Hierarchical Light Transformer Ensembles for Multimodal Trajectory Forecasting (Supplementary material)

Notations	Meaning
$\mathcal{D} = \{\mathbf{X}^i_{1:T}\}$	The set of $\lvert\mathcal{D}\rvert$ agents' trajectories lasting T time steps
A	The number of agents
T	The number of time steps
K	The number of modes, *i.e.*, components in the mixture distribution
$K^\star$	The number of meta-modes, *i.e.*, components in the meta-mixture distribution
K'	The number of modes within a meta-mixture component
a, t, k	The indexes of the current agent, the current time step, the current mode
$X_i = \mathbf{X}^i_{1:t}$	The observed trajectories assuming t steps of context
$Y_i = \mathbf{X}^i_{t+1:T}$	The target trajectories (*i.e.*, ground-truth of the forecasts) assuming t steps of context
H	The number of attention heads in multi-head attention layers
M	The number of estimators in an ensemble, *i.e.*, ensemble size
θ_k	The set of weights of the kth-component of a parametric probabilistic mixture model
α	The width-augmentation factor of HLT-Ens.
P_θ	The probability density function of a parametric model where θ are the parameters
$\boldsymbol{\mu}_{\theta_k}$	The mean of the kth-component of a Laplace mixture distribution parametrized by θ
$\mathbf{b}_{\theta_k}$	The scale vector of the kth-component of a Laplace mixture distribution parametrized by θ
$\boldsymbol{\pi}_\theta$	The probability vector corresponding to the a mixture weights
$\bar{\boldsymbol{\mu}}_{\theta_k}$	The mean of the kth *meta-mode* of a Laplace mixture distribution parametrized by θ
$\bar{\mathbf{b}}_{\theta_k}$	The scale vector of the kth *meta-mode* of a Laplace mixture distribution parametrized by θ
$\boldsymbol{\Delta}^C$	The probability simplex in the $\mathbb{R}^C$ space

Table 1. **Summary of the main notations of the paper.**

Contents

A. Notations

Tab. 1 summarizes the main notations used throughout this paper.

B. Implementation and Training Details

This section details our models' implementation and training procedure for the experiments. We expect to release the corresponding code upon acceptance and validation by our industrial sponsor. We implement all networks using the PyTorch framework and train them using an Nvidia RTX A6000, except for experiments on Wayformer where a Nvidia Tesla V100 has been used. For both datasets, we used the same simple preprocessing:

1. We transform all agents and lanes coordinates into a scene-centric view, which is centered and oriented based on the last observed state of a focal agent, *i.e.*, an agent having a complete track over the scene duration.
2. We keep only the 6 closest agents to the origin.
3. We keep only the 100 closest lanes to the origin.

Tab. 2 summarizes the hyperparameters of all our experiments. Concerning the $\varepsilon - WTA$ loss, we used $\varepsilon = 0.05$ as specified in [3]. For the EWTA loss, we set up the decay over the top-n modes as described in Appendix B.

C. HWTA Loss Details

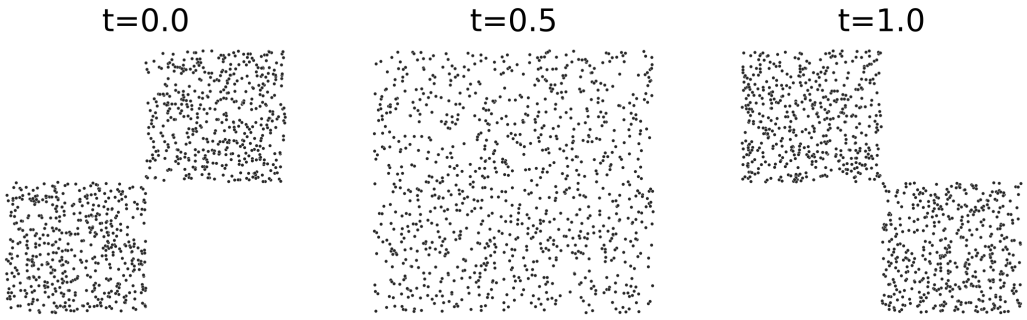
This section provides more explanations and insights on the HWTA loss. First, we present how classification loss terms are encompassed within the *meta-mode* and MWTA losses. Then, we illustrate our loss behavior on a 2D toy dataset (Fig. 1).

