đ

Abstract-As a safety critical task, autonomous driving for safe motion planning, particularly under challenging con-a degraded performance on the challenging scenarios, mainly challenging scenarios closer together in the feature space during training to trigger information sharing among them for more robust learning. These methods, however, primarily rely on ಫ the motion participation of the second participation of ോ), enderbourde ender of the generated trajectories. In this paper, we propose to incor-ط are split into clusters based on the model's output errors, using the training dynamics information, and a prototype is computed within each cluster. Second, we retrain the model using the prototypes in a contrastive learning framework. We conduct empirical evaluations of our approach using two large-scale naturalistic datasets and show that our method achieves state-of-the-art performance by improving accuracy and scene compliance on the long-tail samples. Furthermore, we perform experiments on a subset of the clusters to highlight the additional benefit of our approach in reducing training bias.

I. INTRODUCTION



In summary, our **main contributions** are as follows: 1) We propose TrACT, a Training dynamics Aware ContrasTive

¹University of Toronto. Work done during internship at Huawei Technologies Canada. arielle.zhang@mail.utoronto.ca

²Noah's Ark Laboratory, Huawei Technologies Canada.

 $[*]Corresponding \ author \ {\tt Mozhgan.Pourkeshavarz@huawei.com}$

II. RELATED WORKS

A. Trajectory Prediction

Trajectory prediction is a fundamental problem in autonomous driving. Accurate forecasting requires an understanding of the surrounding agents and the interactions among them as well as the scene configuration. To effectively process such multimodal information, existing methods rely on various architectures, including recurrent networks [8], [12], [13], CNNs [14], [15], GNNs [16]–[22], and more recently transformers [2], [23]–[25]. One of the key challenges in prediction is the uncertainty of future behaviors. To address this problem, models resort to generating a diverse set of trajectories using different approaches [26]–[29], one of which is CVAE [17], [30]–[32]. We use [7], which is a variant of a CAVE-based model [17] as the backbone.

B. Long-tail Learning

Trajectory prediction models are often evaluated on largescale datasets [3]–[5], [11] and the performance is averaged over all samples in the data. Despite achieving promising results on the benchmarks, models trained on these datasets underperform on challenging scenarios [7], [8], [22]. Such an issue is largely due to the long-tail nature of these datasets as they are more biased towards the common scenarios and contain much smaller number of challenging cases [6].

The long-tail phenomenon is incurred by the imbalanced number of more frequent samples and less frequent samples in the dataset. There are many studies on improving the long-tail learning on classification tasks offering techniques, such as data resampling [33]–[35], loss reweighting [36]–[38], boundary adjustment [39], [40], and more recently feature and label distribution smoothing [41].

Some recent works address the long-term problem in trajectory prediction [7], [8]. The authors of [7] learn long-tail samples by reshaping the feature space to better distinguish the head and tail samples through contrastive learning. The work in [8] includes an additional step of offline clustering to obtain the pseudo labels for the prototypical contrastive learning framework. However, in [7], [8], the authors argue that tail samples contain more evasive maneuvers, resulting in more complicated trajectory shapes. Therefore the learning is purely based on the motion patterns of individual



Fig. 2: Error evolution during training for one sample from each one of the *hard*, *confusing*, *easy*, and *trained* clusters on nuScenes.

III. METHOD

A. Problem Formulation

្ cont of the properties of the

B. Overview

ো 你ങ has defined of defi

C. Constructing Dataset Map

束 papp p



ే

D. Clustering

By analyzing the training behavior of each individual sample, we discover that in a dataset, samples that have low *errors* may have different *variances* which shows whether the model parameters are being updated correctly during training in order to learn these samples. This also holds for high *errors*, hence taking both dimensions into consideration, we propose a **4-Quarter** clustering method.

E. Prototype Computation

F. Single Level Prototypical Contrastive Learning

$$L_{ProtoNCE} = L_{ins} + L_{proto}, \tag{1}$$

where L_{ins} is the instance-wise term that penalizes large distances of the same cluster samples in the feature space and L_{proto} is the instance-prototype term that penalizes large distances between the sample and its cluster prototype in the feature space.

