University of Padova

Department of Mathematics "Tullio Levi-Civita"

Master Thesis in Data Science

Expanding Latent Ideology Estimation via Machine Learning for User Classification in Online Social Media

Supervisor
Prof. Alessandro Galeazzi
University of Padova

Co-supervisor
Prof. Giovanni Da San Martino
University of Padova

Master Candidate
Susla Diana

Academic Year
2025-2026

Abstract

Social media platforms have become central arenas for political discourse, yet understanding the ideological positions of users remains challenging. This thesis presents a comprehensive study of ideology inference from Twitter data using advanced machine learning techniques. We developed and compared 10 different models across three categories: text-based models, graph-based models, and hybrid models that combine both modalities.

Our approach extends traditional ideology estimation methods by incorporating network structure through enhanced graph features, including centrality measures, community detection, and ideology-specific metrics.

The thesis contributes: (1) comprehensive comparison of deep learning architectures for ideology inference; (2) novel graph feature engineering for social networks; (3) interpretable attention mechanisms for understanding model predictions; (4) scalable implementation for graphs with millions of nodes. Our findings have implications for understanding political polarization, echo chambers, and information diffusion in social networks.

Results show that the graph-based GNN models, particularly GAT, achieve the strongest performance, confirming that structural position in the retweet network is a stronger signal of ideology than tweet text alone. Hybrid fusion models consistently outperform single-modality baselines, demonstrating that text and graph signals provide complementary information: network structure captures ideological sorting through homophilic connections (who users interact with), while text reveals specific policy positions and argumentative framing (what users express). Combining these modalities is particularly valuable when one signal is weak or unavailable.

Contents

Abstract v

List of figures xii

List of tables xv

Listing of acronyms xvii

1 Introduction 1
1.1 Motivation . . . 1
1.2 Research Questions . . . 2
1.3 Contributions . . . 3
1.4 Thesis Structure . . . 4

2 Literature Review 7
2.1 Political Polarization in Social Media . . . 7
2.1.1 Early Studies of Online Political Networks . . . 8
2.1.2 Twitter as a Platform for Political Discourse . . . 8
2.1.3 Echo Chambers and Filter Bubbles . . . 9
2.1.4 Affective Polarization and Animosity . . . 10
2.1.5 Platform Algorithms and Polarization . . . 11
2.1.6 International Perspectives and Cross-Platform Comparisons . . . 11
2.1.7 Consequences of Online Polarization . . . 12
2.1.8 Implications for This Work . . . 12
2.2 Network-Based Approaches to Ideology Detection . . . 13
2.3 Text-Based Methods for Ideology Classification . . . 14
2.4 Graph Representation Learning . . . 15
2.5 Graph Neural Networks . . . 16
2.6 Multimodal and Hybrid Approaches . . . 17
2.7 Research Contributions . . . 19

3 Data and Experimental Setup 21
3.1 Source Dataset and Data Collection . . . 21
3.1.1 Data Collection Methodology . . . 21
3.1.2 Dataset Composition and Scale . . . 22

3.1.3 Ground Truth Ideology Labels: Latent Ideology Estimation 23
3.1.4 Dataset Characteristics 26
3.1.5 Dataset Scope and Limitations 27
3.2 Data Preprocessing 28
3.2.1 Text-based Features Construction 28
3.2.2 Enhanced Text Feature Extraction 28
3.2.3 Retweet Network Construction 29
3.3 Feature Engineering: Graph Features 29
3.3.1 Graph Features Statistics 34
3.4 Train/Test Split and Evaluation Protocol 40
3.4.1 Evaluation Metrics 41

4 Models and Methods **43**
4.1 Text-Based Models 43
4.1.1 TF-IDF + Ridge Regression 43
4.1.2 BERT Embeddings Extraction 44
4.1.3 BERT + MLP 44
4.1.4 BERT + LSTM 47
4.1.5 BERT + CNN 49
4.2 Graph-Based Models 50
4.2.1 XGBoost on Graph Features 50
4.2.2 GCN (Graph Convolutional Network) 50
4.2.3 GraphSAGE 53
4.3 Hybrid Fusion Models 55
4.3.1 Simple Concatenation Fusion 55
4.3.2 Attention-Based Fusion 57
4.3.3 Gated Fusion 59
4.4 Training Infrastructure 61

5 Results and Analysis **63**
5.1 Overall Performance Comparison 63
5.2 Text-Based Model Results 66
5.2.1 Model Performance 66
5.3 Graph-Based Model Results 68
5.3.1 Model Performance 68
5.3.2 Feature Importance Analysis 70
5.3.3 Comparison with Text-Based Models 76
5.4 Hybrid Model Results 77
5.4.1 XGBoost on Concatenated Features: Best Overall Performance . . . 77
5.4.2 Neural Fusion Strategies 78
5.4.3 Traditional ML vs. Neural Fusion: A Comparison 80

6 Conclusion **83**
6.1 Key Findings 83
6.2 Scenario-Based Recommendations 84
6.3 Practical Considerations 88
6.4 Limitations and Future Work 89
6.5 Final Remarks 89

References **91**

Acknowledgments **97**

Listing of figures

3.1 Ideology score distribution 25
3.2 Graph feature distributions 36
3.3 Graph feature distributions by ideology 38
3.4 Ideology and neighbor ideology correlation 40

4.1 BERT-MLP residual plot 45
4.2 BERT-MLP training curves 45
4.3 BERT fine-tuned MLP residual plot 46
4.4 BERT fine-tuned MLP training curves 47
4.5 BERT-LSTM residual plot 48
4.6 BERT-LSTM training curves 48
4.7 BERT-CNN residual plot 49
4.8 BERT-CNN training curves 50
4.9 GCN residual plot 52
4.10 GCN training curves 52
4.11 GraphSAGE residual plot 54
4.12 GraphSAGE training curves 54
4.13 Simple fusion residual plot 56
4.14 Simple fusion training curves 56
4.15 Attention fusion residual plot 58
4.16 Attention fusion training curves 58
4.17 Gated fusion residual plot 60
4.18 Gated fusion training curves 60

5.1 XGBoost feature importance rankings 71
5.2 SHAP feature importance (bar plot) 73
5.3 SHAP feature impact (beeswarm plot) 74

Listing of tables

3.1 Dataset summary statistics from Baqir et al. (2025) 23
3.2 Retweet network statistics 29
3.3 Graph-based features for ideology inference 33
3.4 Descriptive statistics of graph features 35

5.1 Comprehensive model performance comparison 64
5.2 Text-based model performance comparison 66
5.3 Graph-based model performance comparison 68
5.4 Hybrid multimodal model performance 77
5.5 Traditional ML vs. neural fusion comparison 81

Listing of acronyms

API . . . . . Application Programming Interface

BERT . . . . . Bidirectional Encoder Representations from Transformers

CLS . . . . . Classification Token (special BERT token)

CNN . . . . . Convolutional Neural Network

DistilBERT . . . . Distilled Bidirectional Encoder Representations from Transformers

GAT . . . . . Graph Attention Network

GCN . . . . . Graph Convolutional Network

GNN . . . . . Graph Neural Network

GPU . . . . . Graphics Processing Unit

LSTM . . . . . Long Short-Term Memory

MAE . . . . . Mean Absolute Error

MLP . . . . . Multi-Layer Perceptron

MSE . . . . . Mean Squared Error

NLP . . . . . Natural Language Processing

PyG . . . . . PyTorch Geometric

ReLU . . . . . Rectified Linear Unit

RMSE . . . . . Root Mean Squared Error

RNN . . . . . Recurrent Neural Network

SHAP . . . . . SHapley Additive exPlanations

TF-IDF . . . . . Term Frequency - Inverse Document Frequency

XGBoost . . . . . Extreme Gradient Boosting

1 Introduction

1.1 Motivation

Social media platforms have transformed the way people interact and consume information. Twitter, in particular, has become an important place where politicians, journalists, and regular citizens discuss political issues. Understanding political opinions on social media is important because it helps us study how people form their views and how information spreads online and also because politics and political choices drives society.

Traditional methods for studying political ideology, like surveys or voting records, have limitations. Surveys are expensive and time-consuming, making it hard to study large groups of people. As well as they are subject to self-selection bias. Voting records only show us information about politicians, not regular citizens. In contrast, social media provides access to huge amounts of behavioral data that can reveal people's political views. However, inferring someone's ideology from their social media activity is challenging. A retweet might mean agreement, or it could be sharing something to criticize it. User-generated text is often sarcastic, uses informal language, and may discuss topics with no clear political meaning. The large volume of data - millions of users producing billions of messages - requires computational methods that can handle this scale.

This thesis develops and evaluates machine learning approaches to **infer latent ideology** from Twitter data. Specifically, we examine whether different types of information - text con-

tent, network structure, and combinations of both - can be used to infer users' political orientations. This is not just a technical problem; it helps us understand fundamental questions about political behavior online: How much people's social connections reveal their political identities? Can we detect ideology from how people write, not just what they write about? How do different information sources work together?

There is growing concern that social media may increase political polarization. Unlike traditional media, which exposed everyone to similar news, social media allows users to choose their information sources. This personalization, combined with algorithms that show users content they are likely to engage with, may create "echo chambers" where people only see viewpoints they already agree with. If different groups of people consume completely different information, it becomes harder to have productive political discussions.

Our approach combines three different methods that have usually been studied separately. Text-based methods from natural language processing treat ideology detection as a classification problem, learning patterns in vocabulary and writing style that correlate with political orientation. Network-based methods use the principle of homophily - the idea that similar people tend to connect with each other - to infer users' positions from their social connections. Graph neural networks provide a way to combine these approaches by learning from both text content and network structure simultaneously.

The integration of these approaches is motivated by both theory and practice. Theoretically, political ideology manifests in multiple ways: through what people say (captured by text), through who they connect with (captured by network structure), and through patterns of behavior. Empirically, we show that different types of features capture different information, so models combining text and network information perform much better than models using only one type.

This work also contributes to understanding political polarization in social media. By measuring how predictive different network features are - especially measures of ideological homophily and echo chamber structure - we provide evidence about how much Twitter networks are ideologically segregated. The finding that a user's average neighbor ideology is the most predictive feature, accounting for 89% of the model's predictive power (feature importance), shows that network position provides a very strong signal about latent political orientation.

1.2 Research Questions

This thesis addresses the following research questions:

1. **Text-based inference:** How well can latent political ideology be inferred from the content users produce on Twitter? How do different neural architectures (BERT, LSTM, CNN) perform?

2. **Network-based inference:** How well can ideology be inferred from structural positions in the retweet network? Which graph features are most informative?

3. **Graph neural networks:** How do Graph Neural Networks (GCN, GraphSAGE), which jointly learn from node features and network structure, compare to traditional feature-engineered methods for ideology inference?

4. **Combining information:** How much does combining text and network information improve inference compared to using just one type?

5. **Leveraging political homophily:** How does incorporating ideological homophily and echo chamber features improve ideology inference, particularly in data-scarce scenarios with limited labeled users?

1.3 Contributions

The main contributions of this work are:

- **Comprehensive methodology comparison:** This work provides a systematic evaluation of text-based, network-based, and hybrid approaches to ideology inference using modern deep learning methods. The comparison includes traditional machine learning baselines (TF-IDF + Ridge, XGBoost) and state-of-the-art neural architectures (BERT variants, GNNs), establishing performance benchmarks across different information modalities.

- **Enhanced multimodal feature extraction:** This work delivers an enhanced text feature extraction method that integrates multiple textual sources (tweets, user profiles, locations) using semantic tagging, improving text-only model performance. Additionally, we provide a scalable framework for computing twelve structural and homophily-based graph features from large-scale retweet networks. Following standard practice in homophily analysis, ideology-based features are computed on the undirected version of the network to capture bidirectional influence patterns.

- **Feature importance analysis for data-scarce scenarios:** This work provides a detailed analysis of feature importance, revealing that average neighbor ideology is the dominant

feature, accounting for 89% of the model's predictive power (measured by XGBoost feature importance), with ideology homophily contributing 3.4% and echo chamber score 1%. This analysis is particularly valuable for data-scarce scenarios where labeled data is limited (only 3% of users in our dataset are labeled), as it demonstrates that graph-based features enable semi-supervised learning by propagating ideology information through network connections.

- **Graph Neural Network implementation and evaluation:** This work delivers implementations of GCN and GraphSAGE architectures adapted for ideology inference, demonstrating that graph neural networks achieve comparable performance to feature-engineered approaches while offering inductive learning capabilities for new users joining the network.
- **Empirical evidence on multimodal fusion:** This work provides empirical findings on combining text and network features through multiple fusion strategies (concatenation, attention-based, gated), quantifying the complementary nature of different information sources.

1.4 Thesis Structure

This work is organized as follows:

Chapter 2 reviews related work on ideology measurement, political polarization in social media, and machine learning applications in political science. We discuss how research has evolved from early network analysis to modern deep learning approaches.

Chapter 3 describes the dataset, including how it was collected, preprocessing steps, and how ideology labels were created. We provide detailed statistics about the Twitter retweet network and explain the correspondence analysis method used to generate ideology scores. We also discuss why this dataset is particularly suitable for studying ideology inference in data-scarce scenarios.

Chapter 4 presents the methodology. We describe text-based models (BERT with various neural architectures), graph-based models (traditional machine learning on graph features and graph neural networks - GCN, GraphSAGE) and hybrid models. We explain implementation details and training procedures.

Chapter 5 presents experimental results, including performance comparisons across all models. We analyze which features are most important for inference, discuss advantages and disadvantages of different approaches, and connect findings back to the research questions. Special

attention is given to how models perform in data-scarce scenarios with limited labeled examples.

Chapter 6 concludes with scenario-based recommendations for practitioners, discussing which approaches to use in different situations (e.g., when network data is available vs. unavailable, when labeled data is scarce, when interpretability is required). We also discuss limitations and directions for future research.

2

Literature Review

Understanding political ideology through social media data has become increasingly important as these platforms play a central role in modern political discourse. This chapter reviews existing research on ideology detection, tracing the evolution from early network analysis to modern deep learning approaches. We organize the discussion around key methodological developments: network-based methods that leverage social connections, text-based approaches that analyze linguistic content, graph representation learning techniques, and recent advances in graph neural networks. Throughout this review, we highlight how these different strands of research inform and motivate our own multimodal approach.

2.1 Political Polarization in Social Media

The study of political ideology in online spaces builds on a foundational observation from sociology: people tend to associate with others who are similar to them. This principle, known as homophily, was systematically documented by [1] in their comprehensive review of social network research. Their work established that similarity breeds connection across numerous dimensions - race, education, age, and crucially for our purposes, political beliefs. This insight provides the theoretical foundation for why network structure can reveal ideological positions: if people preferentially connect with like-minded others, then analyzing who connects with whom should tell us something about their political orientations.

2.1.1 Early Studies of Online Political Networks

Early empirical work confirmed that this pattern holds in online political discussions. Adamicраздел a and Glance [2] analyzed the political blogosphere during the 2004 U.S. presidential election and found remarkably clear partisan divisions. Liberal and conservative blogs formed distinct clusters with relatively few cross-ideology links. When connections did cross ideological boundaries, they were often framed as criticism rather than genuine engagement. This work was important not just for its findings, but also for establishing methods, community detection, network visualization, that would become standard tools in the field.

Building on this foundation, Hargittai et al. [3] examined whether ideological segregation online reflected deliberate self-selection or structural features of the web. Their analysis of blog citation patterns found that cross-ideological citations did occur but were systematically different in nature from within-ideology citations: they tended to be shorter, less substantive, and more likely to be critical. This suggested that exposure to opposing views does not necessarily translate into genuine engagement or persuasion.

2.1.2 Twitter as a Platform for Political Discourse

The emergence of Twitter as a major platform for political discourse provided new opportunities to study these dynamics at scale. Unlike blogs, which required technical expertise to maintain, Twitter lowered barriers to political expression and created a more diverse and representative sample of the politically engaged public. Conover et al. [4] examined political polarization during the 2010 U.S. midterm elections and found striking patterns in how information flows through the network. While users from different ideological camps did mention each other occasionally (often in arguments), retweets, a stronger signal of endorsement, occurred almost exclusively within ideological boundaries. Over 90% of retweets stayed within the same political community. This suggested that while awareness of opposing views might exist, actual information sharing and trust networks are heavily segregated.

Subsequent work has consistently documented ideological homophily and network polarization across various political contexts. Himelboim et al. [5] analyzed Twitter discussions during the 2012 U.S. presidential debates and found that despite the shared event creating a common conversational space, interactions remained largely within partisan clusters. Yardi and Boyd [6] examined abortion debates on Twitter and found similar patterns of polarization, with cross-ideological interactions being rare and typically contentious when they did occur.

The prevalence of these patterns led researchers to investigate their causes and consequences.

Barberá and Rivero [7] used Twitter data from six different countries to examine whether polarization was a universal feature of political social media or specific to particular political cultures. Interestingly, they found substantial variation: U.S. Twitter networks exhibited extreme polarization, while German and Spanish networks showed more cross-ideological connections. This suggests that while homophily is a general tendency, its strength depends on the broader political context, including party system structure and cultural norms around political disagreement.

2.1.3 Echo Chambers and Filter Bubbles

The observation that online political networks exhibit strong ideological clustering raised important questions about "echo chambers" and "filter bubbles." Sunstein [8] introduced the echo chamber concept, arguing that the internet enables people to insulate themselves from disagreement, potentially leading to more extreme views and decreased understanding of opposing perspectives. Pariser [9] popularized the related concept of filter bubbles, focusing on the role of algorithmic personalization: as platforms learn users' preferences and show them more of what they like, users may become trapped in informational bubbles where they encounter primarily viewpoints they already agree with.

These concerns gained empirical attention as researchers attempted to quantify the extent of ideological isolation in social media. Bakshy et al. [10] addressed this question with a large-scale study of Facebook users' news consumption, analyzing 10.1 million people's content exposure and engagement. Their analysis revealed a nuanced picture: while algorithms do play a role in shaping exposure to diverse viewpoints, individual choices matter more. Users predominantly select content aligned with their existing views even when exposed to diverse options. This suggests that homophily in online networks reflects genuine preferences rather than just algorithmic sorting.

Flaxman et al. [11] extended this line of inquiry by comparing online and offline news consumption patterns. Contrary to the filter bubble hypothesis, they found that internet news readers consume more diverse sources and encounter more ideological variety than those who rely primarily on traditional media. However, this overall diversity is unevenly distributed: the most politically engaged online users actually exhibit more polarized consumption patterns than moderate users, suggesting that the internet may simultaneously broaden exposure for casual news consumers while deepening divisions among the politically engaged.

More recent work has attempted to quantify the strength of these echo chambers. Boutyline and Willer [12] developed metrics to measure ideological homophily in Twitter networks, go-

ing beyond simple binary divisions to capture the degree of ideological segregation. They found substantial variation across different topics and communities - some discussions show extreme polarization while others maintain more balanced networks. Their quantitative approach provides tools that we adapt in our own feature engineering, particularly the echo chamber score that measures what proportion of a user's neighbors share similar ideological positions.

Dubois and Blank [13] challenged the prevailing narrative by conducting interviews with social media users about their information exposure. They found that most users were aware of echo chambers and actively took steps to expose themselves to diverse viewpoints, such as following accounts from across the political spectrum. However, mere exposure to diverse content does not guarantee engagement: users might follow opposing voices but pay little attention to their content, or engage only to argue rather than to learn. This highlights the distinction between network-level exposure (what appears in users' feeds) and individual-level attention (what users actually read and think about).

2.1.4 Affective Polarization and Animosity

Beyond simple ideological disagreement, recent research has focused on affective polarization - the extent to which partisans dislike and distrust those on the other side. Iyengar et al. [14] documented rising affective polarization in the United States, showing that party identification has become a stronger source of social division than race, religion, or other demographic factors. Social media may amplify affective polarization by facilitating outrage and moral grandstanding. Brady et al. [15] found that messages expressing moral-emotional language spread more widely on Twitter, potentially rewarding extreme expressions of partisan identity.

Settle [16] examined how Facebook use relates to political polarization. Using survey and digital trace data, she found that social media use increases affective polarization by exposing users to political content from weak ties and distant acquaintances, which is more likely to provoke partisan reactions than content from close friends. The very features that make social media powerful for information diffusion - broad networks, algorithmic amplification, easy sharing - may also make it effective at spreading divisive partisan content.

Bail et al. [17] conducted an experiment where Twitter users were paid to follow accounts from the opposing political party. Contrary to the hope that exposure to opposing views might reduce polarization, they found that Republicans who followed liberal accounts became more conservative, while Democrats showed no significant change. This suggests that cross-cutting exposure may sometimes backfire by reinforcing existing beliefs rather than moderating them.

2.1.5 Platform Algorithms and Polarization

The role of platform algorithms in shaping political polarization has been a subject of intense debate. Gillespie [18] examined how algorithmic curation shapes what users see, arguing that algorithms are not neutral technical systems but encode particular values and priorities - typically engagement maximization. If controversial, partisan content generates more engagement (likes, shares, comments), algorithmic recommendation systems may preferentially surface such content, potentially exacerbating polarization.

Bozdag [19] analyzed how algorithmic personalization on social media platforms creates "algorithmic filtering," where content is selected based on predicted user preferences rather than journalistic criteria of importance or balance. This can lead to preference bubbles even for users who do not consciously seek ideologically congruent content. However, Haim et al. [20] found that while filter bubbles exist for some users, many maintain relatively diverse news diets, suggesting that individual agency and media literacy play important roles.

Recent work has focused on understanding the precise mechanisms through which algorithms influence polarization. Levy [21] built a theoretical model showing that even small algorithmic biases toward showing users content similar to what they have engaged with previously can, over time, lead to significant segregation. Huszár et al. [22] analyzed Twitter's algorithm and found that it amplifies political content from elected officials, particularly those on the right, more than content from ordinary users. These findings suggest that platform design choices have meaningful impacts on the information landscape, even if users retain agency in what they choose to consume.

2.1.6 International Perspectives and Cross-Platform Comparisons

While much research has focused on U.S. political polarization on Twitter and Facebook, international studies reveal important contextual variation. Barberá [23] analyzed Twitter networks across multiple countries and found that polarization patterns vary substantially depending on political system characteristics. Countries with proportional representation and multiparty systems tend to show less binary polarization than those with two-party systems like the United States.

Hong and Nadler [24] examined Chinese social media (Weibo) and found that while censorship limits overtly political content, polarization still emerges around issues that slip through

censorship filters. This suggests that ideological clustering is a robust phenomenon not dependent solely on Western-style party competition. Similarly, Del Vicario et al. [25] studied Italian Facebook users and found echo chamber effects comparable to those in the United States, indicating that these patterns transcend national boundaries.

Cross-platform comparisons reveal how affordances shape polarization dynamics. Cinelli et al. [26] compared echo chambers across Twitter, Facebook, Reddit, and Gab. They found that all platforms exhibited ideological clustering, but the strength and structure varied: Twitter showed the clearest partisan divisions, while Reddit's subreddit structure created topic-based rather than purely ideological echo chambers. This suggests that platform architecture - whether content is organized by social networks (Twitter, Facebook) or by topic communities (Reddit)-influences the nature of polarization.

2.1.7 Consequences of Online Polarization

The documented patterns of ideological segregation online raise questions about consequences for democracy and public discourse. Prior [27] argued that increasing media choice and personalization have contributed to knowledge gaps: politically interested individuals consume more political information than ever, while the uninterested can easily avoid politics altogether. This creates an increasingly divided electorate in terms of political knowledge and engagement.

Tucker et al. [28] reviewed the evidence on social media and political polarization, concluding that while platforms have likely contributed to affective polarization, the evidence for ideological polarization (actual policy disagreement) is mixed. They note that polarization has been increasing primarily among demographics that use social media least (older adults), suggesting that broader societal trends beyond just social media use drive polarization.

More optimistically, some research suggests that online networks may provide opportunities for bridging divides. Granovetter's [29] classic work on weak ties suggests that cross-cutting connections, even if infrequent, can facilitate information flow between otherwise disconnected groups. Barnidge [30] found that while social media use correlates with exposure to like-minded content, it also increases exposure to political disagreement for some users. The question is whether this exposure leads to productive engagement or simply reinforces divisions.

2.1.8 Implications for This Work

The empirical evidence is clear: social media exhibits strong ideological clustering and evidence of echo chamber effects. What remains debated is whether this represents an unhealthy narrow-

ing of discourse or simply a digital reflection of existing social divisions, and what role platform design plays in these patterns. For our purposes, the key takeaway is that network structure carries a strong signal about political orientation, making it a valuable source of information for ideology inference.

