity and lack of physical cues can spread misinformation, toxicity, and the formation of echo chambers [6]. Prior work has analyzed how conversations derail into toxicity [53]. To maximize the beneficial capabilities of these spaces while mitigating their harms, we require computational methods capable of capturing the essence of online social networks. This can be done by modeling connections between all online entities, such as users, communities, and text.

To understand these dynamics, we focus on Reddit, a platform organized into thousands of distinct communities known as "subreddits" (e.g., `r/science`, `r/nba`). Each subreddit has its own topic, user base, and distinct set of social norms. The challenge for researchers

is to create models that can understand these communities. Critically, these communities are not just as buckets of text, but social entities.

1.1.2 The Technical Challenge: How to Add Context

To improve performance on socially-aware downstream tasks such as toxicity detection and community recommendation, we must look beyond isolated data points. A fundamental challenge in analyzing online networks is unified representation learning: the task of jointly modeling users, posts, and communities to generate dense representations that capture the latent structure of their interactions. The quality of these representations depends entirely on whether they are learned in isolation or in context.

Currently, researchers typically rely on two isolated sources of signal:

- **Text-Only Signals (Content):** These models analyze what is said. They define the similarity between two communities based on their vocabulary. For example, a text model might calculate that `r/science` and `r/physics` are similar because they both frequently use words like "experiment" and "theory".
- **Structure-Only Signals (Interaction):** These models analyze who is interacting. They define distance based on user overlap. If users who post in `r/nba` also tend to post in `r/lakers`, a structural model draws a strong connection between them, regardless of what they are actually talking about.

The Problem: Context Blindness

Both approaches struggle when used in isolation because the meaning of online interaction is constructed by the interplay of both signals. Text-only models are context-blind; they see the semantics but miss the social norms. Structure-only models see the relationships but are blind to the content.

Table 1.1 illustrates this with an example of reclaimed language. The term "redneck" generally carries a negative connotation. A text-only model might flag it as toxic. However, within `r/republicans`, this term is often used as an in-group identifier (reclaimed). A structural model would know that `r/republicans` is socially distinct from `r/democrats`, but without the text, it cannot evaluate the specific comment. To accurately classify this post, a model must simultaneously understand the semantic content of the word and the unique normative profile of the community where it appears. This dynamic is illustrated

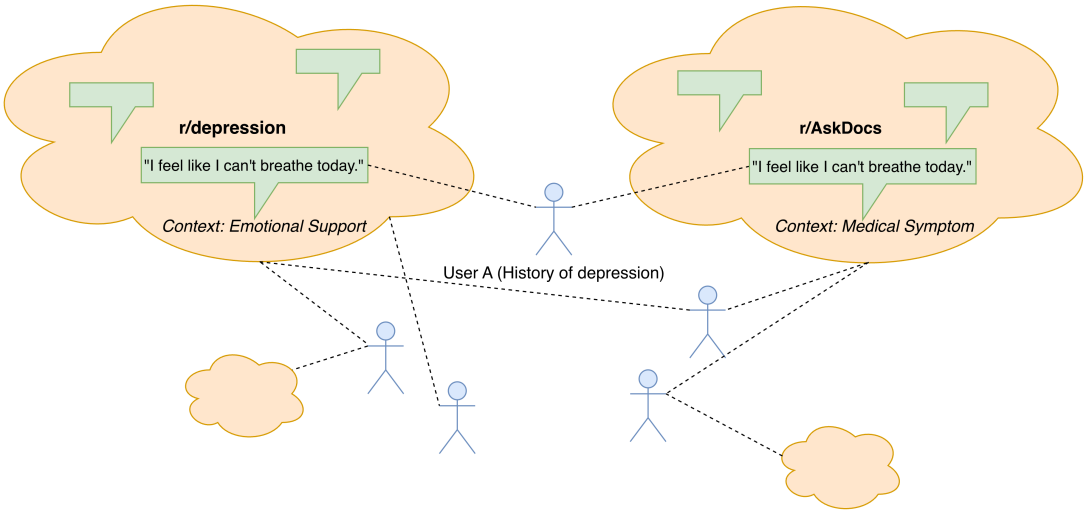


Figure 1.1: The Interplay of Context on Reddit.

in Figure 1.1. The meaning of a post is dynamically constructed by the interplay of User, Text, and Community. We formalize this ecosystem as a heterogeneous graph—a network structure composed of multiple types of nodes and edges. Here, a single user (User A) participates in two distinct communities. While the text content is identical, the community context shifts the pragmatic meaning from an expression of emotional distress (`r/depression`) to a description of a physical symptom (`r/AskDocs`). As shown, the semantic meaning of the phrase 'I feel like I can't breathe' shifts entirely based on the community context, serving as a cry for emotional support in `r/depression` versus a physical symptom description in `r/AskDocs`. A model that ignores this community signal fails to capture the true intent of the post.

Table 1.1: Example of Context-Dependent Labeling: Reclaimed Language

Text Content	**Community**	**Label**
"Strange how culturally superior leftists in SF would want to move to a place full of "dumb rednecks"."	r/republicans	**Non-Hateful**
	r/democrats	**Hateful**

Recent hybrid models attempt to unify these signals but often fail to integrate them effectively. A common shortcoming is that some models treat the community as a derivative of its text. They initialize a community's representation (the starting state of its vector

before training) by simply averaging the vectors of all posts written within it. This flattens the rich, complex identity of a community into a generic semantic summary, washing out the unique structural signal that differentiates a support group from a hate group, even if they discuss similar topics. This also makes these models susceptible to the averaging problem of feature embeddings [13].

We argue that to truly model online networks, the community must be treated as a first-class entity—not a derivative of its text, but a complex, collaboratively learned representation that evolves based on the unique intersection of its user base, its structural position in the network, and its social norms.

1.2 A Data Science Perspective

This thesis approaches the problem of community modeling through the interdisciplinary lens of Data Science. Where researchers may focus solely on algorithmic efficiency or pure social science, which focuses on qualitative theory, data science operates at the intersection of **computational capability** and **societal implications**.

Our work exemplifies this dual focus. On the computational side, we leverage massive, unstructured datasets (millions of Reddit interactions) and engineer high-performance architectures using Heterogeneous Graph Transformers (HGTs). Heterogeneous Graph Transformers are deep learning models designed to handle complex networks with multiple types of nodes and edges. These models convert our data from noise into signal. On the inference side, we design these architectures specifically to capture latent sociological signals, such as normative distance, converting raw data into actionable insights. By encoding social signals into an embedding space, GASTON serves as a bridge between the raw scale of big data and the nuanced requirements of human-centric applications like mental health monitoring.

1.3 Problem Statement

There is a need for a pre-training approach that learns holistic representations for all network entities and the relationships between them. By pre-training on the massive scale of unlabeled data available on platforms like Reddit, we can learn robust general-purpose features that generalize better to small, specific downstream datasets than models trained from scratch. The core problem this thesis addresses is: How can we develop

a heterogeneous graph framework that leverages the community as a contextual anchor, allowing us to generate richer user and text representations that improve performance on socially-aware downstream tasks?

1.4 Contributions

In this thesis, we propose **GASTON** (Graph-Aware Social Transformer for Online Networks), a novel heterogeneous graph learning framework. Rather than treating the community as a secondary feature, GASTON establishes the community as a foundational contextual entity. By learning a distinct structural representation for the community and fusing it with user and text embeddings, we provide the necessary context to ground online discourse. This allows the model to generate robust representations for every node in the graph, ultimately serving as a powerful pre-trained model for diverse downstream applications.

- **Conceptual Contribution: Community as a First-Class Entity.** We propose *Collaborative Community Representations*, where a community's identity is modeled as a distinct learnable entity rather than a derivative of its text content. Content-based approaches often fail to differentiate between semantically similar communities with vastly different social norms. We challenge this paradigm by defining a community's representation through its structural signature within a heterogeneous graph. Crucially, these representations are actively refined during the graph pre-training phase to fuse structural and semantic context. These optimized embeddings are then transferred directly to downstream tasks, serving as a rich, pre-computed community profile that enhances predictive performance.
- **Methodological Innovation: The GASTON Architecture.** We introduce a novel pre-training framework designed to solve the technical challenge of fusing dense semantic features with sparse structural signals. Merging these distinct modalities is complex; naive fusion methods (like averaging) often cause the dominant text signal to wash out the subtler structural cues. To overcome this, we introduce **Contrastive Initialization**. By pre-training community embeddings using a Bayesian Personalized Ranking (BPR) objective, which optimizes the space by pulling connected communities closer and pushing disjoint ones apart, we create a strong structural prior that prevents the text from overwhelming the social signal.

- **Empirical Validation on Social Tasks:** We demonstrate through extensive experiments that this community-centric approach outperforms state-of-the-art baselines. We specifically highlight improvements in nuanced social tasks, such as detecting hate speech and norm violations. In these domains, the meaning of a post is heavily dependent on the place in which it is posted.

Our results validate the effectiveness of this approach. For instance, on the context-heavy norm violation detection task, GASTON achieves an F1 score (the harmonic mean of precision and recall, balancing false positives and false negatives) of 0.97, significantly outperforming the text-only baseline of 0.70 — a relative improvement of nearly 40%. Beyond classification, our model also excels in structural tasks, achieving the highest performance in community recommendation. Overall, GASTON achieves state-of-the-art results in 3 out of 5 downstream tasks. This quantitative success, paired with qualitative evidence that our model effectively distinguishes between supportive and toxic environments, demonstrates that community context is not merely supplementary, but essential for accurate social understanding.

1.5 Thesis Outline

The remainder of this thesis is structured to guide the reader from the building blocks of our architecture to the experimentation results of our proposed solution.

Chapter 2 introduces the foundational concepts of representation learning, detailing the Transformer and Graph Neural Network architectures that underpin our model. **Chapter 3** situates our work within the broader landscape of computational social science, analyzing the limitations of prior text-only and structure-only approaches.

Chapter 4 formally presents the GASTON architecture. It details how we construct the heterogeneous graph and the specific mechanisms used to generate our rich community representations. **Chapter 5** describes our experimental framework and presents results across diverse downstream tasks, followed by a discussion on the societal implications of using these representations for user transparency. Finally, **Chapter 6** concludes the thesis and suggests avenues for future work.

Chapter 2

Background

This chapter introduces the foundational concepts required to understand the proposed model. We operate within the field of Deep Learning, a subset of machine learning that uses multi-layered neural networks to model complex patterns in data. We will first define how embeddings enable these networks to process discrete data into meaningful features. We will then detail the key neural network architectures, namely Transformers and Graph Neural Networks (GNNs), that form the building blocks of our system. We will explain the pre-training and fine tuning strategy that allows a single model to generalize across diverse downstream tasks. Lastly, we will explain contrastive learning in the context needed to understand its usefulness in GASTON.

2.1 Embeddings

An embedding is a dense, low-dimensional vector representation of a discrete unit of data, such as a word, a sentence, or a node in a graph. Its primary function is to transform sparse unit data, which is typically high-dimensional, into a continuous vector space where semantic or relational meanings are encoded by geometric space. This means that the proximity or distance between two embeddings in our embedding space reflects the similarity of the items they represent. For example, words that appear in similar contexts [32] will be mapped close to each other, allowing for vector arithmetic (e.g., King - Man + Woman = Queen). This allows the data to be machine readable in a way that captures complex, latent relationships or features, allowing deep learning models to process more information more efficiently. These embeddings serve as the foundation upon which more

complex neural network architectures, such as Transformers and Graph Neural Networks, are built.

2.2 Transformers

The Transformer architecture, introduced by Vaswani et al. [43], represents a fundamental paradigm shift in sequence modeling. Prior models such as Recurrent Neural Networks (RNNs) [29] and Long Short-Term Memory Models (LSTMs) [11] rely on recurrent layers where data is processed sequentially. This approach is standard for sequence modeling: the task of predicting the next element in a series, such as the next word in a sentence. However, sequential processing limits parallelization. Transformers introduced a novel mechanism known as Self-Attention, which solves these bottlenecks. Self-Attention allows the model to excel at processing sequential data, and is core to the Transformer's success, making it the current foundational model for state-of-the-art applications.

Self-Attention operates on sequences of tokens, where each token is a discrete unit of input, such as words in a sentence. Self-Attention allows a model to weigh the relative importance of all other elements in a sequence when processing a single element. This is achieved by assigning three learnable vectors to each token: a **Query (Q)**, a **Key (K)** and a **Value (V)**. The Query vector represents the current token's request for information. The Key vectors of all other tokens in the sequence act as labels for the information they hold. The Value vectors contain the actual information being shared.

As illustrated in Figure 2.1, the input token x_i is first projected into query (q_i), key (k_i), and value (v_i) vectors via learned linear transformations. The attention mechanism then computes the dot product between the query q_i and the keys of all other tokens K, essentially asking 'how relevant is this other token to me?'. These raw scores are scaled and normalized via a Softmax function (an activation function) to produce attention weights. Finally, these weights are used to compute a weighted sum of the value vectors V, resulting in a context-aware representation for the token.

The model calculates a similarity score between the current token's Query and every other token's Key. These scores are normalized into "attention weights" (which sum to 1), which tell the model how much attention the current token should pay to all other tokens in the sequence. The final representation for that token is then computed as a weighted sum of all the Value vectors in the sequence.

Self-Attention provides several key advantages:

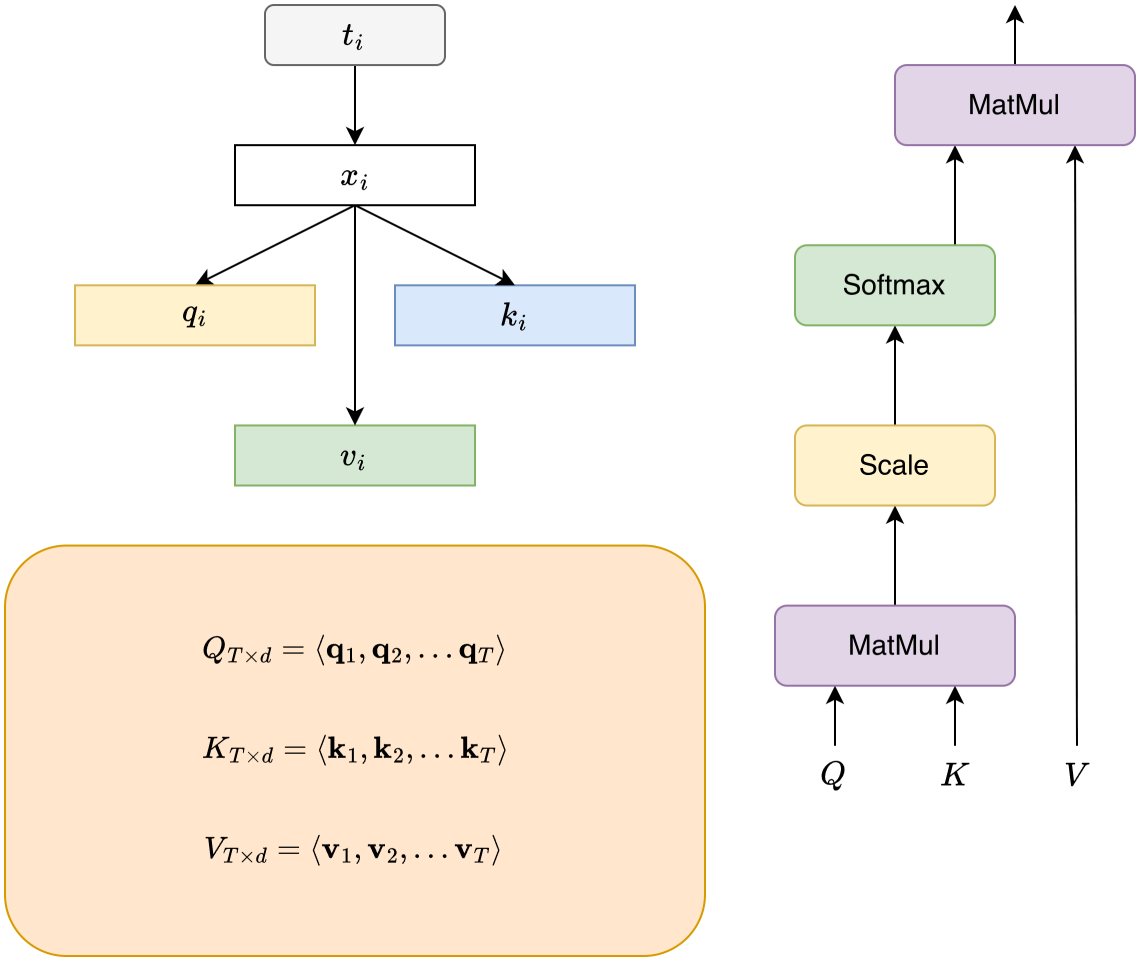


Figure 2.1: The Scaled Dot-Product Self-Attention Mechanism (Adapted from Vaswani et al. [43])

Firstly, it excels at handling long-range dependencies. Unlike RNNs, which process data sequentially and struggle to maintain context over long sequences, self-attention aims to directly model the relationship between two tokens in a sequence, regardless of their distance.