Instance-wise term. The instance-wise loss term L_{ins} in Equation 1 is designed to speed up convergence by attracting

instances belonging to the same cluster:

$$L_{ins} = -\sum_{i=1}^{r} \frac{1}{N_{po_i}} \sum_{i_{+}=1}^{N_{po_i}} log \frac{exp(v_i \cdot v_{i_{+}}/\tau)}{\sum_{j=1}^{r} exp(v_i \cdot v_j/\tau)},$$
 (2)

where *r* denotes the batch size, N_{po_i} is the number of same-cluster samples i_+s of an arbitrary sample *i*, and *v* is the feature embedding. $j \in [1,...,r]$ represents all available samples in the batch and τ is the contrastive temperature.

Instance-prototype term. In our method, we put more emphasis on the overall difficulty of each training sample instead of solely the motion patterns. Hence, we do not need a hierarchy of clusters to accommodate for granularity. Here, unlike the original PCL loss [43], our implementation of the L_{proto} term in Equation 1 only has one level of cluster:

$$L_{proto} = -\sum_{i=1}^{r} log \frac{exp(v_i \cdot c_i/\phi_i)}{\sum_{j=1}^{4} exp(v_i \cdot c_j/\phi_j)},$$
(3)

where for an arbitrary sample *i*, *c* is the cluster prototype, ϕ is the cluster density, and $j \in [1, ..., 4]$ represents all four clusters. The density ϕ of the cluster is given by,

$$\phi = \frac{\sum_{z=1}^{Z} ||v_z - c_z||_2}{Zlog(Z + \alpha)},$$
(4)

where Z is the number of samples belonging to the cluster, z is an arbitrary sample from the cluster, and α is a smoothing term, which is set to 10 following [43].

In the end, our final training objective is described as,

$$L = L_{reg} + \lambda L_{ProtoNCE}, \qquad (5)$$

where λ is the control weight of the prototypical loss.

IV. EXPERIMENTS

A. Setup

្<list-item>

ETH and UCY are pedestrian datasets with five subsets, including ETH, Hotel, Univ, Zara1, and Zara2, all containing different number of scenes for training and testing. Each scene contains a recording of pedestrians interacting scenario with varying time length. Following [7], [8], [24], we perform 5-fold cross validation on the five subsets.

Metrics. We use common evaluation metrics [8], [17], [24]: Average Displacement Error (ADE), which is the average L_2 distance between all predicted states and the ground truth and Final Displacement error (FDE), which is the L_2

ף والمعاصص والمعاصص والمعاصص والمعاصص والمعاص ومعاصص والمعاص والمعاص والمعاص والمعاص ومعاصص والمعاص والمعاص والمعاص والمعاص والمعاص والمعاص والمعاص والمعاص ومعاصص والمعاص ومعاصص والمعاص ومعاص والمعاص ومعاص والمعاص وولم والمعاص وول والمعاص والمعاص وولم والمعاص وول والمعاص والمعاص وا

ങ: bioot

т

ل

非

Dataset	Method	Top 1%		Top 2%		Top 3 %		Top 4%		Top 5 %			All						
nuScenes	Traj++ EWTA*	1.33	3.09	-	1.02	2.35	-	0.87	2.00	-	0.80	1.80	-	0.74	1.64	-	0.19	0.32	-
	+contrastive*	1.28	2.85	-	0.97	2.15	-	0.83	1.83	-	0.76	1.64	-	0.70	1.48	-	0.18	0.30	-
	+FEND*	1.21	2.50	-	0.92	1.88	-	0.79	1.61	-	0.72	1.43	-	0.66	1.31	-	0.17	0.26	-
	Traj++ EWTA	1.73	4.43	11.72	1.36	3.54	10.02	1.17	3.03	8.80	1.04	2.68	7.83	0.95	2.41	7.21	0.19	0.32	-0.14
	+contrastive	1.33	3.09	8.91	1.04	2.44	7.81	0.90	2.08	7.05	0.81	1.85	6.37	0.75	1.68	5.96	0.18	0.30	-0.11
	+TrACT (ours)	1.23	2.65	7.22	0.98	2.11	6.27	0.85	1.82	5.54	0.78	1.64	4.98	0.72	1.49	4.62	0.19	0.31	-0.21
	Traj++ EWTA	0.98	2.54	8.71	0.79	2.07	5.45	0.71	1.81	4.89	0.65	1.63	4.10	0.60	1.50	3.53	0.17	0.32	-0.42
ETH-UCY	+contrastive	0.92	2.33	7.85	0.74	1.91	5.02	0.67	1.71	4.54	0.60	1.48	3.71	0.55	1.32	3.13	0.17	0.32	-0.22
	+FEND	0.84	2.13	-	0.68	1.68	-	0.61	1.46	-	0.56	1.30	-	0.52	1.19	-	0.17	0.32	-
	+TrACT (ours)	0.80	2.00	3.39	0.65	1.63	2.52	0.61	1.46	2.34	0.56	1.31	2.11	0.52	1.18	1.93	0.17	0.32	-0.25