Sec. 4 defines both terms in HWTA loss as the NLL loss of the meta-mixture and the best meta-mode mixture, respectively. Yet, in practice, these terms are combined with a classification loss. Indeed, we leverage the loss formulation in [1] to define $\mathcal{L}^{\mathcal{C}}_{meta}$ and $\mathcal{L}^{\mathcal{C}}_{\text{MWTA}}$:

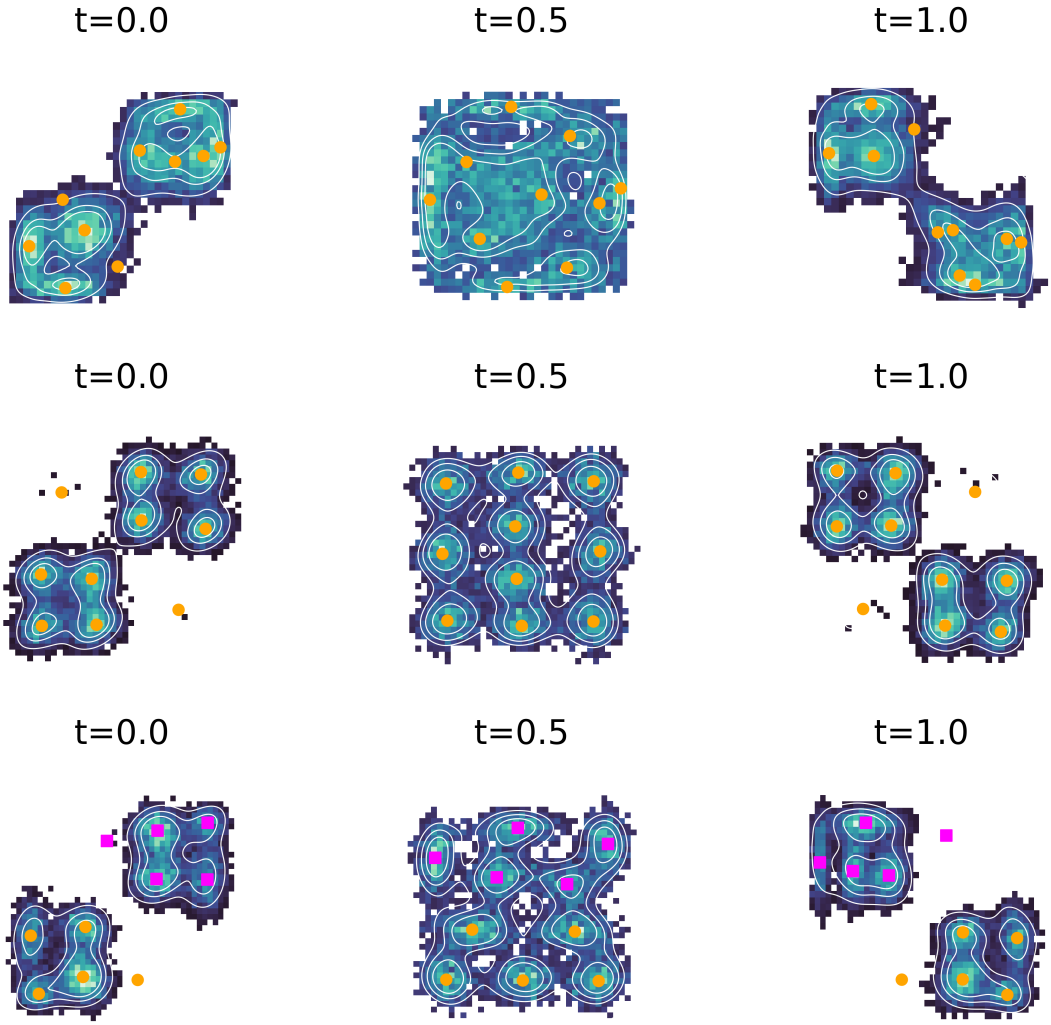
$$\mathcal{L}^{\mathcal{C}}_{meta} = \frac{1}{K} \sum_{Z} D_{\text{KL}}(Q(Z)||P_\theta(Z|Y,X)) \tag{1}$$

$$\mathcal{L}^{\mathcal{C}}_{\text{MWTA}} = \frac{1}{K'} \sum_{Z'} D_{\text{KL}}(Q(Z')||P_\theta(Z'|Y,X)), \tag{2}$$

where Z and Z' are discrete latent variables corresponding to the meta-modes and their respective modes. $D_{\text{KL}}(.||.)$