The documented strength of political homophily provides the theoretical foundation for our network-based approaches: if users preferentially connect with ideologically similar others, then a user's position in the network - who they retweet, who retweets them, the ideological composition of their neighborhood - should be highly predictive of their own ideology. The finding that over 90% of retweets stay within ideological boundaries [4] suggests that retweet networks are particularly strong signals of ideological affiliation.

2.2 Network-Based Approaches to Ideology Detection

Given the strong association between network position and political orientation, researchers have developed various methods to infer ideology from social network structure. These approaches operate on the premise that examining who someone connects with can reveal their political beliefs, even in the absence of explicit political content.

One influential framework comes from Barberá [23], who introduced a Bayesian approach to estimating ideological positions from Twitter following patterns. The key insight is to treat follows as endorsements: when a user follows political accounts, they signal agreement with those positions. Using Item Response Theory (originally developed for educational testing), the method positions both users and political figures on a common ideological spectrum. The model learns these positions such that users are more likely to follow accounts close to them ideologically. While computationally intensive, this approach has become influential in political science applications and demonstrates the power of network-based inference.

Beyond follower networks, various graph-theoretic features can capture ideologically relevant structural positions. Community detection methods, particularly modularity optimization [31], identify cohesive groups within networks - clusters that in political networks often correspond to ideological camps. Centrality measures tell us about different forms of influence: PageRank identifies broadly influential users, betweenness centrality highlights bridge positions between communities, and degree centrality captures basic connectivity. Tang et al. [32] showed how these structural features correlate with user attributes in social networks,

providing a methodological foundation for using graph features in prediction tasks.

Our work builds directly on this tradition but extends it in important ways. Beyond standard centrality measures, we compute features that explicitly capture ideological patterns. The most powerful of these is average neighbor ideology: for users with known political positions, we calculate the mean ideology of their network neighbors. This feature captures the homophily pattern directly and, as we will see, proves to be remarkably predictive - accounting for over 40% of feature importance in our models. We also implement the echo chamber score concept from Boutyline and Willer [12], measuring what proportion of a user's neighbors share the same ideological sign (left vs. right).

An important limitation of purely network-based approaches is their reliance on existing labels. Methods like those of Barberá [23] require known political figures to anchor the ideological space, while our neighbor-based features need training labels to be computed. This creates a semi-supervised setting where we leverage both labeled and unlabeled data through the network structure. The success of these methods depends on homophily being strong enough that network position reliably indicates ideology - an assumption that empirical work consistently validates.

2.3 Text-Based Methods for Ideology Classification

While network structure provides powerful signals, the content that users share should also reveal their political orientations - political ideology also represents in language use, topic selection, and framing choices. Text-based approaches attempt to learn these patterns, treating ideology detection as a classification or regression problem on linguistic features.

Early neural network approaches showed promise for this task. Iyyer et al. [33] applied recursive neural networks to political speeches, achieving around 70% accuracy in classifying congressional statements as liberal or conservative. Their model learned to identify ideological patterns in syntax and word choice, demonstrating that neural networks could capture linguistic patterns of political orientation. However, this work focused on formal political speech rather than social media text, where challenges like brevity, informality, and noise are more pronounced.

Extending text-based analysis to Twitter proved more difficult. Cohen and Ruths [34] documented these challenges systematically, showing that simple approaches yield inconsistent results. Tweets are short, often lack clear political content, employ sarcasm and irony, and may discuss non-political topics entirely. These characteristics make purely textual analysis chal-

lenging and motivate the integration of network information.

Despite these difficulties, more sophisticated approaches have made progress. Preoțiuc-Pietro et al. [35] moved beyond binary classification to predict a seven-point scale scores from Twitter text. Their regression approach achieved moderate performance (correlation between the set of predicted values and the user-reported score ≈ 0.369), identifying linguistic markers associated with different points along the liberal-conservative spectrum. This work established useful baselines and demonstrated the value of treating ideology as not binary. In our work's approach we will use continuous political scores rather than binary ones.

The major breakthrough in text processing came with transformer-based language models. BERT (Bidirectional Encoder Representations from Transformers), introduced by Devlin et al. [36], made revolution in natural language processing through pre-training on massive text corpora. Unlike earlier approaches that learned task-specific representations from scratch, BERT learns general linguistic patterns that transfer to downstream tasks. The model uses bidirectional attention mechanisms to build contextual representations, where each word's meaning depends on both preceding and following context.

For ideology detection, BERT offers several advantages. First, pre-training on large corpora provides broad linguistic knowledge before fine-tuning on specific tasks. Second, the attention mechanism can identify which words or phrases are most relevant for ideological classification. Third, contextual representations handle nuanced language use better than bag-of-words or simple word embedding approaches. We leverage BERT in our text modality by fine-tuning it for ideology prediction, allowing the model to adapt general language understanding to political content.

However, text-based approaches face inherent limitations. Users may not explicitly discuss politics in their tweets, political content may be ambiguous or ironic, and some users curate their public statements carefully. These challenges suggest that combining textual analysis with network structure, where connections themselves signal political alignment, could give us more robust predictions than either modality alone.

2.4 Graph Representation Learning

Traditional graph features like centrality measures capture specific structural properties, but recent work has explored learning more comprehensive node representations automatically. Graph representation learning, or network embedding, aims to encode nodes as low-dimensional vectors that preserve network structure. These learned representations can then feed into down-

stream machine learning tasks.

The foundational work in this area came from Perozzi et al. [37], who made a clever connection between social networks and natural language. Just as Word2Vec [38] learns word embeddings by predicting context words in sentences, DeepWalk learns node embeddings by predicting context nodes in random walks. The algorithm performs random walks on the graph, generating sequences of nodes, then applies Word2Vec to these sequences. The resulting embeddings capture structural proximity: nodes that appear in similar network contexts receive similar representations.

While DeepWalk uses uniform random walks, Grover and Leskovec [39] introduced Node2Vec, which extends this approach with biased random walks controlled by parameters p and q that interpolate between breadth-first and depth-first exploration. For social networks with strong homophily, Node2Vec can emphasize local neighborhoods to capture tight-knit ideological communities. However, preliminary experiments indicated that Node2Vec's computational cost outweighed its marginal performance gains (2-4% improvement in R^2) over explicit graph features, leading us to focus on faster alternatives.

Alternative approaches include LINE [40], which optimizes explicit proximity measures, and Struc2Vec [41], which focuses on structural similarity. These methods demonstrate the diversity of approaches to graph representation learning, each with different trade-offs between computational cost and performance.

2.5 Graph Neural Networks

Graph neural networks represent a significant departure from the methods discussed so far. Rather than computing fixed graph features or learning embeddings in a separate step, GNNs perform end-to-end learning directly on graph-structured data. They learn how to aggregate information from network neighborhoods, adapting the aggregation to the specific prediction task.

The important work came from Kipf and Welling [42], who introduced Graph Convolutional Networks (GCN). The core idea is: update each node's representation by aggregating features from its neighbors, then apply a neural network transformation. This process repeats for multiple layers, allowing information to propagate through the network.

This architecture is particularly well-suited for semi-supervised learning in networks with strong homophily. If labeled nodes tend to have similar labels to their neighbors (which political networks do), then aggregating neighbor information helps classify unlabeled nodes.

Each GCN layer effectively smooths representations across edges, so that connected nodes end up with similar representations. For ideology prediction, this means that users connected to known liberals will develop representations similar to other liberals, even if their own label is unknown.

Scalability concerns led Hamilton et al. [43] to develop GraphSAGE (SAmple and aggreGatE). Instead of using all neighbors, GraphSAGE samples a fixed number from each node's neighborhood. This makes training feasible on large graphs and enables inductive learning - generating embeddings for previously unseen nodes. For our application with millions of Twitter users, this scalability is important.

More sophisticated GNN architectures have been proposed, including Graph Attention Networks (GAT) [44], which learn attention weights to determine neighbor importance rather than treating all neighbors equally. However, for large-scale networks like ours (2.6 million users, 4.3 million edges), memory constraints become significant. GAT's attention mechanism requires storing attention coefficients for all edges, substantially increasing memory requirements compared to GCN's fixed aggregation scheme. Given these practical constraints and the strong baseline performance of GCN and GraphSAGE on our homophilic political network, we focus our empirical evaluation on these more memory-efficient architectures.

One consideration with GNNs is their inductive bias toward homophily. If the network exhibits homophily, GNNs excel by smoothing labels across edges. This aligns perfectly with political networks where ideological similarity drives connection formation. However, this same bias means GNNs can struggle with heterophilic networks where connected nodes tend to be dissimilar. Fortunately, political homophily is empirically well-established, making GNNs a natural fit.

2.6 Multimodal and Hybrid Approaches

While network and text modalities each provide valuable signals, they capture different aspects of political behavior. Network connections reveal who users trust and align with, while text content shows what they say and how they frame issues. Intuitively, combining these complementary sources should improve prediction accuracy beyond what either modality achieves alone.

Early work on multimodal user modeling demonstrated this potential. Benton et al. [45] developed multiview embedding methods that jointly learn from textual content and network structure. Their approach finds shared latent spaces where representations from different views

of the user (what they write, who they connect with) can be meaningfully compared and combined. This work established that different modalities capture genuinely different information rather than simply redundant signals.

More recently, researchers have explored combining modern text and graph models. Zhang et al. [46] introduced heterogeneous graph neural networks that handle multiple node and edge types, enabling richer representations of complex networks. This architecture could, for example, distinguish between different relationship types (retweet vs. reply vs. mention) or incorporate both user nodes and content nodes in a single graph.

The work most relevant to our approach comes from Yu et al. [47], who developed multimodal graph convolutional networks. Their key insight is that different modalities should not just be concatenated - they should be fused with learned weights that can adapt to which modality is more informative. They propose attention-based fusion mechanisms where the model learns to weight text and graph representations differently for different users or tasks.

Our work implements three fusion strategies to explore this design space. The simplest approach is concatenation: we combine BERT text embeddings and graph features by simply stacking them into a longer feature vector. This baseline lets both modalities contribute but treats them as equally important.

More sophisticated is gated fusion, where we learn scalar weights for each modality. The model can emphasize text when network information is weak or unreliable, and vice versa. This provides some adaptation while remaining computationally efficient.

In our work we also use attention fusion, inspired by Yu et al. [47]. Here we compute attention weights over the text and graph representations, allowing the model to dynamically combine them based on their content. For some users with very informative network positions, the model learns to rely heavily on graph signals; for others with distinctive language use but less informative networks, text may dominate.

These findings carry implications beyond specific numbers. First, they confirm that multimodal approaches offer genuine improvements over single-modality methods. Second, they reveal that network structure provides the stronger signal for ideology in our Twitter data - perhaps because connections are harder to manipulate or misrepresent than text content. Third, they demonstrate the value of sophisticated fusion mechanisms that can adapt to varying information quality across modalities.

2.7 Research Contributions

Having reviewed the landscape of ideology detection research, we can now articulate how our work contributes to this field. While existing studies have explored network methods, text analysis, and deep learning approaches, several important gaps remain.

First, most prior work examines these modalities in isolation or with limited integration. Network-based studies like those of Barberá [23] focus exclusively on follower patterns, while text-based work like those of Preoțiuc-Pietro et al. [35] analyzes content without considering network structure. The few multimodal studies that exist often use simpler models or do not systematically compare different fusion strategies. We address this gap through comprehensive experiments that evaluate text-only, graph-only, and hybrid approaches on identical data, enabling direct comparison of their relative contributions.

Second, while graph features are commonly used in network analysis, we incorporate ideology-specific features that directly capture homophily patterns. Our average neighbor ideology, echo chamber score and ideology homophily metrics operationalize theoretical concepts from political science in ways that improve prediction. This demonstrates the value of domain-informed feature engineering even in an era of end-to-end deep learning.

Third, we employ state-of-the-art architectures that have not been fully exploited for ideology detection. Combining BERT with graph neural networks (GCN and GraphSAGE) represents a modern technical stack that balances performance with computational feasibility for large-scale networks. Our attention-based fusion mechanism allows these powerful models to combine adaptively, with the model learning to weight modalities based on their informativeness for each prediction.

Our work thus sits at the intersection of multiple research traditions. From network science, we draw tools for analyzing social structure and the theoretical framework of homophily. From natural language processing, we adopt modern transformer-based representations. From machine learning, we employ graph neural networks and attention mechanisms. The synthesis of these approaches, combined with careful empirical analysis, advances our understanding of how ideology manifests in social media and how we can detect it algorithmically.

3

Data and Experimental Setup

3.1 Source Dataset and Data Collection

The primary data source for this work is a large-scale Twitter dataset collected and analyzed by Baqir et al. [48], who studied the drivers of active participation in polarized social media debates. Their study focused on the controversial debate about whether countries should provide military assistance to Ukraine during its 2022-2023 conflict with Russia. This section describes the data collection procedure, the characteristics of the resulting dataset, and the methodology used to generate ideology labels that serve as ground truth for our supervised learning experiments.

3.1.1 Data Collection Methodology

The dataset was collected using Twitter's Standard API over a four-month period from November 22, 2022 to March 1, 2023. This timeframe captured a period of intense international debate about military support for Ukraine, including discussions about providing tanks (Abrams, Leopard, Challenger), aircraft, munitions, and missile systems (HIMARS) to Ukrainian forces.

Keyword-based sampling: The data collection used a keyword-based search query that identified tweets mentioning any of the following terms: "military aid", "military support", "tanks", "abrams", "leopard", "challenger", "jet", "aircraft", "munitions", "HIMARS", "rockets", and "missile". This approach ensured that the dataset focused on the specific policy debate

about military assistance rather than the broader conflict.

The collection process yielded over 17 million tweets posted by more than 5.2 million users. A notable feature of this dataset is the inclusion of impression counts - a metric introduced by Twitter in December 2022 that quantifies how many times a tweet appeared on users' screens. While Baqir et al. used impression counts to study active versus passive consumption, we focus on the retweet network structure and textual content for ideology inference.

Data filtering: The original dataset underwent several filtering steps. First, tweets without valid impression counts were removed (impression count should always be greater than or equal to any active action metric). Second, only English-language tweets were retained to ensure consistent text analysis. Third, the analysis focused on original tweets rather than retweets, replies, or quotes, as these represent primary content creation rather than secondary engagement.

Retweet network construction: From the filtered dataset, Baqir et al. constructed a directed retweet network where nodes represent users and edges represent retweet actions. An edge from user u to user v indicates that user u retweeted content authored by user v at least once during the observation period. Retweets were chosen as the basis for the network because they represent endorsement - sers amplify content they find credible or agreeable, making retweet patterns a behavioral manifestation of ideological alignment.

3.1.2 Dataset Composition and Scale

The complete dataset consists of several interrelated components that together provide a comprehensive view of political behavior in this Twitter debate:

- **Tweet corpus**: Contains over 17 million tweets posted by users discussing the military aid debate. Each tweet includes the text content, author identifier, timestamp, and engagement metrics (retweet count, reply count, like count, quote count, and impression count).

- **User metadata**: Contains profile information for the 5.2+ million user accounts that appear in the dataset. Available fields include biographical descriptions, locations, display names, follower counts, and other profile attributes.

- **Retweet network**: The final interaction network contains 2,585,169 nodes (users) and 7,152,688 directed edges (retweet actions). Edge weights reflect the number of times one user retweeted another, though preliminary analysis found that binary edges (presence/absence of connection) performed comparably to weighted networks for ideology prediction.

- **Ideology labels**: Contains ideology score estimates for a subset of users generated using the Latent Ideology Estimation method described below. These scores serve as the ground truth labels for supervised learning.

Table 3.1 summarizes the key statistics of the dataset used in our analysis.

Component	Count
Total tweets collected	17,000,000+
Total users in dataset	5,200,000+
Users in retweet network	2,585,169
Retweet edges	7,152,688
Collection period	Nov 22, 2022 – Mar 1, 2023

Table 3.1: Dataset summary statistics from Baqir et al. (2025)

3.1.3 Ground Truth Ideology Labels: Latent Ideology Estimation

A central challenge in ideology inference research is obtaining reliable ground truth labels. Baqir et al. [48] employed Latent Ideology Estimation, a method based on correspondence analysis that infers users' political positions from their retweeting behavior. This approach, originally developed by Barberá [23] for analyzing follower-following relationships, was adapted to use retweet networks as the basis for ideology inference.

Correspondence Analysis Methodology

The Latent Ideology Estimation method operates on the principle of homophily: users who retweet similar sets of influential accounts likely share similar ideological orientations. The method requires first identifying a set of "influencers" - accounts that represent distinct ideological positions in the debate.

Influencer selection: Baqir et al. identified 204 influencer accounts through a multi-step process. First, they ranked users by in-degree (number of unique users who retweeted them) and manually selected prominent accounts from three categories: (i) Russian and Ukrainian politicians, (ii) official media accounts (journals, TV channels), and (iii) political activists. This seed set included accounts with clear positions on military aid - for example, @NATO, @SecBlinken,

and Ukrainian president @ZelenskyyUa on the pro-aid side, and accounts associated with the Russian government or pro-Russia positions on the opposing side.

The seed set was then expanded using Twitter's "Who to follow" recommendations, which suggest similar accounts. This process continued until no new accounts were suggested. The final set was refined by excluding accounts with in-degree below 100 and those whose content was unrelated to the Ukrainian conflict. The resulting 204 influencers represented both supporters and opponents of military assistance to Ukraine.

Correspondence analysis algorithm: Given the set of influencers, the method constructs a matrix $\mathbf{A}$ where element A_{ij} represents the number of retweets user i directed toward influencer j. This matrix is then normalized and decomposed using singular value decomposition (SVD).

The normalization proceeds as follows. First, the matrix is divided by the total number of retweets to obtain a probability matrix:

$$\mathbf{P} = \frac{\mathbf{A}}{\sum_{ij} A_{ij}}$$

Then, row and column sums are computed:

$$\mathbf{r} = \mathbf{P}\mathbf{1}, \quad \mathbf{c} = \mathbf{1}^T\mathbf{P}$$

where $\mathbf{1}$ is a vector of ones. The normalized matrix is:

$$\mathbf{S} = \mathbf{D}_r^{-1/2}(\mathbf{P} - \mathbf{r}\mathbf{c}^T)\mathbf{D}_c^{-1/2}$$

where $\mathbf{D}_r = \text{diag}(\mathbf{r})$ and $\mathbf{D}_c = \text{diag}(\mathbf{c})$ are diagonal matrices. This transformation ensures that the analysis is not dominated by highly active users or very popular influencers.

Finally, singular value decomposition is performed: $\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. The ideology score for user i is taken as the i-th entry of the first column of $\mathbf{U}$. The first singular vector captures the primary axis of variation in retweeting behavior, which in polarized political debates typically corresponds to the left-right ideological spectrum. Influencer ideology scores are estimated by taking the median ideology of their retweeters.

Figure 3.1 displays the distribution of ideology scores for both users and influencers. The distribution exhibits a clear bimodal pattern with distinct clusters corresponding to supporters (negative scores) and opponents (positive scores) of military aid to Ukraine. The left-leaning cluster contains approximately 93% of users, while the right-leaning cluster contains 7%, indicating that supporters of Ukraine aid are 13 times more numerous than opponents in this dataset.

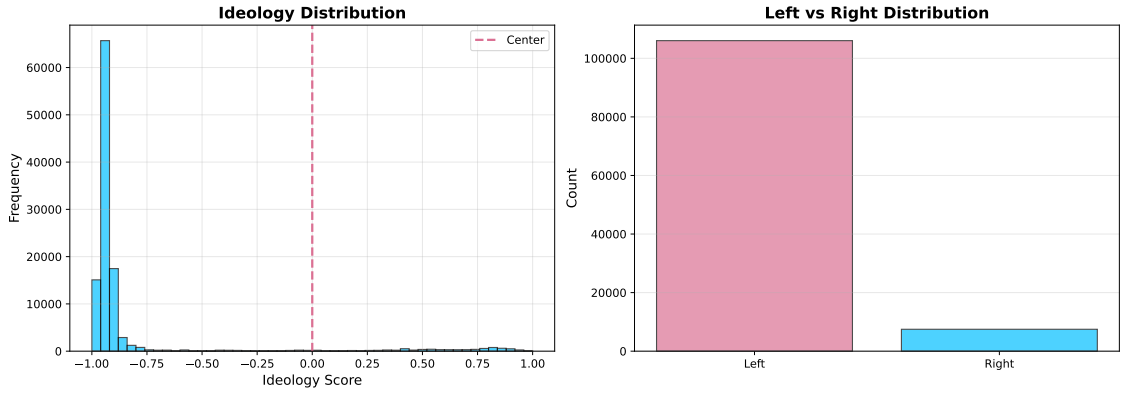

Figure 3.1: Distribution of ideology scores for users and influencers. The bimodal distribution reflects strong polarization around the military aid debate. Negative scores represent support for military aid to Ukraine, while positive scores represent opposition. The asymmetry reflects the larger number of aid supporters in the dataset.

The validity of these ideology scores was confirmed through examination of influencer positions. Accounts with strongly negative scores included NATO, the U.S. Secretary of State, and the Ukrainian president - all entities clearly supporting military aid. Accounts with strongly positive scores were associated with the Russian government or had explicit pro-Russia stances. This correspondence between estimated positions and known political alignments validates that the first dimension captures the relevant ideological divide in this debate.

The bimodal distribution and clear separation between the two groups indicate a highly polarized discussion where most users unambiguously belong to one ideological camp. Baqir et al. categorized users into two groups based on their estimated ideology scores: supporters of Ukraine aid (ideology score ≤ 0) and opponents (ideology score > 0). This polarization is consistent with broader patterns observed in online political discourse, where controversial issues tend to fragment into opposing echo chambers with limited cross-ideological interaction.

Coverage and Scope

It is important to note that the Latent Ideology Estimation method can only assign scores to users who have sufficient retweeting activityspecifically, users who have retweeted at least one of the identified influencers. Users who rarely retweet, or who only retweet non-political content, cannot be reliably positioned in the ideological space. This creates a natural focus on politically engaged users who actively participate in the debate by sharing content.

The dataset we use in our analysis contains ideology labels for users who meet these criteria. These labeled users represent those most actively engaged in discussing and sharing content about military aid to Ukraine, making them an appropriate population for studying how ideology manifests in both text and network structure.

3.1.4 Dataset Characteristics

The Baqir et al. dataset is particularly well-suited for this specific work on ideology inference for several reasons:

1. **Focused debate:** The dataset captures discussion of a specific, well-defined policy issue (military aid to Ukraine) rather than general political discourse. This focus ensures that ideological positions are salient and clearly expressed in both text and network behavior.

2. **Temporal specificity:** The four-month collection period (November 2022 to March 2023) represents a time of intense international debate about Ukraine policy, ensuring high engagement and clear ideological divisions. This temporal focus also controls for potential confounds from shifting political contexts.

3. **Scale appropriate for this work:** With 2.5 million users in the retweet network and 7.1 million edges, the dataset is large enough to exhibit meaningful structural patterns (communities, echo chambers, homophily) while remaining computationally tractable for graph neural network training on available hardware.

4. **Polarization structure:** The bimodal ideology distribution and clear separation between opposing camps (as documented by Baqir et al.) indicate that this is a polarized debate where ideological differences are strongly manifested. This provides a favorable setting for testing whether models can learn to distinguish political orientations and measure the strength of homophily effects.

5. **Multiple modalities for comparison:** The availability of both textual content (tweets, user profiles) and network structure (retweet graph) allows us to directly compare and combine different types of information, addressing our research questions about multimodal approaches and their relative contributions.

6. **Data scarcity scenario: Critically for this work**, the dataset contains a large amount of unlabeled data alongside limited labeled examples. Out of 2.5+ million users in the retweet network, only approximately 76,993 have ideology labels (roughly 3% coverage). This data scarcity scenario is realistic and reflects typical conditions in social media analysis, where obtaining ground truth labels is expensive. The sparse labeling allows us to evaluate semi-supervised approaches that leverage network structure to propagate ideology information from labeled to unlabeled nodes, which is a key contribution of graph-based methods.

7. **Validated labels:** The Latent Ideology Estimation methodology has been extensively validated in prior work [23], and the face validity of influencer positions in this dataset (correlation between estimated scores and known political alignments) provides confidence in the quality of our ground truth labels.

The combination of scale, polarization, multimodal features, and realistic data scarcity makes this dataset ideal for evaluating ideology inference methods under conditions commonly encountered in practice. The 3% labeling rate is particularly valuable because it enables testing of semi-supervised and graph-based approaches that can leverage unlabeled data through network propagation, which is one of the key advantages explored in this thesis.

3.1.5 Dataset Scope and Limitations

For the purposes of this work, we treat the ideology scores from Baqir et al. as ground truth labels, acknowledging that they are themselves estimates with associated uncertainty. This is standard practice in ideology inference research, where obtaining objective measures of political beliefs is fundamentally difficult.

