When Annotators Disagree, Topology Explains: Mapper, a Topological Tool for Exploring Text Embedding Geometry and Ambiguity

Nisrine Rair[1,2], Alban Goupil[1], Valeriu Vrabie[1], Emmanuel Chochoy[2]

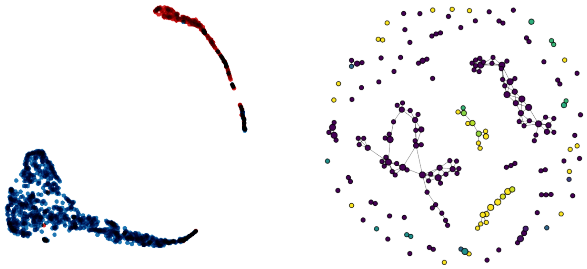
[1]CReSTIC, Université de Reims Champagne-Ardenne, Reims, France,
[2]Chochoy Conseil, Reims, France
nisrine.rair@univ-reims.fr

Abstract

Language models are often evaluated with scalar metrics like accuracy, but such measures fail to capture how models internally represent ambiguity, especially when human annotators disagree. We propose a topological perspective to analyze how fine-tuned models encode ambiguity and more generally instances. Applied to RoBERTa-Large on the MD-Offense dataset, Mapper, a tool from topological data analysis, reveals that fine-tuning restructures embedding space into modular, non-convex regions aligned with model predictions, even for highly ambiguous cases. Over 98% of connected components exhibit $\geq 90\%$ prediction purity, yet alignment with ground-truth labels drops in ambiguous data, surfacing a hidden tension between structural confidence and label uncertainty. Unlike traditional tools such as PCA or UMAP, Mapper captures this geometry directly uncovering decision regions, boundary collapses, and overconfident clusters. Our findings position Mapper as a powerful diagnostic tool for understanding how models resolve ambiguity. Beyond visualization, it also enables topological metrics that may inform proactive modeling strategies in subjective NLP tasks. For reproducibility, all code and experiment configurations are released[1].

1 Introduction

Ambiguity is a persistent challenge in Natural Language Processing (NLP) (Pavlick and Kwiatkowski, 2019) when instances often permit multiple plausible meanings or evoke differing judgements across annotators (Wan et al., 2023). This is the case in offensive language detection or natural language inference, where annotators may disagree (Leonardelli et al., 2021; Nie et al., 2020). While prior work has focused on modeling the ambiguity using soft labels or confidence estimates (Swayamdipta et al., 2020; Fornaciari et al., 2021), less is known about how ambiguous examples are internally structured in representation space.



(a) UMAP projection (b) Mapper graph

Figure 1: Illustration of geometric (UMAP) and topological (Mapper) views of model representations. Figure(a) UMAP shows apparent separation between offensive and non-offensive instances. (b) Mapper reveals a more fragmented and connected structure, with offensive and non-offensive regions interwoven, highlighting spatial organization that UMAP flattens.

Fine-tuning of language models is known to rearrange data within the embedding spaces, increasing cohesion and alignment with task-relevant categories (Zhou and Srikumar, 2021; Rajaee and Pilehvar, 2021). These changes are often analyzed using geometric tools such as PCA and UMAP, and quantified using metrics like cosine similarity, silhouette score, and anisotropy (Ethayarajh, 2019; Coenen et al., 2019; Cai et al., 2021). Yet such tools can oversimplify high-dimensional structure. PCA's global projections compress variance into a limited number of axes (Wold et al., 1987), while UMAP enforces local Euclidean neighborhoods (McInnes et al., 2020), often at the expense of capturing nonlinear or hierarchical relationships. In contrast, linguistic embedding space has been shown to exhibit natural hyperbolic geometry (Nickel and Kiela, 2017), a structure that topological methods can preserve without relying on Euclidean assumptions. This is particularly important in high-dimensional settings, as dimensionality

[1]https://github.com/NisrineRair/tda-nlp-representations

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 8469–8492
November 4-9, 2025

increases, points tend to become nearly equidistant, blurring the notion of proximity and undermining the interpretability of distance-based measures (Donoho, 2000; Aggarwal et al., 2001). As a result, familiar geometric intuitions about neighborhoods and separation begin to break down, making it easy to misread the structure of the embedding space.

This limitation is illustrated in Figure 1a where UMAP applied to the final-layer `[CLS]` embeddings of a RoBERTa-Large model fine-tuned on binary offensive language detection, presents the offensive class as a single, compact cluster. On the contrary, topological tools construct graphs based on point connectivity rather than distances, revealing a more fragmented and nuanced organization as in Figure 1b, where offensive instances are spread across multiple distinct yet connected regions, suggesting that geometric projections may flatten meaningful topological structure. This example highlights a key limitation of classical tools: visual simplicity can come at the cost of structural fidelity. This limitation reflects the embedding space's non-Euclidean, hyperbolic structure.

In this work, we use Mapper, a topological tool, to analyze how fine-tuned language models embed ambiguous instances and how this structure manifests in the geometry of embedding space. Focusing on offensive language detection, we explore how fine-tuning reshapes topological organization, why models exhibit overconfidence in ambiguous inputs, and how these behaviors relate to human disagreement. Our results show that fine-tuning imposes topological regularity: even instances with annotation disagreements are mapped into smooth, consistent regions. Rather than ignoring uncertainty, language models appear to spatially reorganize it, resolving ambiguity into reliable structures.

2 Related Work

Ambiguity, Disagreement, and Uncertainty in NLP. Ambiguity in NLP has been studied from multiple angles. Early work often treated disagreement as label noise to be filtered (Zhang et al., 2017), whereas more recent studies argue that it reflects genuine linguistic uncertainty and advocate for distributional supervision to capture annotator variability (Pavlick and Kwiatkowski, 2019; Mostafazadeh Davani et al., 2022). Building on this perspective, several studies investigated model behavior on ambiguous inputs. (Swayamdipta et al., 2020) show that ambiguous examples exhibit unstable training dynamics, while Liu et al. (2023) argue that models tend to be overconfident on such inputs, failing to reflect underlying uncertainty in their predictions. Other research has challenged the use of majority vote in subjective tasks. Leonardelli et al. (2023) argue that majority aggregation suppresses meaningful disagreement signals, motivating soft-label and annotator aware modeling. These approaches assess ambiguity at the output level, leaving open the question of how it is structurally encoded within the model.

Fine-Tuning and Embedding Space. Fine-tuning is known to reshape representation in the embedding space. Studies using distance-based metrics show that fine-tuned models for classification tasks increase intra-class cohesion and expand inter-class separation (Zhou and Srikumar, 2021). Spectral analyses reveal that variance becomes concentrated in relevant directions, reducing isotropy of embedded instances (Cai et al., 2021; Ethayarajh, 2019). Probing classifiers further confirm that linguistic features become more linearly separable after fine-tuning (Hewitt and Manning, 2019). However, distance-based metrics oversimplify or distort high dimensional structure, potentially missing connectivity patterns and latent organization.

Topological Methods in NLP. Topological Data Analysis (TDA) provides powerful tools for studying high-dimensional data beyond geometry, focusing on how points connect rather than how far apart they are. Two key tools are **(1) Mapper** (Singh et al., 2007), which constructs graph networks by projecting data through a lens into lower dimensions, and **(2) Persistent homology** (Carlsson, 2009), which quantifies topological features like clusters or holes that persist across scales. While both methods have demonstrated significant success in different applications, for example in identifying novel cancer subtypes from gene expression data (Lum et al., 2013), their application to NLP remains limited, with few applications that can be categorized into three main directions: **(1) Lexical-level analysis**, Rathore et al. (2023) employed Mapper to track topological transformations in word embedding spaces during fine-tuning, revealing how semantic relationships reorganize at different training stages. **(2) Summarizing documents**, (Guan et al., 2016) demonstrated that homological simplification can remove noise while preserving structural patterns and effectively extract keyphrases by preserving the topological structure of document

semantic graphs using TF-IDF weighted representations. **(3) Model-internal analysis**, Fitz et al. (2024) applied persistent homology to compare the topological complexity of LSTM and Transformer activation patterns, finding recurrent architectures exhibit greater representational redundancy. Complementary, Gardinazzi et al. (2024) utilized zigzag persistence to track the evolution of topological features across transformer layers, enabling principled model pruning based on structural redundancy analysis. While these prior works demonstrate the feasibility of topological tools in NLP, they focus largely on pretrained models, lexical structure, or model internals.

Prior work offers valuable insights into how models handle ambiguous data, how fine-tuning reshapes representation spaces, and how topological methods extract structures in high dimensions. However, these strands remain disconnected. Ambiguity is typically evaluated at the output level, fine-tuning effects are described through geometric metrics, and TDA is rarely applied beyond lexical or pretrained settings. Our work bridges these gaps by using Mapper TDA-algorithm to explain how fine-tuned models internally structure ambiguous instances revealing predictive regularities that persist even for annotator disagreement.

3 The Mapper Algorithm

This section provides a brief overview of the Mapper algorithm, the foundation of our topological analysis. Mapper (Singh et al., 2007) constructs a graph-based summary of high-dimensional data by combining filtering, covering, and clustering. This process reveals global patterns while preserving local structure, offering a topological perspective on how data are organized. The algorithm steps, illustrated in Figure 2, are described below.

Filtering. Because high-dimensional data can be viewed from many angles, a *lens function* allows it to be projected to highlight a specific aspect of interest. Formally, a lens is a function $f : \mathbb{R}^d \to \mathbb{R}^k$, mapping each point in the original d-dimensional space to a k-dimensional space, where $k \ll d$. While any value of k can be used, implementations typically use $k \leq 3$ to balance computational efficiency with interpretability. The lens function defines a coordinate system by which the data can be more easily partitioned in the next step. This flexibility allows Mapper to be tailored to different analysis goals. For example, projecting onto one or several principal components emphasizes directions of high variance, while ignoring details attached on the other directions. The lens can be any user-defined function, including geometric descriptors such as eccentricity or centrality, density-based scores, or task-specific measures, depending on what aspect of the data one aims to reveal.