(a) Ground truth distribution



(b) Predicted distributions of small MLP models using the mixture NLL loss (*top*), the WTA loss with the NLL (*center*), and the HWTA loss with two meta-modes (in magenta and orange) (*bottom*)

Figure 1. **Temporal 2D distributions**

Parameter	Description	Argoverse 1		Interaction	
		AutoBots	*ADAPT*	*AutoBots*	*ADAPT*
d	Hidden dimension used in all model layers	128	128	128	128
Batch size	Batch size during training	128	128	128	128
Epochs	Number of epochs during training	30	36	60	36
Learning rate	Adam Optimizer initial learning rate	7.5e-4	7.5e-4	7.5e-4	7.5e-4
Decay	Multiplicative factor of learning rate decay	0.5	0.15	0.5	0.15
Milestones	Epoch indices for learning rate decay	5,10,15,20	25,32	10,20,30,40,50	25,32
Dropout	Dropout rate in multi-head attention layers	0.1	0.1	0.1	0.1
K	Number of modes of the baselines	6	6	6	6
H	Number of attention heads	16	8	16	8
$K^\star$	Number of meta-modes for our approaches	3	2	2	2
K'	Number of modes within a meta-mode for our approaches	2	3	3	3
α	Width factor of HLT-Ens	1.5	2.0	1.5	2.0
γ	Tradeoff of the HWTA loss	0.6	0.6	0.6	0.6

Table 2. **Hyperparameters summary** for both AutoBots and ADAPT backbones across all two datasets

Backbones	**Argoverse 1**	**Interaction**
AutoBots	5,10,15,20,25	10,20,30,40,50
ADAPT	5,10,15,20,25	5,10,15,20,25

Table 3. **Hyperparameter for the EWTA loss**. This table reports the epoch indices where the number of modes (n) to update is decremented by 1. At initialization $n = 6$.

is the Kullback-Leibler divergence between the approximated posterior and the actual posterior. As in [1], we set $Q(Z) = P_{\theta_{\text{old}}}(Z|Y,X)$ and $Q(Z') = P_{\theta_{\text{old}}}(Z'|Y,X)$, where θ_{old} are the parameters before the optimization step. Eq. (7) becomes:

$$\mathcal{L} = \gamma\times(\mathcal{L}_{\text{meta}}+\mathcal{L}^{\mathcal{C}}_{meta})+(1-\gamma)\times(\mathcal{L}_{\text{MWTA}}+\mathcal{L}^{\mathcal{C}}_{\text{MWTA}}) \quad (3)$$

Inspired by [3], we utilize their custom toy dataset, which comprises a two-dimensional distribution evolving over time $t \in [0,1]$. They achieve this by dividing a zero-centered square into 4 equal regions and transitioning from having high probability mass in the lower-left and top-right quadrants to having high probability mass in the upper-left and lower-right ones. Following their notation, the sections are defined as:

$$S_1 = [-1,0[\times[-1,0[\subset\mathbb{R}^2 \quad (4)$$
$$S_2 = [-1,0[\times[0,1]\subset\mathbb{R}^2 \quad (5)$$
$$S_3 = [0,1]\times[-1,0[\subset\mathbb{R}^2 \quad (6)$$
$$S_4 = [0,1]\times[0,1]\subset\mathbb{R}^2 \quad (7)$$
$$S_5 = \mathbb{R}^2\backslash\{S_1\cup S_2\cup S_3\cup S_4\} \quad (8)$$

and their respective probabilities being $P(S_1) = P(S_4) = \frac{1-t}{2}$, $P(S_2) = P(S_3) = \frac{t}{2}$, and $P(S_5) = 0$. Whenever a region is selected, a point is sampled from it uniformly. Fig. 1a illustrates such distribution for $t \in \{0, 0.5, 1\}$. To better illustrate the loss dynamics, we train a basic three-layer fully connected network with 50 neurons in each hidden layer and ReLU activation function, similar to [3]. Given the time t, we are interested in modeling a two-dimensional distribution using $K = 10$ modes. We visually compare the effect of the mixture NLL loss, the WTA loss combined with the NLL loss, and our HWTA loss (assuming 2 meta-modes here) in Fig. 1b. Although the NLL loss seems to capture the underlying distribution better, the mode diversity is compromised by some redundant modes, diminishing their coverage. The WTA loss on the mixture NLL creates a Voronoï tesselation of the space, *i.e.*, the modes are efficiently placed for coverage. Yet, it provides no information on what mode to keep if we subsequently reduce the number of modes. On the contrary, the hierarchy in our loss enables us to give 2 levels of modeling. One can directly take the meta-modes to maximize coverage and diminish the overall number of modes. We illustrate the benefit of this strategy in Appendix F.