Second, this architecture enables parallel processing. Attention computations for all tokens in a sequence can be performed simultaneously, rather than token-by-token. This compatibility with modern parallel hardware, like GPUs, significantly reduces training time and increases efficiency on massive datasets. As a result, this makes Transformers highly scalable, in both dataset size and computational resources. This enables the training of Large Language Models (LLMs), with billions of parameters, and makes this architecture well suited to problems such as online community modeling, where there are millions of users and posts, among tens of thousands of communities.

Transformers have also demonstrated significant versatility. While originally built for Natural Language Processing (NLP), self-attention has successfully been adapted to other domains, such as computer vision and bioinformatics [10, 23]. As a result, transformers can also be attributed to the rise of multimodal AI systems that can interpret and integrate different types of data, such as text and images.

Finally, Transformers effectively learn for the purposes of transfer learning, which is central to this thesis. The transfer learning paradigm is defined as pre-training a model on a large generic dataset to learn general features, and then fine-tuning it on a smaller dataset for a specific downstream task. Models that are later fine-tuned on smaller, task specific datasets are able to achieving state-of-the-art results with a fraction of the original computational cost. In this work, we leverage pre-trained language models as the backbone text encoder.

2.3 Language Models

The Transformer architecture described above powers modern language models (LMs). While LMs can be built on various architectures, modern iterations are typically Transformers trained on a massive, web-scale corpus of text, often with a self-supervised objective such as predicting the next word in a sentence or filling in masked-out words. These are most commonly Large Language Models (LLMs).

Early models in this paradigm, such as BERT [9] and its successor XLNet [49], marked a fundamental shift in the field. They demonstrated a strong ability to understand language nuance and context, and established the pre-training and fine-tuning paradigm that now

dominates the field. In this approach, a general-purpose LM is first pre-trained on a massive, unlabeled dataset, and can then be fine-tuned with a small amount of labeled data to achieve state-of-the-art performance on specific tasks.

However, these models were primarily designed for full model tasks, where the entire network is fine-tuned to produce a specific output (like a classification). A different application for these models is to use them simply as feature extractors, where their final-layer representations are used as static text embeddings. While effective, this approach can be suboptimal, as the model was not explicitly trained for the goal of generating high-quality, dense vector representations.

This has led to the development of a new class of specialized embedding models. These models are distinct from general-purpose LMs; their architecture and training objective are engineered for the purpose of converting text into high-quality, semantically rich vector representations.

EmbeddingGemma [45] is a specialized embedding model. While models like BERT are also bidirectional encoders, meaning they process text by attending to both past and future context simultaneously, they are typically optimized for token-level classification. EmbeddingGemma is instruction-tuned specifically to generate high-quality sentence-level vectors suitable for retrieval tasks, such as semantic search or document ranking, making it superior for representing full social media posts. Its strengths lie in the below areas:

- **Specialized Architecture:** Unlike decoder-only models which process text sequentially for the purpose of text generation (e.g., GPT [34]), EmbeddingGemma is based on an encoder-only architecture, which is used for tasks that require semantic understanding. This allows it to use bi-directional attention to capture the full context of the input simultaneously, creating richer semantic representations.
- **High Performance-to-Size Ratio:** Despite its small size (approx. 300M parameters), it achieves state-of-the-art performance on embedding benchmarks (e.g., MTEB [33]), outperforming many models twice its size. This efficiency is critical for research and on-device applications.
- **Task-Specific Optimization:** The model's performance is maximized by using instructional prompts that prime the model for a specific task. For example, text may be prefixed with a `retrieval-document` prompt to ensure the model generates an embedding optimized for that specific use case.
- **Flexible Output Dimensions:** It employs Matryoshka Representation Learning (MRL) [28], a technique that structures the information within the embedding vector

hierarchically. By front-loading the most critical semantic information into the earlier dimensions, MRL allows the native 768-dimensional embeddings to be truncated to smaller sizes (e.g., 256) with minimal performance loss, offering a powerful trade-off between performance and computational cost.

This shift from using general-purpose LMs (like XLNet) to specialized, high-performance embedding models (like EmbeddingGemma) represents a technical improvement in this domain of research. However, these models are not without limitations. Their performance can be sensitive to the use of correct instructional prompts, and their smaller context windows (e.g., 2048 tokens for EmbeddingGemma) may be a limitation for very long documents compared to larger, general-purpose LMs. For the task of encoding social media posts, which are typically short, this trade-off works in our favour, as it provides a more potent and efficient text representation to ground the social graph.

2.4 Graph Neural Networks

Graph Neural Networks (GNNs) are deep learning models designed to operate on non-Euclidean data structured as graphs [25] [15]. Graphs are comprised of a set of objects (nodes) and relationships between them (edges). Traditional deep learning architectures, such as Convolutional Neural Networks (CNNs) [27] and RNNs [29] assume regular grid-like or sequential data structure. In contrast, graphs have a topology with no fixed ordering or distance, making traditional neural network architectures ineffective. GNNs leverage the graph's connectivity to perform computation.

The core principle of nearly all GNNs is an iterative process of message passing (see Figure 2.2). This process is usually broken down into two steps per layer:

1. **Aggregate:** For a given node, the GNN gathers feature vectors (AKA 'messages') from immediate neighbours. The aggregation can be as simple as a sum or mean, or can be a more complex learned function.

2. **Update:** The GNN then combines the aggregated neighbourhood information with the node's own current feature vector (which is derived in the previous layer). This combined representation is then passed through a non-linear transformation to produce the node's new feature vector, which is used in the next layer.

By stacking these layers, a node's representation becomes progressively more aware of its wider structural context, incorporating information from nodes one-hop, two-hops, three-hops away, and so on. This allows the GNN to learn a representation for each node that captures its structural context within the graph.

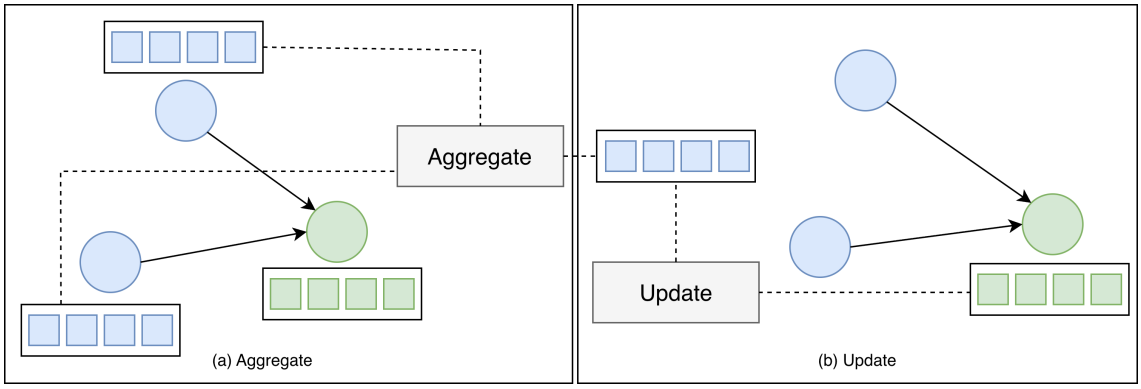


Figure 2.2: The Message Passing Paradigm in Graph Neural Networks. Information is aggregated from the local neighborhood to update the central node's representation.

2.5 Heterogeneous Graphs

Traditional GNNs ([25] [15]) are designed for homogeneous graphs. These are graphs where all nodes and edges are of the same type. This is a significant drawback for modeling complex, real-world systems.

Online social networks, in particular, are inherently heterogeneous. A heterogeneous graph is a graph that contains multiple types of nodes and/or multiple types of edges. For example, the social graph modeled in this research is not uniform; it is a complex structure composed of:

- **Node Types:** `User`, `Text` (posts/comments), and `Community` (subreddits).
- **Edge Types:** The graph explicitly models the following connections:
  - `Text-posted-by-User`
  - `Text-belongs-to-Community`
  - `User-posted-in-Community`

Figure 2.3 illustrates this heterogeneous structure within our specific domain. Distinct node types (`User, Text, Community`) are connected by specific relation types, forming a topology where the semantic meaning of an edge depends on the types of nodes it connects.

This capability is equally important in domains outside of social media, such as academic knowledge graphs. Consider a network consisting of `Authors`, `Papers`, and `Venues`.

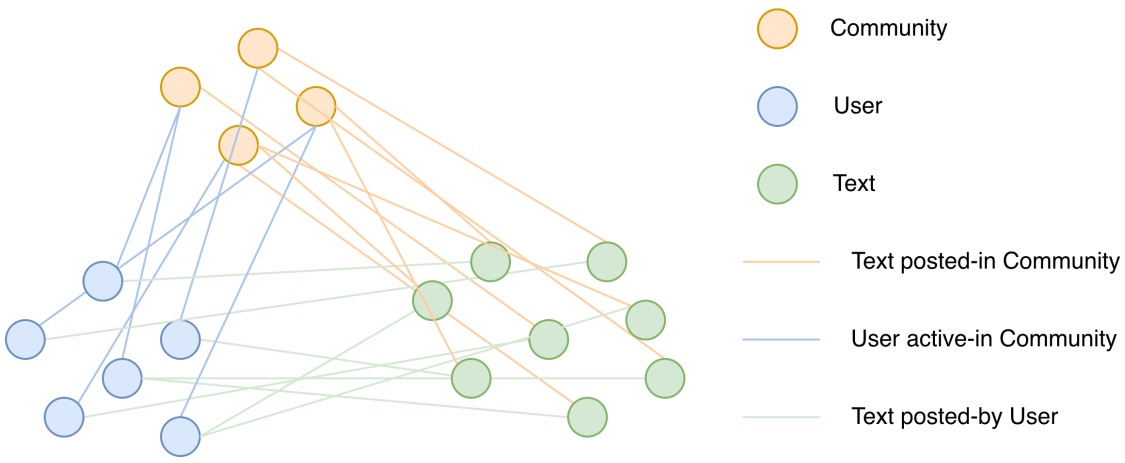


Figure 2.3: A heterogeneous social graph, highlighting the distinction between node types.

The semantic relationship of an `Author` writing a `Paper` is fundamentally different from a `Paper` citing another `Paper`. A homogeneous model would treat both simply as connections. In contrast, an HGT learns distinct parameters for these relations. This allows the model to understand that aggregating information via a "citation" edge should inform the paper's topic, while aggregating via a "published in" edge should inform the paper's venue type, without confounding these distinct signals.

To effectively learn from such multi-relational data, a more specialized architecture is required, one that can understand and leverage this heterogeneity.

2.6 Heterogeneous Graph Transformers

Online social networks consist of multiple types of nodes (e.g., `Users`, `Posts`, and `Communities`) and different edges that represent different relationships. This relationship could be a user posting in a community, a user submitting a post, a post replying to another post, and so on. Applying a simple GNN would require ignoring rich context and structural information and would treat all nodes and edges identically.

Heterogeneous Graph Transformers (HGTs) [21] are a class of GNNs designed to solve this problem, and build upon the attention mechanism introduced in Graph Attention Networks (GAT) [44]. HGTs extend the self-attention mechanism of Transformers to the heterogeneous graph structure. Rather than learning a single function for all message

passing, an HGT learns a distinct set of transformation matrices and attention-weighting functions for each edge type.

This allows the model to learn highly context-specific message passing. For example, the way a `User` node aggregates information from a `Post` it has authored is computationally different from how it aggregates information from the `Community` it is a part of. This allows the model to learn a unified embedding space while still respecting the rich, multi-relational nature of the data. This architecture is foundational to this thesis.

2.7 Graph Pre-Training and Fine Tuning

Inspired by the success of the pre-training and fine-tuning paradigm in NLP, this strategy has been adapted for GNNs. The idea is to first train a GNN on a large, general purpose, unlabeled graph (or a collection of graphs) using some self-supervised task. This pre-training phase allows the model to learn broad, transferrable knowledge about structural patterns, node relationships, and attribute distribution. A foundational study by Hu et al. [20] systematically explored strategies for GNN pre-training, which can be broadly categorized into:

1. **Node-Level Strategies:** These tasks involve masking node features or attributes and training the GNN to reconstruct them from the local neighborhood, analogous to BERT's Masked Language Model (MLM) task [9]. In MLM, a percentage of input tokens are hidden (masked), and the model must predict the missing words based on the surrounding context. Similarly, in Graph Masking, we hide node features and ask the model to reconstruct them.

2. **Structure-Level Strategies:** These tasks focus on topology. They include link prediction, where the model predicts whether an edge exists between two nodes based on the similarity of their embeddings, and contrastive learning, which is explained below, where the model is trained to pull representations of similar nodes (e.g., nodes from the same local subgraph, or nodes reachable via a random walk) together in the embedding space, while pushing dissimilar nodes apart.

While powerful, GNN pre-training is notoriously difficult and, unlike in NLP, is not a silver bullet. A critical challenge identified by Hu et al. [20] is negative transfer. This occurs when the pre-training task is not well-aligned with the downstream task, causing the learned representations to actually hurt performance. For example, a GNN pre-trained on a

generic, platform-wide link prediction task might learn that "users who post in community A also post in community B" is a dominant pattern. However, for a downstream task like norm violation detection, this pattern is irrelevant. What matters is the semantic content of a post and if it aligns with the specific, unique norms of community A. The pre-trained model, in this case, may have learned the wrong priors and could be actively disadvantaged (i.e., suffer from negative transfer) during fine-tuning.