Covering. The codomain, i.e. the image $\mathbb{R}^k$ of the lens f, is then covered by overlapping hypercubes. Mapper divides each dimension into r equal-length intervals, forming a grid of hypercubes that partition the image of f. This grid defines a covering $\mathcal{U} = \{U_i\}$, where each U_i is a hypercube of side length $1/r$. For 1D lens, the covering consists of intervals, for $k = 2$ it consists of rectangles, and in general, for arbitrary k, the covering is a collection of overlapping hypercubes in $\mathbb{R}^k$. The parameter r controls the granularity of this partition: higher values reveal local structure, while lower values favor generalization. To encourage continuity between hypercubes, Mapper introduces a fractional overlap $\epsilon \in [0, 1)$ between adjacent bins. This expands each hypercube slightly by an amount $\epsilon \cdot \frac{1}{r}$ in each dimension, so that neighboring hypercubes intersect. A point $f(x)$ may therefore fall into multiple overlapping bins, and the corresponding data point x will be included in multiple preimage sets $f^{-1}(U_i)$. This overlap is essential for connectivity, since it facilitates the linking of topologically related clusters, preserving continuity and coherence of the underlying data structure in the final graph.

Clustering. For each hypercube $U_i \in \mathcal{U}$, Mapper identifies the corresponding subset $f^{-1}(U_i) \cap X$ in the original high-dimensional space $\mathbb{R}^d$ and applies a clustering algorithm. Each cluster corresponds to a node in the resulting Mapper graph. This local clustering avoids reliance on global geometry and is well-suited to nonlinear data. The clustering algorithm can be chosen based on the task. Common choices include DBSCAN, HDBSCAN, or other density or distance-based methods. If a data point appears in multiple overlapping bins and is assigned to several clusters, the corresponding nodes in the Mapper graph are connected by an edge. These edges reflect shared membership across clusters and allow Mapper to summarize the data as a graph that captures both local structure and global connectivity.

Graph construction. The final output of Mapper is a graph where each node corresponds to a cluster and edges connect nodes that share one or more data points. A graph may contain multiple con-

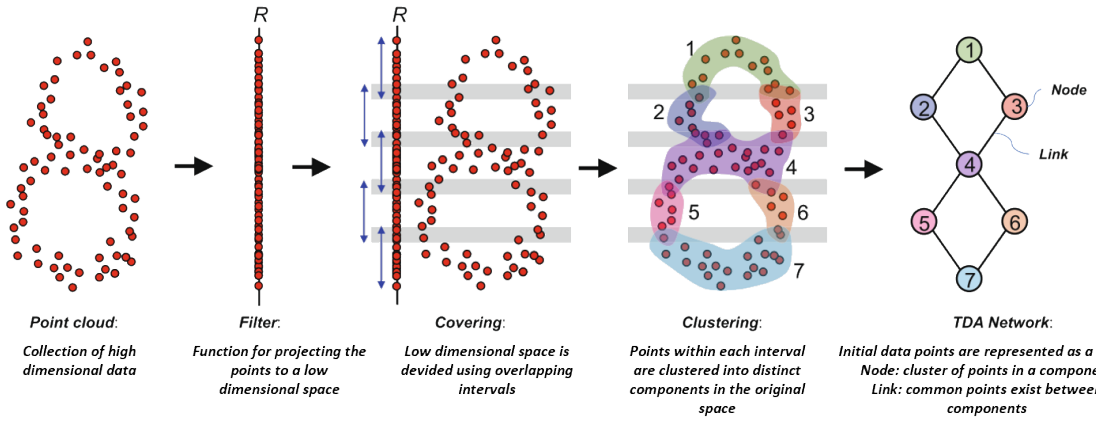


Figure 2: Illustration of the Mapper algorithm. A high-dimensional data is first projected using a lens function (filter) to a low-dimensional space. The projection is covered by overlapping hypercubes (intervals here), and each preimage is clustered in the original space. Nodes in the resulting graph represent clusters, and edges indicate shared points due to overlap. Adapted from Figure 1 of Liao et al. (2019), available under CC BY 4.0.

nected components, reflecting distinct regions of the dataset that remain disconnected under the lens and clustering choices. Analyzing the size, homogeneity, or connectivity of these components can reveal meaningful structures in the data.

Mapper's output depends on three key design choices: the lens function, which determines the aspect of the data being emphasized, the cover resolution and overlap, which control how finely the space is segmented and how regions connect, and the clustering algorithm, which determines how data points are grouped into nodes. Each of these elements shapes the granularity, connectivity, and interpretability of the resulting graph. Choosing values that balance structural detail with readability is essential. For a broader comparison of these elements and their impact, see Madukpe et al. (2025).

4 Methodology

4.1 Dataset

We use the MD-Offense dataset (Leonardelli et al., 2021), designed for the analysis of annotator disagreement in offensive language detection. It contains 10,753 instances, English tweets annotated by five U.S. annotators per instance. The data spans three socially and politically salient domains: COVID-19, the 2020 U.S. presidential election, and racial justice, providing rich variability in both content and labeling agreement. Each instance is assigned a binary label $c = \{\texttt{offensive}, \texttt{non-offensive}\}$, obtained via majority vote from five annotators, and denoted thereafter ground-truth label. In addition, each instance is labeled with an agreement level: A++ (5/5), A+ (4/5), or A0 (3/5), which serves as a proxy for linguistic ambiguity (Pavlick and Kwiatkowski, 2019; Mostafazadeh Davani et al., 2022). A++ examples are considered unambiguous, while A0 reflects high annotator disagreement and thus greater ambiguity. The dataset is split into training, validation, and test sets, stratified by domain and agreement level, with full distributions provided in Appendix A. Prior studies report a sharp decline in classification performance as agreement decreases, with F1 scores dropping from approximately 0.91 for A++ to 0.62 for A0 (Leonardelli et al., 2021). We replicate this trend on several models, reported in Appendix B, underscoring the difficulty of modeling highly ambiguous inputs.

4.2 Model and Embedding Extraction

We focus our analysis on RoBERTa-Large, examining both its pretrained and fine-tuned representations. While several encoder-only architectures were fine-tuned as part of our broader experimentation, we focus our analysis on RoBERTa-Large due to strong performance across agreement levels as shown in Appendix B and because focusing on a single model provides clarity of analysis. This makes it a suitable choice for showcasing Mapper's interpretability rather than conducting model benchmarking. Fine-tuning was performed on the full training set using the ground-truth labels, including all agreement levels from A0 to A++ to reflect the complete distribution of annotator disagreement. While prior work has shown that restricting fine-tuning to A+ and/or A++ subsets can

yield higher classification accuracy (Leonardelli et al., 2021), our goal is not to optimize predictive performance but to analyze how models structure data in representation space. Fine-tuning was performed using standard hyperparameters: 3 training epochs, a batch size of 8, and a learning rate of $2 \cdot 10^{-5}$. To capture model-internal representations, we extract the `[CLS]` embedding from the final hidden layer for each test instance because it serves as input to the classifier and encodes task-relevant information.

4.3 Mapper Parameters

We implemented the Mapper algorithm using the `KeplerMapper` Python library (Veen et al., 2019; van Veen et al., 2021); This library facilitated the projection, covering, and construction of Mapper graphs. To analyze and visualize the resulting networks, we used the `NetworkX` library (Hagberg et al., 2008), which enabled efficient computation of graph-theoretic and topological metrics.

Lens Function We chose six lens functions that project high-dimensional instance embeddings of the `[CLS]`, i.e. $d = 1024$ into 1D intervals ($k = 1$). Each lens captures a distinct geometric or semantic property of the embedding space. The centroid projection captures alignment with the decision boundary between offensive and non-offensive classes. PCA highlights global variance, while eccentricity defined as the maximum cosine distance to training points detects peripheral or outlier instances. The L2 norm reflects activation magnitude, potentially linked to confidence. Two random projections serve as baselines to distinguish signal from noise. All lens statistics are computed using the training set only to avoid data leakage. Full definitions are provided in Appendix C. The use of 1D lenses simplifies comparisons across ambiguity levels by ensuring consistency in Mapper parameters. All graphs are constructed using the same number of intervals, overlap, clustering algorithm, and layout. While higher-dimensional or task-specific lenses may uncover richer structures, we defer such extensions to future work. While our lenses are geometric, Mapper's topological structure is not derived from the projections themselves. The lens simply partitions the data into overlapping regions, clustering is then performed in the original high-dimensional space. As a result, the Mapper graph reflects how data is connected, not merely how it appears in projection. This shifts the focus from geometric simplification to topological structure, preserving branching, overlaps, and ambiguity patterns often flattened by projection-based methods.

Covering We use a fixed resolution of $r = 40$ with 30% overlap ($\epsilon = 0.3$) for all lenses. These hyperparameters were selected after inspection of Mapper graphs across lenses and ambiguity levels. Increasing the number of intervals (e.g., $r > 50$) led to overly fragmented graphs with many disconnected or singleton nodes, while coarse coverings (e.g., $r < 30$) oversimplified the structure, obscuring fine-grained connectivity. A resolution of $r = 40$ offered a middle ground between interpretability and granularity to preserve neighborhood structure without excessive redundancy. The impact of covering resolution on graph topology is provided in Appendix D. Similarly, for overlap, we tested values in the range $\epsilon \in [0.2, 0.9]$. We avoided higher overlaps as they introduced visual artifacts such as redundant nodes due to duplicate points appearing in adjacent regions, which distorted the underlying topological signal. At the same time, too little overlap caused artificial fragmentation. The choice of $\epsilon = 0.3$ consistently preserved connectivity without inflating the graph. Appendix E provides evidence of how overlap settings influence graph structure.

Clustering We use HDBSCAN (McInnes et al., 2017) as our clustering algorithm to avoid imposing a fixed number of clusters. Unlike methods such as k-means or agglomerative clustering that require a global k, HDBSCAN adapts to the local density of points in each interval, making it well-suited for capturing the heterogeneous structures that Mapper is designed to surface. Fixed-k clustering often resulted in artificial splits within homogeneous regions, leading to unstable or redundant nodes. In contrast, HDBSCAN dynamically adjusts to local structure and identifies outliers as noise, improving robustness and interpretability. We use cosine distance, appropriate for normalized embedding spaces, and set the minimum cluster size to 2, enabling the detection of fine-grained patterns without excessive fragmentation. Compared to DBSCAN, HDBSCAN requires less manual tuning and generalizes better to varying densities across intervals. The proportion of excluded instances remained moderate and did not meaningfully distort the structure for most lenses, as shown in Appendix F, which reports detailed noise rates across lenses and ambiguity levels.

