Annealed Winner-Takes-All for Motion Forecasting

Yihong Xu, Victor Letzelter, Mickaël Chen, Éloi Zablocki, Matthieu Cord

Annealed Winner-Takes-All for Motion Forecasting

Yihong Xu[1*] Victor Letzelter[1,2*] Mickaël Chen[1] Éloi Zablocki[1] Matthieu Cord[1,3]

Abstract— In autonomous driving, motion prediction aims at forecasting the future trajectories of nearby agents, helping the ego vehicle to anticipate behaviors and drive safely. A key challenge is generating a *diverse* set of future predictions, commonly addressed using data-driven models with Multiple Choice Learning (MCL) architectures and Winner-Takes-All (WTA) training objectives. However, these methods face initialization sensitivity and training instabilities. Additionally, to compensate for limited performance, some approaches rely on training with a large set of hypotheses, requiring a post-selection step during inference to significantly reduce the number of predictions. To tackle these issues, we take inspiration from annealed MCL, a recently introduced technique that improves the convergence properties of MCL methods through an annealed Winner-Takes-All loss (aWTA). In this paper, we demonstrate how the aWTA loss can be integrated with state-of-the-art motion forecasting models to enhance their performance using only a minimal set of hypotheses, eliminating the need for the cumbersome post-selection step. Our approach can be easily incorporated into any trajectory prediction model normally trained using WTA and yields significant improvements. To facilitate the application of our approach to future motion forecasting models, the code is made publicly available: `https://github.com/valeoai/MF_aWTA`.

I. INTRODUCTION

Motion forecasting is essential for the safe deployment of autonomous driving systems as it allows vehicles to anticipate the possible behaviors of nearby agents and plan their actions accordingly. A key requirement for motion forecasting is that models should propose multiple plausible futures for each agent. This diversity is essential for downstream tasks since anticipating even rare cases is crucial for the safety of autonomous agents [1], [2], [3], [4]. However, for any given situation, only one realization of a future trajectory is observed in the data. This poses a challenge because the vast majority of current methods are data-driven [5], [6], [7], [8], [9], [10], [11], [12].

The challenge of training a multi-hypothesis model with only single observations is commonly addressed using Multiple Choice Learning (MCL) with a Winner-Takes-All (WTA) loss [13], [14]. In this framework, multiple hypotheses — corresponding to possible futures (modes) — are predicted and scored by the model. With WTA training, only the best-performing prediction head (or *hypothesis*) is selected and updated during training. However, as an example shown in Figure 1, the naive use of WTA can lead to instabilities, such as mode collapse during training [15]. To mitigate this,

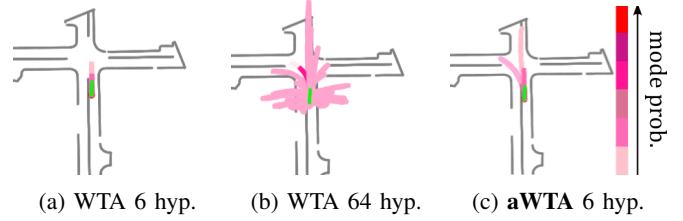(a) WTA 6 hyp.   (b) WTA 64 hyp.   (c) **aWTA** 6 hyp.

Fig. 1: **Issues of WTA, and our proposed aWTA.** Motion forecasting models trained with WTA and a small number of hypotheses (e.g., 6 hyp.) suffer from mode collapse (a), causing a performance drop. A naive solution to improve performance is to increase the training hypotheses (b), but a cumbersome post-selection is required. Alternatively, aWTA (c) covers more effective modes while consistently using the same minimum number of hypotheses for both training and inference, discarding the post-selection and achieving better performance, for example, -9.41% minADE and -36.67% MissRate with MTR [7] versus (b) after selection.

many motion forecasting models increase the number of proposals [16] or use 'intention points' [7] to compensate for the performance drop from mode collapsing. However, to be effective, these approaches generally use a very large number of proposals. To reduce the number of predictions at inference time, a selection method like non-maximum suppression (NMS) or clustering is often applied [7], [16], [17]. These strategies make model performance highly sensitive to the proposal selection method and are computationally inefficient. While this issue has been explored in other fields, leading to several proposed solutions [15], [18], [19], these findings have been scarcely applied to autonomous driving problems [20], [21].

In this work, we take inspiration from a recently introduced training scheme for MCL, called *annealed* Multiple Choice Learning [22]. This method employs the 'annealed WTA' (aWTA) loss, a theoretically grounded approach that has shown promising results primarily on synthetic or toy datasets and small-scale models. Motivated by its potential, we propose to adopt the aWTA loss for motion forecasting models. In contrast to the simpler use cases explored in the original work [22], we demonstrate that this optimization scheme is particularly well suited for the trajectory prediction task. We show that this learning scheme can be effectively integrated into larger models, real-world experiments, and large-scale datasets. We find that aWTA systematically improves over current alternatives [20], [15], [21] across various recent forecasting models [7], [16] and

[1] Valeo.ai, Paris, France; Email: `firstname.lastname@valeo.com`

[2] LTCI, Télécom Paris, Institut Polytechnique de Paris, France.

[3] Sorbonne Université, Paris, France.

[*] Equal contribution.

Corresponding author: Y. Xu, `yihong.xu@valeo.com`

two large-scale real-world datasets [23], [24]. Furthermore, the use of aWTA drastically reduces the necessary number of hypotheses, eliminating the need for cumbersome test-time selection methods. Importantly, our proposal can be easily incorporated into any trajectory prediction model that normally uses WTA, which includes the vast majority of modern motion forecasting models.