This high risk of negative transfer, especially in a domain as context-dependent as social media, motivates the central hypothesis of this thesis: the need for a carefully designed, community-centric pre-training strategy, as proposed in GASTON, to ensure the learned representations are genuinely useful for socially-aware downstream tasks.

2.8 Contrastive Learning and Ranking

Contrastive learning is a self-supervised learning paradigm that focuses on learning representations by comparing data points, rather than reconstructing them. The core intuition, formally introduced by Hadsell et al. [14], is that a robust embedding space should map semantically similar items (positive pairs) close together while pushing dissimilar items (negative pairs) apart, as illustrated in Figure 2.4. Contrastive learning often optimizes a noise-contrastive estimation (InfoNCE) objective [42].

This approach gained widespread adoption in NLP through the Word2Vec algorithm [32], specifically via its Negative Sampling objective. Instead of predicting a word directly over a massive vocabulary, Word2Vec learns to differentiate between true context words (positives) and randomly drawn noise words (negatives). The mechanism of learning by contrasting valid signals against noise allows models to capture rich semantic relationships efficiently and serves as a foundational concept for the structure-aware learning objectives employed in GASTON.

Unlike classification, which requires predefined labels (e.g., "Cat" vs. "Dog"), contrastive learning operates on the principle of instance discrimination. For example, in a social network, a user and a community they frequent form a positive pair, while that same user and a random community form a negative pair. By optimizing the model to distinguish between these pairs, the embedding space naturally organizes itself to reflect the underlying structure of the data, clustering related entities even without explicit class labels.

2.8.1 Bayesian Personalized Ranking (BPR)

A specific formulation of contrastive learning widely used in recommender systems is Bayesian Personalized Ranking (BPR), introduced by Rendle et al. [37]. BPR is designed specifically for implicit feedback scenarios. This is where we observe positive interactions (e.g., a user posting in a subreddit) but rarely observe explicit negative ones (e.g., a user stating they dislike a subreddit).

Standard binary classification treats unobserved interactions as negatives, which can be noisy. BPR instead uses a pairwise ranking loss. It creates training triplets (u, i, j), consisting of:

- A user u.
- An item i that the user *has* interacted with (positive).
- An item j that the user *has not* interacted with (negative).

The objective of BPR is to maximize the probability that the user prefers item i over item j. The optimization criterion is defined as:

$$L_{BPR} = \sum_{(u,i,j) \in D} - \ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) \tag{2.1}$$

where σ is the sigmoid function and $\hat{x}_{ui}$ is the model's predicted score (e.g., dot product) for the user-item pair. By minimizing this loss, the model learns to rank observed interactions higher than unobserved ones, creating a semantic ordering in the embedding space that is robust to the sparsity of social data.

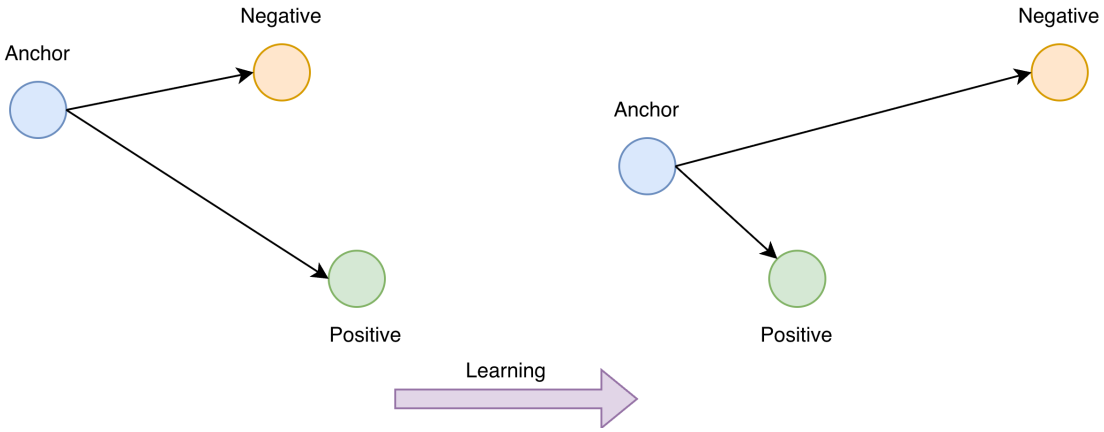


Figure 2.4: Contrastive Learning Intuition. The model optimizes the embedding space by pulling positive pairs closer while pushing negative pairs apart.

Chapter 3

Related Work

3.1 Representing Online Interactions

The challenge of modeling online communities has been approached from three primary directions:

1. Models that focus only on the network structure (section 3.1.1),
2. models that focus only on the textual content (section 3.1.2), and
3. hybrid models that attempt to unify both (section 3.1.3).

3.1.1 Structure-Only Representations: community2vec

A foundational approach to modeling communities is to focus entirely on the network structure. Approaches like node2vec [12] and community2vec [31] exhibit this methodology. The community2vec algorithm is built on the same core intuition as the Word2Vec model [32]. The model learns a dense, low-dimensional vector (typically $d = 100$ to 300) for each community. While effective for structure, this approach completely ignores the semantic content of the discussions.

Martin [31] treats communities as 'words' and users as 'contexts'. A user who comments in `r/nba` and `r/lakers` creates a co-occurrence pair, linking these two communities. By building a co-occurrence matrix from all user participation on Reddit, the model learns

a dense, low-dimensional vector for each community. The paper demonstrates that these structure-only representations are able to capture real-world semantic relationships. For instance, the model can solve analogies, such as finding that the vector for `r/berlin` minus `r/germany` plus `r/unitedkingdom` is the closest vector for `r/london`. This work powerfully establishes that user co-membership data is a rich source of information for modeling community relationships.

3.1.2 Text-only Representations for Social Media

On the other hand, there exist models that focus entirely on textual content to understand or predict online behaviour, leveraging the power of language models. This approach treats online posts as a large corpus of text, well-suited for models like BERT [9]. Research in this area often fine-tunes a pre-trained transformer model for a given social media task. Similarly, MentalBERT [22] applies this domain-adaptive strategy to mental healthcare, pre-training on millions of Reddit posts to capture psychiatric vocabulary. The primary strength of this approach is its deep understanding of language nuance and domain-specific terminology, allowing it to detect subtle linguistic cues that general-purpose models might miss.

For example, models like HateBERT [2] are BERT models further pre-trained on large scale, abusive language datasets. These models achieve state-of-the-art performance on text-based hate speech detection tasks. The primary strength of this approach is its deep understanding of language nuance, semantics, and syntax. However, these models are often context-blind. They are designed to classify a piece of text in isolation, with no awareness of the community dynamics, who the author is, or what post it is replying to. This can lead to errors such as flagging a scientific discussion of a sensitive topic as hateful, or failing to capture the hateful subtext of a seemingly harmless comment. For example, as noted by Hebert et al. [17], a text-only model often misclassifies reclaimed slurs used supportively within a specific community because it lacks the social norm context to interpret the word correctly.

3.1.3 Unified Hybrid Models

More recently, research has focused on fusing these siloed approaches into a single, unified model. Ma et al. [30] introduced One-Model-Connects-All (OMCA), a state-of-the-art unified framework. The key innovation of OMCA is the creation of a single, large-scale heterogeneous graph, composed of three distinct nodes: `Communities`, `Users`, and `Text`

(post/comments). These nodes are connected by edges representing their relationships, such as Text-posted-by-User and Text-belongs-to-Community. While OMCA unifies structure and semantics, it relies on text embedding averaging for initialization. Our framework, GASTON builds upon this hybrid foundation but replaces the averaging mechanism with a contrastive learning objective to better capture distinct community identities, adding user-membership structural information.

The heterogeneous structure of OMCA allows an HGT to pass messages between different types of entities, learning their representations simultaneously. The model is pre-trained on two large-scale, self-supervised tasks: Text Reconstruction (a node-level task to predict masked text) and Edge Generation (a structure-level task to predict link validity).

To initialize this graph, `Text` nodes are encoded with a Language Model (XLNet) [49], while `User` and `Community` nodes are initialized by averaging the text embeddings they are connected to. This unified framework is shown to be highly effective, achieving strong results after being fine-tuned on a diverse set of downstream tasks. The hybrid approach of OMCA establishes a powerful and flexible blueprint for a single, pre-trained model to learn from the complex interactions and structural context in online networks.

3.1.4 Applications Outside of Social Media

The strategy of fusing network structure with semantic content is powerful paradigm used in other complex domains, most notably in healthcare. For example, Wu et al. [48] demonstrated the utility of incorporating structural knowledge into representation learning. They proposed a graph-based Hierarchical Medical Entity (HME) embedding framework that leverages the topology of medical knowledge graphs (such as the hierarchical relations between diagnoses and treatments) alongside textual descriptions.

Just as Wu et al. found that structural hierarchy was necessary to accurately represent medical entities for downstream prediction tasks, we argue that social structural context is similarly necessary to represent online communities. Both domains require moving beyond flat text representations to capture the latent relationships defined by the underlying graph structure.

3.2 Computational Social Science and Relevant Applications

The models described above serve as the foundation for a wide range of applications, from micro-level tasks like moderating a single comment to macro-level analyses of an entire platform.

3.2.1 Macro-Level Analysis: Quantifying Polarization

Waller and Anderson [46] demonstrated the power of community representations as a tool for large-scale computational social science, specifically for quantifying political polarization. While GASTON shares the goal of modeling the latent social landscape, our focus differs. Rather than explicitly projecting communities onto researcher-defined political axes, GASTON aims to learn a general-purpose socio-semantic representation that captures normative boundaries (such as toxicity and rule adherence) organically. By fusing structure and semantics, we aim to provide a more holistic view of community identity that extends beyond simple political alignment.

Waller and Anderson's core contribution was to project known social dimensions onto this community embedding space. By selecting seed communities (e.g., `r/democrats` and `r/Conservative`), they were able to define a *partisan dimension* and assign a quantitative partisanship score to all 10,000 of the largest communities. This transformed the abstract unstructured data of Reddit into a measurable social landscape. Using this method, they were able to make unique discoveries of the user makeup of communities within Reddit. The dramatic increase in polarization on Reddit around 2016 was not, as some may have believed, driven by existing users becoming more extreme. Rather, it was due to population turnover. A large influx of new, highly-polarized users entered the platform and fundamentally shifted its social center of gravity.

3.2.2 Micro-Level Analysis: Modeling Local Discussion Context

While Waller and Anderson [46] operate at the macro-level of entire communities, other work has shown that graph-based context is equally as important at the micro-level of a single discussion. The Multi-Modal Discussion Transformer (mDT) by Hebert et al. [18] is a highly specialized model designed for the specific task of hate speech detection. Its central hypothesis is that hate speech is deeply contextual.

To capture this, mDT builds a unique graph for every individual discussion thread. This graph is multi-modal, incorporating features from both text (BERT) and images (ViT) [10]. The model uses a Graph Transformer to pass information between comments in a thread, giving it an understanding of the entire conversation. A key innovation is the interweaving of graph-transformer layers with multi-modal fusion layers, which grounds the text and image representations in the discussion's flow. This work, along with its accompanying HatefulDiscussions dataset, used in this thesis, proves that for nuanced social tasks, modeling the local graph context is essential for high-performance classification.

3.3 Qualitative Evidence: Why Text-Only Models Fail

While the quantitative success of models like mDT [18] highlights the value of context, qualitative analysis reveals why it is necessary. Hebert et al. [17] provide a detailed study identifying specific failure modes of comment-only models. For nuanced social tasks, such as hate speech detection or norm violation detection, analyzing content in isolation is insufficient. The authors analyze a Graph Transformer based model (Graphormer) [50] that, unlike traditional classifiers, builds a graph of the entire discussion thread to make its predictions.

The paper's key insight is that conversational context fundamentally changes the meaning of a comment. They demonstrate this through several examples. In one case, a comment in the `r/rupaulsdragrace` community using a slur is correctly identified as non-hateful by the graph model, which sees the supportive, reclaimed language context of the surrounding discussion. In contrast, a text-only BERT model which is blind to context, flags the comment as highly hateful.

This directly motivates GASTON's design: by explicitly modeling the community node, we aim to provide the missing context that prevents these specific types of classification errors. This work establishes that context is not merely supplementary, it is a prerequisite for accurate classification in a social context. The authors show that comment-only methods are prone to false positives (such as misinterpreting reclaimed language) and false negatives (missing hateful subtext). Models with context via graph structure are able to move beyond simple text classification to more advanced reasoning about social dynamics. Just as conversation context is important, in this thesis we aim to show that community context is also important, informing a graph model of social dynamics within and across communities that can inform predictions.

3.4 Limitations of Static Graphs and Temporal Dynamics

The models discussed thus far, including the GASTON architecture proposed in this thesis, treat social networks as static snapshots. They aggregate interaction histories into a single graph. While computationally efficient, this ignores the dynamic nature of online communities.

Research into Temporal Graph Networks (TGNs) by Rossi et al. [39] introduces a generic and highly efficient framework for learning from graphs that are not static, but are represented as a continuous-time sequence of events (e.g., User A joins Community B at time t1, User A posts in Community C at time t2). We review this literature to highlight the primary limitation of our current work and to frame the integration of TGNs as a critical avenue for the future work discussed in Chapter 6.

The core innovation of the TGN framework is its use of a memory module for each node. This memory acts as a compressed representation of the node's history. When a new event (like a user posting) occurs, the TGN uses a graph-based operator to compute a message from the event and updates the memory of the nodes involved. This allows the model to capture evolving patterns, such as a user's drifting interests or a community's changing topic focus, while remaining computationally efficient. This line of research demonstrates the importance of incorporating the when of an interaction, not just the who and what, and represents a major direction for future work beyond static graph models.

Chapter 4

Proposed Solution

4.1 Preliminaries

Our proposed model, GASTON, operates on a large-scale, heterogeneous graph, which contains multiple types of nodes and edges. We formally define this graph as $G = (V, E)$, where V represents the set of all nodes and E represents the set of all edges (links) between them. In this thesis, we utilize Reddit subreddits as our primary case study, but `Communities` could generalize to Facebook Groups, Discord Servers, StackOverflow Tags, or other online platforms where people create communities around shared interests.

This graph is defined by a schema $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of node types and $\mathcal{E}$ is the set of edge types.

Node Types

The set of node types $\mathcal{V}$ consists of three distinct entities central to online social networks:

- **User (U):** Represents individual users who post within communities.
- **Text (T):** Represents a single piece of text content, such as a post or a comment.
- **Community (C):** Represents a specific subreddit where interactions occur [1].

[1] Note that the set of communities is fixed to these predefined boundaries; GASTON learns representations for existing subreddits rather than discovering latent communities via graph clustering.