4.4 Topological Metrics

We denote the resulting Mapper graph as $G = (N, E)$, where N represents the nodes and E the edges between them. Each node $n_i \in N$ represents a cluster containing a subset $X_i = \{x_j^i\}$ of the test instance embeddings. The size of each node is denoted $S_i = |X_i|$. A connected component is a set of nodes $C = \{n_i\}$ such that all nodes in C are connected via edges in E. The full set of data points in a component is denoted $X_C = \bigcup_{n_i \in C} X_i$, with total size $|X_C| = \sum_{n_i \in C} S_i$. Based on this formal structure, we propose the following topological evaluation metrics.

Component Purity (CP): average purity of nodes within a connected component C:

$$\text{CP} = \frac{1}{|X_C|} \sum_{n_i \in C} \sum_{x_j^i \in X_i} \left[y_j^i = c \right], \qquad (1)$$

where y_j^i is the label of sample x_j^i, c is the dominant label (either ground-truth or predicted), and $[P]$ is the Iverson bracket, equal to 1 if P is true and 0 otherwise. CP measures the extent to which a component is dominated by a single class, accounting for the varying sizes of its nodes.

Edge Agreement (EA): proportion of edges whose endpoints share the same majority class label:

$$\text{EA} = \frac{1}{|E|} \sum_{(n_i, n_j) \in E} \left[\text{maj}(n_i) = \text{maj}(n_j) \right], \quad (2)$$

where $\text{maj}(n_i)$ is the majority ground-truth label among all samples assigned to node n_i. EA captures the global consistency of the Mapper graph by checking how often directly connected nodes share the same class.

Majority Match (MM): compares predicted vs. true majority labels at the component level:

$$\text{MM} = \frac{1}{|\mathcal{C}|} \sum_{C \in \mathcal{C}} \left[\text{maj}_{\text{true}}(C) = \text{maj}_{\text{pred}}(C) \right], \quad (3)$$

with $\mathcal{C}$ the set of connected components in G, and $\text{maj}_{\text{true}}(C)$, $\text{maj}_{\text{pred}}(C)$ denote the ground-truth and predicted majority labels in X_C, respectively. MM assesses how well the model's predictions align with human consensus at a regional level.

5 Results and Discussion

Topological Effects of Fine-Tuning. Fine-tuning restructures the embedding space not by collapsing classes into simple, linearly separable clusters, but by organizing them into modular, non-convex regions. As shown in Figure 3a, the base model exhibits a fragmented topology on the A++ subset, with diffuse clusters and inconsistent labels. Fine-tuning, as shown in Figure 3b, consolidates these components into more coherent regions that are better aligned with class boundaries, yet they remain distributed across disconnected areas.

This modularity persists across ambiguity levels. Both component purity and edge agreement increase after fine-tuning in Figures 5 and 6, indicating a twofold structural gain: locally, components become more class-homogeneous, and globally, neighboring regions exhibit stronger label consistency. These improvements suggest that the model learns to organize decision boundaries not only within clusters but also across their connections. Gains are weakest in the A0 subset, where annotator disagreement limits geometric consolidation.

Mapper thus reveals that fine-tuning encodes task-relevant structure without enforcing global convexity. The resulting geometry remains multimodal: the model partitions each class into fragmented, task-aligned subregions. This helps explain why linear probes often fail (Hewitt and Liang, 2019; Voita and Titov, 2020): although widely used, they assume unimodal linear separability and therefore miss the nonlinear, multimodal structure that persists even after training. This highlights the value of topological diagnostics such as Mapper, which are designed to capture the complex organization of decision regions.

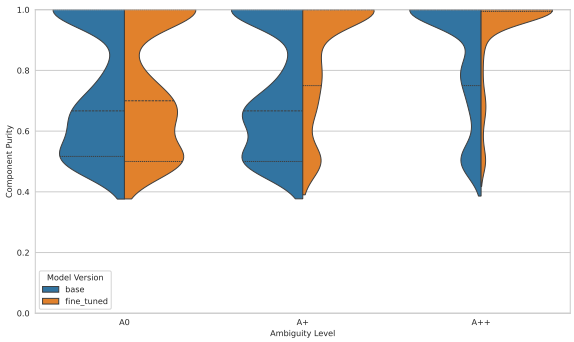
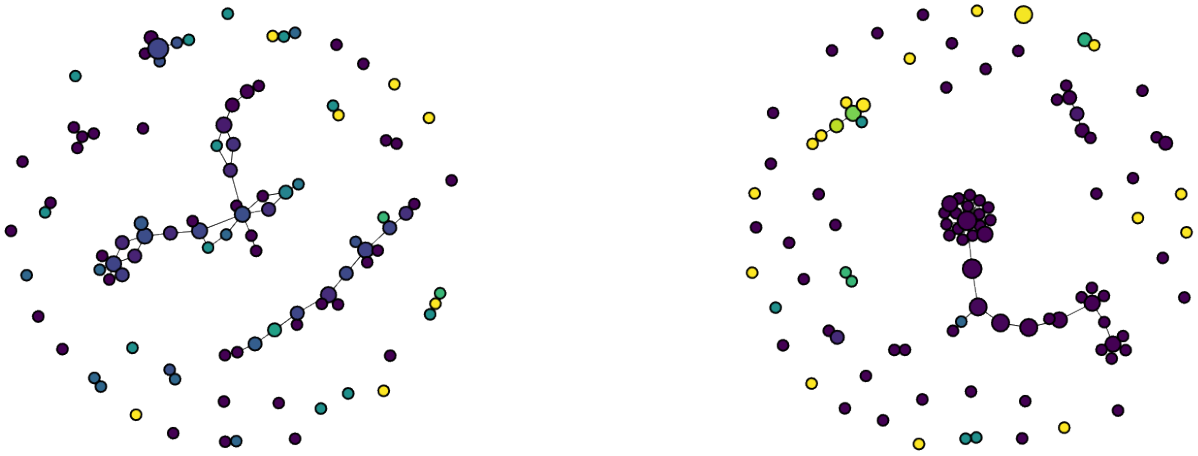


Figure 5: **CP across agreement levels (Base vs Fine-Tuned).** Distribution of component-level purity scores using eq. (1) for each agreement test subset (A0, A+, A++), aggregated across all six 1D lenses. Each point represents the purity of a connected component in a Mapper graph. Fine-tuned representations consistently increase class homogeneity within components, especially in A+ and A++, where label agreement is stronger. Full per-lens distributions are provided in Appendix G.



(a) Before fine-tuning.

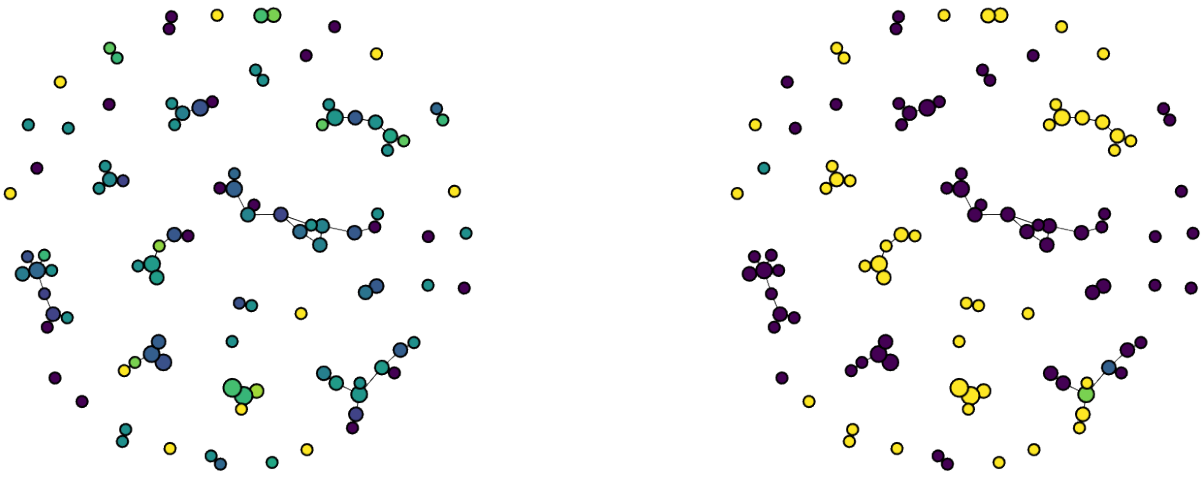
(b) After fine-tuning on the training set.

Figure 3: Mapper graphs of RoBERTa-Large [CLS] embeddings on the A++ test subset, colored by ground-truth labels (non-offensive, offensive, mixed). Both graphs use a PCA-1D lens (40 intervals, 30% overlap) and HDBSCAN clustering. Nodes are considered mixed when no single ground-truth label constitutes a majority among their points. Fine-tuning produces a more structured topology: label purity increases, fragmented components are reduced, and class-aligned subregions emerge, suggesting a modular decision space rather than a single global boundary. Note that visual proximity between nodes does not imply embedding similarity, only edges reflect topological closeness through shared overlap.

Prediction Aligns with Topological Regions. In the most ambiguous setting (A0), where label structure is noisiest, fine-tuned models reveal a striking behavior: prediction regions become geometrically coherent. The base model as shown in Figure 4a produces noisy, fragmented label patterns of embeddings. By contrast, the fine-tuned model as shown in Figure 4b imposes smooth region-level decisions: most connected components yield a single predicted label, suggesting classification operates at the level of regions, not individual points. Only one region, in the bottom right of Figure 4b, shows a gradual transition in predictions: it moves from non-offensive to offensive through a small area with mixed labels. This is a rare example of a visible decision boundary in the embedding space. Most other regions are sharply separated, which suggests that the model makes confident, all or nothing predictions at the level of entire regions, even when the data is ambiguous.