The dataset focuses on English-language tweets, which may introduce geographic and demographic biases. The debate about Ukraine military aid had particularly high salience in English-speaking countries (United States, United Kingdom, other NATO members), and the user population may not represent the global Twitter audience. However, this focus ensures linguistic consistency for our text-based models.

The temporal scope (November 2022 to March 2023) represents a specific period in the Ukraine conflict. Models trained on this data may not generalize perfectly to discussions of the same issue at different times, or to other political debates. However, our goal is not to build a universal ideology detector but to understand how different types of information (text, network structure) can be leveraged for ideology inference and to quantify their relative contributions.

Finally, we note that the retweet network represents only one type of interaction. Users may also follow, mention, or reply to others, and these alternative interaction types may reveal different aspects of ideological structure. The choice to focus on retweets follows standard practice in the field and is justified by the interpretation of retweets as endorsements - a stronger signal of ideological alignment than mere exposure.

3.2 Data Preprocessing

3.2.1 Text-based Features Construction

In our work individual tweets were filtered to retain only those authored by users present in the ideology dataset. User-level text representations were constructed by concatenating all tweets from a given user into a single document. This aggregation strategy is standard in user-level NLP for political classification [35], as individual tweets may be too short and noisy to carry reliable ideological signals, while the aggregate reflects a user's overall communicative style.

Text cleaning applied the following steps: HTML entity unescaping; removal of URLs; removal of punctuation and special characters (preserving `@` and `#`); lowercasing; whitespace normalisation; removal of the "rt" token introduced by Twitter's retweet mechanism. Users whose cleaned text was empty after these operations were excluded from the text-based analysis.

3.2.2 Enhanced Text Feature Extraction

Traditional approaches to ideology inference from social media data typically utilize only tweet content, overlooking valuable information present in user profiles and metadata. In our work we addresses this limitation by integrating multiple textual sources into a unified representation for each user.

We constructs a composite text string by concatenating four distinct components, each prefixed with a semantic tag. The biographical description (`[BIO]`) often contains explicit ideological self-identifications such as political affiliations or policy preferences. The location field (`[LOC]`) captures geographic information that correlates with regional political cultures. The username (`[NAME]`) may include political symbols or references. Finally, the tweet content (`[TWEETS]`) provides behavioral evidence of political stance through expressed opinions and shared content. To manage computational costs while preserving information density, tweet text is truncated to 500 characters per user.

3.2.3 Retweet Network Construction

The retweet network was loaded as a directed graph $G = (V, E)$ using NetworkX, where V is the set of users and $E \subset V \times V$ is the set of directed retweet edges. The network statistics are summarised in Table 3.2.

Property	Value
Number of nodes	2585168
Number of edges	7152687
Weakly connected	No
Strongly connected	No

Table 3.2: Statistical properties of the retweet network. The network contains 2.5+ million users connected by 7.1+ million directed retweet edges. The network is neither weakly nor strongly connected, indicating the presence of multiple disconnected components and unidirectional information flows characteristic of polarized debates.

3.3 Feature Engineering: Graph Features

The retweet network provides rich structural information that complements textual content in ideology inference. This study extracts twelve graph-based features that capture different aspects of network position, connectivity patterns, and ideological clustering. These features are organized into four categories: degree centrality measures, global centrality and cohesion metrics, community structure indicators, and ideology-specific homophily measures.

Degree Centrality Measures

Degree centrality quantifies the immediate connectivity of nodes in the network. For a directed graph $G = (V, E)$ where V is the set of nodes and E is the set of edges, three degree-based metrics are computed for each node $v \in V$.

The **in-degree** $d^{in}(v)$ counts the number of incoming edges to node v:

$$d^{in}(v) = |\{u \in V : (u, v) \in E\}|$$

In the retweet network context, this represents the number of times a user's content has been retweeted by others, serving as a measure of influence or attention received.

The **out-degree** $d^{out}(v)$ counts the number of outgoing edges from node v:

$$d^{out}(v) = |\{u \in V : (v, u) \in E\}|$$

This quantifies the number of other users whose content the focal user has retweeted, indicating engagement with the broader network.

The **total degree** $d(v)$ is the sum of in-degree and out-degree:

$$d(v) = d^{in}(v) + d^{out}(v)$$

This provides an overall measure of connectivity, capturing both information consumption and dissemination behaviors.

The **average neighbor degree** extends the local view by incorporating second-order connectivity:

$$\text{AND}(v) = \frac{1}{d(v)} \sum_{u \in \mathcal{N}(v)} d(u)$$

where $\mathcal{N}(v)$ denotes the set of neighbors of v. This metric captures whether a user connects with highly connected users or with peripheral users, providing insight into their position within the network hierarchy.

Global Centrality and Cohesion Metrics

While degree centrality captures local connectivity, global centrality measures assess a node's importance within the entire network structure.

PageRank [49] computes the probability that a random walker following directed edges arrives at each node. For a node v, the PageRank score $PR(v)$ is defined by the recursive formula:

$$PR(v) = \frac{1 - d}{N} + d \sum_{u \in \mathcal{N}^{in}(v)} \frac{PR(u)}{d^{out}(u)}$$

where $N = |V|$ is the number of nodes, d is the damping factor and $\mathcal{N}^{in}(v)$ denotes the set of nodes with edges pointing to v. This metric identifies users whose content propagates widely through the network with retweet cascades. The iterative computation proceeds until convergence, defined as maximum change in PageRank values falling below a tolerance threshold of 10^{-4}, or until a maximum of 50 iterations is reached.

The **clustering coefficient** measures the tendency of a node's neighbors to form connections among themselves. The clustering coefficient $C(v)$ for node v is:

$$C(v) = \frac{2T(v)}{d(v)(d(v)-1)}$$

where $T(v)$ is the number of triangles (closed triplets) containing node v. A high clustering coefficient indicates that a user's connections form a cohesive, tightly-knit group, which is associated with information filtering and echo chamber formation.

The number of **triangles** $T(v)$ is computed directly:

$$T(v) = |\{(u,w) : u,w \in \mathcal{N}(v) \text{ and } (u,w) \in E\}|$$

This raw count complements the normalized clustering coefficient by providing information about absolute cohesion levels.

The **k-core number** identifies the maximal subgraph in which every node has at least degree k. Formally, a k-core is a maximal subgraph $H \subseteq G$ where $\forall v \in H : d_H(v) \geq k$. The core number $c(v)$ of node v is the highest value of k for which v belongs to a k-core. This metric distinguishes between peripheral nodes in the network (low core numbers) and nodes in the densely connected core (high core numbers), capturing hierarchical network structure.

Community Structure

Community detection identifies densely connected groups of nodes that are sparsely connected to other groups. This study employs the Louvain method [50], which optimizes the modularity measure:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

where A_{ij} is the adjacency matrix, k_i is the degree of node i, m is the total number of edges, c_i is the community assignment of node i, and $\delta(c_i, c_j) = 1$ if $c_i = c_j$ and 0 otherwise. The Louvain algorithm proceeds hierarchically, first assigning each node to its own community, then iteratively merging communities to maximize modularity.

Each node is assigned a **community identifier**, which serves as a categorical feature capturing group membership. Empirical analysis reveals that communities in the retweet network exhibit strong ideological coherence, with users within the same community sharing similar

political orientations. This reflects the well-documented phenomenon of political homophily in social networks.

Ideology-Specific Homophily Measures

The final category of features explicitly incorporates ideology information to quantify homophily and echo chamber effects. These features leverage the known ideology scores of labeled nodes to characterize the ideological composition of each node's neighborhood.

The **average neighbor ideology** $\bar{I}_{\mathcal{N}(v)}$ computes the mean ideology score of a node's neighbors:

$$\bar{I}_{\mathcal{N}(v)} = \frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} I(u)$$

where $I(u)$ is the ideology score of node u, with negative values indicating left-wing orientation and positive values indicating right-wing orientation. This feature has proven to be the single most predictive graph-based feature in preliminary experiments, accounting for approximately 89% of the total feature importance in ideology prediction models. The high predictive value reflects the strong homophily in political networks, where users preferentially connect with ideologically similar others.

The **ideology homophily** measure quantifies the ideological similarity between a node and its neighbors:

$$H(v) = -\frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} |I(v) - I(u)|$$

The negative sign ensures that higher values indicate greater homophily (smaller ideological distance). This metric captures the degree to which a user's network environment aligns with their own ideological position.

The **echo chamber score** measures the proportion of a node's neighbors sharing the same ideological polarity:

$$E(v) = \frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} [\text{sign}(I(v)) = \text{sign}(I(u))]$$

where $[\cdot]$ is the indicator function that equals 1 if the condition is true and 0 otherwise. This binary classification of ideology (left vs. right based on the sign of the ideology score) provides a simplified measure of polarization. An echo chamber score close to 1 indicates that nearly all of a user's connections share their broad ideological orientation, while a score near 0.5 suggests

a more ideologically diverse network.

Important methodological note: Following standard practice in homophily analysis [1], ideology-based features (average neighbor ideology, ideology homophily, echo chamber score) were computed on the undirected version of the retweet network. This choice reflects the bidirectional nature of ideological influence: when user A retweets user B, both the act of retweeting (indicating A's preferences) and the act of being retweeted (indicating B's influence) carry ideological signals. Converting to an undirected graph ensures that both in-links and out-links contribute to neighborhood-based features, substantially increasing the number of available labeled neighbors and better capturing the symmetric nature of ideological similarity.

Table 3.3 summarizes the twelve extracted graph features and their computational complexity.

Table 3.3: Summary of twelve graph-based features extracted from the retweet network for ideology prediction. Features are organized into four categories: degree centrality measures (4 features), global centrality and cohesion metrics (4 features), community structure indicators (1 feature), and ideology-specific homophily measures (3 features). Each feature captures a different aspect of network position and connectivity patterns. The ideology-based features (average neighbor ideology, ideology homophily, echo chamber score) proved most predictive in subsequent modeling.

Feature	Category	Complexity
In-degree	Centrality	$O(\|E\|)$
Out-degree	Centrality	$O(\|E\|)$
Total degree	Centrality	$O(\|E\|)$
Avg. neighbor degree	Centrality	$O(\|E\|)$
PageRank	Global centrality	$O(k\|E\|)$
Clustering coefficient	Cohesion	$O(\|E\|^{1.5})$
Triangle count	Cohesion	$O(\|E\|^{1.5})$
Core number	Hierarchy	$O(\|E\|)$
Community ID	Community	$O(\|E\| \log \|V\|)$
Avg. neighbor ideology	**Homophily**	$O(\|E\|)$
Ideology homophily	Homophily	$O(\|E\|)$
Echo chamber score	Homophily	$O(\|E\|)$

Computational Efficiency and Scalability

The extraction of these twelve features involves varying computational complexity. Degree-based metrics (d^{in}, d^{out}, d, AND) are computed in linear time $O(|E|)$ through edge list traversal. The clustering coefficient and triangle counting require examination of second-order neighborhoods, resulting in $O(|E|^{1.5})$ complexity in practice, though optimized implementations

reduce this substantially. PageRank computation via power iteration scales as $O(k|E|)$ where k is the number of iterations (typically 20-50). Community detection using the Louvain method has quasi-linear complexity $O(|E| \log |V|)$ in practice.

In the final version of the models only 11 features were included - all except the Triangle count since it is computationally high. We work with a large dataset, so it is crucial to use scalable methods.

Feature Preprocessing and Normalization

The features exhibit widely varying ranges: degree values span from 1 to several thousand, PageRank scores are bounded in $(0, 1)$, and clustering coefficients are in $[0, 1]$. Without normalization, gradient-based optimization algorithms struggle to converge effectively.

StandardScaler normalization is applied, transforming each feature x to:

$$\tilde{x} = \frac{x - \mu_x}{\sigma_x}$$

where μ_x and σ_x are the mean and standard deviation of feature x computed on the training set. This transformation ensures zero mean and unit variance for all features, significantly improving model convergence. Empirical experiments demonstrated that Graph Neural Networks trained on unnormalized features achieved $R^2 = 0.11$, while identical architectures with normalized features achieved $R^2 = 0.91$, underscoring the critical importance of this preprocessing step.

For Graph Neural Networks specifically, an additional preprocessing step is required: the addition of self-loops to the adjacency matrix. This ensures that each node includes itself in its neighborhood aggregation, which is necessary for stable training of GCN architecture.

3.3.1 Graph Features Statistics

Table 3.4 presents descriptive statistics for all twelve graph-based features computed on the retweet network. These statistics provide insight into the distribution and scale of network properties.

The high echo chamber score among labeled users (mean = 0.95) is particularly striking. However, this value must be interpreted in context of the dataset composition: 93% of labeled users are supporters (negative ideology) versus only 6% opponents (positive ideology). This strong imbalance inflates the echo chamber metric for the majority group - supporters

Feature	Mean	Std	Min	Max	Median
In-degree	2.77	8.29	0.00	749	1.00
Out-degree	2.77	133.00	0.00	59,006	1.00
Total degree	5.53	133.37	1.00	59,007	1.00
Avg neighbor degree	0.62	20.48	0.00	6,273	0.00
PageRank ($\times 10^{-8}$)	3.87	1.99	3.57	1503.01	3.57
Clustering coefficient	0.02	0.12	0.00	1.00	0.00
Core number	0.00	0.00	0.00	0.00	0.00
Community	0.00	0.00	0.00	0.00	0.00
Avg neighbor ideology	-0.11	0.32	-1.00	1.00	0.00
Ideology homophily	0.00	0.02	-1.82	0.00	0.00
Echo chamber score (all data)	0.03	0.17	0.00	1.00	0.00
Echo chamber score (labled data)	0.95	0.20	0.00	1.00	0.00

Table 3.4: Descriptive statistics for all twelve graph-based features computed on the retweet network. Statistics are computed across all 2.59M nodes in the network. Notable patterns: (1) Degree distributions are highly right-skewed, with most users having few connections (median degree = 1 - 2) but some hubs having thousands (max = 59,007). (2) PageRank values span several orders of magnitude, from 10^{-9} to 10^{-3}, reflecting hierarchical influence structure. (3) Most nodes have low clustering (median = 0), indicating sparse local connectivity. (4) Ideology-based features show different patterns for all data vs. labeled data: when computed across all nodes, values are near zero due to limited label coverage; when computed only for labeled nodes with labeled neighbors, echo chamber scores are very high (mean = 0.95), indicating strong preferential attachment to ideologically similar users.

predominantly connect to other supporters simply because most politically engaged users in the network are supporters. The correlation analysis (r = 0.96) provides a more balanced measure of homophily that accounts for continuous ideology values rather than binary polarity, confirming that homophily is indeed strong but not perfect segregation.

Figure 3.2 visualizes the distributions of selected features using kernel density estimation. The distributions are computed on the 74,789 labeled nodes with at least one labeled neighbor, representing the effective sample used for graph-based inference.

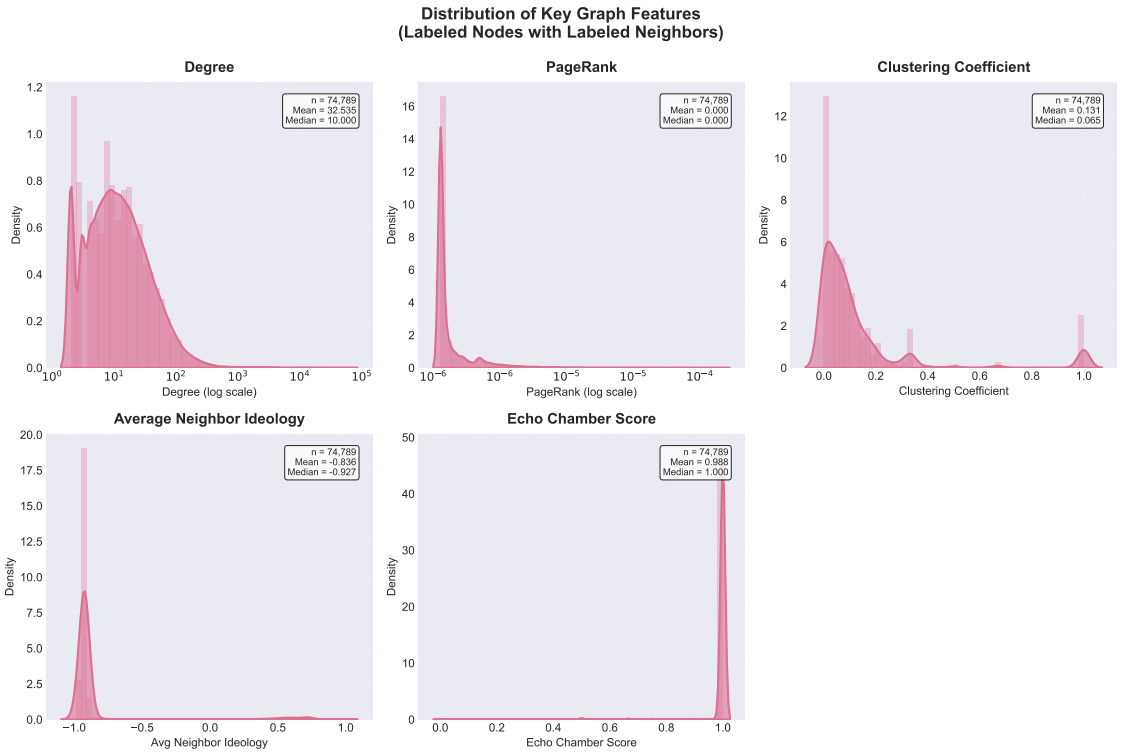

Figure 3.2: Kernel density plots showing distributions of key graph features among labeled nodes with labeled neighbors (n = 74,789, representing 65.9% of all labeled nodes). **Top row:** Structural features. (a) Degree distribution exhibits a power-law pattern with long right tail (mean = 32.5, median = 10), indicating most users have few connections while some hubs are highly connected. (b) PageRank values are highly concentrated near the minimum, with rare influential outliers. (c) Clustering coefficients are low (mean = 0.131, median = 0.065), indicating sparse local connectivity typical of large-scale social networks. **Bottom row:** Ideology-based features. (d) Average neighbor ideology shows strong negative skew (mean = -0.836, median = -0.927), directly reflecting the dataset composition where 93% of labeled users are Ukraine aid supporters with negative ideology scores. (e) Echo chamber scores are remarkably high (mean = 0.988, median = 1.000), indicating near-perfect ideological segregation: the vast majority of users connect almost exclusively with others who share their ideological polarity.

Several patterns emerge from these distributions. First, structural features follow expected patterns for large-scale social networks: degree distributions are heavily right-skewed with power-law characteristics, PageRank values are highly concentrated (most users have near-minimal influence), and clustering coefficients are generally low, indicating that despite global community structure, local neighborhoods tend to be sparse rather than densely interconnected.

Second, and most critically for ideology inference, the ideology-based features reveal extreme polarization. The average neighbor ideology distribution is sharply peaked around -0.9, meaning that most politically engaged users in this network have neighborhoods that are overwhelmingly composed of Ukraine aid supporters. The distribution shows minimal mass near zero

(neutral neighbors) and very little in the positive range (opponent neighbors), confirming that cross-ideological connections are rare.

Third, the echo chamber score distribution is perhaps the most striking result: the distribution is essentially a spike at 1.0, with mean 0.988 and median 1.000. This indicates that for the vast majority of users, all of their labeled neighbors share their ideological polarity. Such extreme segregation, approaching perfect homophilic sorting, is unusual even for political networks and likely reflects the highly polarizing nature of the Ukraine conflict during the dataset collection period. This near-complete segregation means that knowing a user's network neighborhood provides an almost deterministic signal of their ideology.

The strong negative skew in average neighbor ideology (concentrated around -0.85 to -0.95) directly mirrors the dataset's ideological composition (93% supporters vs. 6% opponents). This asymmetry has important implications: majority-view users (supporters) naturally find themselves surrounded by like-minded others simply because most politically active users are supporters, while minority-view users (opponents) face stronger social pressure as they constitute isolated clusters in a predominantly pro-supporter network. This structural asymmetry is precisely what the graph-based models exploit: ideology is predictable from network position because homophilic selection has created a network where connection patterns encode political stance with high fidelity.

Feature Distributions by Ideological Group

Figure 3.3 compares feature distributions between supporters (93% of labeled users) and opponents (7%).

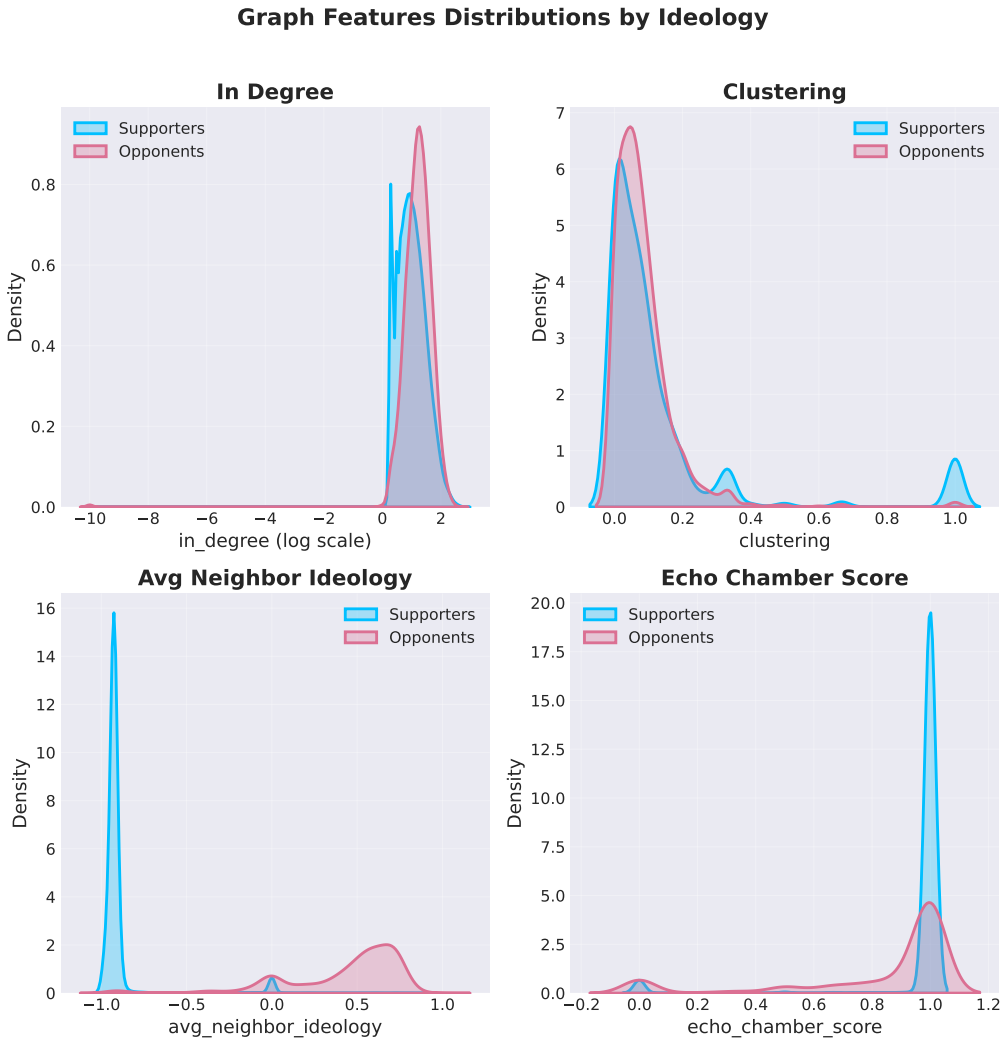

Figure 3.3: Distributions of graph features by ideology. **(a) In-degree** and **(b) Clustering** are nearly identical for both groups, concentrated in 0-2 (log scale) and 0-0.2 ranges respectively. **(c) Avg neighbor ideology** shows complete separation: supporters peak sharply at -1.0, while opponents span -0.3 to 0.9. **(d) Echo chamber score** reveals supporters maintain near-perfect homophily (narrow peak at 1.0), while opponents show more variability (broader distribution).

The key finding is that **structural features (in-degree, clustering) do not differentiate ideological groups**, while **neighborhood ideology features show complete separation**. Both supporters and opponents exhibit similar degree distributions (0–2 range on log scale) and clustering patterns (0–0.2 range), with supporters showing a small secondary peak at clustering = 1.0 representing perfectly connected small groups.