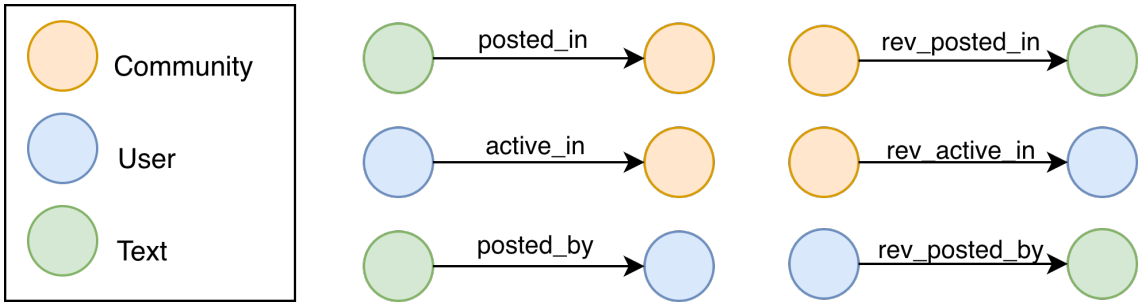


Figure 4.1: The GASTON Heterogeneous Graph Schema. We model three distinct node types and the directed relationships between them.

Each node $v \in V$ belongs to one of these types, defined by a mapping function $\tau : V \to \mathcal{V}$.

Edge Types

The set of edge types $\mathcal{E}$ defines the relationships connecting these nodes. Our graph is defined by the following relationships:

- **Text-PostedBy-User:** An edge $\langle t, \text{posted_by}, u \rangle \in E$ where $u \in U$ and $t \in T$. This edge represents the action of a user authoring a specific piece of text.
- **Text-PostedIn-Community:** An edge $\langle t, \text{posted_in}, c \rangle \in E$ where $t \in T$ and $c \in C$. This edge links a piece of text to the community it was posted in.
- **User-ActiveIn-Community:** An edge $\langle u, \text{active_in}, c \rangle \in E$ where $u \in U$ and $c \in C$. This edge represents a user posting within a community.

Each edge $e = (v_i, r, v_j) \in E$ consists of a source node v_i, a target node v_j, and a relation r. The relation type r belongs to a set of relations $\mathcal{R}$ that define the edge type $\mathcal{E}$. For example, the `posted_by` relation connects `User` and `Text` nodes, forming the edge type $\langle$t, posted_by, u$\rangle$.

As shown in Figure 4.1, our schema explicitly differentiates between posting (a user authoring text) and activity (a user interacting with a community). This distinction allows the model to learn different attention weights for direct authorship versus broader community engagement.

Node Features

Each node $v \in \mathcal{V}$ is associated with an initial feature vector X_v. The specific methods used to generate these initial features are detailed in Section 4.3.1 (for Text nodes) and Section 4.3.2 (for Community nodes). For example, a Text node feature X_T is a dense vector derived from the post content, whereas a Community node feature X_C is a learned vector representing the community's structural identity.

4.2 Graph Construction

The formal graph $G = (V, E)$ defined in Section above is constructed from our curated pretraining corpus (detailed in 5.1). This process transforms the raw data of posts, comments, authors, and subreddits into a unified graph structure. The multi-step graph construction process is detailed below:

1. **Node Instantiation:** We iterate through the pretraining dataset and create the nodes defined in our schema $\mathcal{V}$.
    - For each post or comment in the corpus, a **Text** node is created.
    - For each unique author of a post, a **User** node is created.
    - For each unique subreddit a post belongs to, a **Community** node is created.
2. **Edge Creation:** Once the nodes are instantiated, we create the directed edges that represent the primary actions in the data:
    - A directed **Text-posted_by-User** edge is created to link the `User` node to the `Text` node they authored.
    - A directed **Text-posted_in-Community** edge is created to link the `Text` node to the `Community` node (subreddit) where it was posted.
    - A directed **User-active_in-Community** edge is created if a user has posted at least once in a given subreddit. To keep important user-community relationships, while still maintaining a reasonably sized graph, we keep generating **User-active_in-Community** edges for users who have posted in a given community. While our `Text` node corpus is filtered for quality (as described in Section 5.1.1), we leverage the complete raw interaction logs to generate these structural edges. This ensures we capture a user's full community footprint, even if their specific posts were filtered out of the text processing pipeline.

3. **Undirected Graph Creation:** The heterogeneous graph G is used for message passing, which requires information to flow in both directions. Therefore, after the initial directed graph is built, we make it **undirected** for the HGT. This is achieved by explicitly adding a reverse edge for every canonical relation in our schema $\mathcal{E}$. For example, for the ⟨Text, posted_by, User⟩ relation, we add a corresponding ⟨User, rev_posted_by, Text⟩ relation. This ensures that the HGT model can learn to pass and aggregate messages bi-directionally across all relationships.

Figure 4.2 visualizes this construction pipeline. The top half depicts the raw Reddit log data (submissions, comments, user history). The bottom half illustrates the resulting heterogeneous graph, where the BPR-initialized community nodes (Orange) act as the structural anchors for the text (Green) and users (Blue).

With the graph structure fully constructed and symmetrical, the next critical step is to initialize the feature vectors for each node in V. This process is detailed in the following section.

4.3 Node Embedding Methods

4.3.1 Text Node Embeddings

The features for text nodes, which represent the semantic content of posts and comments, are initialized using pre-trained language models. For GASTON, we used EmbeddingGemma as our language embedding model.

To ensure a rigorous comparison, we independently replicated the OMCA [30] architecture. As the official implementation was not available, we rebuilt the model and adhere strictly to their described methodology, using XLNet [49] as the text encoder. For our simple baselines, BERT is used. The title and text content of each post are combined, cleaned, and passed through the language model. In our replication of OMCA, the final text node embedding is derived by applying mean pooling over the last hidden state of the XLNet outputs. Formally, given a sequence of token vectors $H = \{h_1, ..., h_n\}$, the text representation x_{text} is calculated as the element-wise average: $x_{text} = \frac{1}{n}\sum_{i=1}^{n} h_i$. This text-centric approach is then extended to initialize the social nodes in the OMCA baseline [30]: the initial feature for a User node is computed as the mean of all text embeddings they have authored, and a Community node's feature is the mean of all text embeddings posted within it.

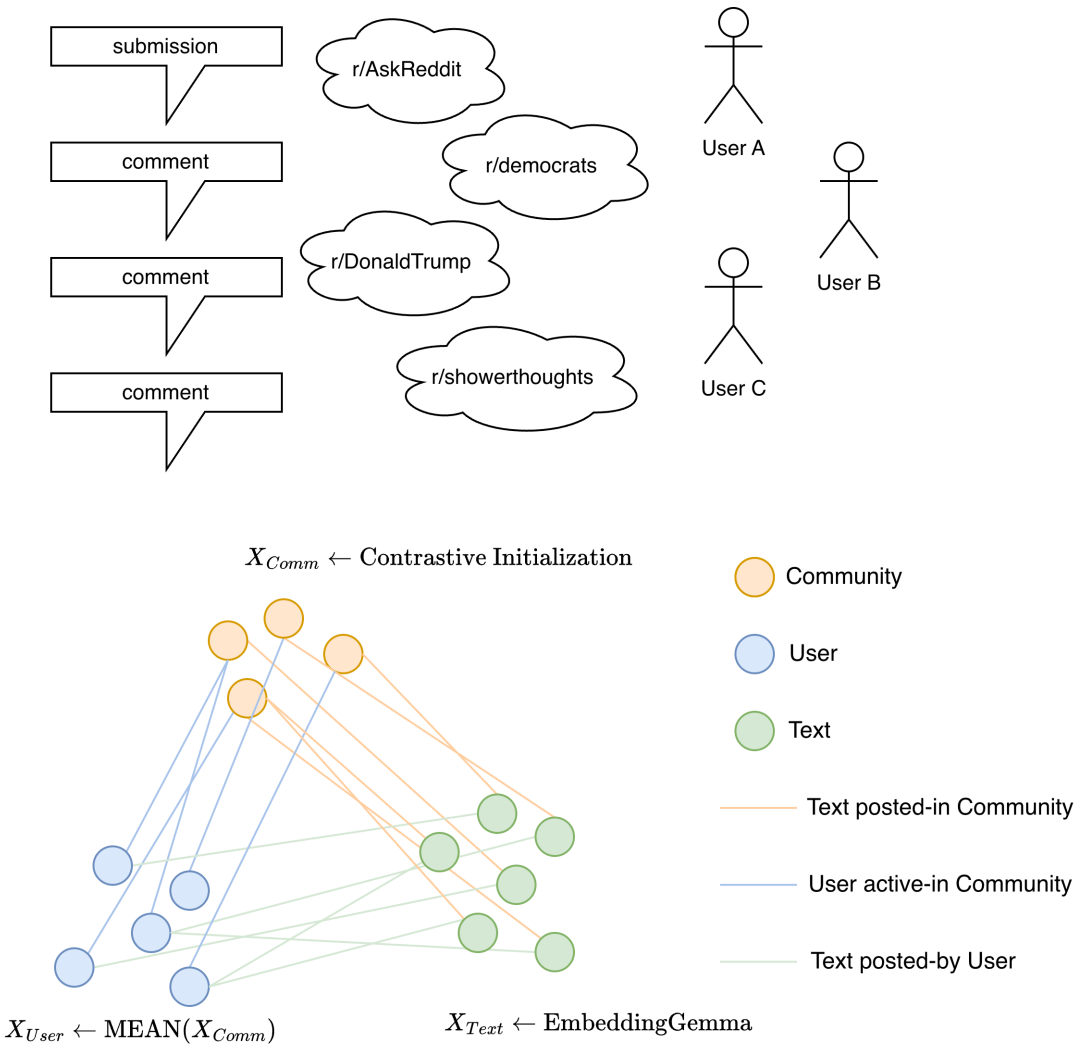


Figure 4.2: GASTON Graph Construction

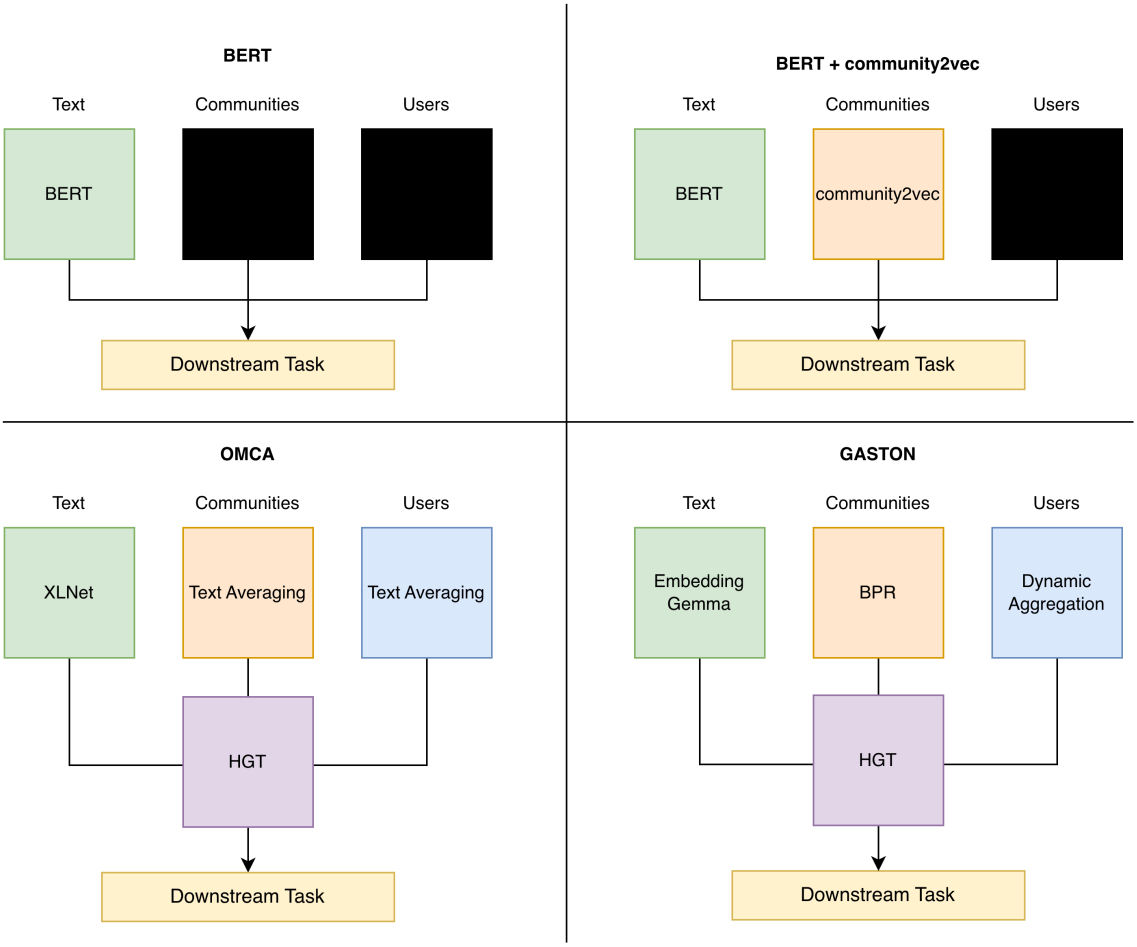


Figure 4.3: High-Level Architecture Comparison.

Figure 4.3 provides a high-level comparison of our architectural approach against standard baselines. We contrast GASTON against text-only and hybrid baselines. BERT is context-blind, ignoring social nodes. BERT + community2vec incorporates structure but ignores user history. OMCA utilizes a full graph but initializes social nodes via naive text averaging. GASTON (ours) introduces a dedicated BPR contrastive initialization for communities and dynamic user aggregation (as detailed in Section 4.3.4), fusing structure and semantics without simple averaging.

To generate high-quality input features, GASTON introduces two key methodological advancements. First, for the text node encoder, we implement EmbeddingGemma [45].This model is specifically optimized for generating dense semantic representations. We combine the title and text content of each post and utilize the model's specialized `retrieval-document` prompt to generate a feature vector optimized for retrieval tasks. Second, we explicitly decouple the initialization of social nodes (Users and Communities) from the text nodes. Our model, GASTON, instead employs a dedicated contrastive learning strategy to learn structure-aware features for these entities, as detailed in the following section.

4.3.2 Community Node Embeddings

We introduce a contrastive initialization strategy for our Community nodes. This is a preparation step designed to pre-train rich structure-aware initial features for our community nodes before they are further trained in our main GNN.

Our contrastive initialization method models the bipartite user-community interaction subgraph. A bipartite graph consists of two disjoint sets of nodes (`Users` and `Communities`) where edges only exist between sets, not within them. Drawing on inspiration from Collaborative Filtering [16], we treat this as a community recommendation problem. We first initialize trainable embedding matrices for the complete set of all users and all communities. The model is then trained independently using a contrastive learning objective, which is based on Bayesian Personalized Ranking (BPR) [37].

The training process is as follows:

1. A positive sample is created by selecting a (user, community) pair that exists in our interaction graph.
2. A negative sample is formed by pairing the same user with a community they have not interacted with.

3. The model computes a similarity score for both the positive and negative pairs.

4. A ranking loss is used to optimize embeddings. This loss function pushes the similarity score of the positive pair to be greater than that of the negative pair.