Table 1 quantifies this trend: after fine-tuning, over 98% of components are prediction-pure, even in A0 subset. Despite annotator disagreement, the model confidently projects a single label per region. Yet predictions are not arbitrary, the majority match rate MM remains high (77.7% in A0, above 94% in A+/A++), showing that component-level predictions still align with dominant ground-truth labels. This behavior highlights two effects. First, the model appears to projects test instances into coarse topological regions that reflect its binary fine-tuning objective. Second, while this reflects structural confidence, it also explains the model's overconfidence in ambiguous regions: predictions appear to be driven by dominant signals within a component, even when the underlying labels are mixed or uncertain. In A0 test subset, the model confidently assigns a single prediction to all components.This region-level coherence is not visible in scalar metrics such as accuracy or F1 score, but emerges clearly through Mapper's topological lens. Appendix J presents qualitative examples of low-purity components. More broadly, Mapper provides a diagnostic tool to investigate how models group sentences. For instance, whether such groupings reflect meaningful features or unintended artifacts. While we do not pursue this question here, our results highlight it as a promising direction for future work.

Robustness Across Lenses and Models. To ensure our findings are not tied to a particular setup, we conducted two robustness checks. First, applying 2D lenses produced Mapper graphs and purity trends indistinguishable from the 1D case, as detailed in Appendix K. Second, repeating the analysis with other encoder-only models yielded the same topological patterns observed with RoBERTa-



(a) Ground-truth labels. (b) Model prediction labels.

Figure 4: Mapper graphs of [CLS] embeddings on the A0 test subset (RoBERTa-Large fine-tuned on the training set, PCA-1D lens, $r = 40$, $\epsilon = 0.3$, HDBSCAN). Left: nodes colored by ground-truth labels, right: same graph colored by model prediction labels, (non-offensive, offensive, mixed). Nodes are considered mixed when no single ground-truth label constitutes a majority among their points. Although ground-truth labels appear fragmented (left), the model's predicted labels (right) exhibit smoother, regionally consistent patterns reflecting the model's binary classification training objective. Each topological component tends to predict a consistent class, with fewer intra-cluster flips. This suggests that the model performs prediction at the level of coarse topological regions, even in ambiguous data.

Large: modular prediction aligned regions, near-perfect prediction purity, and the largest prediction label gap in A0 as shown in Appendix L. Together, these checks indicate that our observations are not artifacts of a particular projection or model, but reflect consistent topological behaviors.

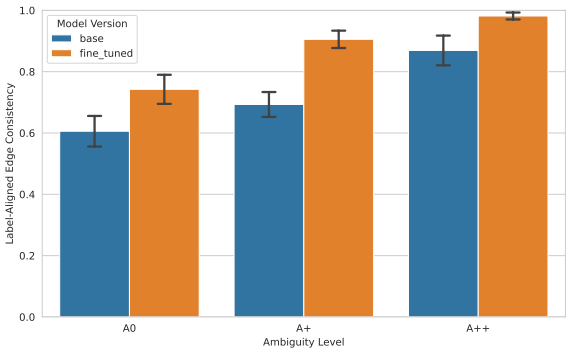


Figure 6: **EA across ambiguity levels (Base vs Fine-Tuned).** Bar plots show the average EA using eq. (2), computed per Mapper graph and aggregated across all 1D lenses for each greement test subset (A0, A+, A++). Fine-tuned embeddings exhibit consistently higher agreement, especially for less ambiguous subsets (A+ and A++), indicating smoother class-consistent transitions between neighboring clusters. Error bars denote standard deviation across lenses. Per-lens edge agreement results are reported in Appendix H, showing that the overall trend holds consistently across all lenses.

6 Conclusion

Mapper reveals that fine-tuning not only compacts and cleans the topology of embedding space, boosting component purity and edge agreement, but also uncovers a modular, non-convex regions that align closely with model predictions. This reorganization is most visible in unambiguous data (A++), where high agreement among annotators leads to clearly defined, coherent regions. In contrast, ambiguous cases (A0) remain fragmented and structurally noisy, yet are often grouped into confident, homogeneous prediction regions. This suggests that models do not merely reflect label noise, they impose their own structured certainty, even in regions where human disagreement dominates. These findings defy common assumptions about model failure in ambiguity. Rather than randomly guessing, models exhibit systematic region-level behavior, suggesting that ambiguity is encoded and resolved in consistent, if not always accurate ways. Whether this resolution reflects true generalization or an overfit to misleading surface patterns remains an open question. Topological analysis thus complements scalar metrics by exposing how models internalize uncertainty and reshape representation space. It highlights not only where models succeed, but how they choose to generalize, whether by col-

Ambiguity	CP (GT) (%)	CP (Pred) (%)	MM (%)
A0	31.5 ± 10.1	98.0 ± 2.4	77.7 ± 20.5
A+	54.8 ± 15.1	98.0 ± 3.2	94.1 ± 12.0
A++	81.8 ± 10.6	99.5 ± 1.2	97.1 ± 3.9

Table 1: Percentage of connected components that exceed the 90% purity threshold, computed using ground-truth labels (CP (GT)) and model predictions labels (CP (Pred)), i.e using eq (1) with corresponding labels. The column MM reports the proportion of components where the predicted majority label matches the ground-truth majority label (see eq. (3)). Results are averaged across all 1D lenses and reported as mean $\pm$ std. Full per-lens results and additional details about variation are provided in Appendix I.

lapsing disagreement into a dominant class, or by carving complex substructures to navigate ambiguous content. Mapper is a promising foundation for developing more reliable and interpretable NLP systems. Beyond post-hoc interpretation, these structures could inform proactive modeling strategies, guiding subset selection, fine-tuning regimes, or robustness evaluations focused on ambiguous regions. By linking embedding geometry, prediction consistency, and human disagreement, Mapper enables a more principled understanding of how models internalize and act on language representations. Leveraging these insights to inform both model development and annotation practices remains a rich direction for future work.

Limitations

While our topological analysis provides novel insights into how fine-tuned models encode class structure and ambiguity, several limitations should be acknowledged.

Mapper Sensitivity and Interpretability. The Mapper algorithm is highly sensitive to its hyperparameters, particularly the lens function, cover granularity, and clustering strategy. Although we use principled heuristics and conduct lens-wise robustness checks, the interpretability of resulting graphs still depends on these design choices. Topological patterns should therefore be interpreted in relative terms (e.g., trends across training conditions), not as fixed geometric ground truths.

Task and Dataset Specificity. Our experiments focus on a single dataset and task (offensive language classification on MD-Offense). While this setup allows controlled analysis, it limits generalizability. It remains an open question whether similar topological phenomena, such as disconnected decision regions or regional predictive consistency, would emerge in other tasks (e.g., natural language inference, sentiment analysis) or with datasets exhibiting different forms of ambiguity.

Lens Expressiveness and Task-Specificity. Our analysis primarily relied on simple, interpretable lenses. While we extended the study to 2D projections , our focus was on general purpose constructions such as PCA, centroid distances, and eccentricity. Task specific or adaptive lenses for example, saliency guided projections, uncertainty aware mappings, or semantics driven nonlinear transformations may expose different structures, particularly in regions where ambiguity arises from subtle pragmatic or contextual cues. Identifying lens designs tailored to the sources of ambiguity in language remains an open direction for future work.

Metric Limitations. The structural metrics we propose, component purity, edge agreement, and majority match, capture interpretable geometric trends, but are Mapper-specific and do not directly reflect classification accuracy or generalization. Moreover, they are inherently correlational: improvements in structure after fine-tuning may reflect better task learning, but could also result from overfitting or feature collapse.

Toward Generalizable Topological NLP. These limitations are not unique to our study but reflect broader challenges in adapting TDA to NLP. They point to the need for hybrid pipelines combining topological summaries with statistical validation, comparisons to alternative graph structures, and multi-model, multi-task replication.

References

Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim. 2001. On the Surprising Behavior of Distance Metrics in High Dimensional Space. In Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, Jan Van Den Bussche, and Victor Vianu, editors, *Database Theory — ICDT 2001*, volume 1973, pages 420–434. Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. ISOTROPY IN THE CONTEXTUAL EMBEDDING SPACE: CLUSTERS AND MANIFOLDS.

Gunnar Carlsson. 2009. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308.

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. Visualizing and Measuring the Geometry of BERT. *arXiv preprint*. ArXiv:1906.02715 [cs].

David Donoho. 2000. High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. *AMS Math Challenges Lecture*, pages 1–32.

Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Stephen Fitz, Peter Romero, and Jiyan Jonas Schneider. 2024. Hidden Holes: topological aspects of language models. *arXiv preprint*. ArXiv:2406.05798 version: 1.

Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 2021. Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2591–2597, Online. Association for Computational Linguistics.

Yuri Gardinazzi, Giada Panerai, Karthik Viswanathan, Alessio Ansuini, Alberto Cazzaniga, and Matteo Biagetti. 2024. Persistent Topological Features in Large Language Models. *arXiv preprint*. ArXiv:2410.11042 [cs].

Hui Guan, Wen Tang, Hamid Krim, James Keiser, Andrew Rindos, and Radmila Sazdanovic. 2016. A topological collapse for document summarization. In *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5, Edinburgh, United Kingdom. IEEE.

Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. 2008. Exploring Network Structure, Dynamics, and Function using NetworkX. pages 11–15, Pasadena, California.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. *Preprint*, arXiv:1909.03368.

John Hewitt and Christopher D. Manning. 2019. A Structural Probe for Finding Syntax in Word Representations. 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)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Elisa Leonardelli, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, Alexandra Uma, and Massimo Poesio. 2023. SemEval-2023 Task 11: Learning with Disagreements (LeWiDi). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2304–2318, Toronto, Canada. Association for Computational Linguistics.

Elisa Leonardelli, Stefano Menini, Alessio Palmero Aprosio, Marco Guerini, and Sara Tonelli. 2021. Agreeing to Disagree: Annotating Offensive Language Datasets with Annotators' Disagreement. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10528–10539, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tianhua Liao, Yuchen Wei, Mingjing Luo, Guo-Ping Zhao, and Haokui Zhou. 2019. tmap: an integrative framework based on topological data analysis for population-scale microbiome stratification and association studies. *Genome Biology*, 20(1):293.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2023. We're Afraid Language Models Aren't Modeling Ambiguity. *arXiv preprint*. ArXiv:2304.14399 [cs].