II. RELATED WORK

Motion forecasting consists of predicting the future trajectory $y \in \mathcal{Y}$ for an agent, e.g., an autonomous vehicle, given a road map and a history of states as grasped in $x \in \mathcal{X}$. Current motion forecasting methods [5], [6], [7], [8], [16], [25], [26], [27], [28], [29], [1], [30], [31], [32], [33], [34] focus on designing architectures to better take into account the map information surrounding the agent (i.e., vehicle to forecast), the behaviors of neighboring agents, and the history of their own trajectories. For example, LaPred [6] enforces the use of map information when predicting the future trajectories of agents of interest. This is achieved by integrating the closest vectorized lane information and the past trajectories of neighboring agents into the agent features. More recent methods are primarily based on transformers. For instance, Wayformer [16] stacks transformer attention modules to encode the road graph, the agent history, and the agent interaction. Different modalities are then aggregated by a scene encoder, and the merged features are used for trajectory decoding, with randomly initialized and learnable mode queries and fully-connected layers. MTR [7] follows a similar transformer-based structure but introduces 'intention points' that provide prior information on the future endpoints to initialize their queries.

Different methods to learn future trajectories have been proposed. In particular, goal-based methods [35] define sparse anchors and learn to predict their confidence scores. Based on these scores, some anchors are selected as goals, and the method completes the trajectory for each goal. Heatmap-based methods [36] follow a similar idea but predict a heatmap with score responses that indicate the goals. However, most current state-of-the-art methods still follow the more straightforward approach of regression, which directly learns to predict plausible future trajectories using the commonly used Winner-Takes-All (WTA) loss. These methods achieve better performance compared to other heuristic approaches [37], [38].

Yet, while the WTA loss is known to be unstable during training [15], only a handful of works have examined this issue in motion forecasting pipelines [15], [21], [20]. For example, • **Relaxed WTA (RWTA)** [15] which updates both the winning head and the non-winning ones, with the latter being scaled by a small constant $\varepsilon = 0.05$, • **Evolving WTA (EWTA)** [21] which updates the top-n heads instead of just the winner, with n following a decreasing schedule during training, • **Divide and conquer (DAC)** [20] which recursively splits the set of hypotheses indices into two subsets until a certain depth, which depends on the training step. These WTA variants can be unified under

TABLE I: **Expression of** $q_t(f_k, x, y)$ **in WTA variants, with** $\mathcal{L}_t$ **from** (6). Here, $\mathbf{1}_{\text{WTA}}(k) \triangleq \mathbf{1}[k \in \text{argmin}_s \ell(f_s(x), y)]$, where $k^\star = \text{argmin}_s \ell(f_s(x), y)$. $\mathcal{T}_n(\theta, x, y)$ refers to the best n hypotheses for (x, y), while $\mathcal{P}(d_t)$ is the hypotheses set at depth d_t (as described in [20]). Z^{-1} ensures the weights normalization when needed. The column 'Schedule' ('Sched.') indicates if the weight q_t depends on the step t. aWTA is unique in having an infinite, uncountable set $\mathcal{V}$ for q_t values, offering more flexibility.

Method	$q_t(f_\theta^k, x, y)$	Sched.	Values $\mathcal{V}$
WTA	$\mathbf{1}_{\text{WTA}}(k)$	✗	$\{0, 1\}$
RWTA [15]	$(1 - \frac{K}{K-1}\varepsilon)\mathbf{1}_{\text{WTA}}(k) + \frac{\varepsilon}{K-1}$	✗	$\{1 - \varepsilon, \varepsilon/(K-1)\}$ or
EWTA [21]	$Z^{-1}\mathbf{1}[k \in \mathcal{T}_{n(t)}(\theta, x, y)]$	✓	$\{0, Z^{-1}\}$
DAC [20]	$Z^{-1}\sum_{\mathcal{S} \in \mathcal{P}(d_t) \mid k \in \mathcal{S}} \mathbf{1}[k^\star \in \mathcal{S}]$	✓	$|\mathcal{V}| < \infty, \mathcal{V} \subset \mathbb{Q}_+$
aWTA [22]	$Z^{-1}\exp(-\ell(f_k(x), y)/T(t))$	✓	$|\mathcal{V}| = \infty, \mathcal{V} \subset \mathbb{R}_+$

the common formulation with a hypotheses-weighted loss (described in the following section) where the scheduled weight q_t depends on the specific method, as detailed in Table I. Unlike previous works that are limited to small models and mostly simulated trajectories, our work proposes for motion forecasting an alternative called annealed Winner-Takes-All (aWTA). Our approach outperforms WTA and its variants [15], [21], [20] in large-scale datasets with modern motion forecasting models.

III. ANNEALED WINNER TAKES ALL (AWTA) IN MOTION FORECASTING

A. WTA for Motion Forecasting

Current regression-based motion forecasting models leverage WTA as the training objective to generate K diverse and plausible future trajectories. Given a past trajectory and map information x, a motion forecasting model parameterized by θ with K forecasting heads produces K future trajectory predictions $f_\theta^1, \ldots, f_\theta^K$.