Formally, we optimize the model parameters Θ by minimizing the regularized BPR objective. Given a user u, a positive community c_i (observed interaction), and a negative community c_j (sampled unobserved interaction), the total loss $\mathcal{L}$ is defined as:

$$\mathcal{L} = - \sum_{(u,c_i,c_j) \in \mathcal{D}} \ln \sigma(\cos(\mathbf{u}, \mathbf{c}_i) - \cos(\mathbf{u}, \mathbf{c}_j)) + \lambda ||\Theta||^2 \tag{4.1}$$

where σ is the sigmoid function, λ is the regularization hyperparameter, and $\cos(\cdot, \cdot)$ denotes the cosine similarity between the user and community embedding vectors.

This process yields a set of community embeddings that are semantically organized based on user co-membership patterns, capturing structural relationships independently of any text content. These pre-trained community embeddings serve as the high-quality initial feature representation for the community nodes in our GASTON architecture.

4.3.3 Model Architecture

The core of GASTON, as illustrated in Figure 4.4 is a multi-layer Heterogeneous Graph Transformer (HGT) [21]. Through GASTON, we introduce trainable community parameters that allow the structural identity learned during initialization to be further refined during pre-training.

Trainable Community Parameters

A defining feature of GASTON is the treatment of community features as learnable parameters rather than fixed input vectors. While text nodes rely on the EmbeddingGemma encoder, the community node features $X_{Community}$ are instantiated as **learnable parameters** within the GNN. In baselines like OMCA, the community feature $X_{Community}$ is a static vector derived from averaging text. This locks the community's identity to its content. In contrast, GASTON instantiates $X_{Community}$ as a trainable embedding matrix, seeded with our structural BPR initialization. During the pre-training phase, gradients from both the text reconstruction and edge generation objectives backpropagate to these

embeddings. This allows the representation to drift from its initialization during training, effectively learning a hybrid identity that balances its structural origins with the semantic reality of the discussion. These optimized community embeddings are then frozen and transferred to all downstream tasks, serving as a universal community profile.

4.3.4 Dynamic User Embeddings

From the collaboratively learned community embeddings, we generate dynamic user representations.

User node features are then dynamically generated in each training batch. For every user in a given batch, we identify all communities they are active in using a pre-computed aggregation map. Their input feature is then calculated by performing a mean aggregation operation over the current embedding vectors of their associated communities. We utilize the mean operation as it is permutation-invariant and handles users with varying numbers of community memberships effectively. To ensure efficiency during training, the mapping of users to their communities is pre-computed and stored in an adjacency list before the training loop begins.

Node Type Projections

To process these learnable community embeddings alongside the text embeddings, we implement a **Node Type Tagging** layer. Before the first message-passing layer, every node's input feature vector X_v is projected into a shared hidden dimension d via a type-specific linear transformation. Formally, for a node v of type $\tau(v)$, the projected feature $H_v^{(0)}$ is calculated as:

$$H_v^{(0)} = W_{\tau(v)} \cdot X_v + b_{\tau(v)} \tag{4.2}$$

where $H_v^{(0)} \in \mathbb{R}^d$ is the initial projected hidden state for node v; X_v is the original input feature vector for node v (derived from EmbeddingGemma for text, or BPR initialization for communities); $\tau(v)$ represents the node type of node v (e.g., User, Text, Community); $W_{\tau(v)} \in \mathbb{R}^{d \times d_{in}}$ is the learnable weight matrix unique to that specific node type; and $b_{\tau(v)} \in \mathbb{R}^d$ is the corresponding learnable bias vector. This projection ensures that features from different modalities (text vs. structural embeddings) are mapped into a shared latent space before message passing begins.

Maintaining distinct, trainable weight matrices W_{Text}, W_{User}, and $W_{Community}$ ensures that the model prevents representational collapse and that the semantic signal of a text node and the structural signal of a community node stay distinct, even as they are projected into the same latent space.

Heterogeneous Message Passing

Following this projection, the projected features $H^{(0)}$ are fed into L layers of a Heterogeneous Graph Transformer. Unlike standard GNNs (e.g., GCN) which apply shared parameters, the HGT leverages our schema by maintaining specific parameters for each relation type. This allows GASTON to learn context-specific aggregation, ensuring that the rich, pre-trained community context is effectively propagated to users and posts.

4.4 Pretraining Setup and Tasks

GASTON is pre-trained on the full heterogeneous graph using a multi-task, self-supervised objective, similar to the OMCA framework [30]. The final loss is a weighted combination of two objectives: Text Reconstruction and Edge Generation.

1. **Text Reconstruction (Masked Feature Regression):** This task forces the model to learn context. We randomly mask a fraction of text nodes in each batch, replacing their input features with a single, shared, learnable `mask` embedding. The HGT model must then use the surrounding graph context to predict the original, unmasked feature vector. The output embedding of each masked node is passed through a decoder, and the error between the reconstructed and original feature vectors is minimized.

2. **Edge Generation (Link Prediction):** This task trains the model to capture the intrinsic graph structure and learn the connections among users, text, and communities. This is formulated as a contrastive link prediction objective across all relation types in our graph.

   After the HGT model produces final node embeddings for a batch, a similarity score (via dot product) is computed between relevant sets of source and target nodes. The model is then trained to distinguish between positive edges and negative edges in the batch.

We use true connections as positive samples and all other potential connections within the batch as negative samples, using cross entropy loss to directly optimize the model to increase the similarity score for positive pairs, while decreasing it for negative pairs. This unified objective effectively maximizes the likelihood of generating the correct graph structure, ensuring the final embeddings are structurally aware.

The final training objective is a weighted sum of these two components: $L = \alpha \cdot L_{\text{recon}} + (1 - \alpha) \cdot L_{\text{edge}}$. Here, α is a hyperparameter that balances the importance of the semantic reconstruction task versus the structural link prediction task. This ensures the model learns to balance semantic understanding with structural awareness.

Figure 4.4 illustrates how our architectural choices drive the learning of community identity. Crucially, note the Community Node (Orange) is not fed fixed features like the text nodes. Instead, it is a trainable parameter (highlighted in the figure) that is updated by gradients from both the text reconstruction loss ($\mathcal{L}_{recon}$) and the edge prediction loss ($\mathcal{L}_{edge}$). This forces the community embedding to act as a bridge: it must be structurally consistent to satisfy the edge task, but semantically informative enough to help reconstruct masked text in its neighborhood.

4.4.1 Pretraining Metrics

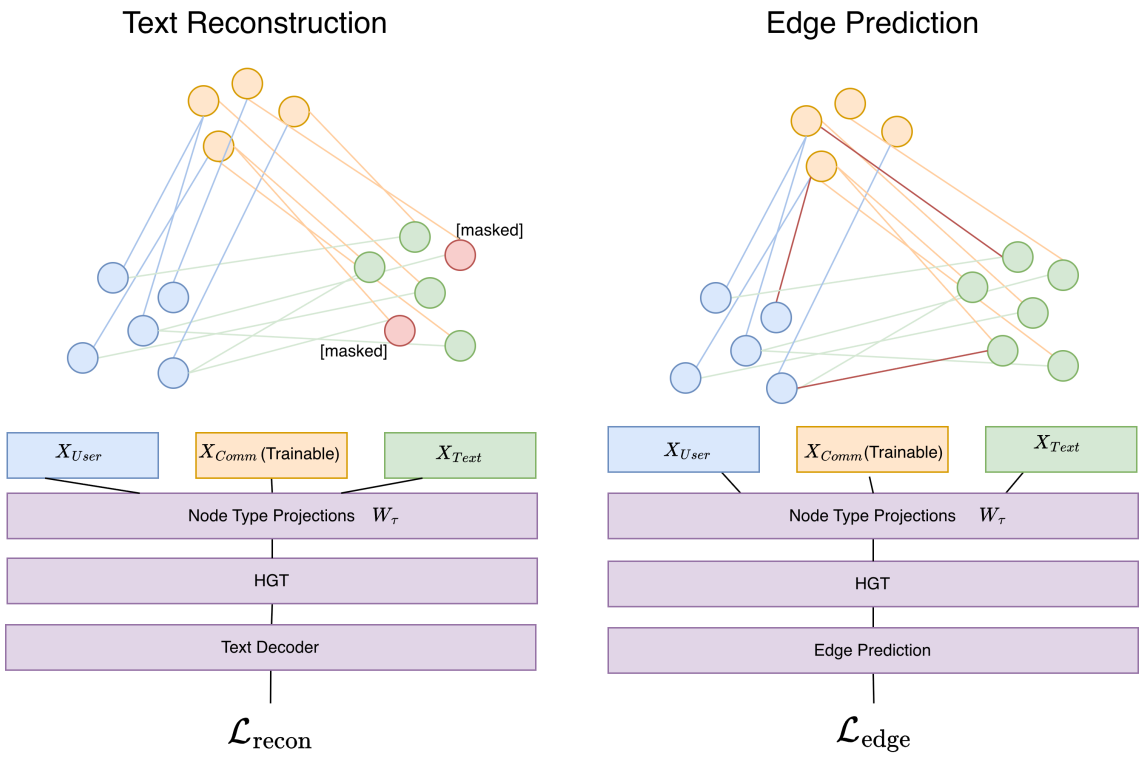
As defined in our loss function, the model optimizes for Text Reconstruction and Edge Link Prediction simultaneously. We use the following metrics to track convergence and representation quality.

Text Reconstruction Metrics

For the masked language modeling task, we employ **Mean Squared Error (MSE)** to quantify the semantic preservation of the node features. Given a batch of masked text nodes, we compare the reconstructed feature vector $\hat{x}_i$ (output by the decoder) against the ground-truth feature vector x_i (from the frozen EmbeddingGemma encoder). For a set of masked nodes M, the loss is defined as:

$$L_{recon} = \frac{1}{|M|} \sum_{i \in M} ||x_i - \hat{x}_i||_2^2 \tag{4.3}$$

Minimizing MSE ensures that the GNN aggregates sufficient neighborhood context to accurately recover the semantic content of a post without seeing its text.



$$\mathcal{L}_{total} = \alpha \mathcal{L}_{\text{recon}} + (1 - \alpha)\mathcal{L}_{\text{edge}}$$

Figure 4.4: The GASTON Pre-training Architecture showing Text Reconstruction and Edge Prediction tasks. The loss weighting parameter α is set to 0.5 to balance the semantic and structural objectives equally.

Edge Generation Metrics

For the link prediction tasks, we explicitly monitor performance on all forward edge types: $\langle Text, posted_by, User \rangle$, $\langle Text, posted_in, Community \rangle$, and $\langle User, active_in, Community \rangle$. We evaluate the model using **Cross Entropy Loss**.

As our edge generation objective treats link prediction as a binary classification problem (distinguishing true edges from negative samples), this loss provides a direct measure of how well the model discriminates between valid and invalid structural connections.

We compute the loss separately for each forward edge type to ensure the model is learning all primary generative relationships effectively. For a given batch of N edges, where $y_i \in \{0, 1\}$ represents the true label (1 for real edges, 0 for negative samples) and $\hat{y}_i$ represents the model's predicted probability, the loss is calculated as:

$$L_{edge} = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \tag{4.4}$$

Minimizing this cross entropy loss indicates that the learned user and community embeddings are successfully aligned in the latent space, effectively maximizing the likelihood of observing the true graph structure while suppressing random pairings.

Chapter 5

Experiments and Discussion

5.1 Data

5.1.1 Pretraining Data

Reddit is a social media platform where individuals can interact in different online communities called subreddits. Each subreddit is topic-specific, and users can read, post, upvote, and downvote other's posts in their interested subreddit. Subreddits inherently possess the characteristics of online communities, which makes Reddit an ideal data source for learning context-aware representations of online discussions.

There are two types of posts on Reddit: submissions and comments. Submissions are top-level posts in a subreddit, and comments are posts made below any submission. Comments are threaded: They can be made directly on submissions or appear as a reply to another comment.

The development of a high-quality representative dataset is a prerequisite for the successful training and evaluation of machine learning models. The Reddit platform, with its vast collection of user-generated content, presents an exceptional data source for the study of online communities. This section details the multi-stage process for constructing a focused and sanitized corpus from Reddit.

Our pre-training data is drawn from a comprehensive historical archive of the Reddit platform, acquired via the Reddit API, consisting of all available user, community, and post data within the selected time range. To address the heterogeneity of topics and community dynamics, a stratified sampling strategy was used. We curated a representative subset of

subreddits using data collected between January 2017 and December 2021. Instead of sampling from the entire platform for that entire date range, which would introduce a significant volume of data, we curated a representative subset of subreddits, and samples a number of posts from each month. This was accomplished by covering all subreddits found in our downstream tasks, in our pretraining dataset. This approach ensures that the resulting corpus captures all subreddits in downstream tasks, and from analysis, we see it covers a broad diversity of topics, ranging from highly technical discussions to general-interest conversations.

To improve data quality and distill meaningful signals from the content, two primary noise reduction heuristics were applied. First, only top-level submissions that received a score of at least 25 were selected. This score-based filter acts as a community-driven signal of quality, as posts with a higher score are more likely to contain substantive content and less likely to be spam or low-effort submissions.

Second, conversational threads were truncated to a depth of four comments. This pruning strategy removes the "long-tail" of discussion, which often consists of off-topic remarks, personal exchanges, or repetitive content. By focusing on the initial, high-quality portions of the threads, this step significantly reduces the presence of conversational noise while retaining the core exchange.

The application of these curation and noise-reduction techniques yielded a clean, focused dataset consisting of about 6100 subreddits and a high-quality selection of posts and their most relevant conversational content.

5.1.2 Downstream Tasks

The purpose of our chosen downstream tasks is to evaluate the generalizability of the pretrained model's representations across different tasks. Moreover, we aim to demonstrate that community and user representations are not just supplementary features, but integral context required to disambiguate the meaning of text in these tasks. Our tasks can be grouped into 2 categories; Node-Level Prediction and Edge-Level Prediction. Our set of tasks span a range of goals, from text prediction to recommendation. Crucially, our framework is not limited to these specific benchmarks. Any social dataset can be adapted as a downstream task, provided it can be reconstructed into a graph representation linking users, text, and communities. A table summarizing our downstream tasks, their categories, and the corresponding datasets can be found in Table 5.1.

Table 5.1: Summary of Downstream Evaluation Tasks

Task	Dataset	Size	Type	Graph Level	Metrics
Mental Health Detection	Dreaddit	4k	Binary Classif.	Node (Text)	Acc, F1
Toxicity Scoring	Ruddit	5k	Regression	Node (Text)	RMSE, r
Norm Violation	NormVio	45k	Binary Classif.	Node (Text)	Acc, F1
Hate Speech Detection	HatefulDisc.	61k	Binary Classif.	Node (Text)	Acc, F1
Comm. Recommendation	Reddit (Ours)	9.8M	Link Prediction	Edge	NDCG, MRR

Dreaddit

One of the key downstream tasks for evaluating the efficacy of our unified graph pre-training model is Mental Health Stress Detection using the Dreaddit dataset [41]. This is a classic binary classification problem that requires the model to predict whether a post from a mental health subreddit expresses stress or no stress. We enriched the original text-only dataset by retrieving the corresponding user and community data, transforming the isolated posts into a connected graph. The primary challenge lies in the nuanced, context-dependent nature of expressing mental distress online. Within our heterogeneous graph framework, we fine-tune a simple classification head on the learned text node embedding. The text node embedding captures the post's content and its local social context (from the author and the post's community), along with information flowing from the community node embedding, which provides a high-level representation of the subreddit's overall thematic and behavioral profile. We hypothesize that by utilizing this rich contextual information, our approach will surpass pure text-only models by implicitly incorporating both the specific social interaction and the expected thematic content of the community itself.