P. Y. Lum, G. Singh, A. Lehman, T. Ishkanov, M. Vejdemo-Johansson, M. Alagappan, J. Carlsson, and G. Carlsson. 2013. Extracting insights from the shape of complex data using topology. *Scientific Reports*, 3(1):1236.

Vine Nwabuisi Madukpe, Bright Chukwuma Ugoala, and Nur Fariha Syaqina Zulkepli. 2025. A Comprehensive Review of the Mapper Algorithm, a Topological Data Analysis Technique, and Its Applications Across Various Fields (2007-2025). *arXiv preprint*. ArXiv:2504.09042 [math].

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, and James Melville. 2020. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint*. ArXiv:1802.03426 [stat].

Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. 2022. Dealing with Disagreements: Looking Beyond the Majority Vote in Subjective Annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110. Place: Cambridge, MA Publisher: MIT Press.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. *arXiv preprint*. ArXiv:1705.08039 [cs].

Yixin Nie, Xiang Zhou, and Mohit Bansal. 2020. What Can We Learn from Collective Human Opinions on Natural Language Inference Data? *arXiv preprint*. ArXiv:2010.03532 [cs].

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent Disagreements in Human Textual Inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694. Place: Cambridge, MA Publisher: MIT Press.

Sara Rajaee and Mohammad Taher Pilehvar. 2021. How Does Fine-tuning Affect the Geometry of Embedding Space: A Case Study on Isotropy. *arXiv preprint*. ArXiv:2109.04740 [cs].

Archit Rathore, Yichu Zhou, Vivek Srikumar, and Bei Wang. 2023. TopoBERT: Exploring the topology of fine-tuned word representations. *Information Visualization*, 22(3):186–208.

Gurjeet Singh, Facundo Mémoli, Gunnar E Carlsson, et al. 2007. Topological methods for the analysis of high dimensional data sets and 3d object recognition. *PBG@ Eurographics*, 2:091–100.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset Cartography: Mapping and Diagnosing Datasets with Training Dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics.

Hendrik Jacob van Veen, Nathaniel Saul, David Eargle, and Sam W. Mangham. 2021. Kepler Mapper: A flexible Python implementation of the Mapper algorithm.

Hendrik Jacob van Veen, Nathaniel Saul, David Eargle, and Sam W. Mangham. 2019. Kepler Mapper: A flexible Python implementation of the Mapper algorithm. *Journal of Open Source Software*, 4(42):1315.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online. Association for Computational Linguistics.

Ruyuan Wan, Jaehyung Kim, and Dongyeop Kang. 2023. Everyone's Voice Matters: Quantifying Annotation Disagreement Using Demographic Information. *arXiv preprint*. ArXiv:2301.05036 [cs].

Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1):37–52.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. *arXiv preprint*. ArXiv:1611.03530 [cs].

Yichu Zhou and Vivek Srikumar. 2021. DirectProbe: Studying Representations without Classifiers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5070–5083, Online. Association for Computational Linguistics.

Appendix

A Data Representation

Table 2 summarizes the number of samples and the proportion of offensive examples across dataset splits (train, validation, test) and agreement levels (A0, A+, A++) based on the ground-truth labels. Offensive content is more frequent in the A0 subsets, which may help explain the higher levels of annotator disagreement observed in these cases.

Split	Agreement Level	Total Samples	Offensive Ratio (%)
Train	A0	1884	45.2
	A+	1930	32.9
	A++	2778	17.1
Validation	A0	322	46.9
	A+	317	37.2
	A++	465	25.6
Test	A0	856	45.1
	A+	909	39.6
	A++	1292	21.1

Table 2: Distribution of offensive labels across MD-Offense, grouped by split and agreement level.

B Model-Level Summary

Table 3 presents test set performance and embedding structure across ambiguity levels (A0, A+, A++) for all fine-tuned models. Accuracy and Macro F1 reflect standard classification performance, while silhouette score variance and Davies–Bouldin index (lower is better) capture structural coherence in embedding space. All metrics are averaged over three random seeds. Best accuracy values are highlighted in bold. RoBERTa-Large consistently achieves strong results across ambiguity levels, motivating its selection for our topological analysis.

Model	Agreement Level	Accuracy	F1 Score	Silhouette (±) ↑	DB Index (±) ↓
BERT-Base	A0	0.607	0.602	0.327 ± 0.010	5.07 ± 0.39
	A+	0.810	0.801	0.465 ± 0.008	1.54 ± 0.03
	A++	0.934	0.904	0.782 ± 0.014	0.83 ± 0.04
DeBERTa-Base	A0	0.602	0.596	0.315 ± 0.002	5.58 ± 0.11
	A+	0.819	0.812	0.478 ± 0.012	1.41 ± 0.02
	A++	0.943	0.916	0.801 ± 0.014	0.73 ± 0.04
DistilBERT	A0	0.606	0.598	0.329 ± 0.006	5.43 ± 0.30
	A+	0.811	0.801	0.472 ± 0.012	1.56 ± 0.03
	A++	0.941	0.912	0.789 ± 0.007	0.84 ± 0.02
E5-Base	A0	0.612	0.608	0.431 ± 0.003	5.60 ± 0.15
	A+	0.809	0.800	0.489 ± 0.021	1.70 ± 0.11
	A++	0.935	0.906	0.798 ± 0.027	0.92 ± 0.11
MiniLM	A0	0.600	0.596	0.312 ± 0.003	4.56 ± 0.13
	A+	0.790	0.783	0.461 ± 0.002	1.43 ± 0.03
	A++	0.942	0.916	0.805 ± 0.009	0.65 ± 0.03
RoBERTa-Large	A0	**0.615**	0.608	0.441 ± 0.001	4.45 ± 0.36
	A+	**0.822**	0.814	0.502 ± 0.041	1.16 ± 0.06
	A++	**0.954**	0.932	0.825 ± 0.045	0.51 ± 0.04

Table 3: Test performance and representation metrics for each fine-tuned model.

C Lens Details

We define the 1D lens functions $f(x) : \mathbb{R}^d \to \mathbb{R}$ used to project test embeddings $x \in \mathbb{R}^d$ into scalar values for Mapper analysis. These lens functions serve as the filtering mechanism that guides Mapper's overlapping cover of the data space. Each function highlights a specific geometric or semantic property of the representation space, such as class separation, density, or position relative to cluster centroids. All lens values are computed using statistics from the training set and rescaled to the $[0, 1]$ interval to ensure consistent binning and comparability across lenses.

We use a diverse set of lens types: supervised projections based on class centroids, unsupervised structure via PCA, local and global density indicators like eccentricity or L2 norm, and random directions that simulate generic perspectives of the space. This variety enables a multifaceted topological analysis, helping ensure that observed Mapper structures are not artifacts of a particular projection choice.
Let $x \in \mathbb{R}^d$ be thereafter the embedding of a test instance.

- **Centroid Projection (`centroid_1d`)** Projects x onto the vector connecting the means of the embeddings (centroids) of the two training classes:

$$f(x) = \frac{x^\top(\mu_1 - \mu_0)}{\|\mu_1 - \mu_0\|},$$

  with μ_0 and μ_1 the centroids, i.e. the mean vectors, of the non-offensive $(y = 0)$ and offensive $(y = 1)$ training embeddings subsets, respectively:

$$\mu_y = \frac{1}{|\mathcal{X}_y|} \sum_{x_i \in \mathcal{X}_y} x_i,$$

  where $\mathcal{X}_y \subset \mathbb{R}^d$ is the subset of training embeddings labeled either $y = 0$ or $y = 1$. The resulting scalar $f(x)$ reflects how aligned the embedding of the test instance x is with the class separation direction.

- **PCA First Component (`pca_1d`)** Projects x onto the first principal component u_1 of the training set:

$$f(x) = u_1^\top x,$$

  where $u_1 \in \mathbb{R}^d$ is the direction of maximal variance estimated by applying PCA on the entire training dataset in the embeddings space. This projection captures the dominant geometric trend in the data and often reflects task-relevant variation.

- **Eccentricity (`eccentricity_1d`)** Measures how distant x is from the training set, using maximum cosine distance:

$$f(x) = \max_{x_i \in \mathcal{X}_{\texttt{train}}} \left(1 - \frac{x \cdot x_i}{\|x\| \, \|x_i\|}\right),$$

  where $\mathcal{X}_{\texttt{train}}$ denotes the set of all training embeddings. High values of $f(x)$ indicate that the test point is geometrically distant (i.e., eccentric) from the training distribution.

- **L2 Norm (`l2norm_1d`)** Measures the Euclidean norm (magnitude) of x :

$$f(x) = \|x\|_2 = \sqrt{\sum_{i=1}^{d} x_i^2}.$$

  This lens reflects how far the point lies from the origin in embedding space and may capture differences in representation scale or intensity.

- **Random Projections (`random1_1d` and `random2_1d`)** Projects x onto random direction

$$f(x) = r^\top x$$

  where $r \in \mathbb{R}^d$ is a fixed random unit vector sampled once before projection. This lens selects a random direction in the embedding space and projects each test point along it. Different random seeds yield different perspectives of the geometry.

D Varying Mapper Covering Resolution

Figure 7 shows the effect of increasing the number of intervals r in the Mapper covering on the same dataset subset (A++) using the `random1_1d` lens. All graphs were generated with the same HDBSCAN configuration and a fixed overlap of $\epsilon = 0.3$. As r increases from 10 to 90, the representation space becomes more finely partitioned. Low values (e.g., 10–20) result in overly coarse graphs that merge distinct regions and obscure local structure. In contrast, high values (e.g., 80–90) lead to significant fragmentation, with many disconnected or singleton components. This reflects how excessive resolution can distort the underlying topological signal. A middle setting, such as r = 30 or 40, consistently offers a good trade-off between local granularity and global connectivity. While this trend holds across models and ambiguity levels, the degree of fragmentation also depends on the lens and the noisiness of the embedding space. These hyperparameter choices are thus best calibrated in relation to both the geometry of the data and the interpretability needs of the analysis.