D. Loss Parameter Sensitivity Study

This section showcases the effect of γ on our HWTA loss $\mathcal{L}$. In particular, Fig. 2 illustrates the performance variation for different γ values. Note that for $\gamma = 0.0$ the *meta-mode* loss $\mathcal{L}_{\text{meta}}$ is not used and for $\gamma = 1.0$ the loss $\mathcal{L}_{\text{MWTA}}$ is annealed. From our experiments, $\mathcal{L}_{\text{meta}}$ is necessary for better accuracy on most confident predictions as we observe a dramatic decrease in mADE_1 and mFDE_1

E. Importance of the Width Factor

Our efficient ensembling architecture depends on two hyperparameters. M corresponds to the ensemble size, and

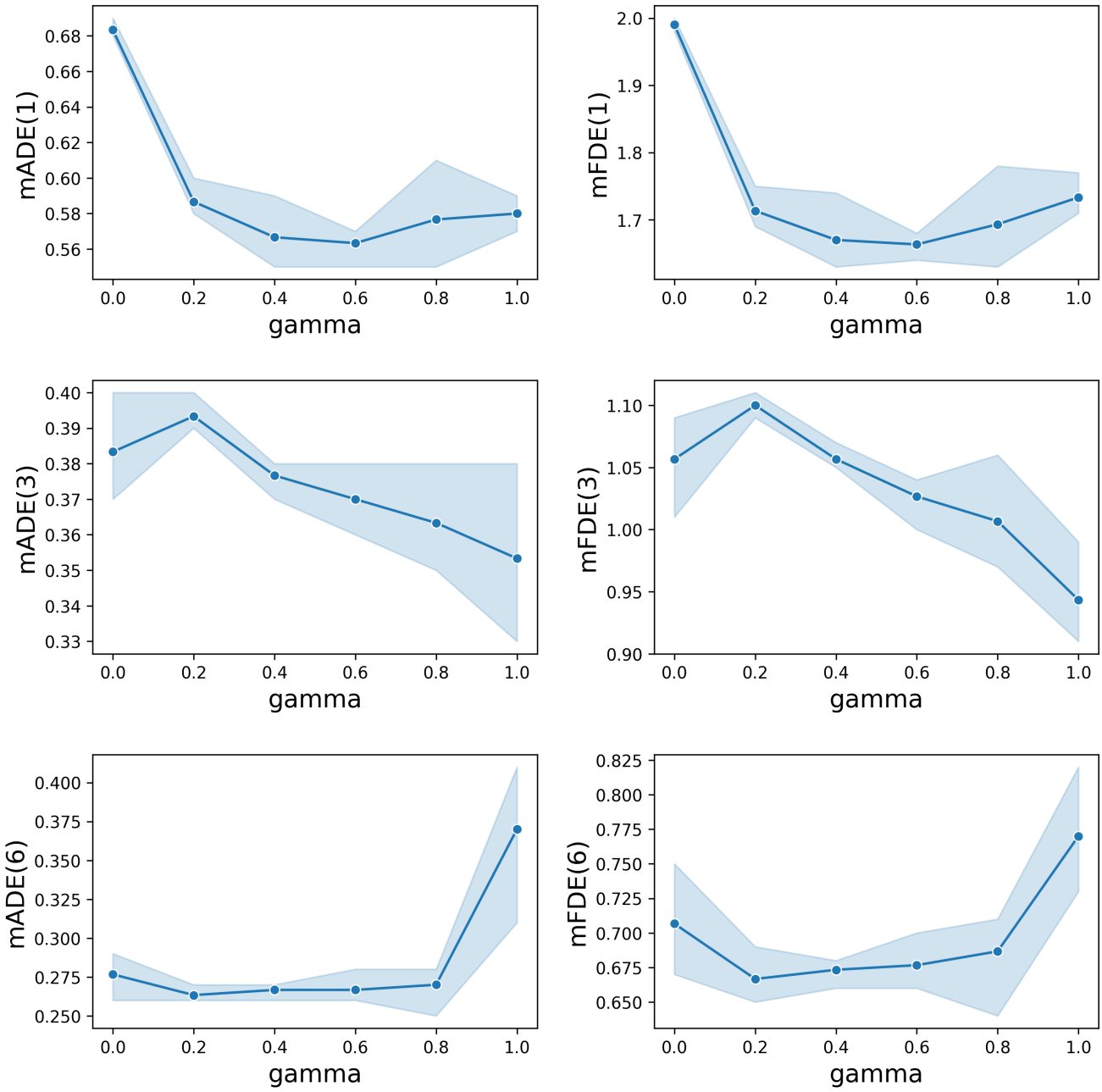


Figure 2. **Sensitivity analysis on γ (gamma)** using AutoBots backbone on the Interaction dataset.

α, as a factor on the embedding size, controls the width of the DNN. We evaluate the sensitivity of HLT-Ens to these parameters by training models using the ADAPT backbone on the