During training, knowing its corresponding future ground-truth trajectory y, the WTA loss only optimizes the head that produces the prediction with the minimum cost ℓ:

$$k^\star = \text{argmin}_k \ell(f_\theta^k(x), y), \tag{1}$$

The $k^\star$-th head is then trained to minimize:

$$\mathcal{L}(\theta) = \ell(f_\theta^{k^\star}(x), y), \tag{2}$$

Typically, ℓ is the Average (euclidean) Distance Error (ADE) [5], [39],

$$\ell\left(f_\theta^k(x), y\right) = \frac{1}{L}\sum_{j=1}^{L}(f_\theta^k(x)_j - y_j)^2, \tag{3}$$

where L is the number of steps of the future trajectory. Alternatively, a Gaussian negative log-likelihood loss is also often used [5], [7], [16], considering each prediction as the component of a Gaussian mixture, which remains applicable with WTA and aWTA. The WTA optimization procedure can be interpreted as an input-dependent gradient-based variant of the Llyod algorithm [40], [15], which is known to be sensitive to initialization and prone to mode collapse.
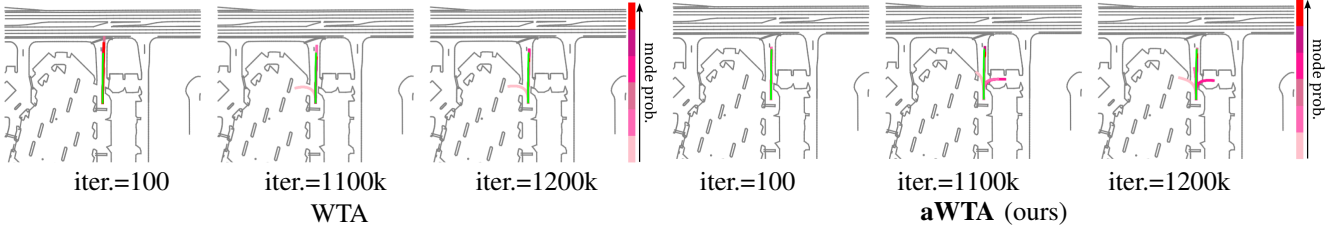
Fig. 2: **Evolution of mode distribution during training.** The predictions are obtained with Wayfomer [16] using WTA (left) or aWTA (right) on Waymo Open Motion Dataset (WOMD) [24]. We observe that the effective number of hypotheses increases with the training step with aWTA. The ground-truth trajectories are shown in green.

B. Incorporating aWTA to Motion Forecasting

To alleviate sensitivity to initialization and mode collapse issues, we take inspiration from annealed MCL [22] and incorporate aWTA into motion forecasting model training.

In theory, aWTA can be applied as a substitution to any WTA-based forecasting backbones. Specifically, modern motion forecasting methods [7], [16] are mostly transformer-based. They model the forecasting heads as learnable queries followed by a decoding network with fully connected layers. The queries are initialized from either intention points, obtained from clustering the end locations of trajectories in the training datasets [7], or random noise [16]. The queries interact through the attention mechanism with the map, the trajectory history, and the other agents. After that, the attended queries predict the future trajectories $f_\theta^k(x)$ and the associated mode probability via the decoding network. Note that while this query-based Transformer architecture can be seen as a straightforward implementation of Multiple Choice Learning, prior work analyzing MCL [14], [41] only experimented with Multi-Layer Perceptrons with multiple heads. As a result, the applicability of their findings to Transformer-based models remains empirically unverified.

aWTA softens the hard assignment between the best forecasting prediction $f_\theta^k(x)$ and the ground-truth y, described in (1), by using a softmin function, replacing the min operation in the selection of the Winner hypothesis. The soft assignment can be written in the form:

$$q_t\left(f_\theta^k \mid x, y\right) \triangleq \frac{1}{Z_{x,y}} \exp\left(-\frac{\ell\left(f_\theta^k(x), y\right)}{T(t)}\right), \quad (4)$$

$$Z_{x,y} \triangleq \sum_{s=1}^{K} \exp\left(-\frac{\ell\left(f_\theta^s(x), y\right)}{T(t)}\right), \quad (5)$$

where t is the training step. After the stop gradient operator being applied to q_t, we replace the WTA training objective (2) with a weighted sum of all $\ell\left(f_\theta^k(x), y\right)$ predicted from different heads or, in our case, queries:

$$\mathcal{L}_t(\theta) = \sum_{k=1}^{K} q_t(f_k, x, y)\ell(f_\theta^k(x), y), \quad (6)$$

Weight scheduling. Note that q_t is scheduled by controlling the temperature $T(t)$ of the current training step t. Ideally, at first, each of the hypotheses should equally contribute with $\lim_{t\to 0} q_t(f_k, x, y) = \frac{1}{K}$: in this

case, the predictions-target assignment is very *soft*. In this phase, the hypotheses converge toward a conditional mean, and the *effective* number of hypotheses is equal to 1 [42]. As training proceeds and the temperature decreases, $\mathcal{L}_t$ converges to the standard WTA training objective, where $\lim_{t\to\infty} q_t(f_k, x, y) = \mathbf{1}[k \in \arg\min_s \ell(f_s(x), y)]$ and the number of effective modes increases. Using the Boltzmann distribution for q_t is justified in Proposition 2 of [22], as the optimal way of constraining the soft assignment to a specific level of entropy. In practice, we observe in Figure 2 that the number of modes effectively increases during training. This is because the hypotheses better cover the space of possible future trajectories.

In practice, to allow q_t to slowly converge to the loss of WTA at the end of training, we control the temperature $T(t)$ using an exponential temperature scheduler:

$$T(t) = T_0 \rho^t, \quad (7)$$

where t is the current number of training steps. The initial temperature T_0, and the speed of temperature decay ρ are carefully ablated in the following section. Remarkably, as aWTA helps gradually distribute the predictions to cover all plausible trajectories, we significantly reduce the number of training hypotheses (i.e., number of queries, from 64 to 6) and discard the selection process, yielding a better forecasting performance with fewer training queries. Experimental proofs and ablation studies are given in Section IV.