E Varying Mapper Overlap

Figure 8 illustrates the effect of increasing Mapper overlap from 10% to 90%, using the `random1_1d` lens on the A++ subset and a fixed resolution of r = 40. All graphs were generated with identical HDBSCAN settings to isolate the effect of overlap alone. At very low values (10%–20%), the covering becomes overly rigid, leading to severe fragmentation: adjacent regions are treated as disconnected, resulting in a large number of small or isolated components. As overlap increases to 30%–40%, local continuity improves and previously disconnected regions begin to merge into coherent graph structures. This is the range where Mapper begins to meaningfully recover the underlying geometry without inflating the graph. Beyond 50%, however, the increasing redundancy from duplicated points across overlapping bins causes a rapid inflation in graph complexity. This leads to spurious edges, tightly packed node clusters, and unnatural fan-like structures, particularly visible at 80%–90%. These artifacts distort the topology and reduce interpretability by over-amplifying minor variations. Overall, the 30%–40% overlap range provides the best trade-off: it allows components to remain connected without over-smoothing the embedding space. This setting is therefore used consistently in our Mapper analyses.

F HDBSCAN Noise Exclusion by Lens

Table 4 reports the proportion of test points labeled as noise by HDBSCAN across 1D lenses and ambiguity levels. Eccentricity consistently yields the lowest exclusion rates often under 5%, reflecting its inclusive behavior due to its definition as the maximum cosine distance to the training set. It captures structurally atypical inputs without discarding them. Random projections also result in minimal exclusion, as expected from their agnostic geometric alignment.

In contrast, task-oriented lenses such as `centroid_1d` and `pca_1d` exhibit noticeably higher noise after fine-tuning. These lenses emphasize class-separating directions, and the increase in excluded points likely reflects sharper representational boundaries introduced during supervised optimization. Ambiguous or boundary-adjacent instances are more frequently labeled as noise in these projections.

These patterns align with our overall claims: fine-tuning enhances topological regularity but also increases exclusion in class-aligned projections. Across lenses, noise remains within acceptable bounds and does not distort Mapper structure. HDBSCAN thus effectively balances robustness and interpretability in our setup.

G Violin plots per lens for component purity

This figure complements Figure 5 by breaking down purity distributions per lens. Despite variation in Mapper construction, the core trend holds: fine-tuning improves the local class consistency (purity) of topological components across all lens types. This robustness supports the interpretation that observed topological shifts reflect meaningful representational changes rather than artifacts of a specific projection.

Lens	Base Noise (%)			Fine-tuned Noise (%)		
	A0	A+	A++	A0	A+	A++
centroid_1d	20.1%	9.5%	8.3%	26.6%	27.4%	20.0%
pca_1d	20.7%	13.9%	9.3%	26.0%	26.7%	20.3%
eccentricity_1d	10.0%	8.5%	5.7%	3.8%	2.1%	2.1%
l2norm_1d	21.3%	12.4%	19.1%	5.6%	5.8%	9.4%
random1_1d	2.9%	1.7%	1.2%	7.8%	9.2%	10.5%
random2_1d	2.2%	1.9%	2.3%	6.3%	6.5%	7.1%

Table 4: Proportion of test points labeled as noise by HDBSCAN for each 1D lens, reported inline by ambiguity level (A0, A+, A++). Values correspond to final RoBERTa-Large representations and are not averaged across seeds.

H Edge Agreement per Lens

Figure 10 displays edge agreement scores disaggregated by lens type. While the main paper aggregates across lenses to report overall trends, this breakdown reveals how lens choice influences topological structure. All lenses benefit from fine-tuning, but the extent of improvement varies: more structured lenses like `eccentricity_1d` and `l2norm_1d` show clearer class-consistent transitions than random projections. These differences highlight the importance of lens selection when using Mapper to interpret model geometry.

I Per-Lens Analysis of Component Purity and Prediction Alignment

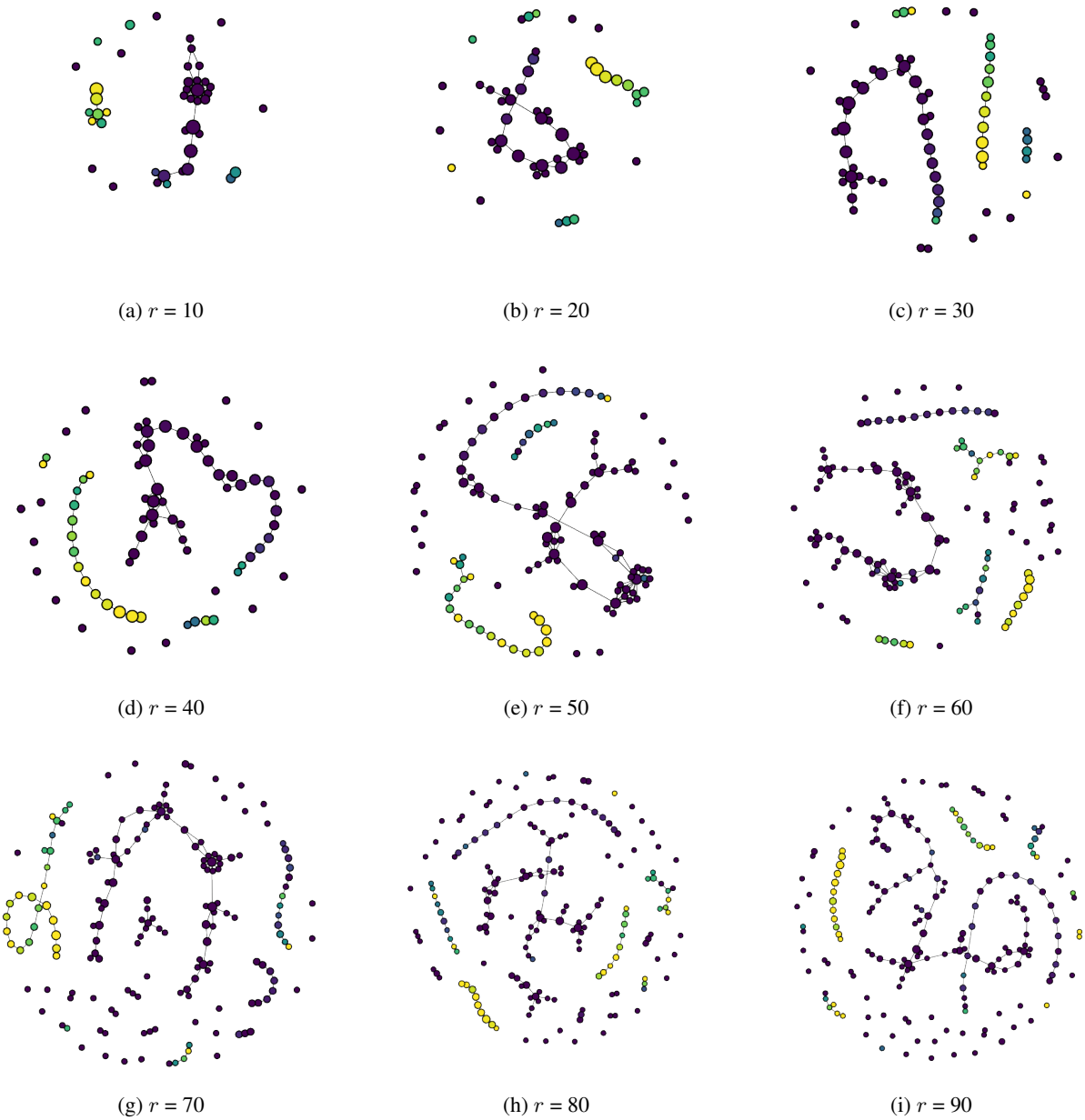
Table 5 reports component-level metrics broken down by ambiguity level and Mapper lens type. While the main paper focuses on averages aggregated across all lenses (Table 1) for robustness, this detailed view helps interpret the observed variance and highlights lens-specific behavior.

Not all lenses yield equally structured Mapper graphs. For example, `pca_1d` and `centroid_1d` consistently produce high-quality decompositions, with strong purity and high majority match across ambiguity levels. In contrast, lenses like `l2norm_1d` or `random2_1d` sometimes produce overly coarse or fragmented graphs, especially on ambiguous data (A0). This leads to volatile purity scores and sharp drops in alignment with ground truth. For example, only 50% majority match for `l2norm_1d` in A0. Interestingly, even lenses like `eccentricity_1d`, which exhibit consistently high prediction purity and perfect majority match, can do so while capturing only limited ground-truth variation. These cases underscore that high predictive agreement does not always imply faithful alignment with human annotations, especially in ambiguous regions.

Such differences illustrate why lens choice matters: while Mapper preserves more structure than projection-based methods, the specific lens determines which aspects of the geometry are emphasized. This explains the standard deviations reported in Table 1 and motivates our decision to report aggregated trends in the main analysis, while offering full per-lens breakdowns here for transparency and interpretability.

Ambiguity	Lens	CP (GT) (%)	CP (Pred) (%)	MM (%)
A0	centroid_1d	40.0	100.0	81.2
	pca_1d	42.5	97.5	75.0
	eccentricity_1d	28.6	100.0	100.0
	l2norm_1d	15.4	100.0	50.0
	random1_1d	36.4	95.5	100.0
	random2_1d	26.3	94.7	60.0
A+	centroid_1d	50.0	100.0	100.0
	pca_1d	60.6	100.0	100.0
	eccentricity_1d	33.3	93.3	100.0
	l2norm_1d	50.0	100.0	100.0
	random1_1d	79.2	100.0	94.7
	random2_1d	55.6	94.4	70.0
A++	centroid_1d	82.9	100.0	96.6
	pca_1d	89.8	100.0	100.0
	eccentricity_1d	66.7	100.0	100.0
	l2norm_1d	83.3	100.0	100.0
	random1_1d	95.2	100.0	90.0
	random2_1d	72.7	97.0	95.8

Table 5: Per-lens breakdown of component-level metrics: percentage of components exceeding the 90% purity threshold for ground-truth and model predictions, and prediction–label majority match among pure components (see eq. (3)).

