Enhancing Extreme Multi-label Classification: Integrating Explainability and Adversarial Robustness into MACH

B22ai026 - khushi Bhardwaj
B22ai039 - Siramsetty Indusri
IIT JODHPUR

Abstract

Extreme multi-label classification (XML) requires assigning multiple labels among millions of classes and poses significant challenges with respect to scalability and reliability. MACH is a recent algorithm that efficiently compresses the high-dimensional label space using a Count-Min Sketch (CMS) and trains independent networks to predict each CMS column. In this paper, we extend MACH with model explainability, via Integrated Gradients, and enhance robustness via adversarial training. Our experiments on publicly available XML datasets (notably Amazon-670K) demonstrate competitive performance along with improved interpretability and resilience against adversarial perturbations.

Keywords

Extreme Multi-label Classification, MACH, Integrated Gradients, Adversarial Robustness, Count-Min Sketch, Model Interpretability

ACM Reference Format:
B22ai026 - khushi Bhardwaj and B22ai039 - Siramsetty Indusri. 2025. Enhancing Extreme Multi-label Classification: Integrating Explainability and Adversarial Robustness into MACH. In *Proceedings of group 28 (YourConf 'XX')*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Classification is one of the most fundamental tasks in machine learning. In conventional settings, a single label is assigned to each sample, but many applications require multi-label classification where samples belong to multiple classes. With the number of labels growing to hundreds of thousands or millions, the problem becomes one of extreme classification (XML).

MACH (Medini et al., 2019) addresses XML by compressing the high-dimensional label vector (which can exceed 670K dimensions) into a low-dimensional bucket space using multiple hash functions and a Count-Min Sketch (CMS). Independent models predict the values in each bucket, greatly reducing memory and computational requirements. However, the original MACH implementation does not offer the advantages of model explainability or robustness against adversarial perturbations—a concern in many practical applications.

YourConf 'XX', City, Country

ACM ISBN 978-1-4503-XXXX-X/2025/XX
https://doi.org/10.1145/nnnnnnn.nnnnnnn

In this paper, we extend MACH along two significant dimensions:

(1) **Model Explainability:** We integrate Integrated Gradients to provide feature-level attributions, highlighting which input features are responsible for the predictions.
(2) **Adversarial Robustness:** We incorporate adversarial training using the Fast Gradient Sign Method (FGSM) to mitigate the impact of small but malicious input perturbations.

Our approach is evaluated on publicly available datasets from the Extreme Classification Repository, particularly Amazon-670K.

2 Related Work

Extreme classification has been addressed with approaches based on tree structures and label embeddings (Bhatia et al., 2015). MACH [1] innovates through the use of CMS to tackle the enormous label space. Recently, methods for model explainability, such as LIME, SHAP, and Integrated Gradients (Sundararajan et al., 2017), have provided insights into neural network predictions. At the same time, adversarial robustness (Goodfellow et al., 2014) has emerged as an important theme to safeguard classifiers against input attacks. Our work merges these trends to build a more interpretable and robust XML system.

3 Methodology

3.1 MACH Framework and Extensions

MACH compresses a high-dimensional label vector into a lower-dimensional space using a Count-Min Sketch, with multiple independent classifiers predicting each bucket. In our implementation, a two-layer feed-forward network processes a feature vector of dimension 50,000 (after feature hashing) and outputs predictions for a CMS with 2,000 buckets.

3.2 Model Explainability with Integrated Gradients

To improve interpretability, we adopt Integrated Gradients. By constructing a linear path from a baseline (e.g., an all-zero vector) to the input, Integrated Gradients assign attributions to each feature based on the interpolated gradients. These attributions provide insights into which features most influence the decision-making process and ensure that the model's focus aligns with domain-relevant aspects.

3.3 Adversarial Training

Adversarial examples are generated by perturbing the input in the direction of the gradient of the loss (FGSM). Our training procedure augments the clean loss with an adversarial loss term, weighted by a hyperparameter. This dual-loss approach ensures that the model

becomes robust to small input perturbations while maintaining its overall performance on clean data.