IV. EXPERIMENTAL RESULTS

In this section, we first describe the experimental protocol and the training details in Section IV-A. We then demonstrate the benefits of aWTA in motion forecasting in Section IV-B. Then in Section IV-C, we compare aWTA with existing WTA variants, showcasing its superior performance for recent motion forecasting models in large-scale datasets, followed by an ablation study of important aWTA components.

A. Experimental protocol and implementation details

Evaluation protocol. For fair comparisons, we follow the standardized evaluation protocol proposed in UniTraj [12]. Models are tasked to predict future trajectories (up to 6 trajectories) for the next 6 seconds at 10Hz (i.e., $L = 60$), based on the scene (e.g., HD maps) and agent trajectories from the past 2 seconds. We use the following metrics:
- **minADE**$_k$, minimum over $k = 6$ predictions of the

3

TABLE II: **Comparison between aWTA and the standard WTA.** The use of aWTA, over the default WTA loss, consistently improves MTR [7] and Wayformer [16] over Argoverse 2 and WOMD, across all metrics. Results on the validation sets.

	Argoverse 2 [23]				**WOMD** [24]			
	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
MTR [7] $w/$ WTA (default)	2.12	0.85	1.68	0.30	2.19	0.76	1.74	0.31
MTR [7] $w/$ **aWTA (ours)**	**2.08** (- **1.89%**)	**0.77** (- **9.41%**)	**1.46** (- **13.10%**)	**0.19** (- **36.67%**)	**1.98** (- **9.59%**)	**0.63** (- **17.11%**)	**1.34** (- **22.99%**)	**0.18** (- **41.94%**)
Wayformer [16] $w/$ WTA (default)	2.19	0.79	1.57	0.24	2.10	0.66	1.46	0.22
Wayformer [16] $w/$ **aWTA (ours)**	**2.16** (- **1.37%**)	**0.78** (- **1.27%**)	**1.53** (- **2.55%**)	**0.22** (- **8.33%**)	**2.09** (- **0.48%**)	**0.65** (- **1.52%**)	**1.45** (- **0.68%**)	**0.21** (- **4.55%**)

Average Distance Error (the average of point-wise L^2 distances between the prediction and ground-truth forecasts), • **minFDE**$_k$, minimum over k = 6 predictions of the Final Distance Error, • MissRate **MR**$_{k@x}$, ratio of forecasts with minFDE$_k$ > 2 meters, and lastly, • **brier-FDE**, sum of minFDE$_k$ and $(1 - \delta)^2$, with δ the hypothesis score. Performance is reported on the validation sets.

Models and datasets. We integrate aWTA into two recent motion forecasting models: MTR [7] (60M parameters) and Wayformer [16] (16M parameters), provided in UniTraj [12]. Vanilla MTR is trained with 64 hypotheses and selects 6 of them with NMS, while Vanilla Wayformer uses only 6 hypotheses for training. We replace all WTA objectives (for both final and intermediate predictions) in the original implementation by aWTA with an exponential temperature scheduler, and the number of hypotheses is fixed to 6 for training and inference. We note that the hypothesis score loss (i.e., cross-entropy loss) remains unchanged. We conduct experiments on two large-scale real-world datasets: Argoverse 2 [23], containing 180k trajectories, and Waymo Open Motion Dataset (WOMD) [24], with 1.8M trajectories.

Training details. All models are trained until their convergence based on **minFDE**$_k$ (around 100 epochs). Training time with aWTA is ~4 minutes (MTR) or 3 minutes (Wayformer) per epoch for Argoverse 2 and 49 minutes (MTR) or 29 minutes (Wayformer) for WOMD, on a node of 8x A100 GPUs. The total batch size is fixed to 256 for MTR and 128 for Wayformer, as in [12]. We perform a grid search on the initial temperature and temperature decay, selecting the best-performed hyper-parameters, as discussed in Section IV-C. The forecasting performance is not highly sensitive to these hyperparameters within a reasonable range. For WTA variants, we follow the descriptions in their respective papers and reimplement them to MTR [7] and Wayformer [16].

B. Main results: benefits of aWTA.

Using aWTA boosts forecasting performances. We present the performance of MTR [7] and Wayformer [16], with the default WTA loss and our aWTA proposal, on Argoverse 2 [23] and WOMD [24] in Table II. The first conclusion is that the use of aWTA, in place of standard WTA, systematically improves both methods, on the two datasets, across all metrics. For instance, using aWTA yields a decrease of 23% in minFDE for MTR on WOMD, and a decrease of 8.3% in MissRate for Wayformer on Argoverse 2. This benefit is consistently observed and particularly visible
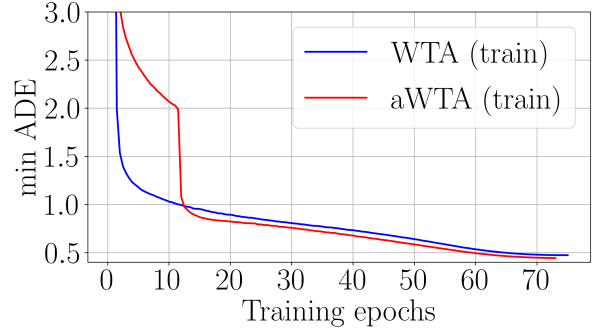

Fig. 3: **Phase transition with aWTA Loss.** Evolution of (averaged) minADE during training of Wayformer [16] on Argoverse 2 [23], comparing WTA (blue) and aWTA (red) training setups. A sudden drop in error is observed around epoch 12, consistent with the expected behavior of the deterministic annealing procedure. Here, we see that the aWTA converges to a better training fit compared to WTA.