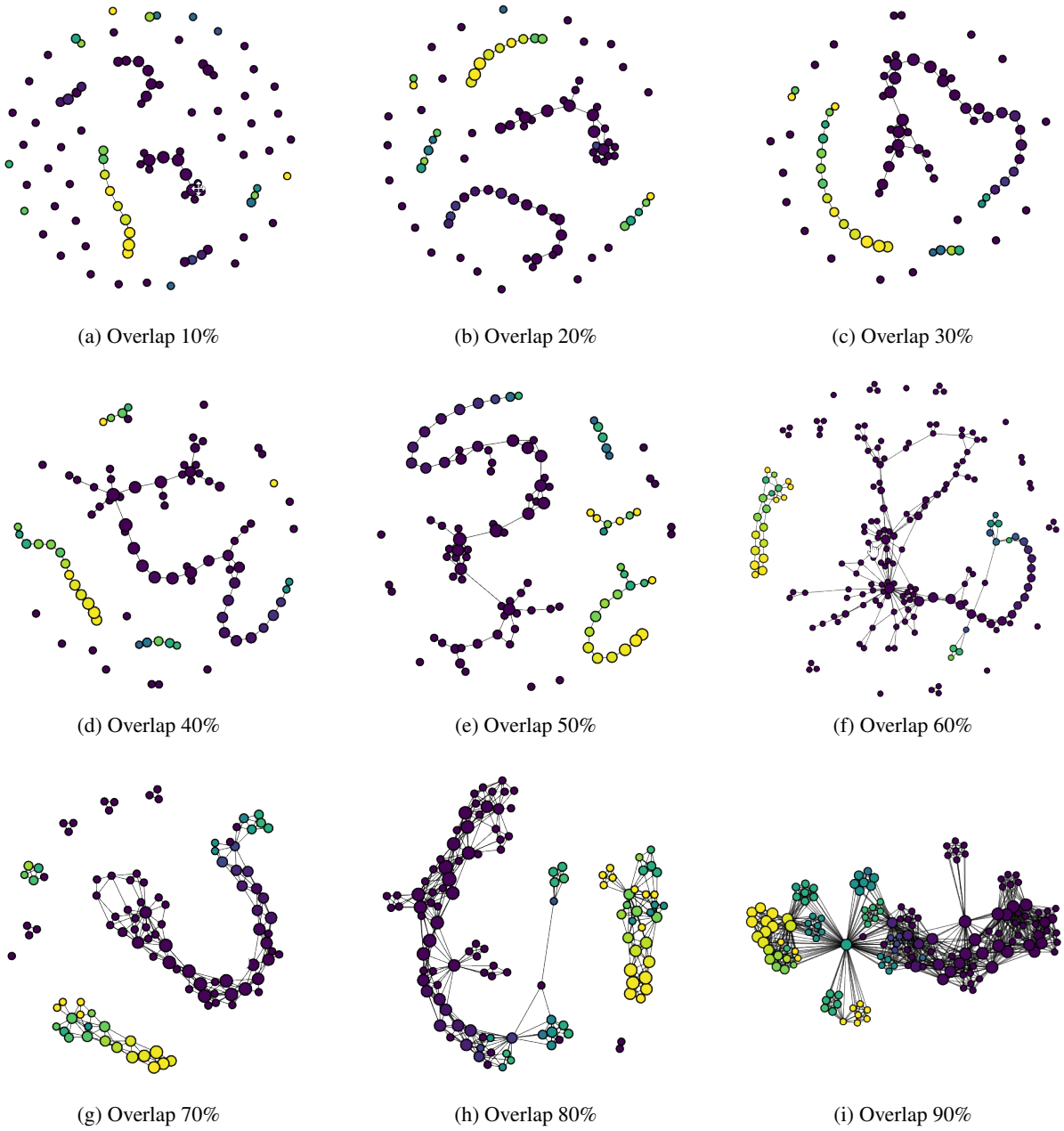


(a) $r = 10$ (b) $r = 20$ (c) $r = 30$

(d) $r = 40$ (e) $r = 50$ (f) $r = 60$

(g) $r = 70$ (h) $r = 80$ (i) $r = 90$

Figure 7: Effect of increasing the number of intervals r in the Mapper covering. Higher resolution introduces finer granularity and reveals smaller connected regions, but can also lead to fragmentation and spurious nodes. All graphs use the same lens and fixed overlap ($\epsilon = 0.3$).



(a) Overlap 10% (b) Overlap 20% (c) Overlap 30%

(d) Overlap 40% (e) Overlap 50% (f) Overlap 60%

(g) Overlap 70% (h) Overlap 80% (i) Overlap 90%

Figure 8: Effect of varying Mapper overlap ϵ from 0.1 to 0.9 on the A++ subset, using the `random1_1d` lens and fixed resolution ($r = 40$). Low overlap causes fragmentation, while high overlap leads to inflated graphs. Overlap around 30% provides the best trade-off.

Figure 9: Component purity distributions across 1D lenses. Each subplot shows the purity of connected components for base vs fine-Tuned models, separated by ambiguity level (A0, A+, A++). Fine-tuning consistently increases purity across all lenses, particularly for less ambiguous data (A+ and A++). Improvements are visible even for random lenses, suggesting the effect is not lens-specific. However, A0 remains harder to organize topologically, with more dispersed purity scores.

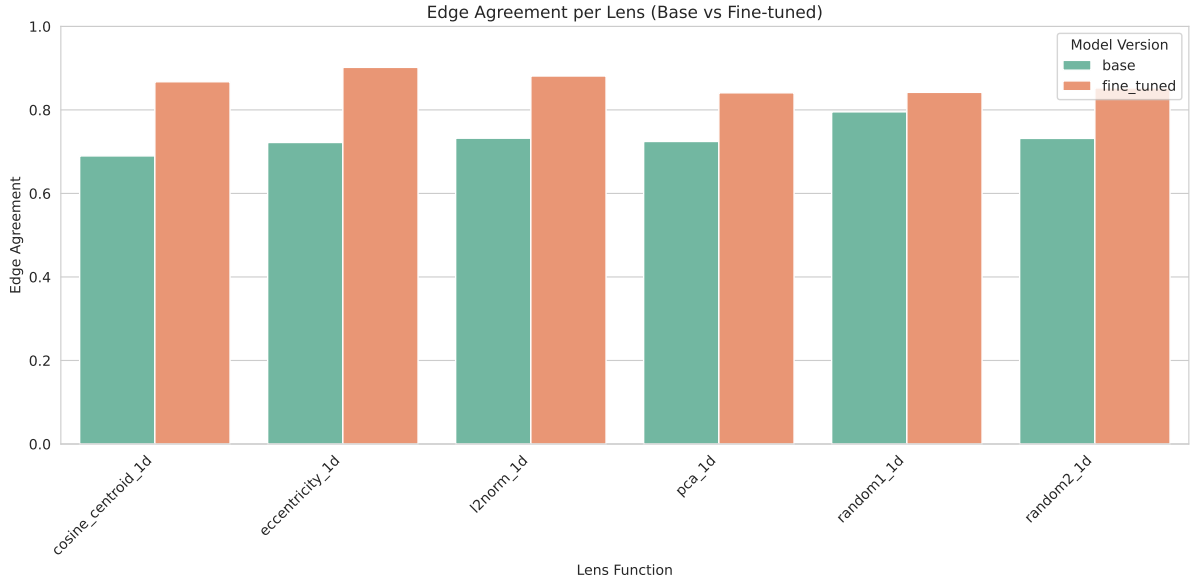


Figure 10: Edge agreement across lenses for base and fine-tuned RoBERTa-Large models. Each bar reports the average proportion of edges connecting nodes with the same majority class label, across Mapper graphs constructed using different 1D lens functions. Fine-tuned models consistently exhibit higher edge agreement across all lenses, reflecting smoother topological class boundaries.

J Qualitative Examples from Cluster Analysis

To better understand how models handle ambiguous or low-consensus regions, we examine individual examples from Mapper components with low ground-truth purity (i.e., clusters mixing offensive and non-offensive samples). All predictions are from the fine-tuned RoBERTa-Large model.

These examples show that consistent model predictions often emerge even when annotators disagree. In some cases, this may reflect the model's reliance on strong surface cues, in others, it may suggest that distinct linguistic phenomena (e.g., sarcasm, political critique, racial discourse) are projected into overlapping regions of embedding space.

Mapper allows us to identify and explore such structures. Whether these regions reflect meaningful generalization or model overconfidence remains an open question. By surfacing how instances are grouped and predicted together, Mapper provides a useful entry point for deeper error analysis and for assessing how models internalize disagreement.

Example 1 (Domain 1 - Social Issues/Race)
Text: *"Does this black life matter to BLM? [...] BLACK AMERICA, look at what Sharpton is doing to you, YOU MEAN NOTHING TO HIM."*
Annotations: `1,1,0,1,0` (3 offensive, 2 non-offensive)
Ground-truth label: Offensive (1)
Model prediction label: Non-offensive (0) with **97.2% confidence**

Example 2 (Domain 0 - Political Discourse)
Text: *"President Trump's rallies [...] Have you not even seen Dementia-Joe??"*
Annotations: `0,0,1,0,1` (2 offensive, 3 non-offensive)
Ground-truth label: Non-offensive (0)
Model prediction label: Non-offensive (0) with **97.1% confidence**

Example 3 (Domain 2 - COVID-19/Health)
Text: *"We will not survive 2 more years of Gov DeSantis; he's really out here tryna kill people [...]"*
Annotations: `1,1,1,0,0` (3 offensive, 2 non-offensive)
Ground-truth label: Offensive (1)
Model prediction label: Non-offensive (0) with **96.4% confidence**

K Robustness with 2D Lenses

While the main study emphasized one-dimensional lenses for their interpretability and exploratory value, here we provide supplementary results using two-dimensional projections. Consistent with the 1D case, fine-tuned embeddings yield near-perfect prediction purity, while label purity decreases most sharply in A0. These trends hold across all five 2D lenses tested (Table 6), with component purity distributions and edge agreement metrics (Figures 12, 11) confirming that, for this dataset, the structural organization revealed by Mapper is stable across projection dimensionality. For 2D lenses, we set Mapper hyperparameters to $r = 20$ and $\epsilon = 0.2$ to ensure graphs were neither too fragmented nor too connected, following the same consistency checks as in the 1D case. While additional or task-specific lenses (e.g., 3D or uncertainty-aware projections) might expose different structures, our results suggest that the core topological behavior is consistent across 1D and 2D lenses in this setting.