Interaction dataset with various settings. Tab. 4 and Tab. 5 showcase the effect of α for 4 and 8 subnetworks respectively. Increasing α enables better results until it reaches a plateau at the cost of more parameters.

F. Robustness Analysis over Mode Number

The number of modes is a critical hyperparameter significantly impacting the model's performance. Predicting more modes and choosing the most confident ones is often preferred to cover the true multimodal distribution. Yet, doing so might decrease the diversity in the predicted modes as the most confident mode is likely to have duplicates. Using hierarchy in the mixture distribution, we showcase

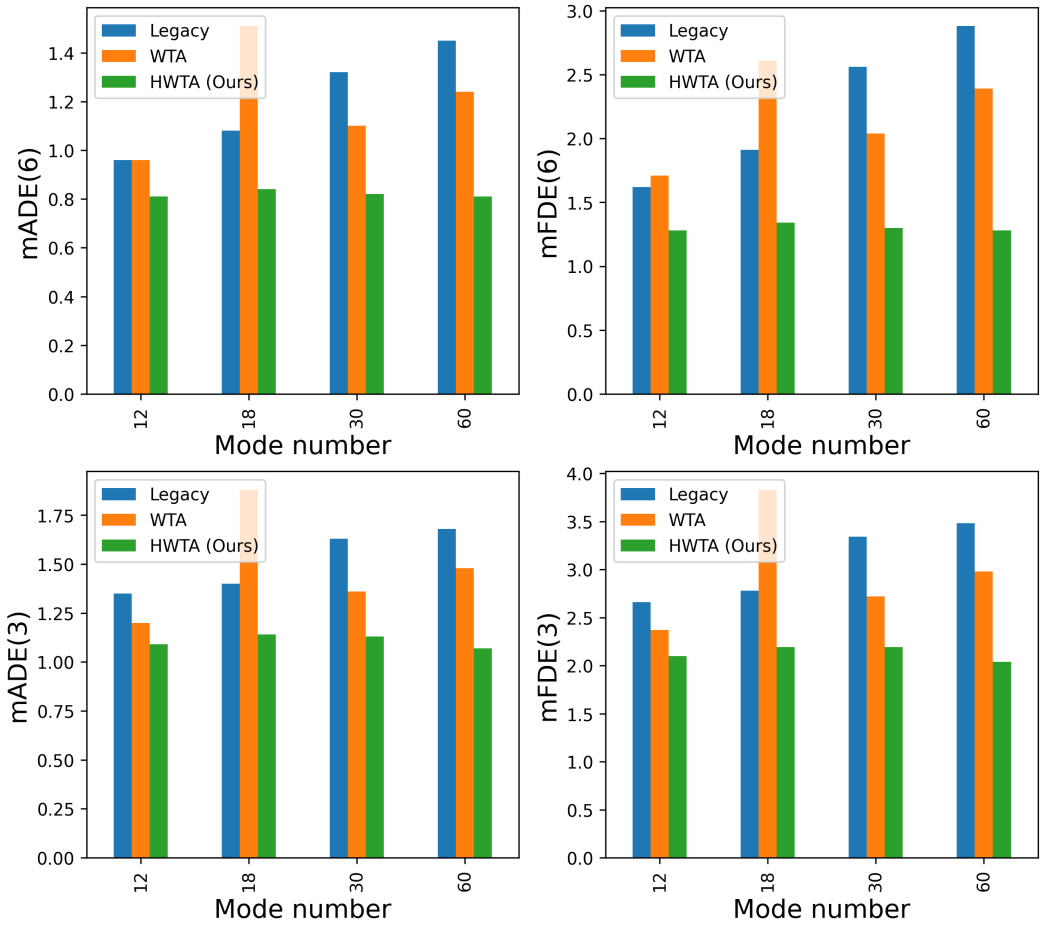


Figure 3. **Performance comparison under variations of the number of modes.** We train three AutoBots using the *Legacy* loss, the WTA loss, and the HWTA loss on Argoverse 1. We report their performance in terms of mADE_3, mFDE_3, mADE_6 and mFDE_6. For our loss, we considered 6 *meta-modes*.