In contrast, average neighbor ideology distributions are completely separated. Supporters' neighborhoods are overwhelmingly negative (sharp peak at -1.0), reflecting near-exclusive con-

nections with other supporters. Opponents show a broad distribution (-0.3 to 0.9), indicating substantial heterogeneity in their network environments. Some opponents have neighborhoods leaning toward support (values near -0.3), while others connect primarily with fellow opponents (values around 0.9). This heterogeneity reflects the challenges faced by a 7% minority embedded in a network dominated by the opposing view.

Echo chamber scores reinforce this asymmetry. Supporters show an extremely narrow peak at 1.0, meaning nearly all supporters connect exclusively with same-polarity users. Opponents exhibit a broader distribution with a lower peak, suggesting more variable exposure to cross-ideological connections - a structural consequence of minority status.

These patterns explain the feature importance hierarchy: avg_neighbor_ideology accounts for 89.1% of predictive power because it captures the only dimension along which supporters and opponents truly differ. Structural features (degree, clustering) contribute only 3.9% because they are nearly identical across groups. The heterogeneity in opponents' neighborhoods does not undermine predictive accuracy because the model learns the full distribution - network position remains highly informative despite asymmetry between majority and minority groups.

The critical insight is that ideology is encoded in whom users connect to (homophily), not how they connect (structure). Both ideological camps exhibit similar structural patterns, but they differ fundamentally in the ideological composition of their neighborhoods.

Ideological Homophily: Correlation Analysis

To further quantify the strength of ideological homophily, we examine the direct relationship between users' own ideological positions and the average ideology of their network neighbors. Figure 3.4 presents this relationship for the subset of labeled users.

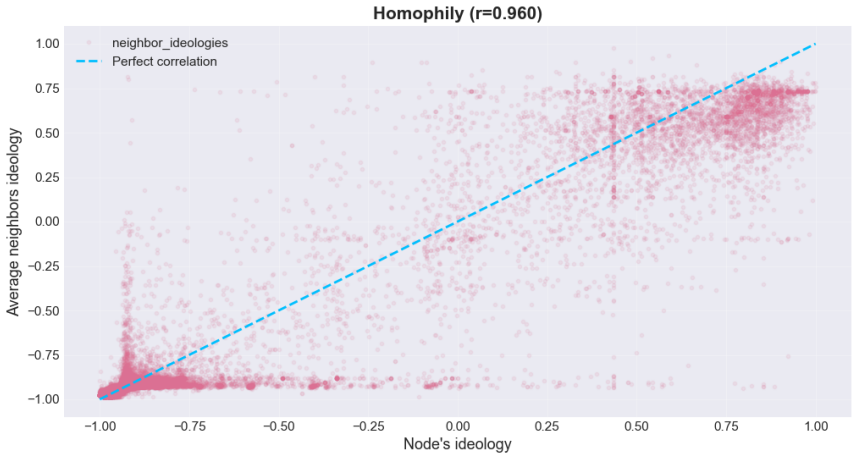

Figure 3.4: Scatter plot showing the relationship between users' own ideology scores and the average ideology of their labeled neighbors. Each point represents one user. The strong positive correlation (Pearson $r = 0.96, p < 0.001$) demonstrates substantial homophily: users predominantly connect with others who share their ideological position. The red line shows the linear fit, while the dashed diagonal represents perfect homophily (where neighbor ideology exactly matches own ideology). The concentration of points along the diagonal confirms that network position is highly predictive of individual ideology, which justifies the use of graph-based features for ideology inference.

The correlation between a user's own ideology and their average neighbor ideology is remarkably strong: Pearson $r = 0.96$ ($p < 0.001$). This finding strongly validates the use of graph-based features for ideology inference: even with sparse labeling (3% coverage), the network structure encodes substantial information about individual ideological positions through homophilic connection patterns.

This correlation of 0.96 aligns with and quantitatively confirms the high echo chamber score (0.95) observed in Table 3.4. Together, these metrics demonstrate that the retweet network exhibits pronounced ideological segregation, where users cluster into homogeneous communities based on their stance on Ukraine aid. This strong homophily is what makes network-based inference methods particularly effective for this dataset: if we know the ideological positions of a user's neighbors, we can predict the user's own position with high accuracy, enabling ideology estimates to propagate effectively through the partially labeled graph.

3.4 Train/Test Split and Evaluation Protocol

All models were evaluated using an 80/20 train-test split stratified to deal with data unbalance. The split was performed with a fixed random seed (SEED = 28) for reproducibility across all

experiments.

3.4.1 Evaluation Metrics

All models solve a regression task, predicting continuous ideology scores, and are evaluated on the following metrics:

- R^2 **(coefficient of determination)** – the proportion of variance in ideology scores explained by the model's predictions.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.1}$$

  where y_i is the true value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean of the observed values, and n is the number of observations. The value of R^2 typically ranges from 0 to 1, where 1 indicates perfect predictions and 0 corresponds to a model that performs no better than predicting the mean. Negative values may occur when the model performs worse than this baseline.

- **MSE (Mean Squared Error)**. The mean squared error measures the average squared difference between the true values and the predicted values.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.2}$$

  Because the errors are squared, this metric penalizes large deviations more strongly, making it sensitive to outliers.

- **RMSE (Root Mean Squared Error)**. The root mean squared error is the square root of the mean squared error.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3}$$

  RMSE has the advantage of being expressed in the same units as the target variable, which makes it easier to interpret.

- **MAE (Mean Absolute Error)**. The mean absolute error measures the average absolute difference between the true values and the predicted values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.4}$$

Unlike MSE and RMSE, MAE does not square the errors and therefore is less sensitive to outliers.

4

Models and Methods

This chapter describes all models implemented and compared in the thesis. They are organised into three types: text-based models (Section 4.1), graph-based models (Section 4.2), and hybrid models (Section 4.3).

4.1 Text-Based Models

Text-based models use the features based on tweet and bio information of each user as the input for ideology prediction.

4.1.1 TF-IDF + Ridge Regression

Firstly we tried classical sparse text representation model and choose it as a baseline. The `term frequency-inverse document frequency` (TF-IDF) vectoriser converts each user's text document into a high-dimensional sparse vector, weighting each term by how often it appears in that user's documents relative to how commonly it appears across all users.

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \log \frac{N}{|\{d : t \in d\}|} \tag{4.1}$$

where: t - term (word), d - document (user's tweet), N - total number of documents (tweets), TF - term frequency

We trained Ridge Regression model on the TF-IDF features to predict continuous ideology scores. Ridge Regression is a linear regression with L2-regularisation.

$$\min_{\beta} \|y - X\beta\|_2^2 + \alpha\|\beta\|_2^2 \tag{4.2}$$

The regularisation coefficient $\alpha = 1.0$ is used. Despite its simplicity, this model establishes a baseline with which we will compare all the other models.

4.1.2 BERT Embeddings Extraction

For the three neural text models, a shared *DistilBERT* encoder [36] is used to produce 768-dimensional contextual embeddings for each user's aggregated data. DistilBERT [36] is a distilled version of BERT-base that retains majority of BERT's performance while using fewer parameters, making it usable for large-scale inference. Embeddings are extracted by passing tokenised text through the model and taking the `[CLS]` token representation from the final layer. We tried both approaches for embedding extraction: using BERT with frozen weights and fine-tuning BERT. These embeddings are used as input features.

4.1.3 BERT + MLP

The `BERT_MLP` model appends a two-layer multi-layer perceptron to the DistilBERT encoder. The hidden layer sizes are [256, 64] with ReLU activations, batch normalisation between layers, and a dropout rate of 0.3. A single linear output neuron produces the predicted ideology score. The model is trained end-to-end on the tokenised tweet sequences using the AdamW optimiser with learning rate 2×10^{-5} for 7 epochs (frozen BERT) and 5 epochs (fine-tuned BERT), with patience 3 for early stopping.

Figure 4.1 shows the residual plot for the frozen BERT + MLP model. The residuals are widely scattered with large variance, indicating poor prediction accuracy. Many predictions deviate substantially from the true values, with residuals frequently exceeding ± 0.5. The training process illustrated in Figure 4.2 shows smooth convergence with validation loss closely tracking training loss, suggesting that the model learns consistently but the frozen embeddings themselves lack the representational capacity for this task.

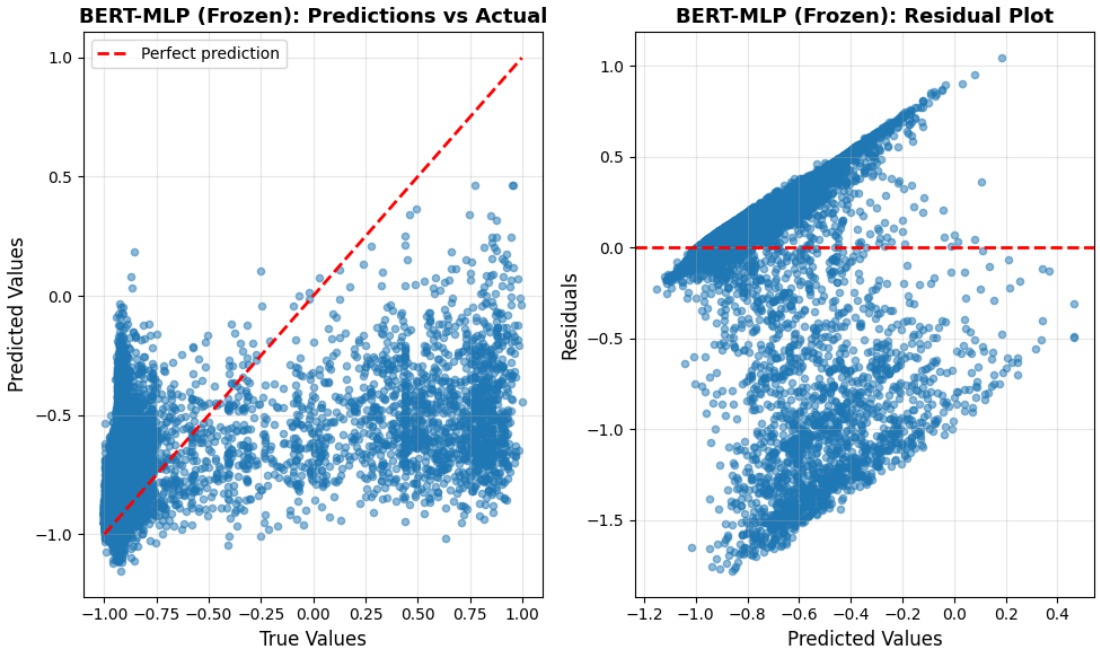

Figure 4.1: Predictions versus actual ideology scores and residual plot for BERT (frozen embeddings) + MLP model on the test set. The x-axis shows predicted ideology scores, while the y-axis shows prediction errors (true - predicted). Residuals are widely scattered with large variance, frequently exceeding ± 0.5, indicating poor prediction accuracy.

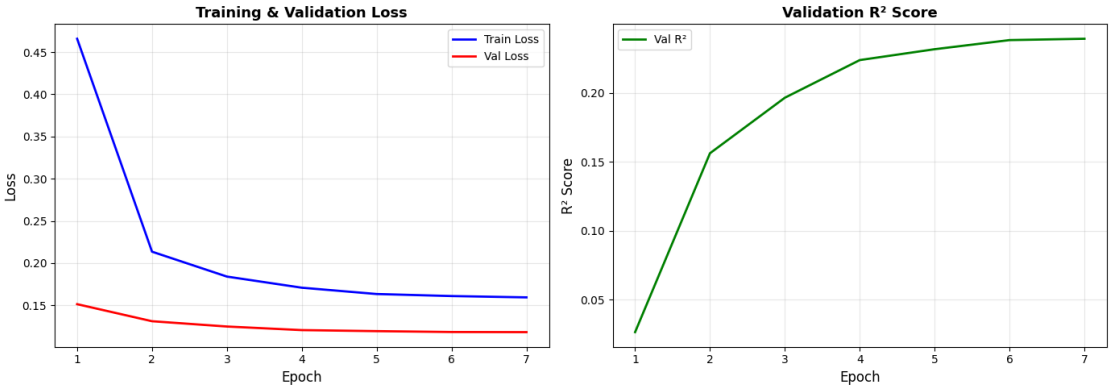

Figure 4.2: Training and validation loss curves for BERT (frozen) + MLP model across epochs. Both curves decrease smoothly and converge, indicating consistent learning without overfitting. However, the final loss values remain relatively high, reflecting the model's limited performance ($R^2 \approx 0.25$) due to the frozen embeddings' inability to capture ideology-relevant features.

As well another approach is used. Model with the same parameters but with fine-tuned DistilBERT is trained for 5 epochs since it requires less time for model to train with fine-tuned

embeddings. The fine-tuned version shows substantially improved performance, as evidenced by the tighter residual clustering in Figure 4.3 compared to the frozen version. Most residuals fall within ± 0.2. The training curves in Figure 4.4 demonstrate faster convergence and higher final $R^2 \approx 0.78$, with both training and validation curves declining smoothly without signs of overfitting. This confirms the benefit of task-specific adaptation.

In the next models only fine-tuned DistilBERT embeddings are used because this training approach shows higher performance.

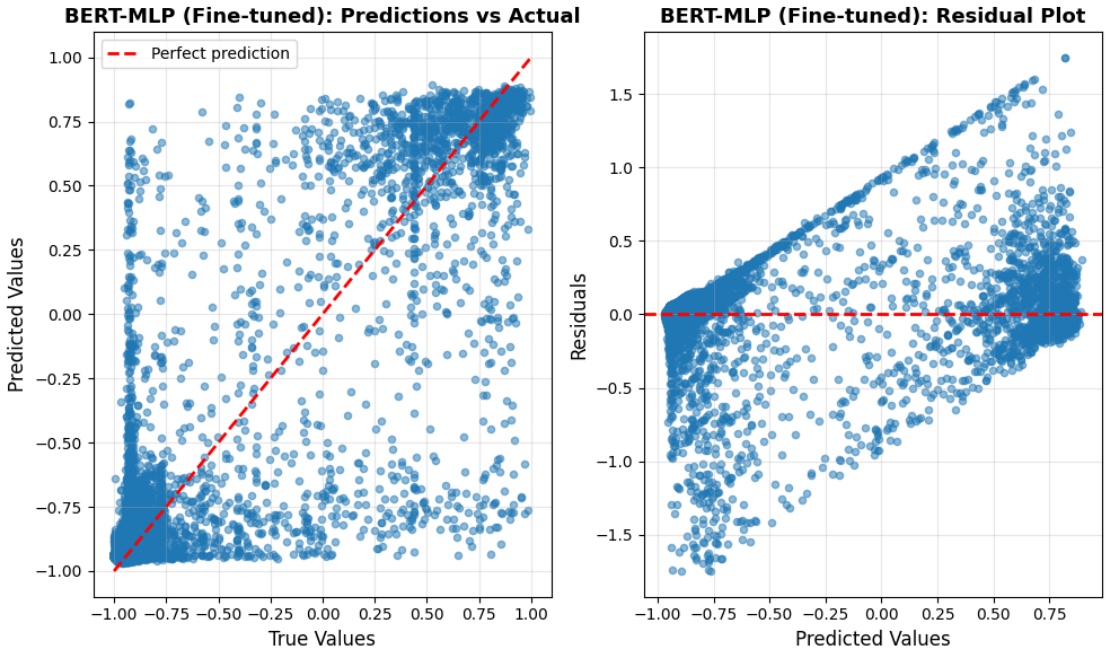

Figure 4.3: Predictions versus actual ideology scores and residual plot for BERT (fine-tuned) + MLP model. Compared to the frozen BERT version (Figure 4.1), residuals show substantially tighter clustering around zero, with most errors within ± 0.2.

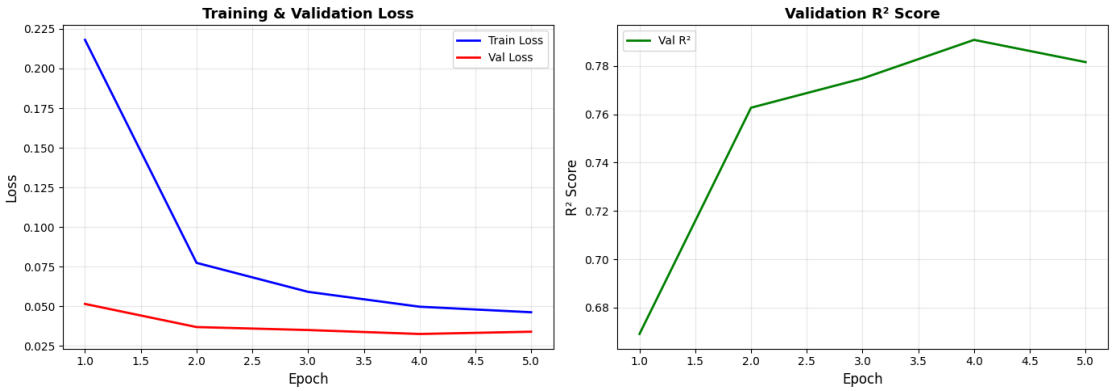

Figure 4.4: Training and validation loss curves for BERT (fine-tuned) + MLP. Fine-tuning the entire BERT encoder (rather than using frozen embeddings) results in substantially lower final losses and better convergence. Both curves decline smoothly without crossing, indicating effective learning without overfitting. The model converges faster than the frozen version, reaching optimal performance in fewer epochs.

4.1.4 BERT + LSTM

The `BERT_LSTM` model passes the sequence of token embeddings from the fine-tuned DistilBERT encoder into a two-layer LSTM with hidden dimension 128 and dropout 0.3. The final hidden state is passed to a linear regression head. This architecture captures sequential structure within the aggregated text, potentially useful for modelling shifts in political framing across a user's tweet history. The LSTM is trained with learning rate 2×10^{-5} for 5 epochs with patience 3.

As shown in Figure 4.5, the LSTM architecture produces residuals similar to the fine-tuned BERT + MLP model, with tight clustering around zero and most errors within ± 0.2. However, the training dynamics in Figure 4.6 reveal a critical difference: the validation loss curve (red) crosses above the training curve (blue) starting around epoch 3, indicating overfitting. The LSTM's additional parameters allow it to memorize training patterns that do not generalize well to validation data. Despite this overfitting tendency, early stopping and dropout regularization prevent severe performance degradation, and the final test performance remains competitive ($R^2 \approx 0.8$).

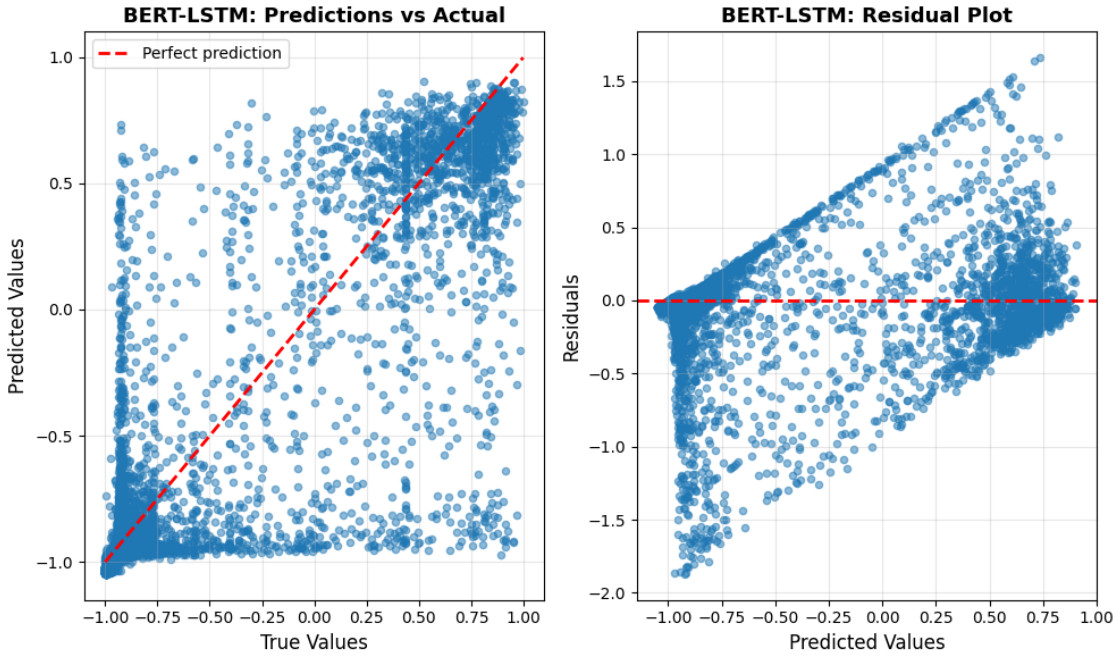

Figure 4.5: Predictions versus actual ideology scores and residual plot for BERT + LSTM model. The LSTM layer processes BERT embeddings sequentially to capture long-range dependencies in concatenated user text. Residuals show patterns similar to the fine-tuned BERT + MLP model, with tight clustering around zero and most errors within ± 0.2.

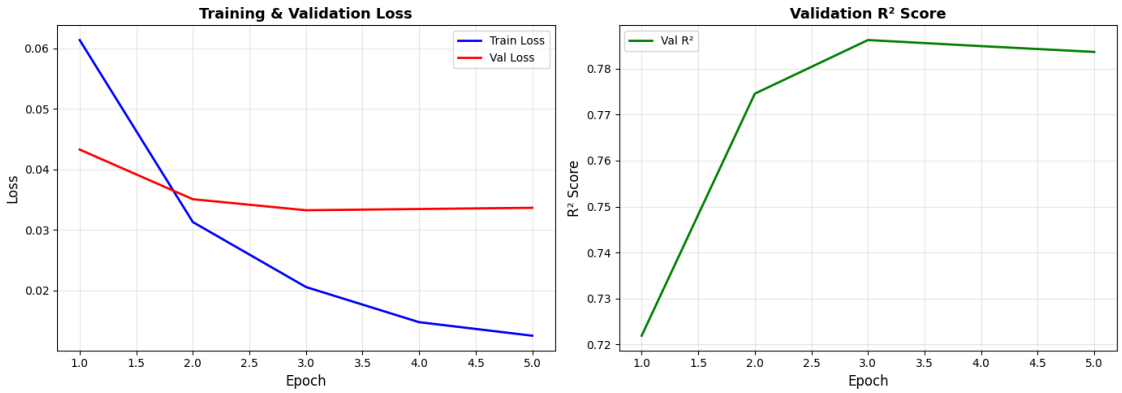

Figure 4.6: Training and validation loss curves for BERT + LSTM. This is clearly visible starting around epoch 3, where the validation curve (red) crosses above the training curve (blue) and remains higher, indicating that the model begins to memorize training-specific patterns - to overfit.

4.1.5 BERT + CNN

The `BERT_CNN` model applies parallel one-dimensional convolutions with filter sizes $\{2, 3, 4, 5\}$ and 128 filters each to the sequence of token embeddings from DistilBERT. Max-pooling over time is applied to each filter bank, and the resulting 512-dimensional feature vector is passed to a linear regression head with dropout 0.3. This architecture has been shown effective for text classification tasks where local n-gram patterns are informative [33]. The model is trained with learning rate 2×10^{-5} for 5 epochs with patience 3.

The residual analysis in Figure 4.7 shows that CNNs achieve performance comparable to LSTM, with residuals distributed around zero and most errors within ± 0.2. However, Figure 4.8 demonstrates that CNNs exhibit even more pronounced overfitting than LSTM: the validation loss begins rising relative to training loss as early as epoch 2. Additionally, the validation R^2 curve shows higher volatility and less stability compared to previous models, with noticeable fluctuations throughout training. This instability, combined with early overfitting, suggests that the CNN's capacity to learn complex local patterns leads it to capture training-specific noise rather than generalizable features. Despite achieving similar final test performance to LSTM, the training instability makes CNN less reliable for this task.