Ruddit

The Ruddit dataset [19] is used as a critical benchmark for evaluating the model's capacity to detect and quantify offensive language and toxicity. Ruddit is unique because it provides a fine-grained, real-valued score for the degree of offensiveness, ranging from -1 (maximally supportive) to +1 (maximally offensive). This enables us to formulate the task as a regression problem, which is more challenging and informative than simple binary classification. The model's objective is to predict this continuous offensiveness score. We use the text node embedding as the primary input to a regression head, with information flowing from the community node embeddings. To generate heterogeneous graphs,

we enriched the dataset by retrieving the corresponding user and community data. We hypothesize the community node embedding helps the model recognize whether the text's tone or topic aligns with the specific, potentially tolerant or aggressive, norms learned for that community. Success on this task will validate the pre-trained embeddings' ability to generalize from structural and semantic pre-training to a nuanced social dimension like toxicity grading.

NormVio

The Norm Violation Detection (NormVio) task, based on the dataset introduced by Park et al. [35], serves as a benchmark for evaluating the model's ability to capture community-specific rules and contextual sensitivity. This task is formulated as a binary classification problem: detecting whether a comment violates that subreddit's rules. The NormVio dataset, compiled from conversations across thousands of subreddits, moves beyond simple toxicity to encompass a wide spectrum of subreddit-specific rules enforced by moderators, such as format, off-topic content, and incivility, often categorized into nine fine-grained classes. Since the acceptability of a comment is highly dependent on the community's established norms, this task directly tests the GNN's ability to leverage global context. We once again reconstructed the missing social context by mapping the text samples to their original authors and subreddits, thereby generating the heterogeneous graph required for GASTON. We fine-tune a binary classification layer on the text node embeddings (representing the comment) with information flowing from connected nodes, in particular, the community node (representing the subreddit's unique normative profile). GASTON and all other baseline models are never explicitly provided with the text of the subreddit's written rules. Instead, it must infer these boundaries entirely from the latent patterns encoded in the community embedding, effectively learning to recognize the "unwritten rules" of the space. We hypothesize that the context of the community node embedding will significantly improve performance over context-agnostic baselines by providing the model with the necessary community nuances to determine if a specific piece of text truly constitutes a violation within that social environment.

HatefulDiscussions

To assess the model's capability for detecting high-stakes, socially sensitive content, we utilize the HatefulDiscussions dataset, a curated corpus derived from the work by Hebert et al. [17]). To suit the needs of this method and model, this task is framed as a binary classification problem, specifically focusing on classifying a comment as hateful or

non-hateful, independent of its submission hierarchy. This setup allows the model to concentrate specifically on the text content, the author, and the community context. We fine-tune a binary classification head atop the pre-trained HGT and the HatefulDiscussions constructed graph. This approach is crucial because the perceived severity and nature of hate speech are often influenced by the norms of the containing community; what is tolerated in one subreddit might be flagged in another. By integrating the community and user context, we hypothesize the model gains the necessary contextual lens to improve the precision of hate speech detection, validating that our graph representations can effectively capture and utilize the nuanced, community-specific social boundaries required for robust moderation tasks.

Recommendation

To evaluate the pre-trained model's proficiency in modeling user behavior and relational patterns, we evaluate the Community Recommendation task, which is framed as an edge prediction problem. This task requires the model to predict novel, yet likely, edges between user nodes and community nodes, effectively predicting what communities a specific user will likely engage with or "enjoy". Crucially, this serves as our primary non-text-based benchmark. Unlike the text-based classification tasks, this task directly assesses the quality of the learned user and community representations as they relate to structural compatibility within the graph. The fine-tuning process is set up as a typical edge prediction task: we train a prediction head to score the probability of an edge existing between the user node embedding and the community node embedding. During evaluation, the goal is to distinguish between a set of true (observed) and false (unobserved) edges. We fine-tune the model on user-community interactions completely disjoint from the pre-training set. High performance on this task will validate that the pre-training objective successfully learned transferable, high-quality relational features, enabling the User and Community embeddings to align in the latent space based on genuine, unobserved user preference, thus validating the pre-trained model's utility in real-world recommender systems.

5.1.3 Evaluation Metrics

To evaluate GASTON across our diverse set of downstream tasks, we employ specific metrics for each type of problem: classification, regression, and recommendation.

Classification Metrics (Binary and Multi-Class)

For node-level classification tasks (Dreaddit, NormVio, HatefulDiscussions), we report **Accuracy** and **Macro-F1 Score**. While Accuracy measures the overall correctness, Macro-F1 is critical for our datasets, which often suffer from class imbalance. The F1-score is calculated as the harmonic mean of Precision (P), the fraction of relevant instances among the retrieved instances, and Recall (R), (the fraction of relevant instances that were retrieved).

Regression Metrics

For the Ruddit toxicity scoring task, which requires predicting a continuous value, we evaluate performance using **Root Mean Squared Error (RMSE)** and **Pearson Correlation Coefficient (r)**.

To calculate RMSE, we first compute the Mean Squared Error (MSE), which quantifies the average squared difference between the predicted toxicity score $\hat{y}$ and the ground truth score y. For a dataset of size N, MSE is defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{5.1}$$

We report the RMSE, which is the square root of the MSE:

$$RMSE = \sqrt{MSE} \tag{5.2}$$

Using RMSE returns the error metric to the original scale of the dataset (toxicity scores ranging from -1 to 1), allowing for a direct interpretability of how far, on average, the model's predictions deviate from human annotations. Additionally, Pearson's r measures the linear correlation between predictions and ground truth, indicating how well the model ranks toxicity relative to human annotators.

Recommendation Metrics

For the User-Community recommendation task, which is a ranking problem, we adopt information retrieval metrics: **Mean Reciprocal Rank (MRR)** and **Normalized Discounted Cumulative Gain (NDCG)**.

MRR evaluates the model's ability to place the correct community high in the predicted list. For a set of users U, where $rank_u$ is the position of the ground-truth community in the model's sorted prediction list:

$$MRR = \frac{1}{|U|} \sum_{u \in U} \frac{1}{rank_u} \tag{5.3}$$

NDCG@k (specifically at $k = 10$) evaluates the quality of the top-k recommendations, giving higher importance to correct predictions at the very top of the list. NDCG@k is calculated as the Discounted Cumulative Gain (DCG) divided by the Ideal DCG (IDCG). DCG measures the quality of the ranking by summing the relevance scores of recommendations, discounted logarithmically by their position in the list. IDCG represents the score of a perfect ranking order.

5.2 Baselines

5.2.1 Naive Baselines

To establish a performance floor and quantify the value of graph modeling, we use two simple text-based baselines. First, BERT-only serves as a context-agnostic benchmark; it is a standard pre-trained BERT classifier fine-tuned solely on the textual content of the post, ignoring all author and community information. Second, BERT + Community represents the simplest method of incorporating social context. In this model, we employ the community2vec algorithm [31] to generate user-membership based community embeddings. During fine-tuning, this community vector is concatenated with the BERT text representation before being passed to the classification head, allowing the model to see where a post was made, but without the complex message-passing dynamics of a GNN.

5.2.2 OMCA

As our primary state-of-the-art comparison, we utilize our replication of the OMCA framework [30]. As described above, OMCA is a heterogeneous graph transformer that unifies users, text, and communities. However, it relies on a text-centric aggregation strategy where community identity is merely a derivative of its content. We aim to shift this paradigm by proposing a community-centric strategy that treats a community as a distinct, first-class structural entity. This baseline allows us to directly test if GASTON's structural

contrastive initialization and modern embedding backbone offer an improvement over the current standard in unified community modeling.

5.3 Experimental Environment

5.3.1 Hardware and Data Split

Experiments [1] were conducted using an RTX 6000 Ada Generation GPU with 160 GB of RAM and 16 CPU cores. All models, including GASTON, baselines, and all finetuned models, were trained and evaluated using PyTorch, PyTorch Geometric, and PyTorch-Lightning. PyTorch-Lightning is a deep learning framework which wraps PyTorch for streamlined model training and evaluation.

The final graph used for training in all pre-trained models consisted of 6.4 million pieces of text, 6132 communities, and 4.7 million users. Consequently, there were 6.4 million ⟨Text, posted_by, User⟩ edges, 6.4 million ⟨Text, posted_in, Community⟩ edges, and 21 million ⟨User, active_in, Community⟩ edges in our final pre-training graph.

The majority of our downstream tasks have a predefined train-test split. For the remaining datasets, we use an 80-10-10 train-test-validation split. We perform train, test, and validation dataset splitting at the user level to prevent data leakage during training. For downstream tasks, we use the provided train/test split where applicable, and use 15% of the training data for validation. To manage dataset imbalance in the downstream task data, we apply weights to each class when calculating Cross Entropy Loss. We apply seeds in all models and data splitting to ensure reproducibility and fair comparison across experiments.

5.3.2 Model Configuration and Training Details

All models shared a consistent hyperparameter configuration. The embedding dimension for all node features and the hidden channel size within the HGT layers was set to $d = 768$. The HGT architecture was composed of 3 layers employing 8 attention heads. In total, the full GASTON model contains approximately 26 million trainable parameters.

[1] All code, including experimental scripts, model and dataset configurations, are publicly available at https://github.com/olhawloch/gaston

For the pre-training phase, we utilized the Adam optimizer with a learning rate of 1e−4 and a batch size of 128. To handle the scale of the full heterogeneous graph, which exceeds typical GPU memory constraints, we employed a `NeighborLoader` to sample local subgraphs for each batch. The loss weighting parameter α, which balances the text reconstruction and edge generation objectives, was set to 0.5 across all experimental setups. This value was selected to assign equal importance to both structural and semantic learning signals, establishing a balanced baseline without biasing the model toward one modality.

Each pre-trained model was trained for a fixed duration of 6 epochs to generate the parameters used for fine-tuning. This procedure is computationally intensive due to the high memory consumption of the graph states; the pre-training of the fully trainable GASTON model required approximately 34 hours to complete on the specified hardware.

5.4 GASTON Performance

Table 5.2 presents the main experimental results across all five downstream tasks. Overall, we see consistent directional improvement in tasks with improved context. GASTON achieves state-of-the-art performance on 3 out of 5 tasks, specifically excelling in socially nuanced classification problems.

In the Norm Violation and Hate Speech Detection tasks, GASTON significantly outperforms both the text-only baselines and the OMCA graph baseline (e.g., +21% F1 improvement over BERT on NormVio). This confirms our hypothesis that community-aware representations are critical for detecting context-dependent violations. In Community Recommendation, GASTON also achieves the highest ranking performance, validating the efficacy of our contrastive initialization.

However, for tasks with smaller datasets like Ruddit (toxicity) and Dreaddit (stress), we observe that the simpler BERT baselines perform competitively with, or slightly better than, the graph models. As detailed below, we attribute this to the risk of overparameterization when applying complex GNNs to data-scarce environments. Notably, even in these cases, the BERT + Community baseline consistently outperforms the text-only BERT baseline (e.g., 0.894 vs 0.854 Accuracy on Dreaddit). This reinforces our central hypothesis: regardless of model complexity, incorporating community context is essential for accurate social prediction.

Table 5.2: We report Accuracy and Macro-F1 for classification tasks, Pearson's r and RMSE for regression, and NDCG@10 and MRR@10 for recommendation. **Higher is better** for all metrics, except **RMSE** where lower is better (↓). Best-performing model is in bold.

Model	Node-Level Tasks								Edge-Level Task	
	Dreaddit		NormVio		HatefulDisc.		Ruddit		Recommend.	
	F1	Acc	F1	Acc	F1	Acc	r	RMSE (↓)	NDCG@10	MRR@10
BERT-only	0.911	0.854	0.696	0.754	0.841	0.828	0.663	0.250	-	-
BERT + Community	**0.939**	**0.894**	0.710	0.761	0.889	0.854	**0.670**	**0.248**	-	-
OMCA (Ma et al.)	0.929	0.878	0.961	0.953	0.916	0.892	0.525	0.292	0.011	0.009
GASTON (Ours)	0.914	0.856	**0.971**	**0.967**	**0.920**	**0.901**	0.598	0.271	**0.055**	**0.061**

5.4.1 Qualitative Analysis and Error Analysis

To better understand the quantitative results presented in Table 5.2, we conduct a qualitative analysis of specific failure and success cases. While the aggregate metrics on smaller datasets (like Ruddit and Dreaddit) may lack statistical significance due to limited sample sizes, this examination reveals a qualitative improvement in the model's reasoning. We find that GASTON consistently captures context-dependent nuances, such as distinguishing support from toxicity, that text-only baselines systematically miss. This suggests that despite the statistical limitations, the model's underlying representations are better aligned with the complex socio-semantics of online interaction.

The Challenge of Small Datasets: Ruddit

While GASTON outperforms OMCA on the Ruddit toxicity regression task (0.271 RMSE vs 0.292 RMSE), we observe that the simpler BERT-based baselines achieve slightly superior performance (0.248 RMSE). This phenomenon highlights the over-parameterization risk inherent in complex Graph Neural Networks when applied to small datasets. The Ruddit dataset contains fewer than 6,000 samples. In a data-scarce task, the structural signal provided by the graph is sparse, and the model struggles to propagate meaningful messages without overfitting. The BERT-only baseline, which relies solely on the semantic content of the text without the overhead of learning graph topology, is less prone to this noise on smaller datasets. This suggests that while GASTON excels at scale, its graph-aware components require a critical mass of data to distinguish valid social signals from noise.

The Challenge of Small Datasets: Dreaddit

We observe a similar trend with the Dreaddit dataset. Through GASTON, we achieved an accuracy of 0.856, which is effectively the same as the basic BERT baseline (0.854) but slightly lower than the simpler BERT + Community method (0.894). With only about 4,000 nodes in the training graph, it is possible that the dataset is too small for a complex Graph Neural Network to be fully effective. While GASTON attempts to learn deep structural relationships, there may not be enough interaction data here to generate a strong signal. In this specific case, the simpler baseline, which just adds a static community embedding to the text, appears to be the more efficient approach, providing helpful context without the need for complex training.

Sparsity in Recommendation Tasks

For the Community Recommendation task, GASTON significantly outperforms OMCA (NDCG@10 of 0.055 vs 0.010). However, the absolute values for these metrics remain low. This is primarily driven by the extreme sparsity of the user-community interaction graph. Our analysis reveals that the average user in the validation set has interacted with only 4.7 communities, with a median of just 2 interactions.

Furthermore, the evaluation metrics (NDCG and MRR) penalize the model for "incorrect" ranking order. However, the ground truth data is binary (interaction vs. non-interaction) rather than an ordered preference list. The model might recommend a highly relevant community that the user hasn't visited yet; in the evaluation, this counts as a false prediction. If the dataset contained richer user histories, we hypothesize that the gap between GASTON and the baselines would widen further, as the collaborative filtering signal would be stronger.