α value	$\text{mADE}_1 \downarrow$	$\text{mFDE}_1 \downarrow$	$\text{mADE}_6 \downarrow$	$\text{mFDE}_6 \downarrow$	#Prm (M)
1	0.78	2.24	0.45	1.18	0.4
2	0.61	1.79	0.34	0.87	1.5
3	0.55	1.58	0.28	0.69	3.1
4	0.51	1.48	0.25	0.62	5.5

Table 4. **Performance of HLT-Ens - ADAPT (averaged over three runs) on Interaction wrt. α.** Our ensemble has $M = 4$ subnetworks, with $K = 2$ meta-modes containing $K' = 3$ modes each.

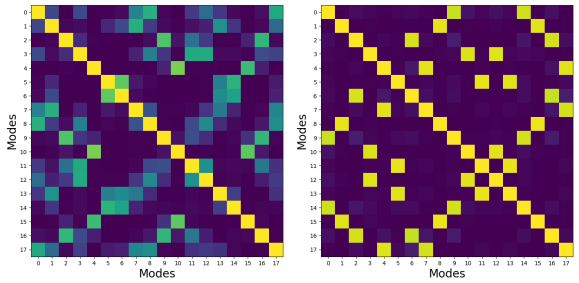
α value	$\text{mADE}_1 \downarrow$	$\text{mFDE}_1 \downarrow$	$\text{mADE}_6 \downarrow$	$\text{mFDE}_6 \downarrow$	#Prm (M)
3	0.62	1.82	0.36	0.93	1.7
4	0.59	1.71	0.32	0.82	2.9
6	0.53	1.53	0.27	0.69	6.3
8	0.51	1.47	0.26	0.65	10.9

Table 5. **Performance of HLT-Ens - ADAPT (averaged over three runs) on Interaction wrt. α.** Our ensemble has $M = 8$ subnetworks, with $K = 2$ meta-modes containing $K' = 3$ modes each.

that we can reduce our mixture complexity while retaining most of its diversity. Fig. 3 illustrates the impact of the number of modes over the performance on diversity metrics (*i.e.*, mADE_6 and mFDE_6). The comparison is done with three different settings (*Legacy* loss, WTA loss, and HWTA loss), all using AutoBots backbone and trained on Argoverse 1. Concerning our loss, instead of taking the 6 most confident trajectories, we use the 6 *meta-modes* as a set of forecasts. Doing so seems to stabilize the performance, suggesting our loss enables more robustness to variations in the number of modes with a fixed number of *meta-modes*.

G. Diversity in Ensembles of Mixtures

Prediction diversity is essential for the performance of ensembles. In [2], the authors present two sources of stochasticity in the training process producing diversity



(a) DE similarity matrix (b) HLT-Ens similarity matrix

Figure 4. **Similarity matrices for DE and HLT-Ens on Argoverse 1.** We use the AutoBots backbone to construct ensembles of size $M = 3$. For HLT-Ens, we set $\alpha = 1.5$.

	Method	$\text{mADE}_1 \downarrow$	$\text{mADE}_6 \downarrow$	$\text{mFDE}_1 \downarrow$	$\text{mFDE}_6 \downarrow$	$\text{NLL}_3 \downarrow$	$\text{NLL}_6 \downarrow$	#Prm ↓
SceneTransformer	**Single model**							
	Legacy	0.66	**0.21**	1.50	**0.39**	18.57	17.23	11.8
	HWTA (Ours)	**0.48**	0.26	**1.17**	0.52	**-4.74**	**-11.15**	
	Ensemble							
	DE	0.57	**0.21**	1.30	**0.60**	23.09	22.59	35.4
	HT-Ens (Ours)	0.50	0.23	1.19	0.47	**-7.34**	**-9.94**	35.4
	HLT-Ens (Ours)	**0.49**	0.25	**1.17**	0.53	-4.24	-9.59	**8.9**
Wayformer	**Single model**							
	WTA	1.75	0.51	4.13	1.20	49.92	-11.58	**1.1**
	HWTA (Ours)	**0.81**	**0.40**	**2.24**	**0.95**	**-15.54**	**-49.71**	
	Ensemble							
	DE	1.01	0.48	2.74	1.19	-12.81	-21.23	3.3
	HT-Ens (Ours)	0.74	0.35	2.06	0.80	-25.80	-47.36	3.3
	HLT-Ens (Ours)	**0.68**	**0.33**	**1.87**	**0.71**	**-27.64**	**-49.78**	1.5