ъ

Method	Тор 1%	Тор 2%	Тор 3%	Тор 4 <i>%</i>	Тор 5 <i>%</i>	All		
Traj++ EWTA	6.52 1.22	5.75 1.05	4.24 0.76	3.80 0.61	3.17 0.49	0.22 0.03		
+contrastive	4.65 0.72	3.42 0.65	2.69 0.50	2.17 0.38	1.93 0.33	0.18 0.02		
+TrACT (ours)	4.04 0.56	3.26 0.56	2.59 0.42	2.41 0.36	1.99 0.29	0.23 0.03		

B. Comparison to SOTA

We compare TrACT against baseline Traj++ EWTA, Traj++ EWTA + contrastive, [7] and state-of-the-art FEND [8] methods on nuScenes and ETH-UCY and report the results in Table I. Note that, on nuScenes, two sets of baseline results are reported: the original ones from FEND [8] (*) and the ones we reproduced following instructions in ى 孩子的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们的问题,我们们能知道我们的问题,我们们能知道我们的问题,我们们能知道我们的问题,我们们能知道我们们能知道我们们就能知道我们们们就能知道我们们就能知道我们们就能知道我们们就能知道我们们就能知道我们们们就能知道我们们就能知道我们们就能知道我们们就能知道我们们就能知道我们们就能知道我们们们就能 ഡഡഡഡ a reference. FEND does not report results on NLL-KDE and evaluate FEND on this metric. As shown in Table I, TrACT achieves state-of-the-art performance on all metrics on all challenging subsets, significantly improving performance by up to 22.48% on KDE-NLL on the top 5% challenging subset compared to Traj++ EWTA + constrastive. On distance-based metrics, the largest improvement is achieved on the top 1% challenging subset with 7.52% on minADE and 14.24% on with a very small margin on distance-based metrics while maintaining the best performance on KDE-NLL.

C. Scene Compliance Predictions on Challenging Scenarios

ု)

D. Qualitative Results

ਣ

$\{\theta_e, \theta_{var}\}$	Easy (%)	Confusing (%)	Hard (%)	Trained (%)	Top 1%	Top 5%	All
$\{0.50, 0.10\}$	55.72	20.54	4.43	19.30	1.30 2.50 7.35	0.84 1.51 4.56	0.26 0.39 0.82
$\{0.70, 0.20\}$	65.53	11.06	3.68	19.71	1.23 2.65 7.22	0.72 1.49 4.62	0.19 0.31 -0.21
$\{0.90, 0.10\}$	59.16	8.01	1.00	31.83	1.25 2.54 7.16	0.76 1.46 4.21	0.22 0.35 0.27

Training Dataset (size%)	Top 1%		Top 2%			Top 3%			Top 4%			Top 5%			All			
Full Dataset (100%)	1.73	4.43	11.72	1.36	3.54	10.02	1.17	3.03	8.80	1.04	2.68	7.83	0.95	2.41	7.21	0.19	0.32	-0.14
Removed easy ($\approx 80\%$)	1.37	3.10	9.11	1.06	2.42	8.05	0.92	2.08	7.27	0.83	1.83	6.63	0.76	1.65	6.26	0.19	0.32	0.18



simple but the contextual information is more complicated, requiring the model to rely more on the map information to make safe and scene compliant predictions.