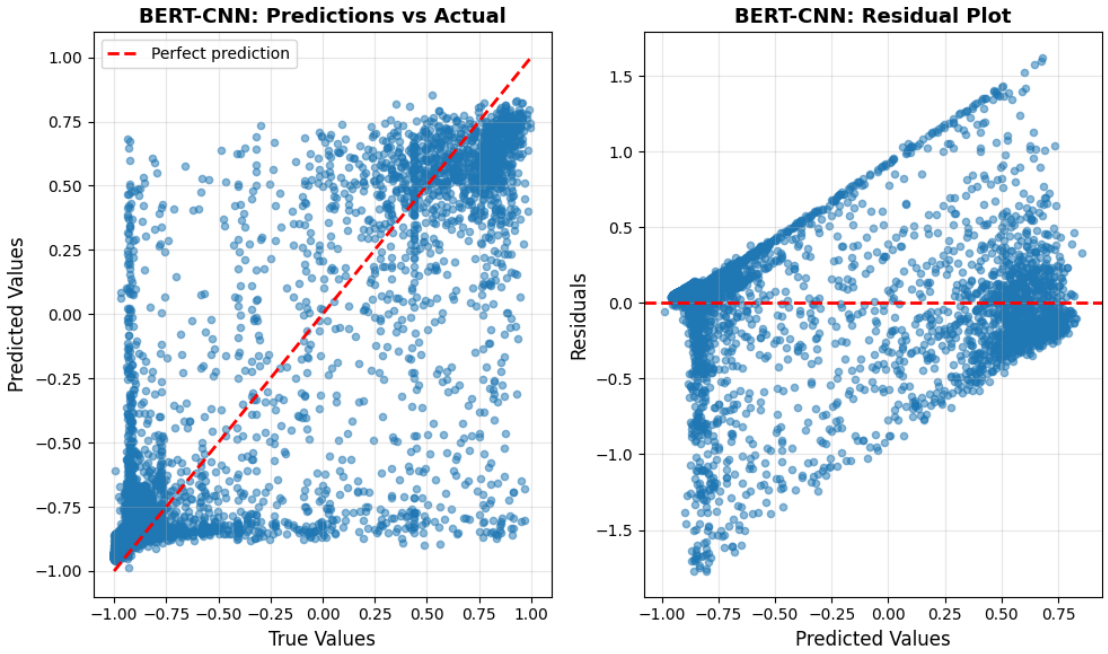

Figure 4.7: Predictions versus actual ideology scores and residual plot for BERT + CNN model. Residuals are distributed around zero with most errors within ± 0.2.

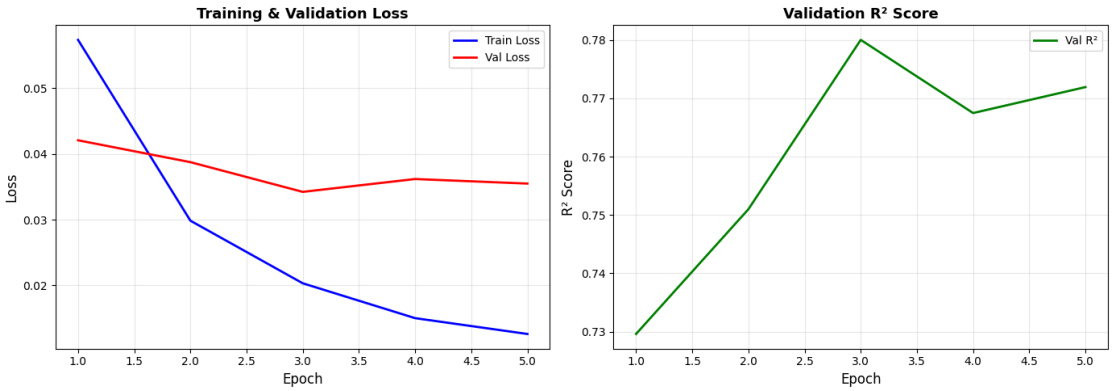

Figure 4.8: Training and validation loss curves for BERT + CNN. The model exhibits pronounced overfitting starting as early as epoch 2, where the validation loss begins diverging from the training loss. The validation R^2 curve displays greater volatility and instability compared to BERT + MLP (Figure 4.4) and BERT + LSTM (Figure 4.6), with noticeable fluctuations throughout training.

4.2 Graph-Based Models

Graph-based models exploit the structure of the retweet network, either through hand-crafted topological features or through graph neural networks that learn representations directly from the graph.

4.2.1 XGBoost on Graph Features

The `XGBoost` baseline for graph-based prediction uses the eleven graph features described in Section 3.3 as inputs to an `XGBRegressor` model. XGBoost [51] is an implementation of gradient boosted decision trees that achieves strong performance on tabular features through efficient second-order gradient optimisation and regularisation. The model is trained with 50 estimators, maximum depth 8, and learning rate 0.1. All features are standardised before training using StandardScaler, which proved critical for achieving good performance (models trained on unnormalized features achieved $R^2 \approx 0.11$ compared to $R^2 \approx 0.92$ with normalization).

4.2.2 GCN (Graph Convolutional Network)

The GCN [42] uses the eleven graph features as initial node features $\mathbf{x}_v \in \mathbb{R}^{11}$ and applies the graph convolution operation:

$$\mathbf{H}^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\,\tilde{D}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right) \tag{4.3}$$

where $\tilde{A} = A + I$ is the adjacency matrix with added self-loops, $\tilde{D}$ is the corresponding degree matrix, and $\mathbf{W}^{(l)}$ is a learnable weight matrix. The addition of self-loops ensures that each node includes itself in its neighborhood aggregation, which is necessary for stable training. Two GCN layers with hidden dimensions [128, 64] are used, followed by a linear regression head. Dropout 0.5 is applied between layers to reduce overfitting in the semi-supervised setting. The model is trained with learning rate 0.01 for up to 300 epochs with patience 40 for early stopping.

The residual analysis in Figure 4.9 demonstrates that GCN achieves substantially tighter error distributions than text-based models, with most residuals clustered very close to zero. This superior performance stems from the model's ability to aggregate ideological information from network neighborhoods through the graph convolution operation. The training dynamics shown in Figure 4.10 reveal rapid initial convergence as the model learns to exploit homophily patterns, followed by gradual refinement. The small gap between training and validation curves indicates effective generalization despite the model's capacity.

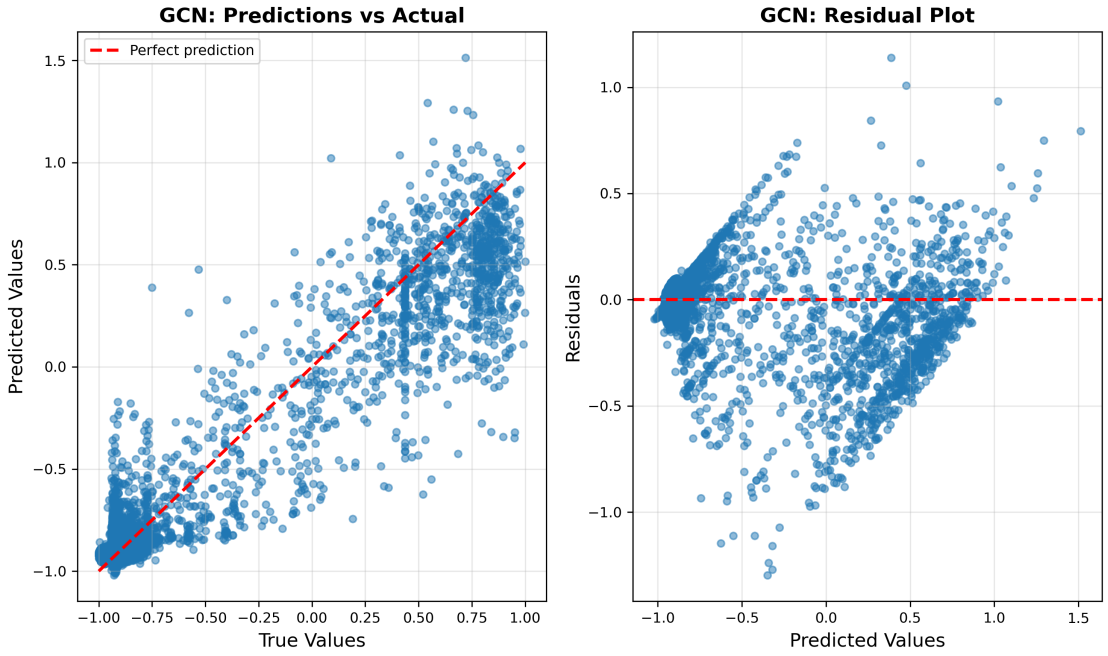

Figure 4.9: Predictions versus actual ideology scores and residual plot for Graph Convolutional Network (GCN) using graph-based features. Residuals are tightly clustered around zero with notably smaller variance than text-only models, indicating superior prediction accuracy. The GCN leverages network structure and homophily to propagate information across connected nodes, resulting in more accurate predictions. The tight residual distribution reflects the strong signal provided by average neighbor ideology (89% feature importance) and echo chamber patterns in the retweet network.

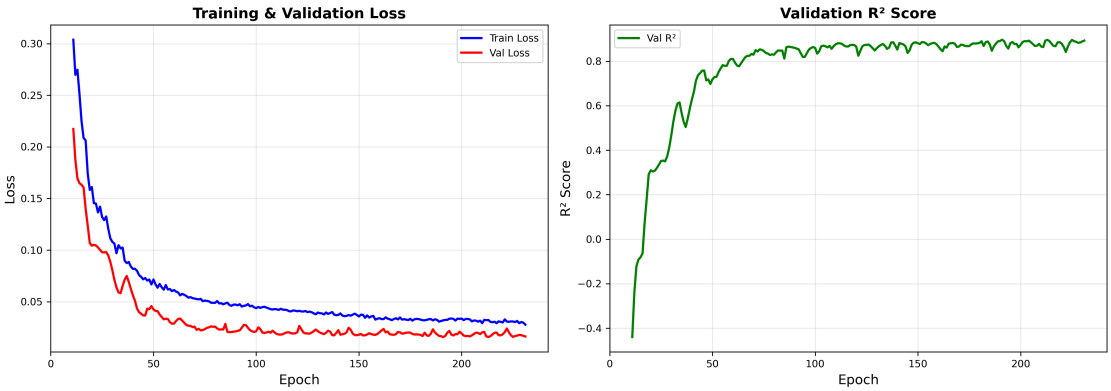

Figure 4.10: Training and validation loss curves for GCN. The curves show rapid convergence in early epochs as the model learns to aggregate neighbor information, followed by gradual refinement. The small gap between training and validation loss indicates effective generalization.

4.2.3 GraphSAGE

GraphSAGE [43] replaces the symmetric normalised convolution of GCN with a neighbourhood *sampling and aggregation* scheme. For each node v, a fixed-size sample $\mathcal{N}(v)$ of its neighbours is drawn, and their features are aggregated using a *mean aggregator*:

$$\mathbf{h}_v^{(l+1)} = \sigma\big(\mathbf{W}^{(l)} \cdot \mathrm{CONCAT}\big(\mathbf{h}_v^{(l)},\ \mathrm{MEAN}_{u \in \mathcal{N}(v)}\, \mathbf{h}_u^{(l)}\big)\big) \tag{4.4}$$

The inductive capability of GraphSAGE is particularly valuable here, as ideology estimation should generalise to users who join the debate after the model is trained. Two GraphSAGE layers with hidden dimensions [128, 64] are used, with dropout 0.3, learning rate 0.05, and patience 40 epochs for early stopping.

As illustrated in Figure 4.11, GraphSAGE achieves residual patterns similar to GCN, though with slightly larger variance due to the stochasticity introduced by neighbor sampling. Despite this, the model maintains excellent prediction accuracy. The training curves in Figure 4.12 show more variability than GCN (Figure 4.10) due to sampling-based gradient estimates, but overall convergence remains stable and reaches comparable final performance.

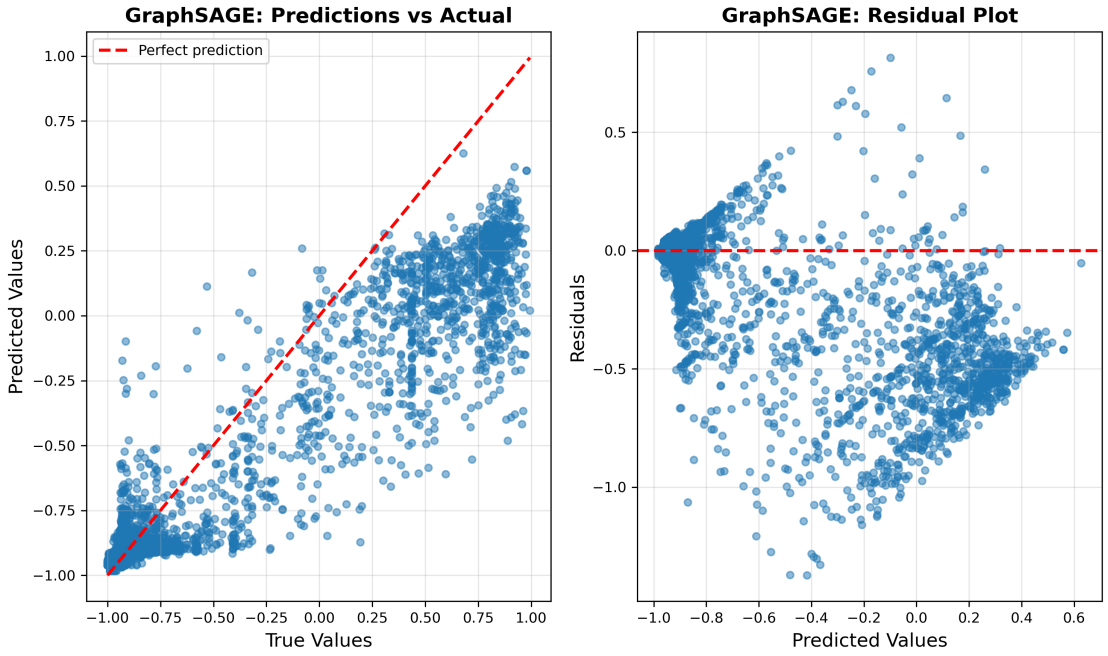

Figure 4.11: Predictions versus actual ideology scores and residual plot for GraphSAGE (Sample and Aggregate) model. Residuals show patterns similar to GCN (Figure 4.9) but with slightly larger variance, as sampling introduces stochasticity in neighbor aggregation. Despite this, GraphSAGE achieves comparable accuracy to GCN while being more computationally efficient for large-scale graphs. The residuals remain less centered at zero across the ideological spectrum.

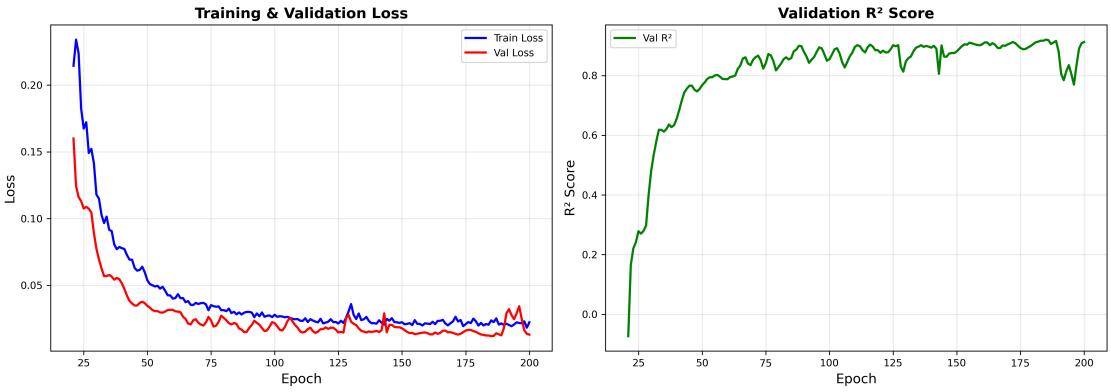

Figure 4.12: Training and validation loss curves for GraphSAGE. The sampling-based approach introduces variance in gradient estimates, leading to slightly noisier training curves compared to full-neighborhood GCN (Figure 4.10). However, the overall trend remains consistent, with both curves decreasing and converging. GraphSAGE's inductive capability (ability to generate embeddings for unseen nodes) makes it suitable for dynamic networks where new users continuously join. The final performance is comparable to GCN ($R^2 = 0.91$).

4.3 Hybrid Fusion Models

Hybrid models combine the BERT-extracted text embeddings ($\mathbf{t} \in \mathbb{R}^{768}$) with the eleven structural graph features ($\mathbf{g} \in \mathbb{R}^{11}$) for users present in both modalities. The motivation is that text and network features capture complementary aspects of ideology: text reveals what users say, while networks reveal whom they choose to align with behaviorally. We evaluate three fusion strategies—simple concatenation, attention-based weighting, and gated mechanisms—to determine whether combining modalities improves upon the strong graph-only baseline (XGBoost $R^2 = 0.92$).

4.3.1 Simple Concatenation Fusion

The simplest fusion approach concatenates the text embedding vector $\mathbf{t}$ with the graph feature vector $\mathbf{g}$ to form a combined representation $[\mathbf{t} \| \mathbf{g}] \in \mathbb{R}^{779}$. This concatenated vector is passed through two models.

First, XGBoost. The model is trained with 50 estimators, maximum depth 8, and learning rate 0.1, achieving the best performance out of all trained models ($R^2 \approx 0.92$).

Second, a two-layer MLP with hidden dimensions [256, 128], dropout 0.3, and a single linear output for ideology prediction. The model is trained end-to-end using AdamW optimizer with learning rate 10^{-4}, patience 5 for early stopping.

Simple concatenation allows the network to learn arbitrary combinations of text and graph signals but provides no explicit mechanism for handling conflicts between modalities or learning modality-specific patterns. The MLP must learn fusion patterns from scratch through gradient descent.

Figure 4.13 shows the predictions versus actual values and residual distribution. The model exhibits systematic bias, with predictions compressed toward the center of the ideology range and poor performance on extreme values. Multiple outliers with residuals indicate the model struggles with certain user types.

Figure 4.14 presents the training dynamics. The validation loss stabilizes quickly after epoch 5, but validation R^2 shows high volatility throughout training. This instability suggests the model has difficulty learning a consistent fusion strategy. The compression of predictions toward neutral values indicates that the simple concatenation approach may be averaging conflicting signals from text and graph rather than learning complementary patterns.

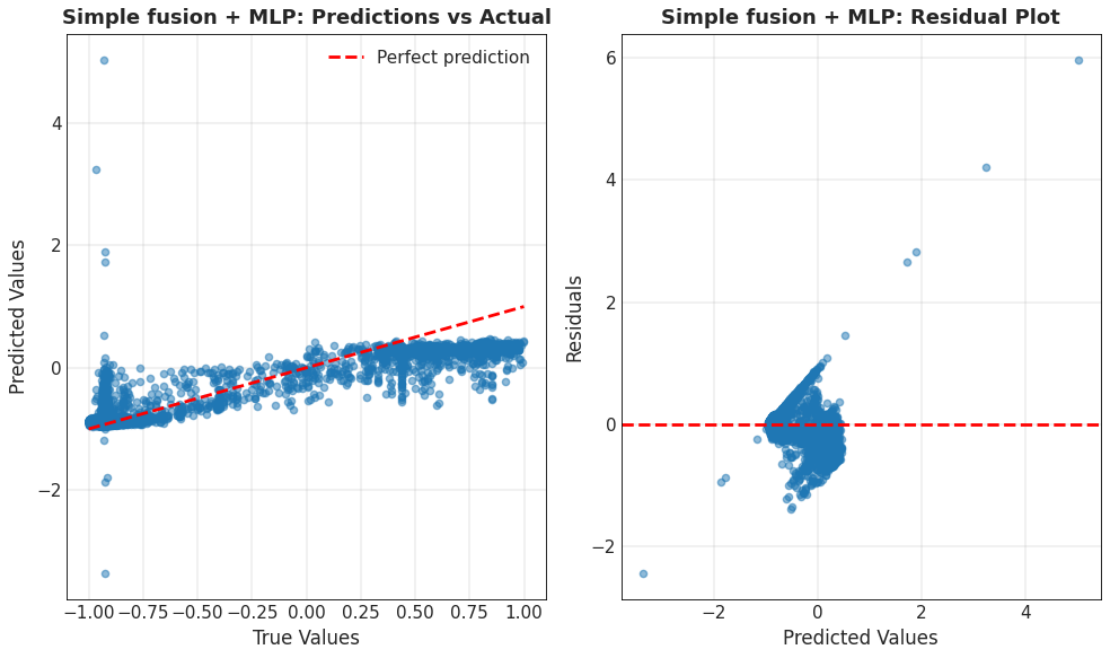

Figure 4.13: Predictions versus actual ideology scores (left) and residual distribution (right) for simple concatenation fusion. The model compresses predictions toward the center, with numerous outliers indicating poor handling of extreme ideological positions. Residuals show systematic bias with predictions rarely exceeding ± 0.5 despite true values spanning $[-1, 1]$.

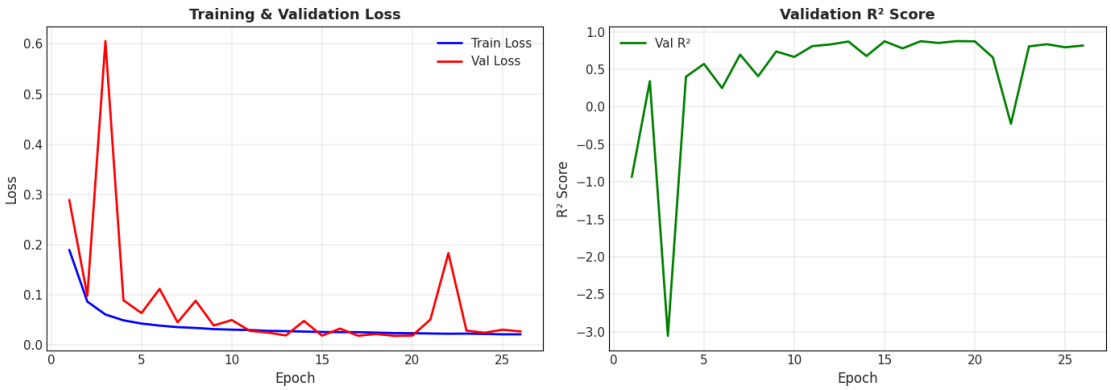

Figure 4.14: Training and validation loss (left) and validation R^2 score (right) for simple concatenation fusion across 27 epochs. Validation R^2 exhibits high volatility, indicating difficulty learning a stable fusion strategy. The spike in validation loss at epoch 20 suggests potential overfitting to noisy patterns.

4.3.2 Attention-Based Fusion

The `AttentionFusion` model implements a learned weighting mechanism to combine text and graph representations. First, both modalities are projected to a common dimension (256):

$$\mathbf{t}' = \mathbf{W}_t\mathbf{t}, \quad \mathbf{g}' = \mathbf{W}_g\mathbf{g} \tag{4.5}$$

Then, attention scores are computed for each modality:

$$\alpha_t, \alpha_g = \text{softmax}([\mathbf{W}_a\mathbf{t}', \mathbf{W}_a\mathbf{g}']) \tag{4.6}$$

where $\mathbf{W}_a$ is a learned attention parameter vector. The final representation is:

$$\mathbf{h} = \alpha_t\mathbf{t}' + \alpha_g\mathbf{g}' \tag{4.7}$$

This weighted combination is passed through an MLP for final prediction. The attention mechanism allows the model to learn instance-specific fusion strategies: for users who express clear ideological positions in text, attention weights favor the text representation; for users with ambiguous text but clear network affiliations, weights favor graph features. The model is trained with learning rate 10^{-4}, dropout 0.3, and patience 5.

Figure 4.15 shows improved prediction quality compared to simple fusion, with predictions spanning the full ideology range and fewer extreme outliers. The residual plot shows approximately centered distribution around zero with reduced systematic bias, though some heteroscedasticity remains (larger errors for extreme predicted values).

Figure 4.16 demonstrates stable training dynamics. Validation loss converges smoothly after epoch 5 and remains stable through epoch 18. Validation R^2 reaches approximately 0.90 by epoch 9 and plateaus with fluctuations between 0.85 and 0.91. The smooth convergence and stable performance indicate the attention mechanism successfully learns meaningful fusion weights, though the model does not surpass the graph-only baseline, suggesting that in this highly homophilic network (echo chamber score ≈ 0.95), graph features already capture most ideology-relevant information and text provides limited complementary signal.