3.4 Data Preprocessing and Datasets

We evaluate our approach using publicly available datasets from the Extreme Classification Repository. The primary dataset is Amazon-670K, which includes millions of examples with sparse feature representations. The raw text data is preprocessed into TFRecord format, which enhances data streaming and reduces GPU idle time during training. Similar preprocessing steps are applied to additional datasets, such as Delicious-200K and Wiki10-31K, to verify scalability.

4 Experimental Setup and Results

4.1 Training Configuration

Our models are implemented in PyTorch and trained using the following key configuration parameters:

- **Input Dimension:** 50,000 (via feature hashing)
- **Label Dimension:** Approximately 670K (Amazon-670K)
- **Network Architecture:** Two-layer feed-forward network (hidden layer with 4096 units)
- **Output (Bucket) Size:** 2,000 buckets
- **Adversarial Parameters:** FGSM perturbation magnitude $\epsilon = 0.01$ and adversarial weight $\lambda = 1.0$
- **Optimization:** Adam optimizer with a learning rate of 0.001 over 5 epochs

Models are trained in parallel on multiple GPUs, and memory usage is carefully managed to accommodate large parameter counts.

4.2 Evaluation Metrics

We quantify model performance using standard XML metrics:

- **Precision@k:** The proportion of correct labels among the top k predictions.
- **NDCG@k:** Normalized Discounted Cumulative Gain, which emphasizes the ranking order of correct predictions.
- **Loss Metrics:** Binary cross-entropy loss is computed for both clean and adversarial inputs.

Our evaluation framework reconstructs the full prediction probabilities from the CMS output and aggregates results from multiple models.

4.3 Results and Discussion

Experiments on Amazon-670K indicate that our extended MACH model achieves competitive Precision@k scores relative to the baseline. Integrated Gradients provide meaningful attributions, confirming that critical input features drive the predictions. Moreover, adversarial training results in a modest increase in loss for adversarial examples but substantially improves the ranking metrics, demonstrating robustness against small input perturbations. Overall, these enhancements contribute to a more dependable extreme multi-label classification system.

5 Conclusion and Future Work

This work extends MACH by incorporating model explainability via Integrated Gradients and adversarial robustness through FGSM-based training. Our experimental results on the Amazon-670K dataset reveal that these modifications yield enhanced interpretability and resilience against adversarial manipulations without sacrificing overall performance. Future work will explore further improvements in adversarial defenses, alternative explainability methods, and scalability to even larger datasets.

Acknowledgments

We thank the authors of MACH [1] for releasing their innovative algorithm, as well as the developers of public XML repositories and associated libraries. Their contributions have been integral to this research.

References

References

[1] Medini, T. K. R., Huang, Q., Wang, Y., Mohan, V., & Shrivastava, A. (2019). Extreme Classification in Log Memory using Count-Min Sketch: A Case Study of Amazon Search with 50M Products. In *Advances in Neural Information Processing Systems*. https://proceedings.neurips.cc/paper_files/paper/2019/file/69cd21a0e0b7d5f05dc88a0be36950c7-Paper.pdf

[2] Bhatia, K., Kumar, D., Soni, P., & Kumar, A. (2015). Extreme Multi-label Learning. *Workshop on Machine Learning for Multimedia*, in conjunction with ACM Multimedia.

[3] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic Attribution for Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*.

[4] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations*.

Additional Resources

- XML Repository: http://manikvarma.org/downloads/XC/XMLRepository.html
- MACH Official Code (NeurIPS 2019): https://proceedings.neurips.cc/paper_files/paper/2019/file/69cd21a0e0b7d5f05dc88a0be36950c7-Paper.pdf
- MACH Implementation in Python: https://github.com/jamescporter/MACH-implementation
- MACH Official Repository: https://github.com/Tharun24/MACH/