E. Ablation Studies

ห



Fig. 5: Parameter sensitivity of λ on minFDE, plotted for the top 1-5% challenging subsets and the rest 95% of the data.

made the choice of $\{\theta_e = 0.70, \theta_{var} = 0.15\}$ for ETH-UCY.

F. Dataset Map for Reducing Training Bias

In this study, we aim to show the benefit of the proposed dataset map for achieving better performance without contrastive learning. For this purpose, we conducted experiments on the full 100% dataset and also on 80% of the data by cluster. Our intuition is that by removing a portion of the *easy* samples, we reduce the overall data bias, as the easy samples are more frequent in the dataset. Hence, the model would focus more on challenging scenarios, and as a result, achieves a more balanced performance without the use of an explicit contrastive objective. As our findings in Table IV suggest, improvements of up to 31.72% on distanced-based metrics and 22.27% on KDE-NLL are achieved across all challenging subsets while the overall performance is unchanged on distance-based metrics. The decline in KDE-NLL metric on all samples can be primarily due to the reduced size of the dataset making the model unsure about the true distributions of the samples.

V. CONCLUSIONS

In this work, we proposed a novel framework for learning long-tail scenarios in the context of trajectory prediction for autonomous driving. Our approach, TrACT, exploits the training dynamics information of the model to cluster samples into groups of different levels of difficulty. The clusters, combined with the model feature embeddings, form prototypes to be used in a prototypical contrastive learning framework.

We conducted empirical studies on two trajectory prediction benchmark datasets and showed that TrACT achieved state-of-the-art performance by significantly improving over past arts across the challenging subsets. Besides achieving improved performance on common metrics, TrACT generates significantly more map compliant trajectories, making it more suitable for practical applications. At the end, we illustrated the benefit of the proposed dataset map construction technique for improving performance on challenging scenarios without explicit use of a contrastive learning objective.

REFERENCES

- D. Zhu, G. Zhai, Y. Di, F. Manhardt, H. Berkemeyer, T. Tran, N. Navab, F. Tombari, and B. Busam, "Ipcc-tp: Utilizing incremental pearson correlation coefficient for joint multi-agent trajectory prediction," in *CVPR*, 2023.
- [2] R. Karim, S. M. A. Shabestary, and A. Rasouli, "Destine: Dynamic goal queries with temporal transductive alignment for trajectory prediction," *arXiv preprint arXiv:2310.07438*, 2023.
- [3] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *CVPR*, 2020.
- [4] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3D tracking and forecasting with rich maps," in *CVPR*, 2019.
- [5] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020.
- [6] C. Chen, M. Pourkeshavarz, and A. Rasouli, "Criteria: a new benchmarking paradigm for evaluating trajectory prediction models for autonomous driving," arXiv preprint arXiv:2310.07794, 2023.
- [7] O. Makansi, Ö. Cicek, Y. Marrakchaibo, and T. Brox, "On exposing the challenging long tail in future prediction of traffic actors," in *ICCV*, 2021.
- [8] Y. Wang, P. Zhang, L. Bai, and J. Xue, "Fend: A future enhanced distribution-aware contrastive learning framework for long-tail trajectory prediction," in *CVPR*, 2023.
- [9] F. Bartoli, G. Lisanti, L. Ballan, and A. Del Bimbo, "Context-aware trajectory prediction," in *ICPR*, 2018.
- [10] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *ICCV*, 2009.
- [11] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," in *CVPR*, 2014.
- [12] A. Rasouli, M. Rohani, and J. Luo, "Bifold and semantic reasoning for pedestrian behavior prediction," in *ICCV*, 2021.
- [13] Y. Xu, Z. Piao, and S. Gao, "Encoding crowd interaction with deep neural network for pedestrian trajectory prediction," in CVPR, 2018.
- [14] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "GOHOME: Graph-oriented heatmap output for future motion estimation," in *ICRA*, 2022.
- [15] M. Ye, T. Cao, and Q. Chen, "TPCN: Temporal point cloud networks for motion forecasting," in CVPR, 2021.
- [16] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Socialstgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *CVPR*, 2020.
- [17] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *ECCV*, 2020.
- [18] J. Li, F. Yang, M. Tomizuka, and C. Choi, "Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning," in *NeurIPS*, 2020.
- [19] S. Casas, C. Gulino, R. Liao, and R. Urtasun, "Spatially-aware graph neural networks for relational behavior forecasting from sensor data," in *ICRA*, 2020.