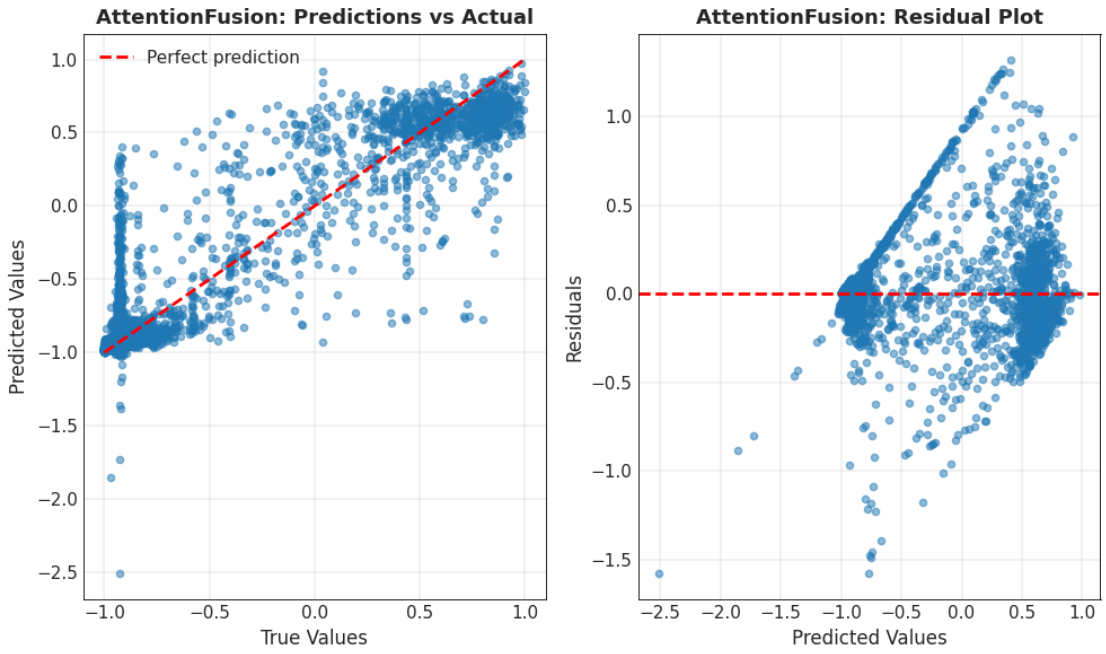

Figure 4.15: Predictions versus actual ideology scores (left) and residual distribution (right) for attention-based fusion. Compared to simple fusion, predictions span the full ideology range with improved alignment to the diagonal. Residuals are more centered around zero, though some heteroscedasticity persists for extreme values.

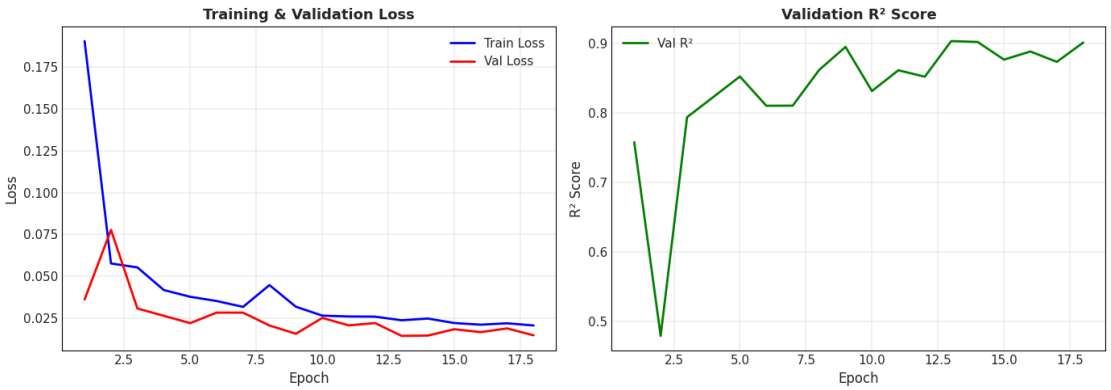

Figure 4.16: Training and validation loss (left) and validation R^2 score (right) for attention fusion across 18 epochs. Validation R^2 stabilizes at approximately 0.90 with minimal oscillation, indicating successful learning of instance-specific fusion weights.

4.3.3 Gated Fusion

Gated fusion uses a gating mechanism inspired by LSTM cells to control information flow from each modality:

$$\mathbf{z}_t = \sigma(\mathbf{W}_{zt}\mathbf{t} + \mathbf{b}_{zt}), \quad \mathbf{z}_g = \sigma(\mathbf{W}_{zg}\mathbf{g} + \mathbf{b}_{zg}) \tag{4.8}$$

where $\mathbf{z}_t, \mathbf{z}_g \in [0, 1]$ are gate values controlling how much information from each modality to pass through. Both modalities are first projected to a common hidden dimension (256). The gated representations are:

$$\mathbf{h} = \mathbf{z}_t \odot \mathbf{W}_t\mathbf{t} + \mathbf{z}_g \odot \mathbf{W}_g\mathbf{g} \tag{4.9}$$

where $\odot$ denotes element-wise multiplication. This gated combination is then processed by a prediction MLP. The gating mechanism allows the model to filter or emphasize different features based on their reliability for each instance. The model is trained with learning rate 10^{-4}, dropout 0.3, and patience 5.

Figure 4.17 shows excellent prediction quality across the full ideology spectrum. Predictions closely follow the diagonal with minimal systematic bias (outliers). The residual plot displays a well-centered distribution around zero with reduced heteroscedasticity compared to attention fusion, indicating more consistent error magnitude across the prediction range.

Figure 4.18 reveals remarkably stable training after few first epochs. Validation loss drops rapidly in the first 5 epochs and remains near-constant thereafter. Most strikingly, validation R^2 reaches approximately 0.92 by epoch 5 and maintains stability through epoch 33, with only a brief dip around epoch 8. This exceptional stability suggests the gating mechanism effectively learns to selectively incorporate text information only when it provides signal beyond what graph features already capture. The final test performance of $R^2 = 0.92$ is the strongest hybrid performance (between MLP-based models), demonstrating that learnable gates provide more effective fusion than fixed attention weights.

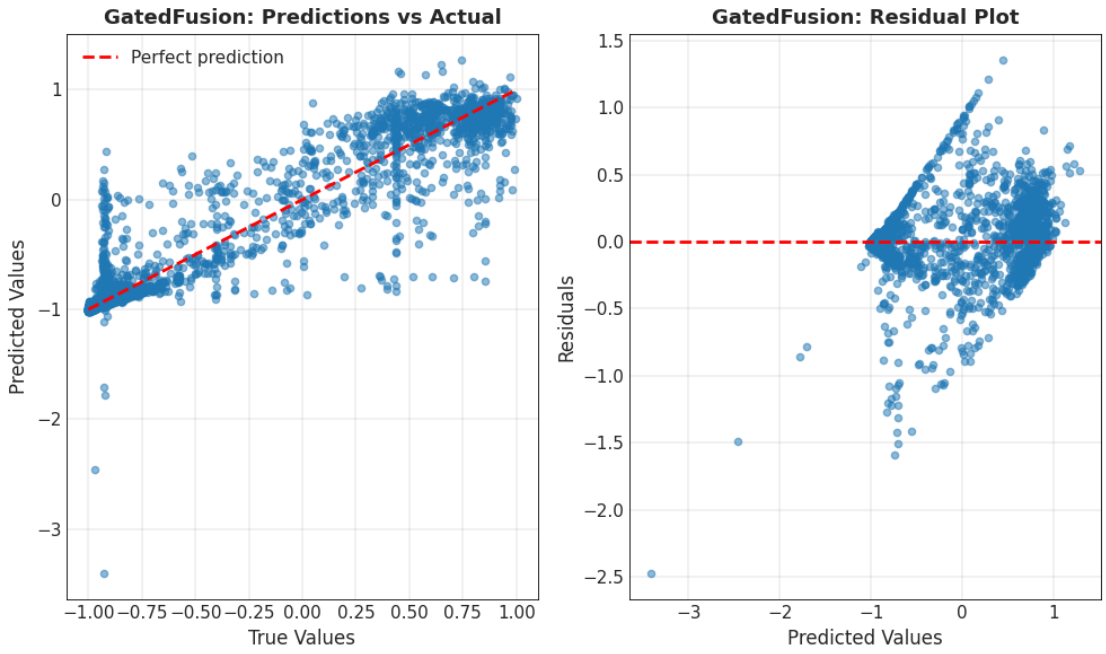

Figure 4.17: Predictions versus actual ideology scores (left) and residual distribution (right) for gated fusion. Predictions closely follow the diagonal across the full ideology range. Residuals are tightly centered around zero with reduced heteroscedasticity, demonstrating the most consistent performance among all hybrid models.

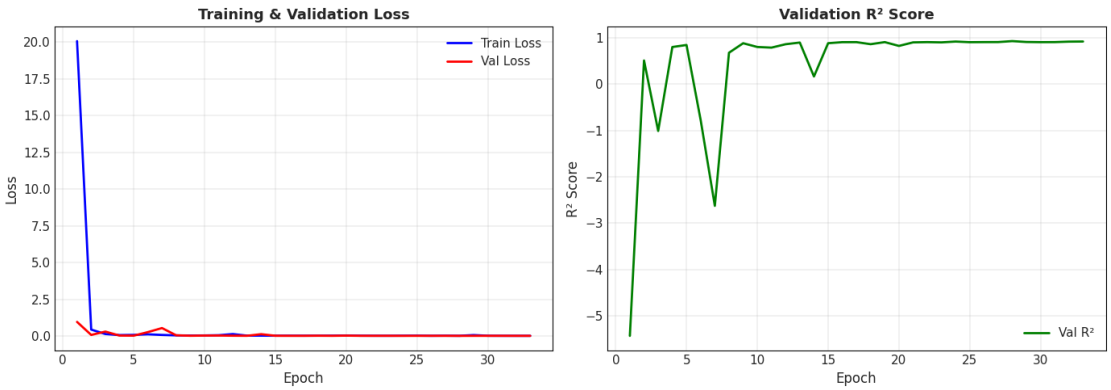

Figure 4.18: Training and validation loss (left) and validation R^2 score (right) for gated fusion across 33 epochs. Validation R^2 reaches 0.92 by epoch 5 and remains remarkably stable, with only a dip at epoch 8. This exceptional stability indicates effective gating that selectively incorporates text information when complementary to graph features.

4.4 Training Infrastructure

All deep learning models were implemented in PyTorch and PyTorch Geometric (PyG). GNN models were trained on the full graph using PyG's `Data` object with node-level train/test masks. Hybrid and text models were trained on mini-batches using PyTorch's `DataLoader`. All experiments were run on GPU (CUDA) when available. The random seed was fixed at 28 for all experiments to ensure reproducibility.

All features were standardized using `StandardScaler` fitted on the training set and applied to both training and test sets, which proved critical for neural network performance. For hybrid models, text embeddings and graph features were standardized separately before concatenation. Self-loops were added to the adjacency matrix for GCN models. Early stopping was employed for all neural models to prevent overfitting.

5

Results and Analysis

This chapter presents the experimental results for all models evaluated on the ideology prediction task. We organize the discussion by model category: text-based models (Section 5.2), graph-based models (Section 5.3), and hybrid multimodal approaches (Section 5.4). For each category, we report performance metrics, analyze patterns in predictions, and discuss the implications of our findings.

All models were trained using a 80/20 train/test split. The primary evaluation metric is R^2 (coefficient of determination), which measures the proportion of variance in ideology scores explained by the model. We also report RMSE (root mean squared error), MSE (mean squared error), and MAE (mean absolute error) for completeness.

5.1 Overall Performance Comparison

Table 5.1 presents a comprehensive comparison of all models evaluated in this work. The results reveal clear performance differences across model categories and fusion strategies, with XGBoost on concatenated hybrid features achieving the best overall performance.

Category	Model	R^2	RMSE	MSE	MAE
Text-Based	TF-IDF + Ridge Regression	0.7342	0.2034	0.1101	0.0414
Text-Based	BERT-MLP (Frozen)	0.2393	0.3441	0.2043	0.1184
Text-Based	BERT-MLP (Fine-tuned)	0.7816	0.1844	0.0738	0.0340
Text-Based	BERT-LSTM	**0.7837**	**0.1835**	**0.0724**	**0.0337**
Text-Based	BERT-CNN	0.7719	0.1884	0.1049	0.0355
Graph-Based	Ridge Regression	0.7984	0.1788	0.0320	0.0704
Graph-Based	XGBoost	**0.9152**	**0.1159**	**0.0134**	**0.0346**
Graph-Based	GCN	0.9101	0.1162	0.0135	0.0590
Graph-Based	GraphSAGE	0.9099	0.1161	0.0135	0.0697
Hybrid	XGBoost + Concatenated Features	**0.9373**	**0.0968**	**0.0094**	**0.0261**
Hybrid	Gated Fusion (Neural)	0.9212	0.1085	0.0118	0.0361
Hybrid	Attention Fusion (Neural)	0.9051	0.1191	0.0142	0.0459
Hybrid	Simple Fusion (MLP)	0.8189	0.1645	0.0271	0.0725

Table 5.1: Complete performance comparison across all thirteen models grouped by category (Text-Based, Graph-Based, Hybrid). Key findings: (1) The best overall model is XGBoost trained on concatenated text and graph features (R^2 = 0.9373), demonstrating that traditional machine learning with explicit features outperforms end-to-end neural fusion. (2) Among neural hybrid models, Gated Fusion achieves the strongest performance (R^2 = 0.9212), while simple MLP concatenation fails (R^2 = 0.8189), highlighting the importance of sophisticated fusion mechanisms in neural architectures. (3) XGBoost successfully exploits concatenated features where MLP fails, suggesting gradient boosting's superior ability to handle heterogeneous feature spaces. (4) Graph-based models consistently outperform text-based models, with the gap ranging from 13-15 percentage points in R^2.

The most striking finding is that XGBoost trained on concatenated text embeddings and graph features achieves the best overall performance ($R^2 = 0.9373$), outperforming both single-modality baselines and neural hybrid models. This 2.2 percentage point improvement over graph-only XGBoost ($R^2 = 0.9152$) demonstrates that text and graph modalities do provide complementary information, but that **the fusion strategy matters critically**. XGBoost succeeds with simple concatenation where a neural MLP fails ($R^2 = 0.8189$), revealing fundamental differences in how these architectures handle multimodal features.

The success of XGBoost with concatenated features can be attributed to several factors. First, gradient boosting excels at learning complex non-linear interactions between heterogeneous features, automatically discovering which combinations of text and graph signals are most informative. Second, XGBoost's tree-based structure naturally handles features of different scales

and types without requiring careful architectural design for fusion. Third, the model's built-in regularization and feature selection mechanisms prevent overfitting to redundant signals, allowing it to extract the complementary information that text provides beyond graph features.

In contrast, the neural MLP with simple concatenation struggles to learn effective fusion. The dimensional imbalance (768-dimensional BERT embeddings vs. 11 graph features) likely causes the network to rely too heavily on one modality or average conflicting signals inappropriately. This highlights a key insight: **for traditional machine learning approaches, simple concatenation suffices when combined with a powerful learner like XGBoost; for neural networks, sophisticated fusion architectures (gating, attention) are essential**.

Within the text-based category, there is considerable variation. The frozen BERT baseline ($R^2 = 0.2393$) performs poorly, suggesting that generic language representations without task-specific adaptation are insufficient. Fine-tuning BERT improves performance dramatically to $R^2 = 0.7816$, a 226% increase, highlighting the importance of domain adaptation.

Among graph-based models, XGBoost ($R^2 = 0.9152$) outperforms graph neural networks (GCN and GraphSAGE, both $R^2 \approx 0.91$) by 0.5 percentage points. This suggests that for this dataset, explicit feature engineering combined with gradient boosting is more effective than end-to-end GNN learning.

Among neural hybrid models, performance varies dramatically based on fusion strategy. Gated Fusion achieves $R^2 = 0.9212$, approaching the XGBoost hybrid baseline through element-wise gating that selectively incorporates complementary signals. Attention Fusion reaches $R^2 = 0.9051$, demonstrating that instance-specific weighting helps but is less effective than fine-grained gating. Simple MLP concatenation fails catastrophically ($R^2 = 0.8189$), performing worse than either single-modality baseline and confirming that neural networks require explicit fusion mechanisms to combine heterogeneous modalities effectively.

The performance hierarchy establishes that: (1) XGBoost on hybrid features > (2) XGBoost on graph features ≈ Gated Fusion > (3) GCN/GraphSAGE ≈ Attention Fusion > (4) Simple MLP Fusion > (5) Text-only models. This ordering reveals that both the base learner (XGBoost vs. neural network) and the fusion strategy (concatenation only works for XGBoost, not MLPs) jointly determine multimodal performance.

5.2 Text-Based Model Results

5.2.1 Model Performance

Text-based models attempt to infer ideology from the content users produce, including tweets, biographical descriptions, and profile information. Table 5.2 shows the performance of five text-based approaches.

Model	R^2	RMSE	MSE	MAE
TF-IDF + Ridge	0.7342	0.2034	0.1101	0.0414
BERT-MLP (Frozen)	0.2393	0.3441	0.2043	0.1184
BERT-MLP (Fine-tuned)	0.7816	0.1844	0.0738	0.0340
BERT-LSTM	**0.7837**	**0.1835**	**0.0724**	**0.0337**
BERT-CNN	0.7719	0.1884	0.1049	0.0355

Table 5.2: Performance comparison of text-based models on the test set. All models use BERT for text encoding but differ in the architecture applied to BERT embeddings. Metrics include R^2 (coefficient of determination, primary metric), MSE (mean squared error), MAE (mean absolute error), and RMSE (root mean squared error). BERT + LSTM achieves the best text-only performance ($R^2 = 0.784$), followed closely by fine-tuned BERT + MLP ($R^2 = 0.782$) and BERT + CNN ($R^2 = 0.772$). TF-IDF + Ridge serves as baseline ($R^2 = 0.734$).

BERT Embeddings: Frozen vs. Fine-tuned

The comparison between frozen and fine-tuned BERT embeddings reveals a critical insight: task-specific adaptation is essential for ideology detection. The frozen BERT-MLP model achieves only $R^2 = 0.2393$, explaining less than 24% of ideological variance. This poor performance occurs because BERT was pre-trained on general text (Wikipedia, BookCorpus) for masked language modeling and next sentence prediction - tasks unrelated to political ideology. The resulting embeddings capture general semantic and syntactic properties but miss the subtle linguistic patterns that distinguish ideological positions.

Fine-tuning the entire BERT encoder on the ideology prediction task improves performance to $R^2 = 0.7816$, a 226% increase. During fine-tuning, BERT learns to pay attention to politically relevant features: specific vocabulary choices, framing patterns, and rhetorical devices associated with different ideological camps. This adaptation allows the model to capture nuanced differences in how supporters and opponents of Ukraine aid express their views.

The substantial gap between frozen and fine-tuned BERT underscores a key limitation of transfer learning: pre-trained representations are not universally effective. Domain adaptation

through fine-tuning is necessary when the target task requires specialized knowledge or when relevant patterns were not emphasized during pre-training.

Architectural Variations

Among models using fine-tuned BERT embeddings, we observe modest performance differences across architectures. BERT-LSTM achieves the best results ($R^2 = 0.7837$), followed closely by BERT-MLP ($R^2 = 0.7816$) and BERT-CNN ($R^2 = 0.7719$). The improvements from using LSTM or CNN architectures are small (0.3-1.2 percentage points), suggesting that most ideologically relevant information is already captured by fine-tuned BERT embeddings.

The LSTM architecture performs best, likely because it can model sequential dependencies in the concatenated text (biographical description, location, tweets). However, the marginal improvement over MLP indicates that for aggregated user-level text, sequential patterns provide limited additional signal. This contrasts with document-level tasks where sentence ordering and discourse structure are critical.

The BERT-CNN model achieves comparable performance to BERT-LSTM while being computationally more efficient due to the parallelizable nature of convolutions. CNNs are effective at detecting local patterns such as specific phrases or terminology, but these patterns are largely captured by BERT's attention mechanism, limiting CNN's added value.

TF-IDF Baseline Performance

Interestingly, the simple TF-IDF + Ridge regression baseline achieves $R^2 = 0.7342$, competitive with sophisticated neural architectures. This result demonstrates that traditional bag-of-words approaches remain effective for ideology detection when the domain vocabulary is strongly associated with political positions. Certain terms and phrases are reliable indicators of ideology, and linear models can learn these associations from TF-IDF representations.

However, TF-IDF + Ridge underperforms fine-tuned BERT-based models by 4-6 percentage points, suggesting that contextual representations do provide benefits. BERT captures semantic nuances and understands context-dependent usage, all of which contribute to improved performance. Nevertheless, the competitive baseline performance indicates that ideology detection from text is partly a lexical task - certain words are diagnostic of political positions regardless of context.

5.3 Graph-Based Model Results

5.3.1 Model Performance

Graph-based models leverage network structure and graph features to predict ideology. Table 5.3 shows that these approaches substantially outperform text-based methods.

Model	R^2	RMSE	MSE	MAE
Ridge Regression (graph features)	0.7984	0.1788	0.0320	0.0704
XGBoost (graph features)	**0.9152**	**0.1159**	**0.0134**	**0.0346**
GCN	0.9101	0.1162	0.0135	0.0590
GraphSAGE	0.9099	0.1161	0.0135	0.0697

Table 5.3: Performance comparison of graph-based models using network structure and graph features only. All graph models substantially outperform text-only approaches, confirming the strong signal provided by network homophily. XGBoost achieves the best graph-only performance ($R^2 = 0.9152$) through explicit feature engineering and gradient boosting. GraphSAGE and GCN show comparable performance ($R^2 \approx 0.91$), with GraphSAGE offering better scalability through neighbor sampling.

The most striking result is that all graph-based models achieve $R^2 > 0.90$, explaining over 90% of ideological variance. This represents a substantial improvement over text-based models ($R^2 = 0.78$), confirming that network structure provides stronger signals about political ideology than textual content.

XGBoost with Graph Features

XGBoost with handcrafted graph features achieves the best graph-only performance ($R^2 = 0.9152$), outperforming graph neural networks by 0.5 percentage points. This model uses the eleven graph-based features described in Chapter 3, including degree centrality measures, PageRank, clustering coefficients, community membership, and ideology-specific homophily metrics.

The success of this approach demonstrates that explicit feature engineering combined with gradient boosting is highly effective for graph-based ideology prediction. XGBoost's ability to model non-linear interactions between features allows it to capture complex patterns. For instance, the combination of high clustering coefficient with specific neighbor ideology values may be more informative than either feature alone.

The superior performance of XGBoost over graph neural networks is noteworthy. While GNNs can learn feature representations end-to-end, they may require more data or more careful tuning to match the performance of explicitly engineered features. For datasets of this size (76,993 labeled nodes), domain knowledge incorporated through feature engineering combined with a powerful tree-based learner appears more effective than purely learned representations.

Importantly, XGBoost on graph features serves as a strong baseline for evaluating hybrid approaches. The fact that adding text features through simple concatenation improves XGBoost performance to $R^2 = 0.9373$ (a 2.2 percentage point gain) demonstrates that text does provide complementary information beyond graph features, though the improvement is modest given the network's extreme homophily.

Graph Neural Networks

Both GCN and GraphSAGE achieve comparable performance ($R^2 = 0.9101$ and $R^2 = 0.9099$ respectively), demonstrating that different GNN architectures converge to similar effectiveness on this task. These models learn node representations by aggregating information from network neighborhoods, allowing them to leverage both node features (the eleven graph metrics) and netxwork structure (the retweet graph).

The near-identical performance of GCN and GraphSAGE is interesting. GCN uses all neighbors for aggregation, while GraphSAGE samples a fixed number of neighbors. The similarity in results suggests that sampling does not significantly harm performance for this graph, possibly because the most informative neighbors (those with strong ideological signals) are likely to be included even in sampled neighborhoods.

One advantage of GraphSAGE over GCN is computational efficiency. For large graphs, sampling makes training feasible, while full-neighborhood aggregation becomes prohibitively expensive. The fact that GraphSAGE matches GCN performance while being more scalable makes it attractive for deployment on even larger Twitter networks.

More sophisticated GNN architectures exist, such as Graph Attention Networks (GAT), which learn to weight neighbor contributions adaptively rather than treating all neighbors equally. However, for our network scale (2.6 million users, 4.3 million edges), GAT's attention mechanism requires storing attention coefficients for all edges, creating prohibitive memory requirements. Given that GCN and GraphSAGE already achieve strong performance ($R^2 > 0.91$) and our network exhibits extreme homophily (echo chamber score = 0.95), where even uniform neighbor aggregation works well, the additional computational overhead of attention

mechanisms was not justified for this application.

5.3.2 Feature Importance Analysis

To understand which graph features contribute most to ideology prediction accuracy, we analyze feature importance using two complementary approaches: (1) XGBoost's built-in gain-based feature importance, which measures the average gain in predictive accuracy from splits using each feature, and (2) SHAP (SHapley Additive exPlanations) values, which quantify each feature's contribution to individual predictions while accounting for feature interactions. Together, these methods provide both global importance rankings and insight into how feature values directionally influence predictions.

XGBoost Feature Importance

Figure 5.1 shows the relative importance of all eleven graph features as measured by XGBoost's gain metric. The results reveal an extremely clear hierarchy, with ideology-based features dramatically outperforming structural features.