TABLE III: **aWTA: better performance, less hypotheses.** We compare training with aWTA and 6 hypotheses to training with WTA with 6 hypotheses or 64 hypotheses after 'Non-Maximum Suppression' post-processing ('+ NMS'). The results are reported in the validation sets of Argoverse 2 [23] and WOMD [24] using MTR [7]. 'time' indicates the training time per epoch in minutes.

MTR $w/$	time (min.)	#hypo. train	#hypo. infer.	**Argoverse 2** [23] Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
WTA	3	6	6	3.34	1.36	3.08	0.56
WTA + NMS	5	64	6	2.12	0.85	1.68	0.30
aWTA (ours)	4	6	6	**2.08**	**0.77**	**1.46**	**0.19**

MTR $w/$	time (min.)	#hypo. train	#hypo. infer.	**WOMD** [24] Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
WTA	41	6	6	3.40	1.27	3.22	0.58
WTA + NMS	61	64	6	2.19	0.76	1.74	0.31
aWTA (ours)	49	6	6	**1.98**	**0.63**	**1.34**	**0.18**

for MTR, as this may be explained by the bigger model size and auxiliary losses in all intermediate layers of MTR.

Phase transitions. As the temperature decreases, we observe abrupt drops in the loss value, as shown in Figure 3. These drops can be theoretically explained as 'phase transitions' [43], [42], [22]. At certain critical temperatures, sudden improvement occurs in the metrics. These phase transitions correspond to moments when the effective number of hypotheses changes. Our observations align with the theoretical insights from [22].

Discussion on the number of hypotheses. Motion forecasting methods trained with WTA often suffer from mode collapse when trained with a small number of hypotheses, e.g., 6, as observed in Figure 1 and quantified with the poor results in Table III (6 hypotheses for both training and inference). To mitigate this issue, methods typically train with more hypotheses, e.g., 64, and then use post-processing techniques, such as Non-Maximum Suppression (NMS) or clustering, to select the top-K best hypotheses (in our case, $K = 6$ following the experimental protocol). However, this approach requires careful design and tuning of the post-processing step. In contrast, with aWTA, we use the same number of hypotheses during training and inference, $K = 6$. This approach yields better performance and save ~20% the training time than methods that artificially increase the number of hypotheses during training and that require post-processing of predictions, as seen in Table III.

TABLE IV: **Comparison between aWTA, WTA variants**, namely RWTA [15], EWTA [21], and DAC [20], and Linear aWTA. Results are reported on the validation sets of Argoverse 2 [23] and WOMD [24] using MTR [7] and Wayformer [16].

MTR [7]	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
	Argoverse 2 [23]			
w/ WTA	2.12	0.85	1.68	0.30
w/ RWTA [15]	2.30	0.83	1.63	0.20
w/ EWTA [21]	2.09	**0.77**	1.49	0.20
w/ DAC [20]	2.10	0.78	1.50	0.20
w/ Linear aWTA	4.44	1.58	3.75	0.52
w/ **aWTA (ours)**	**2.08**	**0.77**	**1.46**	**0.19**
MTR [7]	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
	WOMD [24]			
w/ WTA	2.19	0.76	1.74	0.31
w/ RWTA [15]	2.17	0.70	1.49	0.19
w/ EWTA [21]	2.00	0.65	1.37	0.19
w/ DAC [20]	2.15	0.71	1.57	0.25
w/ Linear aWTA	2.15	0.71	1.57	0.25
w/ **aWTA (ours)**	**1.98**	**0.63**	**1.34**	**0.18**
Wayformer [16]	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
	Argoverse 2 [23]			
w/ WTA	2.19	0.79	1.57	0.24
w/ RWTA [15]	2.30	0.81	1.64	0.25
w/ EWTA [21]	2.20	0.79	1.58	0.24
w/ DAC [20]	2.18	0.78	1.56	0.24
w/ Linear aWTA	3.43	1.31	2.73	0.35
w/ **aWTA (ours)**	**2.16**	**0.78**	**1.53**	**0.22**
Wayformer [16]	Brier-FDE↓	minADE↓	minFDE↓	MissRate↓
	WOMD [24]			
w/ WTA	2.10	0.66	1.46	0.22
w/ RWTA [15]	2.23	0.69	1.55	**0.21**
w/ EWTA [21]	2.12	0.67	1.50	0.23
w/ DAC [20]	2.16	0.68	1.53	0.24
w/ Linear aWTA	5.25	1.66	4.55	0.66
w/ **aWTA (ours)**	**2.09**	**0.65**	**1.45**	**0.21**

C. Baselines and ablation.

a) Comparison to other WTA variants: Although some WTA variants have been proposed to alleviate the instability of WTA, it is worth noting that most of the WTA variants (RWTA [15], EWTA [21] and DAC [20]) have primarily been tested in simulated datasets with small models, e.g., ResNet-18, and have rarely been tested on real-world datasets. To our knowledge, we conducted the first comparison of different WTA variants on large-scale real-world datasets.

From Table IV, we observe that our proposed aWTA consistently outperforms other variants for MTR and Wayformer on Argoverse 2 and WOMD across all metrics. In particular, we find that no other WTA variant improves Wayformer, especially in WOMD, while the use of the aWTA loss does. This might be because these methods are more sensitive to the hypothesis initialization, which is random noise in Wayformer. Moreover, aWTA is unique in having an infinite, uncountable set $\mathcal{V}$ for q_t values, providing more flexibility.