Case Study: The Necessity of Community Context

The most significant performance gain is observed in the Norm Violation (NormVio) task, where GASTON achieves an F1 score of 0.972 compared to the BERT baseline's 0.696. This massive improvement validates our core hypothesis: what constitutes a violation is defined by the community, not just the text.

We highlight two examples from our test set where graph-based models (GASTON and OMCA) succeeded while the text-only baseline failed.

Example 1: Sharing Personal Information

*"Can you add my snap? D**********103 I'm 18"*

Prediction Analysis:

- **BERT-only:** Classifies this as *Non-Violation*. The model analyzes the semantics and finds no toxicity, aggression, or hate speech. It interprets the text as a neutral social request.
- **OMCA:** Classifies this as *Non-Violation*. Despite using a graph structure, the model fails to capture the specific constraints of `r/teenagers`. It prioritizes the benign semantic content of the post over the signal indicating a rule violation.
- **GASTON:** Correctly classify this as a *Violation*. The graph models incorporate the `r/teenagers` community node, which represents a subreddit with strict safety rules against sharing Personally Identifiable Information (PII) or soliciting off-platform contact. By leveraging the community context, the models understand that exchanging Snapchat handles is a normative violation in this specific space.

Example 2: Unconstructive Commentary

"You're pretty enough to get a boyfriend? You're ahead of the rest of us."

Prediction Analysis:

- **BERT-only:** Classifies this as *Non-Violation*. The text contains positive sentiment words like "pretty" and "ahead", which confuses a model trained primarily on toxicity or sentiment. It likely interprets this as a compliment.
- **Graph Models (GASTON & OMCA):** Correctly classify this as a *Violation*. In the context of this being in the BodyDysmorphia subreddit, this comment functions as a backhanded compliment or unconstructive comparison, violating the norm "Be Kind". The graph models perceive the misalignment between this type of post and the community's established norms for supportive interaction.

These examples demonstrate that text-only models suffer from context blindness, and existing graph models miss the nuance in these pieces of text. They fail to detect violations that are defined by rules rather than by vocabulary. GASTON provides the grounding to align the definition of a violation with the specific social environment.

Case Study: The Echo Chamber Effect in Extreme Communities

A failure mode of models like OMCA is the "normalization of toxicity" within echo chambers. If a community's discourse is uniformly extreme, the average semantic vector of that community will reflect that extremism. Thus, a hateful comment might be mathematically close to the community's center, leading the model to interpret it as normative and therefore acceptable.

We observe this phenomenon clearly in examples from `r/TumblrInAction`, a subreddit known for extreme anti-progressive rhetoric that has since been banned from the platform for violating Reddit's hate speech policies.

Consider the following examples from this community which GASTON correctly identified as Hateful, but OMCA misclassified as Non-Hateful:

- *"The act of blowing a tr*nny takes on very, very different meanings depending on the context..."*
- *"ehh... it gets iffy. Some people aren't cool with it even if they have a vagina. [...] If it happened to me I'd be hurt maybe a little offended, but overall I would understand and not blame the [sic]. (unless they went full-on tr*nny hate)"*
- *"She's pretty well-known for being volatile [...]. Screaming at and berating the ref for doing his job [...] shows an astonishing lack of class. She was acting like a selfish brat and ruined the actual winner's big moment by being such a tw*t. I'm appalled anyone is defending her behavior."*

Failure Analysis:

- **OMCA:** OMCA initializes the community representation by averaging the posts within it. In a subreddit like `r/TumblrInAction`, where hostility was the norm, these comments align with the average discourse. The model likely perceived these comments as fitting the community standard and classified them as non-hateful.
- **GASTON:** GASTON learns the community's identity through user co-occurrence. The users active in `r/TumblrInAction` likely overlapped with other controversial or high-toxicity communities. GASTON learned a community embedding that placed `r/TumblrInAction` in a high-risk region of the latent space. Consequently, when the model processed these comments, the Community Node acted as a bias correction, effectively acting as a contextual prior. Crucially, this does not mean the model

labels all speech in this community as hateful. Rather, for ambiguous or borderline content—which might pass as neutral in isolation—the model uses this negative community signal to shift the decision boundary, correctly identifying the hateful intent that text-only models miss.

This provides evidence that GASTON is robust to the "Echo Chamber" effect, capable of detecting toxicity even when it is the dominant norm within a specific subgroup.

5.4.2 Embedding Visualization

To validate the qualitative structure of our learned representations, we project a sample of the final community embeddings into a 2D space using t-SNE (t-Distributed Stochastic Neighbor Embedding), a statistical method for visualizing high-dimensional data by giving each datapoint a location in a two or three-dimensional map. Our community embedding visualization can be seen in Figure 5.1. Figure 5.1 reveals that GASTON can successfully organize the latent space into distinct, semantically coherent clusters. We observe tight groupings of communities based on shared topics and user bases:

- **Mental Health:** Communities such as `r/depression`, `r/suicidewatch`, and `r/anxiety` form a dense cluster, reflecting the high user overlap and shared support function of these spaces.
- **Politics:** Political subreddits, including `r/democrats`, `r/republican`, and `r/politics`, cluster together. Notably, despite their ideological differences, the model places them in the same structural neighborhood, capturing their shared utility as forums for political discourse.
- **Canadian:** Subreddits tailored specifically to Canadians clearly form a cluster in the top left quadrant of the embedding space, showing their contextual similarity amongst other subredddits.
- **Finance & Crypto:** A distinct cluster emerges for financial interests, grouping `r/investing` with `r/bitcoin` and `r/wallstreetbets`.

This geometric organization demonstrates that GASTON has successfully translated user interaction patterns and semantic signals into meaningful proximity, where the distance between two community vectors serves as a proxy for their sociological similarity.

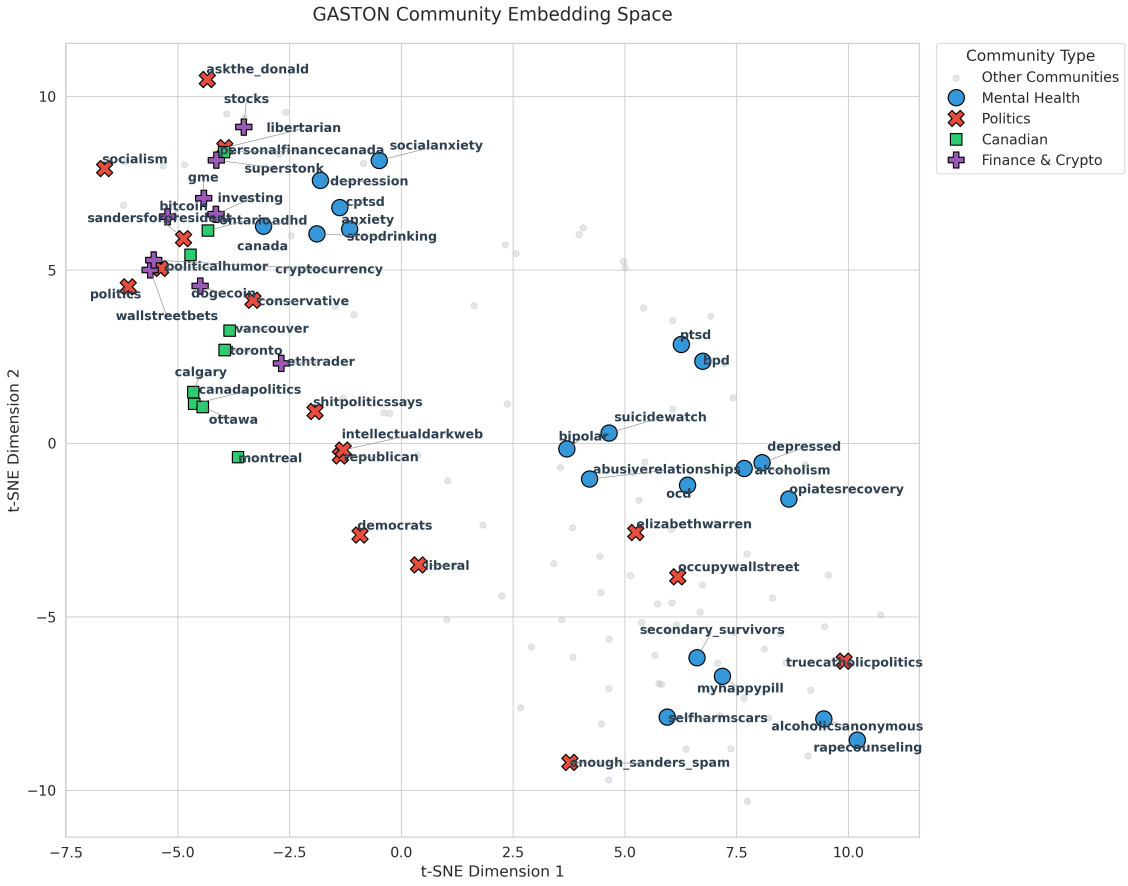


Figure 5.1: GASTON Trained Community Embeddings

Crucially, this structure emerges without the need for manual curation. Unlike Waller and Anderson [46], who mapped the social landscape by explicitly projecting communities onto axes defined by researcher-selected poles, GASTON enables latent sociological dimensions to surface organically. We do not impose a predefined normative framework. Instead, the organization is derived entirely from the user-community and semantic topology of the graph.

5.5 Ablations

To isolate the contributions of specific components within GASTON, we evaluate three variations of our architecture:

1. **GASTON (Frozen Embeddings)**: In this ablation, we initialize community nodes using our contrastive (BPR) method, but freeze these weights during the graph pre-training phase. This tests whether the structural signal learned from user-overlap is sufficient, or if the community representations benefit from being updated based on semantic signals during pre-training.
2. **GASTON (Average Initialization):** This ablation removes our contrastive initialization and replaces it with the text-averaging strategy used by OMCA (initializing a community as the mean of its posts). By keeping the rest of the GASTON architecture identical, this ablation isolates the specific impact of modeling communities based on structure versus semantics.
3. **GASTON (with XLNet):** This ablation replaces our text encoder, EmbeddingGemma, with XLNet, matching the encoder used in OMCA. This allows us to disentangle the performance gains attributed to our graph architecture from those attributed simply to using a more modern, instruction-tuned language model.

Table 5.3: We analyze the impact of the language model (Gemma vs. XLNet) and the community initialization strategy (Trainable Community vs. Frozen vs. Average Initialization). **Higher is better** for all metrics, except **RMSE** where lower is better (↓).

Model Variant	Node-Level Tasks								Edge-Level Task	
	Dreaddit		NormVio		HatefulDisc.		Ruddit		Recommend.	
	F1	Acc	F1	Acc	F1	Acc	r	RMSE (↓)	NDCG@10	MRR@10
GASTON (Ours)	**0.914**	**0.856**	0.971	0.967	**0.920**	**0.901**	**0.598**	**0.271**	0.055	0.061
Community Initialization Ablations										
– with Frozen Embeddings	0.888	0.816	**0.972**	0.977	0.896	0.873	0.422	0.304	**0.057**	0.066
– with Average Init	0.906	0.842	0.754	0.783	0.885	0.861	0.386	0.312	0.052	**0.069**
Language Model Ablation										
– with XLNet (instead of Gemma)	0.889	0.817	0.971	**0.978**	0.913	0.895	0.561	0.277	0.033	0.043

5.5.1 Impact of Language Model Architecture: XLNet vs. EmbeddingGemma

As shown in Table 5.3, replacing XLNet [49] with EmbeddingGemma [45] yields consistent performance gains across all tasks, most notably in the Ruddit regression task and Rec-

ommendation. We attribute this improvement to the fundamental difference in training objectives between the two models. XLNet is a generalized autoregressive pre-training method designed primarily for sequence generation and probability estimation. While effective for many NLP tasks, it is not explicitly optimized to produce dense, semantically meaningful vector representations of full sentences without fine-tuning.

In contrast, EmbeddingGemma is an encoder-only model that has been instruction-tuned specifically for retrieval and representation tasks. This means its latent space is geometrically organized such that semantically similar concepts are clustered closer together before they even enter our graph.

5.5.2 Community Initialization Strategy: Trainable vs. Frozen Weights

A core contribution of GASTON is the use of collaboratively learned community nodes which are initialized using a contrastive BPR objective, then fine tuned during our pre-training task. In this ablation, we investigate whether these initialized weights should remain fixed (acting as a static structural prior) or be trainable (allowing them to evolve).

Frozen Embeddings: When we freeze the community embeddings after the contrastive initialization step, the model still outperforms the Average Initialization baseline significantly (e.g., NormVio F1 of 0.971 vs 0.754). This confirms that the structural "signature" learned from user membership patterns is highly predictive: knowing *who* interacts in a community is more informative than simply averaging *what* they write.

Trainable Embeddings: However, the fully trainable approach achieves the best overall performance. By allowing the initialized weights to be updated during the graph pre-training phase, GASTON benefits from both structure and semantics. The BPR initialization provides a strong prior belief about which communities are related based on user overlap. The graph training then refines this position based on the text content flowing through the network. This allows the community embedding to evolve into a hybrid representation: it starts as a purely structural entity (defined by its user base) and becomes a socio-semantic entity (defined by its users *and* its evolving norms), providing the rich context necessary for nuanced tasks like norm violation detection.

5.6 Societal Implications: From Algorithmic Censorship to Informed Engagement for Mental Health

The quantitative results presented above demonstrate GASTON's performance across standard metrics. However, in the domain of computational social science, and particularly in mental health applications, standard metrics do not fully capture the societal impact of a model.

This section discusses the implications of our findings, moving beyond the traditional goal of automated moderation, as done in previous research and in practice [3]. We argue that instead of building better censorship capabilities, we should use community-aware representations to empower users with the transparency they need to navigate online spaces safely.

5.6.1 The Double-Edged Sword of Online Support

As discussed in Chapter 1, social media acts as a double-edged sword for mental health [1]. Research indicates that while online communities provide vital peer support and identity affirmation for marginalized groups, they also carry significant risks. These include exposure to toxicity, misinformation, and "problematic use" patterns that correlate with anxiety and depression [38, 36].

A critical failure of standard text-only models in this domain is their tendency to flatten this complexity. By treating all negative text as toxic, context-blind models risk dismantling positive support networks. For example, a user venting about distress in a recovery group might be flagged for removal, stripping them of their support system when they are most vulnerable.

5.6.2 Emotional Contagion and User Agency

While support is vital, online communities can also facilitate the formation of echo chambers that normalize distress and amplify emotional contagion, which is the phenomenon where users unconsciously converge on the emotional states of their peers [26].

For vulnerable individuals, the risk is not merely exposure to explicit abuse (which standard models can catch), but exposure to *maladaptive norms*. A community that subtly encourages rumination or normalizes self-destructive behavior can trap users in a negative

feedback loop, often without their realizing it. Traditional context-blind models fail to detect this risk because they analyze posts in isolation, missing the cumulative normative pressure of the environment.

GASTON addresses this gap. By learning a distinct representation for the community itself, our model captures the emotional baseline of a digital space, distinguishing between a support group that fosters recovery and an echo chamber that reinforces pathology. This enables a shift from moderation to user agency, effectively providing the building blocks of an "algorithmic nutrition label" [24] that has the potential to provide users with the ability to assess the health of a community before engaging.