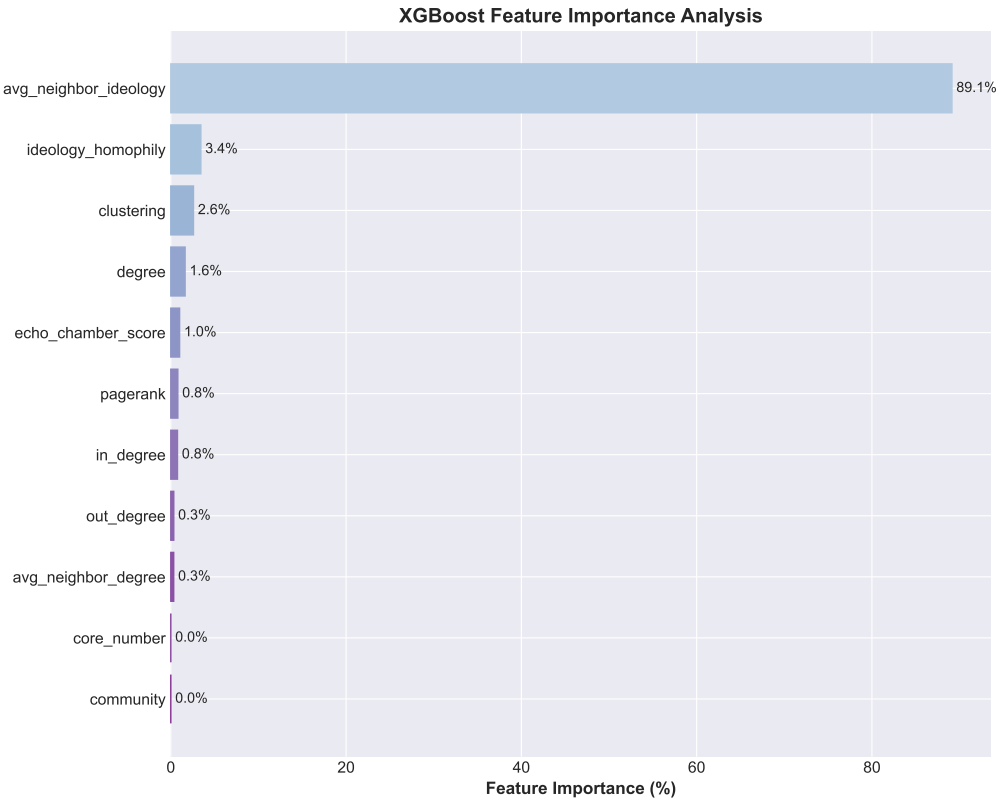

Figure 5.1: Feature importance scores from the XGBoost model, measured as the percentage contribution to total predictive gain across all trees. Features are ranked by importance, with avg_neighbor_ideology overwhelmingly dominant at 89.1%. Ideology-based features (avg_neighbor_ideology, ideology_homophily, echo_chamber_score) collectively account for 93.5% of total importance, while structural features contribute only 6.5%. Community and core number features show zero importance, likely due to computational issues during feature extraction on the full 2.59M-node network.

The feature importance hierarchy is as follows:

1. **Average neighbor ideology (89.1%):** This single feature dominates the model, accounting for nearly 90% of all predictive power. This overwhelming importance directly reflects the strong homophily documented earlier (Pearson $r = 0.875$ between own and neighbor ideology). The average ideological position of a user's network neighbors is by far the strongest signal of their own position, confirming that ideology inference in this network is fundamentally a homophily-driven process.

2. **Ideology homophily (3.4%):** The second most important feature, though contributing only 3.4%, captures similarity to neighbors. While related to average neighbor ideology, this feature provides complementary information by measuring the degree of ideological alignment (continuous similarity) rather than absolute neighbor positions. Its modest contribution suggests it offers some refinement beyond what average neighbor ideology already captures.

3. **Clustering coefficient (2.6%):** Local network density contributes modestly to predictions. Users embedded in denser local neighborhoods may exhibit different ideological patterns than those in sparse regions, though this effect is small compared to direct homophily measures.

4. **Degree (1.6%):** Total connectivity provides weak but non-zero signal. Highly connected users may have different ideological distributions than isolated users, though this structural property is far less informative than neighborhood ideology composition.

5. **Echo chamber score (1.0%):** Surprisingly, this measure of same-polarity neighbor proportion contributes only 1.0%, despite its high mean value (0.95 among labeled users). This likely reflects redundancy with average neighbor ideology: if avg_neighbor_ideology already captures ideological similarity, the binary echo chamber metric (based on sign only) adds little additional information.

6. **Other structural features (PageRank, in-degree, out-degree, avg_neighbor_degree): 0.8–0.3%:** These global and local centrality measures contribute negligibly. Their low importance indicates that a user's influence level or degree of activity in the network provides almost no independent signal about ideology beyond what is captured by neighborhood composition.

7. **Community and core number (0.0%):** These features show exactly zero importance. For core number, this reflects the computational issue documented in Table 3.4, where the feature was constant across all nodes. For community ID, the zero importance is more surprising given theoretical expectations that communities should exhibit ideological coherence. However, this likely indicates that community membership provides no additional signal beyond what avg_neighbor_ideology already captures: if communities are defined by connection patterns, and connections are driven by homophily, then community labels become redundant once neighborhood ideology is known.

The dominance of `avg_neighbor_ideology` is striking and theoretically meaningful. It confirms that ideology inference in this polarized retweet network reduces primarily to a homophily-based label propagation problem: knowing the ideological composition of a user's immediate neighborhood provides sufficient information to predict their own position with high accuracy (as demonstrated by the models. Structural properties of the network - how connected, influential, or central users are - matter far less than who they connect to.

SHAP Feature Importance and Impact Analysis

To complement the gain-based importance scores, we employ SHAP values, which offer two advantages: (1) they account for feature interactions and provide a more nuanced importance

ranking, and (2) they reveal the direction of each feature's effect on predictions. Figure 5.2 shows mean absolute SHAP values, representing each feature's average contribution to prediction magnitude, while Figure 5.3 displays the distribution of SHAP values across individual predictions, colored by feature values.

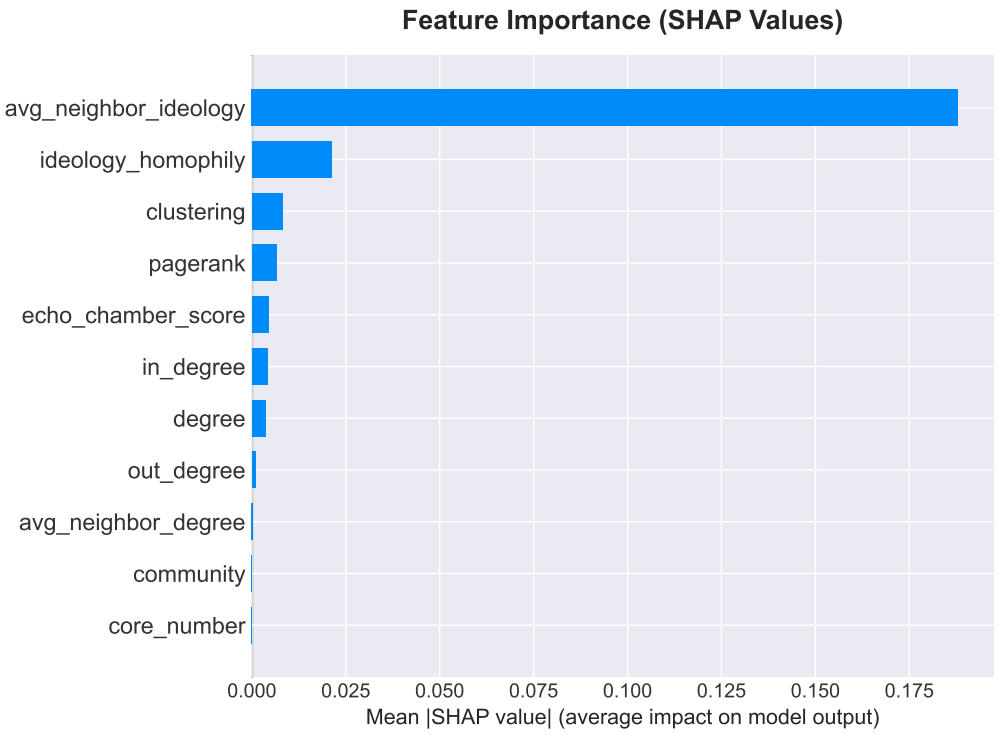

Figure 5.2: Mean absolute SHAP values across all predictions, measuring each feature's average contribution to model output magnitude. The ranking confirms avg_neighbor_ideology as the dominant feature, with ideology_homophily and clustering providing secondary contributions. The ordering largely matches XGBoost's gain-based importance, validating the consistency of importance metrics across methods.

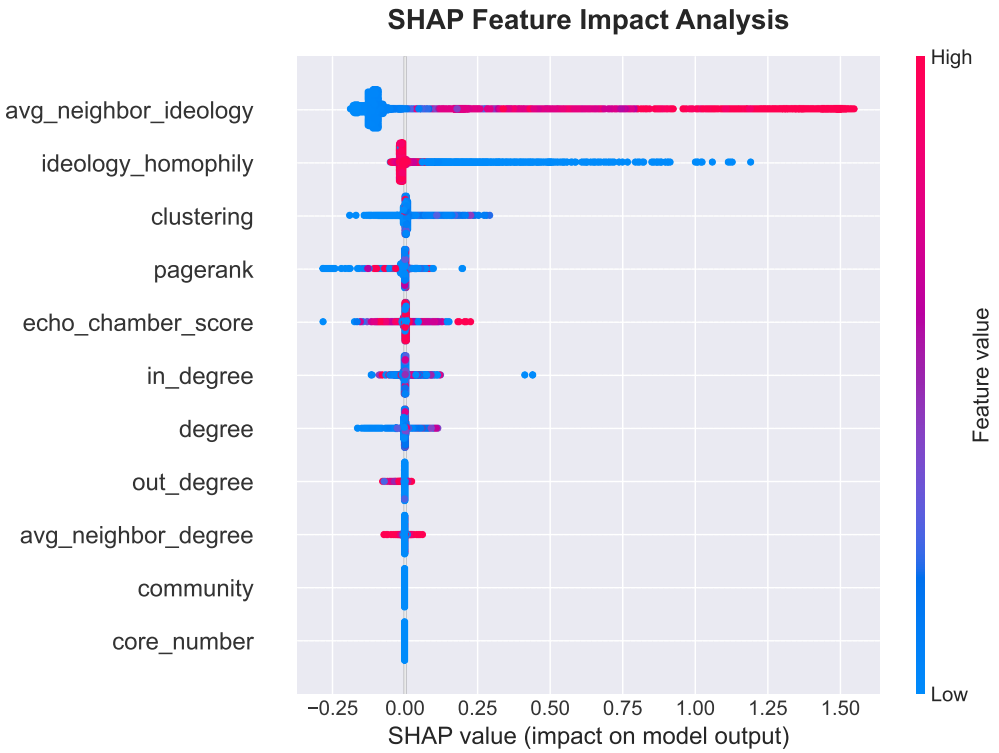

Figure 5.3: Beeswarm plot showing the distribution of SHAP values for each feature. Each point represents one prediction, colored by feature value (red = high, blue = low). For avg_neighbor_ideology (top row), the pattern is clear: high neighbor ideology (red) → positive SHAP → prediction pushed toward opponent side; low neighbor ideology (blue) → negative SHAP → prediction pushed toward supporter side. This demonstrates how the model uses neighborhood composition to infer ideology.

The SHAP bar plot (Figure 5.2) largely confirms the XGBoost importance ranking, with avg_neighbor_ideology far exceeding all other features in mean absolute impact. The consistency between gain-based and SHAP-based importance rankings validates the robustness of our importance estimates.

The SHAP beeswarm plot (Figure 5.3) reveals additional insights about how features influence predictions:

- **avg_neighbor_ideology:** Shows a clear positive relationship between feature value and SHAP value. High neighbor ideology (red points, indicating pro-opponent neighbors) produces large positive SHAP values, pushing predictions toward the opponent side. Low neighbor ideology (blue points, pro-supporter neighbors) produces large negative SHAP values, pushing predictions toward the supporter side. This linear relationship with substantial magnitude confirms the feature's dominance.

- **ideology_homophily:** Displays a more complex pattern, with high values (less negative, indicating greater similarity to neighbors) tending toward positive SHAP values. This suggests that users who are more ideologically similar to their neighbors receive predictions shifted toward their neighborhood's position, reinforcing the homophily-driven inference.

- **Structural features (clustering, degree, pagerank):** Show compressed SHAP distributions near zero, indicating minimal individual impact. The small spread and low magnitude confirm these features' limited contribution to predictions, consistent with the importance rankings.

- **echo_chamber_score:** Despite its high mean value in the data (0.95), shows limited SHAP impact with compressed distribution. This confirms redundancy with avg_neighbor_ideology: the continuous measure of neighbor ideology subsumes the binary same-polarity signal captured by echo chamber score.

The Primacy of Homophily

The feature importance analysis reveals a clear and theoretically significant finding: **ideology inference in this polarized political network is overwhelmingly driven by homophily**, specifically the ideological composition of users' immediate network neighborhoods. The avg_neighbor_ideology feature alone accounts for 89.1% of the model's predictive power, while all structural features combined (degree, PageRank, clustering, core number) contribute only 6.5%.

This dominance has several important implications:

1. **Homophily as the core mechanism:** Political ideology in retweet networks propagates primarily through homophilic connections. Users reveal their ideological positions not through their structural role in the network (influence, connectivity, community membership) but through their choice of whom to connect with. The near-perfect predictability from neighborhood composition ($R^2 = 0.92$) suggests that retweet behavior is strongly selective along ideological lines.

2. **Redundancy of community detection:** The zero importance of community features, despite strong theoretical expectations, indicates that explicit community detection adds no value beyond what is captured by local neighborhood ideology. In networks with strong homophily, communities naturally emerge as clusters of like-minded users, making community labels redundant once individual connections are known. This suggests

that for ideology inference specifically, computationally expensive community detection algorithms may be unnecessary - neighborhood-based features suffice.

3. **Limited role of network position:** Structural centrality measures (degree, PageRank) contribute negligibly to ideology prediction. This contrasts with influence detection or information diffusion tasks, where centrality is paramount. For ideology, being highly connected or influential provides little signal independent of who one connects to. A highly followed user with pro-supporter neighbors is predictable as a supporter; influence level alone reveals nothing.

4. **Implications for methodology:** These results suggest that for ideology inference in politically polarized networks, sophisticated structural analysis may be less critical than accurate measurement of neighborhood ideological composition. Resources are better invested in maximizing label coverage (to expand the number of users with labeled neighbors) than in computing complex structural features. Graph-based inference reduces essentially to a neighborhood averaging problem weighted by homophily strength.

The striking simplicity of the result - that one feature captures 89% of predictive signal - should not obscure its substantive importance. It provides quantitative confirmation of the echo chamber structure of political discourse on social media during polarizing events. Users' network neighborhoods so strongly predict their ideological positions because those neighborhoods are constructed through homophilic selection: people retweet those they agree with, creating a network where connection patterns encode ideology with high fidelity. The XGBoost model, in learning to predict ideology from graph features, has essentially learned to exploit this homophilic structure, with avg_neighbor_ideology serving as a sufficient statistic for the network's ideological signal.

5.3.3 Comparison with Text-Based Models

Graph-based models outperform text-based models by 13-15 percentage points in R^2 (0.91-0.94 vs. 0.78). This performance gap has important implications for understanding how ideology is expressed and manifested in social media.

Network affiliations appear to be more reliable indicators of ideology than textual content. This makes sense from a theoretical perspective: users may moderate their language to maintain plausible deniability or appeal to broader audiences, but their choice of whom to retweet, a form of endorsement, reveals their true alignments. Retweeting is a behavioral action that signals agreement and amplifies particular voices, making it a stronger commitment than merely expressing an opinion in text.

Additionally, network features are more robust to adversarial manipulation. A user can easily change their vocabulary or framing, but altering their network position requires building new connections and abandoning old ones - a more costly and time-consuming process. This makes network-based inference harder to game than text-based inference.

The strong performance of graph-based models also validates decades of social network theory emphasizing homophily as a fundamental organizing principle. The principle that "birds of a feather flock together" is empirically confirmed in digital networks, where users self-organize into ideologically homogeneous clusters through their retweeting behavior.

5.4 Hybrid Model Results

Hybrid models combine text and graph features to leverage complementary information from both modalities. We evaluate four approaches: (1) XGBoost trained on concatenated features (traditional ML), and three neural fusion strategies: (2) simple MLP concatenation, (3) attention-based weighting, and (4) gated fusion. Table 5.4 reports performance for all hybrid models.

Model	R^2	**RMSE**	**MSE**	**MAE**
XGBoost + Concatenated Features	**0.9373**	**0.0968**	**0.0094**	**0.0261**
Gated Fusion (Neural)	0.9212	0.1085	0.0118	0.0361
Attention Fusion (Neural)	0.9051	0.1191	0.0142	0.0459
Simple Fusion (MLP)	0.8189	0.1645	0.0271	0.0725

Table 5.4: Performance of hybrid models combining text embeddings (768-dimensional BERT) and graph features (11 dimensions). XGBoost trained on concatenated features achieves the best performance (R^2 = 0.9373), outperforming all neural fusion strategies. Among neural models, Gated Fusion (R^2 = 0.9212) and Attention Fusion (R^2 = 0.9051) demonstrate that sophisticated fusion mechanisms enable effective multimodal learning, while simple MLP concatenation fails (R^2 = 0.8189). The contrast between XGBoost's success and MLP's failure with concatenation reveals that gradient boosting naturally handles heterogeneous features, while neural networks require explicit architectural support for fusion.

5.4.1 XGBoost on Concatenated Features: Best Overall Performance

The most significant finding is that XGBoost trained on concatenated text embeddings and graph features achieves $R^2 = 0.9373$ - the best performance across all models evaluated in this work. This 2.2 percentage point improvement over graph-only XGBoost ($R^2 = 0.9152$) and

1.6 percentage point gain over the best neural hybrid model (Gated Fusion, $R^2 = 0.9212$) demonstrates several important principles about multimodal learning.

Why XGBoost Succeeds with Simple Concatenation

XGBoost's superior performance with concatenated features stems from properties of gradient-boosted trees that make them naturally suited to heterogeneous multimodal data:

1. **Automatic feature interaction discovery:** Tree-based models learn complex non-linear interactions through recursive splits. XGBoost can automatically discover that certain text features are informative only when specific graph features take particular values, without requiring explicit fusion mechanisms. For example, textual sentiment markers might matter more for users with ambiguous network positions (moderate avg_neighbor_ideology), a pattern gradient boosting discovers through splits.

2. **Feature selection and regularization:** XGBoost's built-in feature importance mechanisms and L1/L2 regularization automatically downweight redundant features. When text and graph provide overlapping information (which occurs frequently in this homophilic network), XGBoost naturally relies more on the stronger graph signal while selectively using text features only when they add value.

3. **Robustness to modality conflicts:** When text and graph signals conflict, XGBoost learns decision rules that resolve the disagreement based on which modality has proven more reliable in similar contexts. Neural MLPs, by contrast, may average conflicting signals inappropriately without explicit fusion logic.

The 2.2% improvement over graph-only XGBoost, while modest, confirms that text features do provide complementary information beyond graph structure. This gain is achieved despite the network's extreme homophily (echo chamber = 0.95) where avg_neighbor_ideology already captures 89% of predictive signal. The complementary value likely comes from: (1) disambiguating ideology for users with heterogeneous network neighborhoods, and (2) capturing linguistic markers of ideological positions that are independent of behavioral retweet patterns.

5.4.2 Neural Fusion Strategies

Neural models show dramatically different performance depending on fusion strategy, ranging from $R^2 = 0.9212$ (Gated Fusion) to $R^2 = 0.8189$ (Simple MLP). This 10 percentage point gap demonstrates that neural networks require explicit architectural support for multimodal fusion, unlike XGBoost which succeeds with naive concatenation.

Simple MLP Fusion: A Failure Mode

Simple MLP fusion, where text and graph features are concatenated and passed through a standard feedforward network, achieves only $R^2 = 0.8189$ - worse than either single-modality baseline (text: 0.78, graph: 0.9152). This catastrophic failure reveals fundamental limitations of neural networks when handling heterogeneous multimodal inputs without explicit fusion design.

Training dynamics reveal the problem. Validation R^2 oscillates wildly across epochs, indicating unstable learning. Predictions compress toward the center of the ideology range (mostly within ± 0.5 despite true values spanning $[-1, 1]$), with numerous outliers. The model appears to average conflicting signals rather than learning to leverage complementary information.

The failure stems from several factors:

- **Dimensional imbalance:** 768-dimensional BERT embeddings dominate the 11-dimensional graph features in the input space, potentially causing the network to over-rely on text despite its weaker signal.
- **Redundancy without discrimination:** When text and graph provide overlapping information (common in homophilic networks), the MLP has no mechanism to recognize redundancy and focus on the stronger signal. It treats all inputs as equally informative.
- **Inappropriate averaging:** When modalities conflict, the MLP may average them inappropriately. A user with pro-supporter language but pro-opponent network connections might receive a neutral prediction that satisfies neither modality.

The stark contrast between Simple MLP's failure ($R^2 = 0.8189$) and XGBoost's success ($R^2 = 0.9373$) with the same concatenated features highlights a critical difference: **gradient boosting inherently handles heterogeneous feature spaces; neural networks do not**.

Gated Fusion: Best Neural Hybrid

Gated Fusion, which employs element-wise gating inspired by LSTM cells, achieves $R^2 = 0.9212$ - the best neural hybrid performance. Unlike simple concatenation, gated fusion computes learnable element-wise gates that control information flow from each modality. This finer-grained control enables selective incorporation of features at the dimension level rather than the modality level.

Training dynamics are almost stable. Validation R^2 reaches approximately 0.92 by epoch 5 and maintains near-perfect stability through epoch 33. Residuals are tightly centered around

zero with minimal heteroscedasticity. The model handles extreme ideological values well, avoiding the compression toward neutral that plagued simple MLP fusion.

Gated Fusion's 1.6 percentage point gap below XGBoost hybrid ($R^2 = 0.9373$) reflects several factors. First, XGBoost's ensemble of thousands of trees may capture more subtle feature interactions than a single neural network. Second, gradient boosting's explicit feature selection through gain metrics may more effectively identify the subset of text features that complement graph signals. Third, neural networks may require more careful tuning of architectural hyperparameters to fully exploit multimodal data.

Nevertheless, Gated Fusion demonstrates that neural architectures can effectively combine modalities when given appropriate fusion mechanisms. The 10 percentage point improvement over Simple MLP ($R^2 = 0.8189 \rightarrow 0.9212$) quantifies the value of architectural support for fusion in neural networks.

Attention Fusion

Attention Fusion, which learns instance-specific scalar weights for each modality, achieves $R^2 = 0.9051$.

Training is stable, with validation R^2 converging smoothly to 0.90 by epoch 5 and plateauing with minimal oscillation. Residuals show reduced systematic bias compared to Simple MLP. However, Attention Fusion falls 1.6 percentage points below Gated Fusion ($R^2 = 0.9212$), suggesting that scalar modality-level weights are less effective than element-wise feature-level gates.

The performance gap reveals a limitation: attention weights must make a binary decision about whether to use text or graph for each instance, whereas gated fusion can selectively use specific dimensions from each modality. When some text features complement graph signals while others are redundant, gating's finer granularity provides advantages.

5.4.3 Traditional ML vs. Neural Fusion: A Comparison

The results establish a clear performance hierarchy: XGBoost Hybrid ($R^2 = 0.9373$) > Gated Fusion ($R^2 = 0.9212$) > Attention Fusion ($R^2 = 0.9051$) > Simple MLP ($R^2 = 0.8189$). This ordering reveals fundamental differences between traditional machine learning and neural approaches to multimodal fusion.

Approach	R^2	Fusion Type	Key Property
XGBoost Hybrid	0.9373	Concatenation	Scale-invariant trees
Gated Fusion (Neural)	0.9212	Element-wise gates	Feature-level control
Attention Fusion (Neural)	0.9051	Scalar weights	Instance-specific
Simple MLP (Neural)	0.8189	Concatenation	No fusion mechanism
XGBoost Graph-Only	0.9152	Concatenation	Single modality

Table 5.5: Comparison of traditional machine learning (XGBoost) and neural approaches to multimodal fusion. XGBoost succeeds with simple concatenation due to properties of tree-based models (scale invariance, automatic feature interaction, built-in regularization). Neural networks require explicit fusion mechanisms (gating, attention) to approach XGBoost's performance, with simple concatenation failing catastrophically.