Table 6. **Performance comparison on Interaction for the SceneTransformer and Wayformer backbones.** All ensembles have $M = 3$ subnetworks and are followed by a KMeans algorithm to form 6 trajectory clusters from which we take the centroids; we highlight the best performances in bold. For our method, we consider $\alpha = 1.5$ for SceneTransformer and $\alpha = 2.0$ for Wayformer. The number of parameters is expressed in millions.

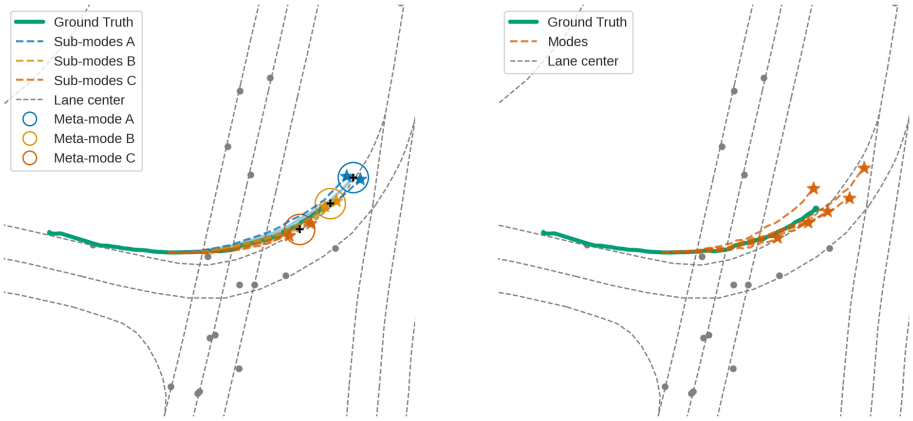
among ensemble members: the random initialization of the model's parameters and the shuffling of the batches. HLT-Ens does not benefit from this last source of stochasticity, yet it has comparable performance. To provide more insight into the effect on the diversity of this source of stochasticity in the training of trajectory forecasting models, we conducted a small experiment on the consistency of the cluster found by the *post-hoc* KMeans algorithm on our ensembles. Fig. 4 presents the similarity matrices for both DE and HLT-Ens. Each cell represents the rate of two modes being clusterized in the same cluster by the KMeans algorithm on the validation set of Argoverse 1. The clustering appears more consistent for HLT-Ens (i.e., the similarity matrix is sparser), highlighting the possibility of clustering only once the modes and applying it without too much performance loss compared to executing a KMeans algorithm for each sample.

H. Wayformer and SceneTransformer Experiments

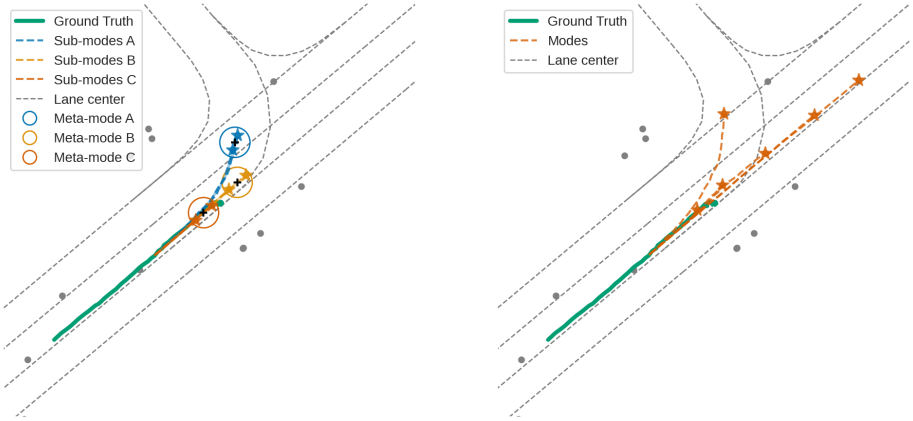
Appendix H presents the performance of our method on the SceneTransformer and Wayformer backbones trained on the Interaction dataset. These results are based on custom re-implementations we developed for both methods. These preliminary results showcase the ability of our approach to improve the most confident forecast accuracy and the quality of the predicted multimodal distribution compared to the original loss presented in both papers. Our method applied to Wayformer outperforms its counterpart on all metrics. Interestingly, **HLT-Ens** even outperforms **HT-Ens**, making it an attractive choice. It only has slightly more parameters than the classic Wayformer. Concerning SceneTransformer, we observe a lack of diversity in our predicted modes. We argue it might be necessary to try out other values of γ (we used $\gamma = 0.6$ here).