This transparency goes beyond simple descriptive statistics by leveraging the implicit signals encoded within GASTON's latent representations. By analyzing the geometric neighborhoods of a community within the embedding space, the system can identify latent associations—such as a gaming community that clusters surprisingly close to political extremist groups—that would be invisible to surface-level analysis. Furthermore, these representations can be augmented with external resources. By projecting external toxicity scores or independent reliability ratings onto the embedding space, we can define semantic axes (e.g., a "hostility plane") that allow us to quantify a community's orientation relative to known safe or harmful clusters, providing users with a multidimensional risk assessment.

Chapter 6

Conclusion

6.1 Conclusion

The central thesis of this work is that the meaning of online interaction is nuanced; it is dynamically constructed through the interplay of content and community norms. While modern language models have made immense strides in understanding what is written, this thesis establishes the necessity of community-centric representation learning. We demonstrate that for socially nuanced tasks, accurate modeling requires capturing not just the semantic content, but the structural identity of where it is written, treating the community itself as the critical lens through which all interaction is interpreted.

In this thesis, we presented GASTON, a unified heterogeneous graph architecture that successfully integrates the semantic power of language models with the structural insights of GNNs powered by transformers. By challenging the convention of initializing strategies, we introduced a Contrastive Initialization technique that prioritizes communities as entities, which are then fine tuned during pre-training to generate meaningful contextual representations for downstream tasks. Our extensive evaluation confirms that this approach yields promising representations. Whether detecting norm violations or recommending communities to users, GASTON proved that a model grounded in the structural reality of the network can read the room that it is in.

Beyond quantitative metrics, this work contributes to a necessary shift in how we approach online safety. We argued that the prevailing strategy of automated moderation, which treats safety as a binary classification task, is insufficient for the complex social landscape of the modern web. Instead, we propose that the true utility of community-centric representation learning lies in transparency.

Mitigating harm in AI systems requires leveraging context to understand suitability. GASTON leverages this principle for social media. By effectively combining structural and semantic signals, GASTON provides a robust foundation for mapping the complexity of our online social landscape. This capacity to model unique social dynamics allows us to represent the inherent nuance of social interactions. We recognize that online discourse is rarely black and white, but exists in a gray area governed by local norms. Ultimately, GASTON demonstrates that by capturing the rich, heterogeneous context of online networks, we can build systems that do not merely process text, but truly understand discourse online.

6.1.1 Limitations and Future Work

Although GASTON established a strong foundation for community-centric representation learning, several high potential avenues remain for future research.

Incorporating Temporality with Dynamic Graphs

The current GASTON model, like most pre-training frameworks, operates on a static snapshot of the graph, aggregating interactions over time. This approach is computationally efficient, but temporally blind, ignoring the dynamic nature of online communities. A significant next step would be to model the graph as an evolving sequence of events. This could be achieved by adapting the GASTON architecture into a Temporal Graph Network (TGN) [39]. In this setup, interactions like a user posting, a user joining a community, or text replying to another would be treated as timed events. The model would learn node representations that evolve over time, potentially using a memory module to maintain a compressed history of each user and community. This would unlock the ability to model community evolution, user interest drift, and the real-time propagation dynamics of a phenomena like misinformation or emerging mental health crises.

Improving the Graph with Richer Relations and Features

The current graph, while effective, could be enriched to capture a more detailed view of social interactions. A key improvement would be adding more granular edge types, such as $\langle text, replying_to, text \rangle$ relations. Modeling the explicit reply-tree structure of conversations would allow the GNN to learn even more conversational context, distinguishing between top-level comments and deep, branching discussions. Furthermore, the nodes themselves could be enriched with additional features beyond their learned embeddings.

Further research may consider augmenting user nodes with features representing their activity frequency or account age, while community nodes could include features for size, post velocity, or other dynamic properties.

Latent Community Detection

While GASTON operates on the explicit, predefined structure of subreddits, online communities often contain distinct sub-groups or cliques. Future work could explore using the learned embeddings to perform latent community detection. This would allow the model to identify cohesive sub-communities within a subreddit (e.g., participants of schools of thought within a political forum) or across subreddits, offering a more granular view of social structure than the platform's boundaries provide.

The Cold Start Problem and Platform Scale

As detailed in Section 5.1, our pre-training strategy ensured that all subreddits appearing in downstream tasks were present in the training graph, allowing the model to utilize their learned structural embeddings. Consequently, communities completely unseen during pre-training would suffer from the cold start problem, as they lack a learned vector representation [52]. Future work should explore methods to generate embeddings for new communities based on their initial text content. However, it is worth noting that GASTON is explicitly designed for large-scale, platform-wide deployment. Its primary strength lies in its ability to digest massive historical logs to model the stable core of a network. In production environments, where the vast majority of traffic occurs within established communities, this platform-wide coverage ensures that the strongest context is available where it is most needed.

Multi-modal Representation Learning

Social media is an inherently multimodal environment, yet our model currently processes text, ignoring images and videos within posts. A powerful extension would be to incorporate the visual modality associated with posts. This would involve transforming GASTON into a Multimodal Graph Learning (MGL) framework, similar to architectures proposed in recommendation domains [47]. In such a model, text node's initial representation would be a fused embedding derived from both its text content and its image content. This may enable the model to capture a deeper understanding of content that relies on the interplay between text and image, such as memes, info-graphics, or visual-based hate speech.

Development of Ethically-Sourced, Large-Scale Datasets

A persistent challenge in this field is the availability of high-quality, large-scale labeled datasets. Many existing datasets for tasks like mental health analysis are either small, rely on proxy-labels (e.g., post removal), or have missing and/or deleted user information. These datasets also often label multiple substrings in a single post, which results in an even smaller dataset after consolidating labels at a post level, which was done in GASTON. Future work should call for the creation and open-sourcing of more nuanced, ethically sourced, and annotated datasets specifically designed for computational social science. These datasets are critical for validating models on complex social tasks and ensuring they generalize across platforms.

Explainability and Social Responsibility

As we propose a socially aware community-modeling tool such as GASTON, the interpretability of the model becomes a critical requirement. GASTON, like most deep learning architectures based on GNNs and Transformers, currently operates as a black box. While effective, its decision-making process remains hidden.

Future iterations of GASTON should incorporate Explainable AI (XAI) mechanisms to further strengthen the relationship between model performance and user trust. Research should focus on adapting GNN explanation methods, such as GNNExplainer [51], to the heterogeneous domain. By identifying which specific neighbors (users or communities) or edge types contributed most to a classification, we can provide users with granular reasoning for a risk score. Furthermore, analyzing attention weights within the Transformer layers [4] could reveal whether the model is reacting to specific features or to the broader normative history of the author.

This transparency is a prerequisite for socially responsible AI, specifically regarding algorithmic bias. Online networks often reflect societal inequalities, and there is a risk that graph embeddings may inadvertently encode these biases. This is known as algorithmic segregation. If a community is predominantly composed of a marginalized group, the model must not learn to associate the *identity* of that group with toxicity simply due to correlation in the training data. Future work must explore Fair GNN Frameworks [7] that use debiasing techniques to ensure that learned community representations do not rely on sensitive attributes, allowing them to remain fair without losing their usefulness for social analysis.

User-Facing Community Transparency

Historically, improved classification models have been used to fuel automated moderation. These models act as censorship machines that hide content deemed unsafe. We propose a fundamentally different application: using GASTON to create a Community Nutrition Label that empowers users to make their own safety decisions.

This tool could provide users with a summary of a community's health and characteristics. For instance, it could visualize a community's nearest neighbors in the embedding space (revealing its thematic alignments), or it could use the embedding as input to downstream classifiers that generate risk scores for toxicity, polarization, or echo-chamber effects. By presenting this community footprint to users, we could empower them to make more informed decisions about their engagement, help them identify potential echo chambers, and allow them to measure the community distance and normative differences between their current environment and new communities they wish to explore.

This algorithmic nutrition label can provide users with:

- **Pre-Exposure Risk Assessment:** Before engaging with a new community, a user could view a "footprint" derived from the community's alignment in the Ruddit and Dreaddit tasks. This might reveal that a specific gaming community has a high hostility score, or that a discussion group has a high distress normalization score.
- **Informed Consent:** This transparency shifts agency back to the user. Rather than an algorithm paternalistically hiding content, the user is equipped with the data to decide what they are emotionally resilient enough to handle at that moment.

This approach parallels recent advancements in "Dataset Nutrition Labels" [5], which demonstrate that raw ingredients alone are insufficient for assessing safety. It is the leveraging of context, specifically how data maps to intended use cases, that mitigates downstream harms. Similarly, GASTON uses the learned context of community norms that has the potential to act as a social nutrition label. Just as a dataset might be well-suited for one model but biased for another, a community's norms may be supportive for one user but toxic for another, a distinction that can only be made visible by surfacing the latent context of the network.

By rendering the invisible social context visible, GASTON transforms the definition of safety from *restriction* to *awareness*, allowing users to navigate the complex mental health landscape of the internet with their eyes wide open.

References

[1] Paul Best, Roger Manktelow, and Brian Taylor. Online communication, social media and adolescent wellbeing: A systematic narrative review. *Children and Youth Services Review*, 41:27–36, 2014.

[2] Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH)*, 2021.

[3] Eshwar Chandrasekharan, Umashanthi Pavalanathan, Anirudh Srinivasan, Adam Glynn, Jacob Eisenstein, and Eric Gilbert. You can't stay here: The efficacy of reddit's 2015 ban examined through hate speech. *Proc. ACM Hum.-Comput. Interact.*, 1(CSCW), December 2017.

[4] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 782–791, 2021.

[5] Kasia S. Chmielinski, Sarah Newman, Matt Taylor, Josh Joseph, Kemi Thomas, Jessica Yurkofsky, and Yue Chelsea Qiu. The dataset nutrition label (2nd gen): Leveraging context to mitigate harms in artificial intelligence. *arXiv preprint arXiv:2201.03954*, 2022.

[6] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences (PNAS)*, 118(9), 2021.

[7] Enyan Dai and Suhang Wang. Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 680–688, 2021.

[8] Munmun De Choudhury, Michael Gamon, Scott Counts, and Eric Horvitz. Predicting depression via social media. In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, 2013.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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 Computational Linguistics (NAACL)*, 2019.

[10] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[11] Klaus Greff, Rupesh K. Srivastava, Jan Koutnik, Bas R. Steunebrink, and Jurgen Schmidhuber. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232, October 2017.

[12] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 855–864, New York, NY, USA, 2016. Association for Computing Machinery.

[13] Vivek Gupta, Ankit Saw, Pegah Nokhiz, Praneeth Netrapalli, Piyush Rai, and Partha Talukdar. P-sif: Document embeddings using partition averaging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[14] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[15] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, pages 1024–1034, 2017.

[16] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. *arXiv preprint arXiv:1708.05031*, 2017.

[17] Liam Hebert, Hong Yi Chen, Robin Cohen, and Lukasz Golab. Qualitative analysis of a graph transformer approach to addressing hate speech: Adapting to dynamically changing content. *arXiv preprint arXiv:2301.10871*, 2023.

[18] Liam Hebert, Gaurav Sahu, Yuxuan Guo, Nanda Kishore Sreenivas, Lukasz Golab, and Robin Cohen. Multi-modal discussion transformer: integrating text, images and graph transformers to detect hate speech on social media. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024.

[19] Iris Hendrickx et al. Ruddit: Norms of offensiveness for English reddit comments. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, 2021.

[20] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

[21] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. Heterogeneous graph transformer. In *Proceedings of The Web Conference 2020*, 2020.

[22] Shaoxiong Ji, Tianlin Zhang, Luna Ansari, Jie Fu, Prayag Tiwari, and Erik Cambria. MentalBERT: Publicly available pretrained language models for mental healthcare. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC)*, pages 7184–7190, Marseille, France, 2022. European Language Resources Association.

[23] John Jumper et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[24] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W Reeder. Standardizing privacy notices: an online study of the nutrition label paradigm. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1573–1582, 2009.

[25] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[26] Adam D I Kramer, Jamie E Guillory, and Jeffrey T Hancock. Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences*, 111(24):8788–8790, 2014.

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[28] Aditya Kusupati, G Bhatt, A Rege, M Wallingford, A Sinha, V Ramanujan, et al. Matryoshka representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[29] Zachary C. Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*, 2015.

[30] Ruoxue Ma, Jiarong Xu, Xinnong Zhang, Haozhe Zhang, Zuyu Zhao, Qi Zhang, Xuanjing Huang, and Zhongyu Wei. One-model-connects-all: A unified graph pre-training model for online community modeling. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15034–15045, Singapore, December 2023. Association for Computational Linguistics.

[31] Trevor Martin. community2vec: Vector representations of online communities encode semantic relationships. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, 2017.

[32] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, volume 26, pages 3111–3119, 2013.

[33] Niklas Muennighoff, Nelson Tiong, Marone Marone, Ryumei Mou, Beidi Jacobsen, and Haque He. Mteb: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023.

[34] OpenAI. GPT-4 technical report, 2023.

[35] Chan Young Park et al. Detecting community sensitive norm violations in online conversations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

[36] Brian A Primack, Ariel Shensa, Jaime E Sidani, Erin O Whaite, Liu yi Lin, Daniel Rosen, Jason B Colditz, Ana Radovic, and Elizabeth Miller. Social media use and perceived social isolation among young adults in the us. *American Journal of Preventive Medicine*, 53(1):1–8, 2017.

[37] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.

[38] Victoria Rideout and Susannah Fox. Digital health practices, social media use, and mental well-being among teens and young adults in the u.s. Technical report, Hopelab and Well Being Trust, 2018.

[39] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637*, 2020.

[40] Constance A Steinkuehler and Dmitri Williams. Where everybody knows your name: Online games as third places. *Journal of Computer-Mediated Communication*, 11(4):885–909, 2006.

[41] Elsbeth Turcan and Kathy McKeown. Dreaddit: A Reddit dataset for stress analysis in social media. In Eben Holderness, Antonio Jimeno Yepes, Alberto Lavelli, Anne-Lyse Minard, James Pustejovsky, and Fabio Rinaldi, editors, *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, pages 97–107, Hong Kong, November 2019. Association for Computational Linguistics.

[42] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2019.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2018.

[45] Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, et al. Embeddinggemma: Powerful and lightweight text representations. *arXiv preprint arXiv:2509.20354*, 2025.

[46] Isaac Waller and Ashton Anderson. Quantifying social organization and political polarization in online platforms. *Nature*, 600(7888):264–268, 2021.

[47] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. MM-GCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1437–1445, 2019.

[48] Tianup Wu, Yuxiao Wang, Yixuan Wang, et al. Leveraging graph-based hierarchical medical entity embedding for healthcare applications. *Scientific Reports*, 11(1):5858, 2021.

[49] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[50] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Chen, and Tie-Yan Liu. Do transformers really perform bad for graph representation? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[51] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

[52] He Yuan and Alexander A. Hernandez. User cold start problem in recommendation systems: A systematic review. *IEEE Access*, 11:136958–136977, 2023.

[53] Justine Zhang, Jonathan P. Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Nithum Thain, and Dario Taraborelli. Conversations gone awry: Detecting early signs of conversational failure. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1350–1361. Association for Computational Linguistics, 2018.