- [20] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *ECCV*, 2020.
- [21] X. Jia, P. Wu, L. Chen, Y. Liu, H. Li, and J. Yan, "Hdgt: Heterogeneous driving graph transformer for multi-agent trajectory prediction via scene encoding," *PAMI*, 2023.
- [22] M. Pourkeshavarz, C. Chen, and A. Rasouli, "Learn tarot with mentor: A meta-learned self-supervised approach for trajectory prediction," in *ICCV*, 2023.
- [23] A. Rasouli and I. Kotseruba, "Pedformer: Pedestrian behavior prediction via cross-modal attention modulation and gated multitask learning," in *ICRA*, 2023.
- [24] L. Shi, L. Wang, S. Zhou, and G. Hua, "Trajectory unified transformer for pedestrian trajectory prediction," in *ICCV*, 2023.
- [25] E. Amirloo, A. Rasouli, P. Lakner, M. Rohani, and J. Luo, "Latentformer: Multi-agent transformer-based interaction modeling and trajectory prediction," arXiv preprint arXiv:2203.01880, 2022.
- [26] W. Mao, C. Xu, Q. Zhu, S. Chen, and Y. Wang, "Leapfrog diffusion model for stochastic trajectory prediction," in CVPR, 2023.
- [27] C. ". Jiang, A. Cornman, C. Park, B. Sapp, Y. Zhou, and D. Anguelov, "Motiondiffuser: Controllable multi-agent motion prediction using diffusion," in *CVPR*, 2023.
- [28] P. Nikdel, M. Mahdavian, and M. Chen, "Dmmgan: Diverse multi motion prediction of 3d human joints using attention-based generative adversarial network," in *ICRA*, 2023.
- [29] M. Wang, X. Zhu, C. Yu, W. Li, Y. Ma, R. Jin, X. Ren, D. Ren, M. Wang, and W. Yang, "Ganet: Goal area network for motion forecasting," in *ICRA*, 2023.
 [30] A. Rasouli, "A novel benchmarking paradigm and a scale-and motion-
- [30] A. Rasouli, "A novel benchmarking paradigm and a scale-and motionaware model for egocentric pedestrian trajectory prediction," *arXiv* preprint arXiv:2310.10424, 2023.
- [31] M. Lee, S. S. Sohn, S. Moon, S. Yoon, M. Kapadia, and V. Pavlovic, "MUSE-VAE: Multi-scale VAE for environment-aware long term trajectory prediction," in CVPR, 2022.
- [32] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "Agentformer: Agentaware transformers for socio-temporal multi-agent forecasting," in *ICCV*, 2021.
- [33] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [34] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: A new oversampling method in imbalanced data sets learning," in Advances in Intelligent Computing, 2005.
- [35] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in ECCV, 2016.
- [36] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *CVPR*, 2019.
- [37] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1263–1284, 2009.
- [38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017.
- [39] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *NeurIPS*, 2019.
- [40] A. K. Menon, S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar, "Long-tail learning via logit adjustment," arXiv preprint arXiv:2007.07314, 2020.
- [41] Y. Yang, K. Zha, Y. Chen, H. Wang, and D. Katabi, "Delving into deep imbalanced regression," in *ICML*, 2021.
- [42] S. Swayamdipta, R. Schwartz, N. Lourie, Y. Wang, H. Hajishirzi, N. A. Smith, and Y. Choi, "Dataset cartography: Mapping and diagnosing datasets with training dynamics," *arXiv preprint arXiv:2009.10795*, 2020.
- [43] J. Li, P. Zhou, C. Xiong, and S. C. Hoi, "Prototypical contrastive learning of unsupervised representations," arXiv preprint arXiv:2005.04966, 2020.
- [44] M. Bahari, S. Saadatnejad, A. Rahimi, M. Shaverdikondori, A. H. Shahidzadeh, S.-M. Moosavi-Dezfooli, and A. Alahi, "Vehicle trajectory prediction works, but not everywhere," in *CVPR*, 2022.