Additionally, we present qualitative examples of predictions given by MTR trained with different WTA variants in Figure 4. We observe that MTR with aWTA tends to cover a wider range of feasible trajectories compared to other WTA variants. In contrast, other variants distribute less widely over different plausible trajectories and struggle to cover the diverse ground-truth trajectories.

b) Ablation study: aWTA employs a temperature scheduling scheme where the temperature gradually decreases during training. Eventually, this scheme converges to an optimization process equivalent to the standard WTA approach. This training scheme introduces two key parameters: (1) the temperature scheduler which determines how quickly the temperature decreases and the pattern of this decrease, and (2) the initial temperature value. We discuss below the choice and impact of these parameters on the performance.
Temperature scheduler. To implement the aWTA loss, we ablate on two types of scheduler, i.e., exponential (7) or linear (8) schedulers:

$$T(t) = T_0 \left(1 - \frac{t}{100}\right) \mathbf{1}(t < 100) + T_f \mathbf{1}(t \geq 100), \quad (8)$$

where $T_f = 1e-8$. Both schedulers start from the same initial temperature T_0 (10 for MTR and 8 for Wayformer) and ρ is set to 0.834 and 0.89 for MTR and Wayformer respectively.

From the results shown in Table IV (Linear aWTA vs. aWTA), we observe that the exponential scheduler, which provides a slower slope in the end, achieves much better empirical performance. Therefore, we choose the exponential temperature scheduler for our implementation.
Initial temperature. We study the sensitivity of the forecasting performance to the initial temperature of aWTA. For these experiments, we use the exponential scheduler with a temperature decay fixed at 0.834 for both MTR and Wayformer. From the results shown in Figure 5, we observe that the performance gains brought by aWTA are maintained across a wide range of initial temperature values, showing that aWTA is relatively insensitive to this hyperparameter.
Temperature decay. Similarly, with a fixed initial temperature (10 for MTR and 8 for Wayformer) we study the performance impact of the rate of the temperature decay, with exponential scheduling. The results are shown in Figure 6.

From Figure 5, Figure 6 and our empirical experience, we observe that aWTA improves over the baselines (dotted lines of the model trained with WTA), as long as the initial temperature is within the order of magnitude, as the average distance between the predictions and the ground truth, and the temperature decay is around 0.8. This demonstrates that
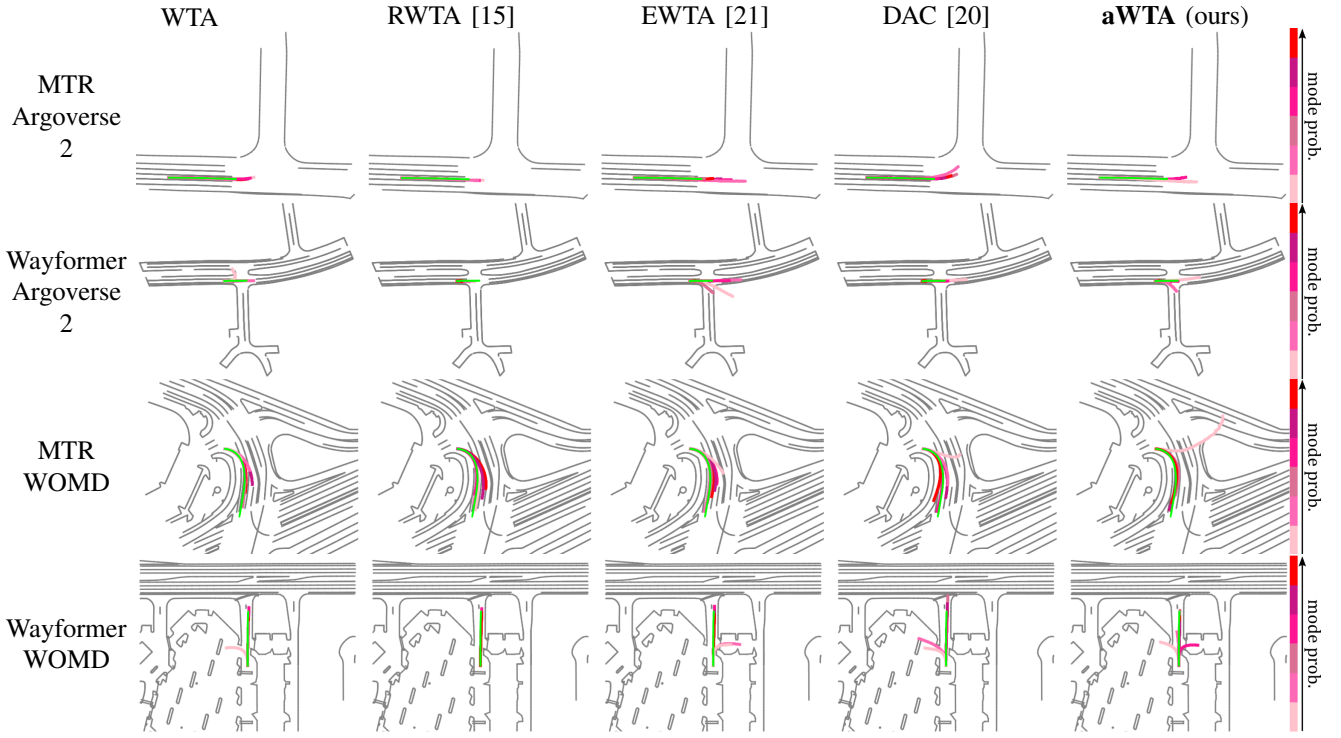
Fig. 4: **Qualitative comparison between aWTA and WTA variants.** Predictions are shown on Argoverse 2 [23] (rows 1 and 2) and WOMD [24] (rows 3 and 4) for MTR [7] (rows 1 and 3) and Wayformer [16] (rows 2 and 4) models. The ground-truth trajectories are shown in green.
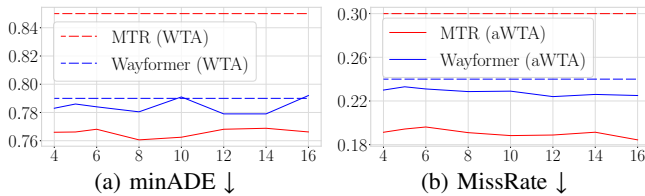

Fig. 5: **Impact of the initial temperature value** T_0 **(x axis) on the performance (y axis)** for aWTA in two different methods, MTR [7] (in red) and Wayformer [16] (in blue). Results are obtained with Argoverse 2 validation set [23]. The dotted lines are baselines trained with default WTA loss.