Key insights from this comparison:

1. **Concatenation suffices for tree-based models:** XGBoost achieves the best performance with naive concatenation. No architectural engineering for fusion is needed when using gradient boosting.

2. **Neural networks need fusion architecture:** The 4-10 percentage point gap between sophisticated neural fusion (Gated, Attention) and Simple MLP demonstrates that neural networks require explicit fusion mechanisms to handle heterogeneous modalities.

3. **XGBoost's inherent advantages:** Scale invariance, automatic feature interaction discovery, and regularization make gradient boosting naturally suited to multimodal data that neural networks struggle with.

4. **Best neural fusion approaches but doesn't surpass XGBoost:** Even with sophisticated gating, Gated Fusion falls 1.6 percentage points short of XGBoost hybrid, suggesting gradient boosting may be fundamentally better suited to this type of multimodal tabular data.

6 Conclusion

This thesis investigated methods for inferring latent political ideology from Twitter data, comparing text-based, graph-based, and hybrid approaches. We analyzed 4.2+ million users discussing military aid to Ukraine, with ideology labels for only 3% of users - a realistic data-scarce scenario.

6.1 Key Findings

Network structure dominates text for ideology inference. The best graph-based model (XGBoost, $R^2 = 0.9152$) substantially outperforms the best text model (BERT-LSTM, $R^2 = 0.7837$), a 13.2 percentage point gap. The best overall model combines both modalities (XGBoost on concatenated features, $R^2 = 0.9373$), though the 2.2 percentage point gain over graph-only reflects limited text contribution in highly homophilic networks.

Political homophily is the primary mechanism. Feature importance analysis reveals that avg_neighbor_ideology alone accounts for 89.1% of predictive power. Ideology-based features (neighbor ideology, homophily, echo chamber score) explain 93.5% of total importance, while structural features (degree, PageRank, centrality) contribute only 6.5%. The echo chamber score of 0.95 indicates near-complete ideological segregation.

Fusion architecture matters for neural networks but not gradient boosting. XGBoost succeeds with simple concatenation ($R^2 = 0.9373$), while neural MLPs fail catastrophically ($R^2 = 0.8189$). Neural networks require explicit fusion mechanisms: Gated Fusion ($R^2 =$

0.9212) and Attention Fusion ($R^2 = 0.9051$) dramatically outperform simple concatenation through learnable modality integration.

Task-specific adaptation is essential for text models. Frozen BERT achieves only $R^2 = 0.2393$; fine-tuning improves performance to $R^2 = 0.7816$ (226% increase). Within fine-tuned models, architectural variations (LSTM, CNN, MLP) differ by only 0.3-1.2 percentage points, suggesting most signal comes from language model adaptation rather than architecture.

Graph methods excel with limited labels. With 3% labeled nodes, graph neural networks achieve $R^2 \approx 0.91$ by propagating information through network structure, making them valuable for semi-supervised learning in data-scarce scenarios.

6.2 Scenario-Based Recommendations

Based on our findings, we provide practical recommendations for different scenarios:

1. **Maximum Performance with Both Modalities**

   Recommendation: Use XGBoost on concatenated BERT embeddings and graph features.

   Rationale: Achieves the best overall performance ($R^2 = 0.9373$), outperforming all neural architectures including sophisticated fusion mechanisms. XGBoost's tree-based structure naturally handles heterogeneous feature spaces without requiring explicit fusion design, succeeding with simple concatenation where neural MLPs fail catastrophically.

   Implementation: Extract BERT text embeddings by fine-tuning on the ideology prediction task. Compute graph-based features (neighborhood ideology metrics, centrality measures, clustering coefficients). Standardize each modality separately before concatenation. Train XGBoost on the combined feature vector.

   Expected performance: $R^2 \approx 0.93 - 0.94$, representing a 2.2 percentage point improvement over graph-only approaches.

   Trade-off: Requires both text and network data. More complex than single-modality approaches but provides best accuracy. Feature importance analysis reveals which modality contributes more for different user subgroups.

   When to use: Maximum accuracy is paramount and both modalities are available. Particularly valuable for users with heterogeneous network neighborhoods (approximately 5% of users in highly polarized debates) where text provides complementary disambiguation beyond network position.

2. **Network Data Available, Sufficient Labeled Examples**

   Recommendation: Use XGBoost with engineered graph features.

   Rationale: Provides excellent performance ($R^2 = 0.9152$) while being simpler to implement than multimodal approaches. Outperforms graph neural networks and offers interpretable feature importance scores. Requires only network structure and graph-based features, not text processing.

   Implementation: Compute graph-based features with particular focus on neighborhood ideology metrics (avg_neighbor_ideology, ideology_homophily), clustering coefficients, and community structure. Standardize features before training. Train gradient boosting on graph features.

   Expected performance: $R^2 \approx 0.91 - 0.92$ for ideology inference.

   Advantage: Simpler than multimodal approaches, provides interpretability, requires less computational resources. Feature importance reveals that avg_neighbor_ideology accounts for 89% of predictive power, confirming homophily as the primary mechanism.

3. **Network Data Available, Very Limited Labels (Data Scarcity)**

   Recommendation: Use Graph Neural Networks (GCN or GraphSAGE) with semi-supervised learning.

   Rationale: GNNs can propagate information from the small set of labeled nodes through the network to unlabeled nodes. The strong homophily in political networks (echo chamber score = 0.95) means that knowing even a few neighbors' ideologies provides strong signals. GraphSAGE offers inductive learning capabilities, generating embeddings for new nodes without retraining.

   Implementation: Start with a small labeled set (even 1-2% of nodes). Use GCN or GraphSAGE to aggregate neighbor information through message-passing. Ensure self-loops are added to the adjacency matrix for GCN. Apply regularization and early stopping to prevent overfitting on the small labeled set. Monitor performance on a held-out labeled validation set.

   Expected performance: With 3% labeled nodes, expect $R^2 \approx 0.89 - 0.91$. Performance degrades gracefully as labeled set shrinks, remaining usable down to 0.5% labeling.

   Note: More sophisticated architectures like Graph Attention Networks (GAT) exist but may not be feasible for large-scale networks (millions of edges) due to memory constraints. GCN and GraphSAGE provide excellent performance with efficient memory usage.

4. **No Network Data, Text Only**

   Recommendation: Use fine-tuned BERT with task-specific architecture (LSTM, CNN, or MLP).

Rationale: When network structure is unavailable (e.g., analyzing deleted accounts, cross-platform analysis, privacy-preserving scenarios), text is the only signal. Fine-tuning is essential; frozen BERT performs poorly ($R^2 = 0.24$ vs. $R^2 = 0.78$ fine-tuned).

Implementation: Extract enhanced text combining biographical descriptions, locations, and user tweets. Fine-tune BERT-based encoder (DistilBERT or full BERT) on the ideology inference task. Add task-specific head (LSTM for sequential modeling, CNN for local patterns, or MLP for simplicity). Apply regularization to prevent overfitting.

Expected performance: $R^2 \approx 0.76 - 0.78$ for ideology inference. This is substantially lower than graph-based methods ($\sim$13 percentage point gap) but may be the only option when network data is unavailable.

Note: Architectural variations (LSTM, CNN, MLP) show minimal performance differences (0.3-1.2 percentage points) when using fine-tuned BERT, suggesting that most signal is captured by the language model adaptation.

5. **Need for Interpretability**

   Recommendation: Use XGBoost with graph features and perform feature importance analysis.

   Rationale: Neural networks (both text-based and GNNs) are black boxes. XGBoost provides feature importance scores that explain which factors drive predictions - in our case, revealing that avg_neighbor_ideology accounts for 89% of predictive power. This interpretability is valuable for understanding mechanisms and for applications requiring justification (e.g., content moderation policies, research publication).

   Implementation: Train XGBoost on graph features. Extract SHAP values or use XGBoost's built-in gain-based feature importance. Analyze which features contribute most for different user groups (e.g., central vs. peripheral users, different ideological ranges, users with heterogeneous vs. homogeneous neighborhoods).

   Expected performance: $R^2 \approx 0.91 - 0.92$ with full interpretability of predictions.

   Insight: Feature importance analysis reveals that ideology-based features (avg_neighbor_ideology, ideology_homophily, echo_chamber_score) account for 93.5% of total importance, while structural features (degree, PageRank, centrality) contribute only 6.5%, confirming that political homophily is the primary mechanism enabling network-based inference.

6. **Deployment to Dynamic Networks (New Users Joining)**

   Recommendation: Use GraphSAGE for inductive learning capabilities.

   Rationale: GCN requires retraining when new nodes join the network. GraphSAGE learns an aggregation function that can generate embeddings for unseen nodes based on their neighbors, making it suitable for evolving networks like Twitter where users continuously join and connections form.

Implementation: Train GraphSAGE on existing labeled nodes using neighbor sampling. When new users join, apply the trained aggregation function to their neighborhood to generate embeddings. No retraining required until model drift is detected (monitor performance on recent labeled examples).

Expected performance: $R^2 \approx 0.91$ for ideology inference, with ability to handle new nodes efficiently without retraining.

Advantage: Inductive learning enables scaling to evolving networks. Neighbor sampling reduces memory requirements and enables mini-batch training on large graphs.

7. **Resource-Constrained Environments**

   Recommendation: Use XGBoost with a reduced feature set focusing on the most important graph features.

   Rationale: The top three features (avg_neighbor_ideology, ideology_homophily, clustering coefficient) account for 95.1% of predictive power, making them sufficient for most applications. This dramatically reduces computation time for feature extraction on large networks, as computing PageRank, community detection, and core decomposition can be expensive.

   Implementation: Compute only the most important features based on feature importance analysis. For avg_neighbor_ideology, use the undirected graph version to maximize labeled neighbor coverage. Train a lightweight gradient boosting model with reduced complexity (fewer trees, shallower depth).

   Expected performance: $R^2 \approx 0.88 - 0.90$ with significantly reduced computational cost (approximately 90% reduction in feature extraction time).

   Trade-off: Small performance degradation (1-3 percentage points) in exchange for dramatic computational savings. Particularly valuable for networks with millions of nodes where full feature extraction is prohibitive.

8. **Maximum Neural Multimodal Performance**

   Recommendation: Use Gated Fusion combining BERT text embeddings and graph features.

   Rationale: Gated Fusion achieves $R^2 = 0.9212$, the best neural hybrid performance, through element-wise gating that selectively incorporates text information at the feature dimension level. While this falls 1.6 percentage points below XGBoost hybrid ($R^2 = 0.9373$), it represents the strongest neural multimodal approach and may be preferred when end-to-end differentiable architectures are required.

   Implementation: Extract BERT text embeddings and graph features. Standardize each modality separately. Use gated fusion architecture with element-wise gates controlling

information flow from each modality: $\mathbf{h} = \mathbf{z}_t \odot \mathbf{W}_t\mathbf{t} + \mathbf{z}_g \odot \mathbf{W}_g\mathbf{g}$, where gates $\mathbf{z}_t, \mathbf{z}_g \in [0, 1]$ are learned. Apply regularization and early stopping.

Expected performance: $R^2 \approx 0.92$ for ideology inference. Note that in highly homophilic networks (echo chamber score > 0.9), the incremental gain over graph-only methods is modest (0.6 percentage points), as text provides limited complementary signal when network position is nearly deterministic of ideology.

In summary, for maximum performance use XGBoost on concatenated text and graph features ($R^2 = 0.9373$); for graph-only scenarios use XGBoost with engineered features ($R^2 = 0.9152$) for interpretability and strong performance; for neural multimodal approaches use Gated Fusion ($R^2 = 0.9212$); and for text-only scenarios use fine-tuned BERT ($R^2 = 0.7837$). Avoid simple concatenation with neural networks. For practitioners working with limited labeled data, Graph Neural Networks provide an effective semi-supervised learning framework that leverages network homophily to propagate ideology information from labeled to unlabeled nodes.

6.3 Practical Considerations

Data preprocessing: Standardization is critical - unnormalized features led to 85% performance drop. For hybrid models, standardize text embeddings and graph features separately before fusion.

Computational resources: GNNs on 2.6M-node graphs require significant GPU memory. GraphSAGE with neighbor sampling or subgraph training provides efficiency. Graph Attention Networks were infeasible due to memory requirements for 7.2M edges.

Temporal and cross-platform limitations: Models trained on November 2022 - March 2023 data may not generalize to later periods as discourse evolves. Cross-platform transfer (Twitter → Facebook, Reddit) likely requires platform-specific tuning, though core findings about graph superiority may generalize.

Ethical considerations: Ideology inference enables both beneficial applications (understanding discourse, detecting coordination) and harmful ones (targeting, surveillance). Deployment requires careful ethical review and safeguards against misuse.

6.4 Limitations and Future Work

Generalization: Our evaluation focused on the Ukraine debate, a highly polarized topic (echo chamber = 0.95). Less divisive topics might show different patterns where text provides more complementary signal. Cross-domain evaluation on healthcare, climate change, or elections would establish broader applicability.

Interaction diversity: We analyzed retweets only. Mentions, replies, and quote tweets may reveal different ideological patterns - antagonistic interactions often cross boundaries, while retweets stay within camps. Multi-relation graphs could improve inference for users with heterogeneous interaction patterns.

Advanced fusion: While we evaluated XGBoost concatenation and three neural fusion strategies, graph transformers with cross-modal attention or deep co-training could potentially improve results. However, given that our best hybrid ($R^2 = 0.9373$) only marginally exceeds graph-only ($R^2 = 0.9152$) due to extreme homophily, incremental benefits may be limited.

Scalability and robustness: Our implementation handled 2.6M users and 7.2M edges; scaling to billion-edge networks requires distributed frameworks. Adversarial robustness (bot networks, strategic language moderation) remains unexplored - network methods may be more robust than text (changing connections is costly), but quantification is needed.

6.5 Final Remarks

The central finding is that political homophily provides the strongest signal for ideology inference in social networks. Average neighbor ideology accounts for 89% of predictive power, while structural position and textual content contribute marginally. This has methodological and substantive implications.

Methodologically: when network data is available, use graph-based methods (XGBoost with features for interpretability, GraphSAGE for dynamic networks). For maximum performance with both modalities, XGBoost on concatenated features achieves $R^2 = 0.9373$. Neural hybrids require gated fusion ($R^2 = 0.9212$) to avoid concatenation failure. Text-only methods reach $R^2 = 0.78$ and should be reserved for scenarios where network structure is unavailable.

Substantively: an echo chamber score of 0.95 means 95% of neighbors share ideological polarity - near-complete segregation. Network position is nearly deterministic of ideology ($r = 0.875$ correlation with neighbor ideology), demonstrating that political discourse occurs within isolated ideological bubbles with minimal cross-cutting exposure.

This structure has implications for information diffusion and polarization. Users' information environments are determined by network formation patterns, not algorithmic filtering. Interventions aiming to increase diverse exposure must address homophilic connection formation, not merely content recommendation.

As political discourse moves to digital platforms central to democratic processes, understanding ideology inference will grow in importance. This work provides both theoretical insight into how ideology manifests in social networks and practical tools for measuring political behavior in digital spaces.

References

[1] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[2] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 U.S. election: Divided they blog," in *Proceedings of the 3rd International Workshop on Link Discovery*, ser. LinkKDD '05. New York, NY, USA: ACM, 2005, pp. 36–43.

[3] E. Hargittai, J. Gallo, and M. Kane, "Cross-ideological discussions among conservative and liberal bloggers," *Public Choice*, vol. 134, no. 1-2, pp. 67–86, 2008.

[4] M. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, F. Menczer, and A. Flammini, "Political polarization on Twitter," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, ser. ICWSM '11, vol. 5, no. 1. AAAI Press, 2011, pp. 89–96.

[5] I. Himelboim, S. McCreery, and M. Smith, "Birds of a feather tweet together: Integrating network and content analyses to examine cross-ideology exposure on twitter," *Journal of Computer-Mediated Communication*, vol. 18, no. 2, pp. 154–174, 2013.

[6] S. Yardi and d. boyd, "Dynamic debates: An analysis of group polarization over time on twitter," *Bulletin of Science, Technology & Society*, vol. 30, no. 5, pp. 316–327, 2010.

[7] P. Barberá and G. Rivero, "Understanding the political representativeness of twitter users," *Social Science Computer Review*, vol. 33, no. 6, pp. 712–729, 2015.

[8] C. R. Sunstein, "Echo chambers: Bush v. gore, impeachment, and beyond," *Princeton University Press*, 2001.

[9] E. Pariser, *The Filter Bubble: What the Internet Is Hiding from You*. Penguin Press, 2011.

[10] E. Bakshy, S. Messing, and L. A. Adamic, "Exposure to ideologically diverse news and opinion on Facebook," *Science*, vol. 348, no. 6239, pp. 1130–1132, 2015.

[11] S. Flaxman, S. Goel, and J. M. Rao, "Filter bubbles, echo chambers, and online news consumption," *Public Opinion Quarterly*, vol. 80, no. S1, pp. 298–320, 2016.

[12] A. Boutyline and R. Willer, "The social structure of political echo chambers: Variation in ideological homophily in online networks," *Political Psychology*, vol. 38, no. 3, pp. 551–569, 2017.

[13] E. Dubois and G. Blank, "The echo chamber is overstated: The moderating effect of political interest and diverse media," *Information, Communication & Society*, vol. 21, no. 5, pp. 729–745, 2018.

[14] S. Iyengar, G. Sood, and Y. Lelkes, "Affect, not ideology: A social identity perspective on polarization," *Public Opinion Quarterly*, vol. 76, no. 3, pp. 405–431, 2012.

[15] W. J. Brady, J. A. Wills, J. T. Jost, J. A. Tucker, and J. J. Van Bavel, "Emotion shapes the diffusion of moralized content in social networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 28, pp. 7313–7318, 2017.

[16] J. E. Settle, *Frenemies: How Social Media Polarizes America*. Cambridge University Press, 2018.

[17] C. A. Bail, L. P. Argyle, T. W. Brown, J. P. Bumpus, H. Chen, M. F. Hunzaker, J. Lee, M. Mann, F. Merhout, and A. Volfovsky, "Exposure to opposing views on social media can increase political polarization," *Proceedings of the National Academy of Sciences*, vol. 115, no. 37, pp. 9216–9221, 2018.

[18] T. Gillespie, "The relevance of algorithms," *Media Technologies: Essays on Communication, Materiality, and Society*, vol. 167, no. 2, 2014.

[19] E. Bozdag, "Bias in algorithmic filtering and personalization," *Ethics and Information Technology*, vol. 15, no. 3, pp. 209–227, 2013.

[20] M. Haim, A. Graefe, and H.-B. Brosius, "Burst of the filter bubble? effects of personalization on the diversity of google news," *Digital Journalism*, vol. 6, no. 3, pp. 330–343, 2018.

[21] R. Levy, "Social media, news consumption, and polarization: Evidence from a field experiment," *American Economic Review*, vol. 111, no. 3, pp. 831–870, 2021.

[22] F. Huszár, S. I. Ktena, C. O'Brien, L. Belli, A. Schlaikjer, and M. Hardt, "Algorithmic amplification of politics on twitter," *Proceedings of the National Academy of Sciences*, vol. 119, no. 1, p. e2025334119, 2022.

[23] P. Barberá, "Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data," *Political Analysis*, vol. 23, no. 1, pp. 76–91, 2015.

[24] S. Hong and D. Nadler, "Political rumors on twitter during the 2010 us mid-term elections: What can we learn from this?" *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media*, 2012.

[25] M. D. Vicario, A. Bessi, F. Zollo, F. Petroni, A. Scala, G. Caldarelli, H. E. Stanley, and W. Quattrociocchi, "Echo chambers: Emotional contagion and group polarization on facebook," *Scientific Reports*, vol. 6, no. 1, p. 37825, 2016.

[26] M. Cinelli, G. D. F. Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini, "The echo chamber effect on social media," *Proceedings of the National Academy of Sciences*, vol. 118, no. 9, 2021.

[27] M. Prior, *Media and Political Polarization*. Annual Reviews, 2013, vol. 16.

[28] J. A. Tucker, A. Guess, P. Barberá, C. Vaccari, A. Siegel, S. Sanovich, D. Stukal, and B. Nyhan, "Social media, political polarization, and political disinformation: A review of the scientific literature," *Political Polarization, and Political Disinformation: A Review of the Scientific Literature (March 19, 2018)*, 2018.

[29] M. S. Granovetter, "The strength of weak ties," *American Journal of Sociology*, vol. 78, no. 6, pp. 1360–1380, 1973.

[30] M. Barnidge, "Explicitly searching for political information: An analysis of political misperceptions and motivated reasoning," *Social Science Computer Review*, vol. 35, no. 6, pp. 746–759, 2017.

[31] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[32] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Dis-*

covery and Data Mining, ser. KDD '09. New York, NY, USA: Association for Computing Machinery, 2009, pp. 807–816.

[33] M. Iyyer, P. Enns, J. Boyd-Graber, and P. Resnik, "Political ideology detection using recursive neural networks," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, 2014, pp. 1113–1122.

[34] R. Cohen and D. Ruths, "Classifying political orientation on Twitter: It's not easy!" in *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, ser. ICWSM '13, vol. 7, no. 1. AAAI Press, 2013, pp. 91–99.

[35] D. Preotiuc-Pietro, Y. Liu, D. Hopkins, and L. Ungar, "Beyond binary labels: Political ideology prediction of Twitter users," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 729–740.

[36] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186.

[37] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. New York, NY, USA: ACM, 2014, pp. 701–710.

[38] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems*, vol. 26. Curran Associates, Inc., 2013, pp. 3111–3119.

[39] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 855–864.

[40] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World*

Wide Web, ser. WWW '15. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[41] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '17. ACM, Aug. 2017, p. 385–394. [Online]. Available: http://dx.doi.org/10.1145/3097983.3098061

[42] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, ser. ICLR '17, 2017. [Online]. Available: https://openreview.net/forum?id=SJU4ayYgl

[43] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," 2018. [Online]. Available: https://arxiv.org/abs/1706.02216

[44] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2018. [Online]. Available: https://arxiv.org/abs/1710.10903

[45] A. Benton, R. Arora, and M. Dredze, "Learning multiview embeddings of Twitter users," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany: Association for Computational Linguistics, 2016, pp. 14–19.

[46] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19. New York, NY, USA: ACM, 2019, pp. 793–803.

[47] P. Yu, C. Fu, Y. Yu, C. Huang, Z. Zhao, and J. Dong, "Multiplex heterogeneous graph convolutional network," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '22. ACM, Aug. 2022, p. 2377–2387. [Online]. Available: http://dx.doi.org/10.1145/3534678.3539482

[48] A. Baqir, Y. Chen, F. Diaz-Diaz *et al.*, "Unveiling the drivers of active participation in social media discourse," *Scientific Reports*, vol. 15, p. 4906, 2025.

[49] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking : Bringing order to the web," in *The Web Conference*, 1999. [Online]. Available: https://api.semanticscholar.org/CorpusID:1508503

[50] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[51] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. ACM, Aug. 2016, p. 785–794. [Online]. Available: http://dx.doi.org/10.1145/2939672.2939785

Acknowledgments

This thesis would not exist without the many people who supported me throughout this journey.

First, I want to thank Professor Alessandro Galeazzi for his guidance and mentorship. Thank you for never pressuring me, for being genuinely supportive, and for inspiring me to pursue an academic path. Your approach to supervision gave me the space to grow as a researcher while always knowing that I had your support and knowledge you shares with me.

To my parents Svetlana and Mihkail, my aunt Ira and my younger brother Nikita: thank you for believing in my decision to study abroad and for supporting me through every step of this journey. Your encouragement, your sacrifices, and your unwavering support across all these kilometers have made everything possible. I am incredibly grateful to have you as my family.

To Daniele, my partner: thank you for being unconditionally by my side through all of this. You went through this university journey with me as if you were doing it all over again yourself - patiently waiting while I struggled to write code without errors, celebrating every small breakthrough, and being there through every frustration and victory.

To Diana: thank you for being the person who first made me believe I could study abroad. You inspired me to take that leap, and your support throughout this whole process has been invaluable. I'm grateful for your friendship across the distance.

To Nadya: thank you for your constant support and friendship despite the kilometers between Moscow and Padova. You've been there for every high and low, and knowing you're always just a message away has meant so much.

To Liza: thank you for the countless cappuccinos and croissants we shared while I talked through my thesis struggles. Those moments made Italy feel like home, and your presence in my daily life has been a gift.

To Ilya: thank you for the many cups of tea you sat through while listening to my complaints about research. Your patience and willingness to listen again and again has been more helpful than you probably realize.

Учёба в магистратуре в Италии и, в частности, этот диплом были для меня большим шагом в сторону новой, лучшей жизни, и я очень благодарна всем, кто был рядом и поддерживал меня!