Ambiguity	Lens	CP (GT) (%)	CP (Pred) (%)	MM (%)
A0	cosine_c0_c1	35.4	100.0	88.2
	eccentricity_2d	35.4	100.0	88.2
	pca_2d	44.2	100.0	68.4
	random_2d	25.7	100.0	77.8
	umap_2d	39.2	95.9	77.8
A+	cosine_c0_c1	71.9	100.0	100.0
	eccentricity_2d	71.9	100.0	100.0
	pca_2d	63.8	100.0	94.6
	random_2d	69.0	100.0	95.0
	umap_2d	66.2	97.5	94.3
A++	cosine_c0_c1	81.1	100.0	100.0
	eccentricity_2d	81.1	100.0	100.0
	pca_2d	82.0	100.0	100.0
	random_2d	73.5	100.0	100.0
	umap_2d	86.3	100.0	100.0

Table 6: Percentage of connected components that exceed the 90% purity threshold, computed using ground-truth labels (CP (GT)) and model predictions labels (CP (Pred)), i.e using eq (1) with corresponding labels. The column MM reports the proportion of components where the predicted majority label matches the ground-truth majority label.

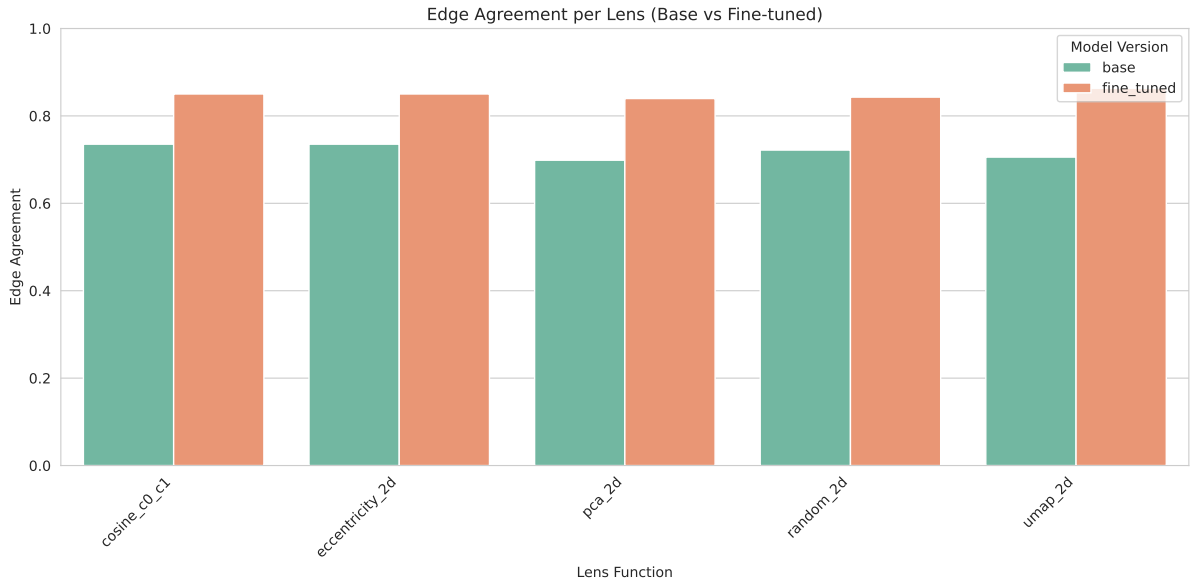


Figure 11: Edge agreement across 2D lenses for base and fine-tuned RoBERTa-Large models. Fine-tuning consistently increases edge agreement, indicating smoother and more coherent decision boundaries.

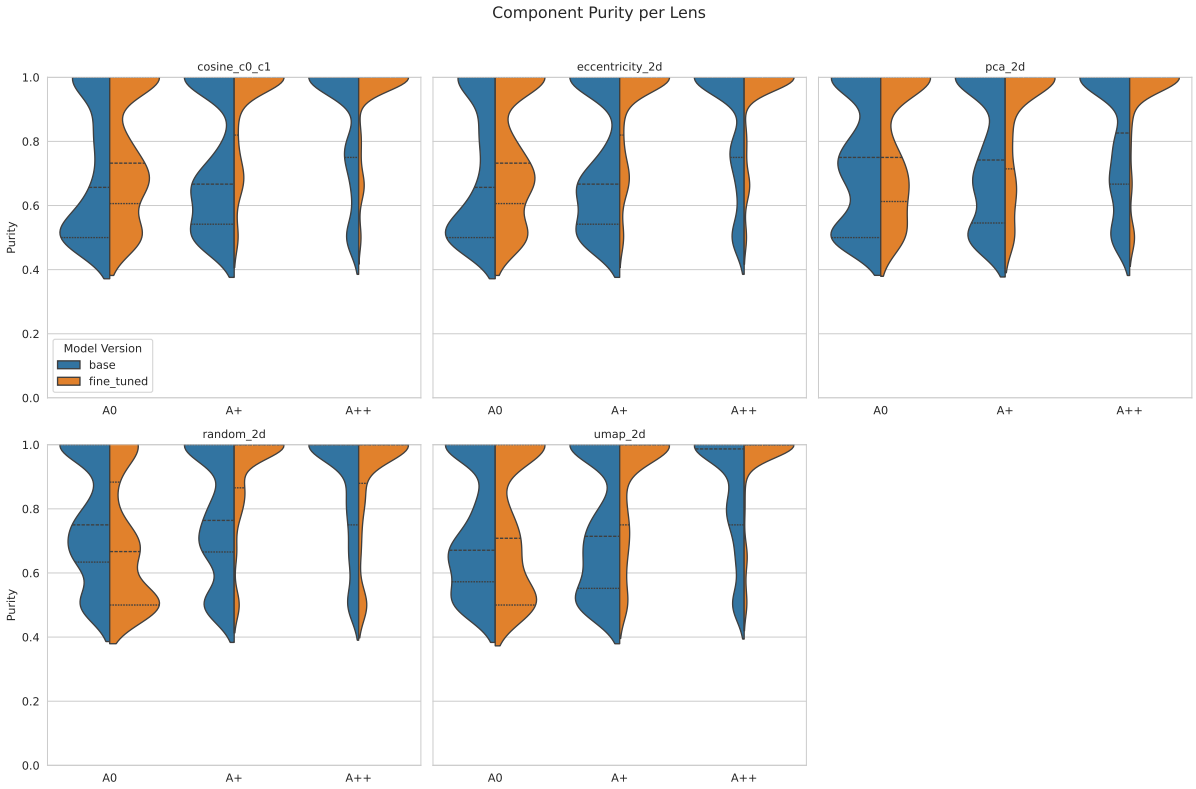


Figure 12: Distribution of component purity across 2D lenses for RoBERTa-Large. Consistent with the 1D case, fine-tuned embeddings yield near-perfect prediction purity, while label purity decreases most sharply in A0.

L Robustness Across Models

To test robustness across architectures, we repeated the Mapper analysis on additional encoder-only models: BERT-Base, DeBERTa-Base, DistilBERT, E5-Base, and MiniLM, alongside RoBERTa-Large. Table 7 reports component purity aggregated across lenses, with values shown as mean ± standard deviation. Across all models, the qualitative pattern is consistent: prediction purity remains higher than label purity, and the largest prediction label gap occurs in A0. This indicates that models tend to impose coherent predictions at the level of topological regions, following dominant signals within components rather than the full variability of annotator labels. While smaller models such as BERT-Base and DistilBERT achieve lower absolute prediction purity, their overall topological organization mirrors that of RoBERTa-Large, confirming that our findings are not model-specific.

Ambiguity	**Lens**	**CP (GT) (%)**	**CP (Pred) (%)**	**MM (%)**
RoBERTa-Large	A0	31.5 ±10.1	98.0 ± 2.4	77.7 ± 20.5
	A+	54.8± 15.1	98.0 ± 3.2	94.1 ± 12.0
	A++	81.8 ±10.6	99.5 ± 1.2	97.1± 3.9
BERT-Base	A0	30.2 ± 10.7	59.2 ± 19.0	67.0 ± 4.9
	A+	42.8 ± 18.4	54.2 ± 17.6	77.0 ± 8.2
	A++	68.1 ± 10.2	68.6 ± 7.2	86.3 ± 8.8
DeBERTa-Base	A0	31.1 ± 9.1	88.4 ± 10.7	68.2 ± 9.4
	A+	52.9 ± 21.7	95.1 ± 5.5	85.8 ± 9.3
	A++	72.4 ± 15.5	95.0 ± 4.6	82.6 ± 8.5
DistilBERT	A0	28.3 ± 12.5	62.8 ± 17.9	54.8 ± 15.4
	A+	48.2 ± 8.2	65.6 ± 6.9	69.1 ± 5.3
	A++	65.0 ± 15.2	66.7 ± 19.7	85.0 ± 10.6
E5-Base	A0	31.7 ± 6.2	78.3 ± 14.1	56.3 ± 6.5
	A+	48.5 ± 15.0	80.8 ± 17.1	68.6 ± 12.6
	A++	69.6 ± 10.2	91.1 ± 6.3	74.9 ± 11.5
MiniLM	A0	34.5 ± 5.4	83.3 ± 13.2	61.5 ± 2.6
	A+	48.6 ± 15.8	93.0 ± 5.5	77.9 ± 7.1
	A++	73.3 ± 16.8	94.5 ± 4.3	86.5 ± 9.6

Table 7: Percentage of connected components that exceed the 90% purity threshold, computed using ground-truth labels (CP (GT)) and model predictions labels (CP (Pred)), i.e using eq (1) with corresponding labels. The column MM reports the proportion of components where the predicted majority label matches the ground-truth majority label (see eq. (3)). Results are averaged across all 1D lenses.