Fig. 6: **Impact of temperature decay rate** ρ **(x axis) on the performance (y axis)** for aWTA with an exponential scheduler in two different methods, MTR [7] (in red) and Wayformer [16] (in blue). Results are on Argoverse 2 valset [23]. The dotted lines are baselines with WTA loss.

aWTA is not too sensitive to such hyperparameters, facilitating its adoption in different motion forecasting methods.

V. CONCLUSION

In the past years, the motion forecasting literature has widely and overwhelmingly adopted the WTA training loss. In fact, this training scheme is both simple and quite effective in capturing the many modes of the output, even when the modes have unbalanced probabilities. However, WTA comes with well-established training instabilities that have been largely ignored in the community. Instead, we collectively adopted cumbersome workarounds that increase the computation cost and make using the predictions more difficult. In this paper, we examine the root issues with WTA for motion forecasting and propose to tackle them directly using a better, while simple, training scheme. In doing so, we show that we do not need an additional number of hypotheses, nor the
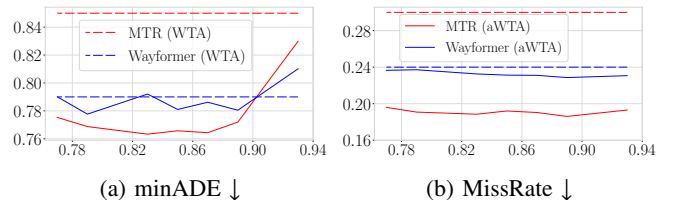
post-selection step, to attain competitive performance. Our work highlights the importance for the community to further investigate and improve upon WTA training, specifically in the context of motion forecasting, as we believe it to be currently a bottleneck for motion forecasting methods.

REFERENCES

[1] Y. C. Tang and R. Salakhutdinov, "Multiple futures prediction," in *NeurIPS*, 2019, pp. 15 424–15 434. 1, 2

[2] S. H. Park, G. Lee, J. Seo, M. Bhat, M. Kang, J. Francis, A. R. Jadhav, P. P. Liang, and L. Morency, "Diverse and admissible trajectory forecasting through multimodal context understanding," in *ECCV*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., 2020, pp. 282–298. 1

[3] A. Cui, S. Casas, A. Sadat, R. Liao, and R. Urtasun, "Lookout: Diverse multi-future prediction and planning for self-driving," in *ICCV*, 2021, pp. 16 087–16 096. 1

[4] L. Calem, H. Ben-Younes, P. Pérez, and N. Thome, "Diverse probabilistic trajectory forecasting with admissibility constraints," in *ICPR*, 2022, pp. 3478–3484. 1

[5] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *ECCV*, 2020, pp. 683–700. 1, 2

[6] B. Kim, S. H. Park, S. Lee, E. Khoshimjonov, D. Kum, J. Kim, J. S. Kim, and J. W. Choi, "Lapred: Lane-aware prediction of multi-modal future trajectories of dynamic agents," in *CVPR*, 2021, pp. 14 636–14 645. 1, 2

[7] S. Shi, L. Jiang, D. Dai, and B. Schiele, "Motion transformer with global intention localization and local movement refinement," *NeurIPS*, pp. 6531–6543, 2022. 1, 2, 3, 4, 5, 6

[8] A. Cui, S. Casas, K. Wong, S. Suo, and R. Urtasun, "Gorela: Go relative for viewpoint-invariant motion forecasting," in *ICRA*, 2023, pp. 7801–7807. 1, 2

[9] Y. Xu, L. Chambon, É. Zablocki, M. Chen, A. Alahi, M. Cord, and P. Pérez, "Towards motion forecasting with real-world perception inputs: Are end-to-end approaches competitive?" in *ICRA*, 2024, pp. 18 428–18 435. 1

[10] Z. Xu, R. Zhou, Y. Yin, H. Gao, M. Tomizuka, and J. Li, "MATRIX: multi-agent trajectory generation with diverse contexts," in *ICRA*, 2024, pp. 12 650–12 657. 1

[11] S. Ettinger, K. Goel, A. Srivastava, and R. Al-Rfou, "Scaling motion forecasting models with ensemble distillation," in *ICRA*, 2024, pp. 4812–4818. 1

[12] L. Feng, M. Bahari, K. M. B. Amor, É. Zablocki, M. Cord, and A. Alahi, "Unitraj: A unified framework for scalable vehicle trajectory prediction," in *ECCV*, 2024, pp. 106–123. 1, 3, 4

[13] A. Guzman-Rivera, D. Batra, and P. Kohli, "Multiple choice learning: Learning to produce multiple structured outputs," *NeurIPS*, vol. 25, 2012. 1

[14] S. Lee, S. Purushwalkam Shiva Prakash, M. Cogswell, V. Ranjan, D. Crandall, and D. Batra, "Stochastic multiple choice learning for training diverse deep ensembles," *NeurIPS*, vol. 29, 2016. 1, 3