I. Additional Qualitative Results

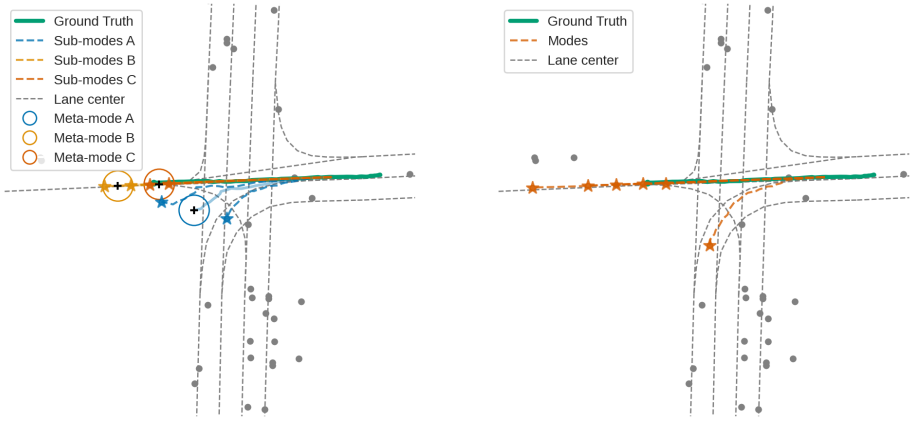
Fig. 5 provides additional trajectory forecasting examples on the Argoverse 1 dataset. We display predictions from an AutoBots trained using our new loss HWTA and a classical AutoBots model. The first thing we observe is that the predictions from our model are less scattered than the other. Indeed, they are closer to each other and the ground truth, which explains why we reached higher performance on mADE_1, mFDE_1 and the NLL_k metrics. We also note that the *sub-modes* belonging to the same *meta-modes* are near each other, as announced in the paper. With this knowledge, we can expect the average of several *sub-modes* (i.e., a *meta-mode*) to be a more robust prediction, as it might be in higher-density areas. Finally, we show a failure example, where due to the bad position of one *sub-mode*, the *meta-mode* A is misplaced. We argue that this issue could occur for likely trajectory candidates and that one could easily compute the intra-mode distances to see whether the cluster is coherent. Moreover, adding more *sub-modes* per *meta-mode* should alleviate this issue as one bad *sub-mode* will have less effect.



(a) Left turn scenario



(b) Intersection scenario



(c) Another intersection scenario

Figure 5. **Qualitative results with AutoBots backbone on Argoverse 1.** We compare an AutoBots model trained with its original loss ($K = 6$) compared with our HWTA loss ($\gamma = 0.8$) with 3 *meta-modes* (i.e., $K^\star = 3$ and $K' = 2$).

References

[1] Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira Ebrahimi Kahou, Felix Heide, and Christopher Pal. Latent variable sequential set transformers for joint multi-agent motion prediction. *ICLR*, 2022. 1, 3

[2] Olivier Laurent, Adrien Lafage, Enzo Tartaglione, Geoffrey Daniel, Jean-Marc Martinez, Andrei Bursuc, and Gianni Franchi. Packed-ensembles for efficient uncertainty estimation. *ICLR*, 2023. 5

[3] Christian Rupprecht, Iro Laina, Robert DiPietro, Maximilian Baust, Federico Tombari, Nassir Navab, and Gregory D Hager. Learning in an uncertain world: Representing ambiguity through multiple hypotheses. In *ICCV*, 2017. 1, 3