[15] C. Rupprecht, I. Laina, R. DiPietro, M. Baust, F. Tombari, N. Navab, and G. D. Hager, "Learning in an uncertain world: Representing ambiguity through multiple hypotheses," in *ICCV*, 2017, pp. 3591–3600. 1, 2, 5, 6

[16] N. Nayakanti, R. Al-Rfou, A. Zhou, K. Goel, K. S. Refaat, and B. Sapp, "Wayformer: Motion forecasting via simple & efficient attention networks," in *ICRA*, 2023, pp. 2980–2987. 1, 2, 3, 4, 5, 6

[17] L. Lin, X. Lin, T. Lin, L. Huang, R. Xiong, and Y. Wang, "Eda: Evolving and distinct anchors for multimodal motion prediction," in *AAAI*, 2024, pp. 3432–3440. 1

[18] E. Ilg, O. Cicek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox, "Uncertainty estimates and multi-hypotheses networks for optical flow," in *ECCV*, 2018, pp. 652–667. 1

[19] M. Firman, N. D. Campbell, L. Agapito, and G. J. Brostow, "Diversenet: When one right answer is not enough," in *CVPR*, 2018, pp. 5598–5607. 1

[20] S. Narayanan, R. Moslemi, F. Pittaluga, B. Liu, and M. Chandraker, "Divide-and-conquer for lane-aware diverse trajectory prediction," in *CVPR*, 2021, pp. 15 799–15 808. 1, 2, 5, 6

[21] O. Makansi, E. Ilg, O. Cicek, and T. Brox, "Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction," in *CVPR*, 2019, pp. 7144–7153. 1, 2, 5, 6

[22] D. Perera, V. Letzelter, T. Mariotte, A. Cortés, M. Chen, S. Essid, and G. Richard, "Annealed multiple choice learning: Overcoming limitations of winner-takes-all with annealing," *arXiv preprint arXiv:2407.15580*, 2024. 1, 2, 3, 4

[23] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, D. Ramanan, P. Carr, and J. Hays, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," in *NeurIPS Datasets and Benchmarks*, 2021. 2, 4, 5, 6

[24] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020, pp. 2446–2454. 2, 3, 4, 5, 6

[25] T. Phan-Minh, E. C. Grigore, F. A. Boulton, O. Beijbom, and E. M. Wolff, "Covernet: Multimodal behavior prediction using trajectory sets," in *CVPR*, 2020, pp. 14 074–14 083. 2

[26] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting," in *ICCV*, 2021, pp. 9813–9823. 2

[27] J. Gu, C. Sun, and H. Zhao, "Densetnt: End-to-end trajectory prediction from dense goal sets," in *ICCV*, 2021, pp. 15 303–15 312. 2

[28] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," in *CoRL*, 2019, pp. 86–99. 2

[29] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. Lam, D. Anguelov, and B. Sapp, "Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction," in *ICRA*, 2022, pp. 7814–7821. 2

[30] H. Cui, V. Radosavljevic, F. Chou, T. Lin, T. Nguyen, T. Huang, J. Schneider, and N. Djuric, "Multimodal trajectory predictions for autonomous driving using deep convolutional networks," in *ICRA*, 2019, pp. 2090–2096. 2

[31] P. Kothari, D. Li, Y. Liu, and A. Alahi, "Motion style transfer: Modular low-rank adaptation for deep motion forecasting," in *CoRL*, 2022, pp. 774–784. 2

[32] 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, pp. 1609–1615. 2

[33] Y. Xu, Éloi Zablocki, A. Boulch, G. Puy, M. Chen, F. Bartoccioni, N. Samet, O. Siméoni, S. Gidaris, T.-H. Vu, A. Bursuc, E. Valle, R. Marlet, and M. Cord, "Valeo4cast: A modular approach to end-to-end forecasting," 2024. [Online]. Available: https://arxiv.org/abs/2406.08113 2

[34] Y. Xu, Y. Yin, T.-H. Vu, A. Boulch, Éloi Zablocki, and M. Cord, "Ppt: Pre-training with pseudo-labeled trajectories for motion forecasting," 2024. [Online]. Available: https://arxiv.org/abs/2412.06491 2

[35] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, *et al.*, "Tnt: Target-driven trajectory prediction," in *CoRL*, 2021, pp. 895–904. 2

[36] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "Home: Heatmap output for future motion estimation," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 500–507. 2

[37] J. Gu, C. Hu, T. Zhang, X. Chen, Y. Wang, Y. Wang, and H. Zhao, "Vip3d: End-to-end visual trajectory prediction via 3d agent queries," in *CVPR*, 2023, pp. 5496–5506. 2

[38] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li, "Planning-oriented autonomous driving," in *CVPR*, 2023, pp. 17 853–17 862. 2

[39] H. Ben-Younes, É. Zablocki, M. Chen, P. Pérez, and M. Cord, "Raising context awareness in motion forecasting," in *CVPR Workshops*, 2022, pp. 4408–4417. 2

[40] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982. 2

[41] K. Lee, C. Hwang, K. Park, and J. Shin, "Confident multiple choice learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2014–2023. 3

[42] K. Rose, "A mapping approach to rate-distortion computation and analysis," *IEEE Transactions on Information Theory*, vol. 40, no. 6, pp. 1939–1952, 1994. 3, 4

[43] K. Rose, E. Gurewitz, and G. Fox, "A deterministic annealing approach to clustering," *Pattern Recognition Letters*, vol. 11, no. 9, pp. 589–594, 1990. 4