2023-01-0865

# Hierarchical Neural Network-based Prediction Model of Pedestrian Crossing Behavior at Unsignalized Crosswalks

Mate Ćorić, Branimir Škugor, Joško Deur (University of Zagreb, Faculty of Mechanical Engineering)

Vladimir Ivanović, H. Eric Tseng (Ford Motor Company)

## Abstract

To enable smooth and low-risk autonomous driving in the presence of other road users, such as cyclists and pedestrians, appropriate predictive safe speed control strategies relying on accurate and robust prediction models should be employed. However, difficulties related to driving scene understanding and a wide variety of features influencing decisions of other road users significantly complexifies prediction tasks and related controls. This paper proposes a hierarchical neural network (NN)-based prediction model of pedestrian crossing behavior, which is aimed to be applied within an autonomous vehicle (AV) safe speed control strategy. Additionally, different single-level prediction models are presented and analyzed as well, to serve as baseline approaches. The hierarchical NN model is designed to predict the probability of pedestrian crossing the crosswalk prior to the vehicle at the high level, and parameters of Gaussian probability distribution of pedestrian entry time to the crosswalk at the low level. On the other hand, the baseline single-level models only provide entry time probability distributions, either in discrete form or in the form of bimodal Gaussian probability distribution. The proposed hierarchical model is validated against the baseline ones for a simplified single-vehicle/single-pedestrian case, where the data obtained through large-scale simulations of a game theory-based pedestrian model and an open-loop driven vehicle are used.

## Introduction

Machine learning algorithms have been proven to be successful in many different fields of industry and science, like computer vision, natural language processing, and computer game playing. Some of key aspects of autonomous vehicles, e.g., lane tracking systems and scene understanding, are already based on machine learning algorithms for pattern recognition. However, complex interactions with other road agents, e.g. pedestrians and cyclists, still pose a for autonomous vehicles [1]. Interactions between vehicles and pedestrians have been in the spotlight of many research papers in recent years [2]. Some of the main factors that influence behavior at crosswalk are for instance gender, age, culture, size of the group, behavioral psychology, type of uncontrolled pedestrian crosswalk, etc. [3], [4]. Uncertainty related to pedestrian crossing behavior should be therefore included in vehicle control strategies while driving near crosswalks with pedestrians [5]. A lot of research studies have shown that prediction of pedestrian intention is very complex due to inherent stochasticity of related decisions [6]–[8]. For instance, authors in [9] propose a large-scale pedestrian intention estimation (PIE) dataset and novel algorithm that uses observed pedestrian trajectory and local visual information for prediction of pedestrian crossing intention. It has been shown in [10] that pedestrians mainly focus on vehicle position and speed when making crossing decisions. The pedestrian crossing behavior prediction approaches could be categorized according to the prediction of crossing intention [11], walking destination [12], movement trajectory [13], etc. The pedestrian movement trajectory prediction is in focus of most recent studies [14], [15], where neural network-based regression models are considered for this task. Difficulties of predicting whole pedestrian trajectories is tackled in authors' previous paper [23], by focusing only on prediction of quantities relevant from the perspective of vehicle, i.e., the pedestrian entry time to and exit time from a crosswalk. Apart from predictive tasks, some references deal with a vehicle-pedestrian interaction modelling for the purpose of simulation-based analyses, such as [16] which propose a game theory-based modelling of both vehicles and pedestrians by using experimental data collected at real uncontrolled crosswalk.

This paper deals with the development of a hierarchical prediction model of pedestrian crossing behavior near unsignalized crosswalks. The considered model consists of two submodels: binary classifier called *high-level model* that predicts probability of pedestrian crossing prior to the vehicle, and *low-level model* that predicts parameters of unimodal Gaussian probability distribution of pedestrian entry time in the hypothetical case of pedestrian opting for crossing decision. This model is built upon the previous work related to single-level model providing discrete probability distributions of entry and exit times [23] (later referred to as the discrete model). A variant of single-level model providing parameters of bimodal Gaussian probability distribution of pedestrian entry time is proposed herein, to serve as an additional baseline along with the discrete model. The bimodal distribution is introduced to capture two distinctive probability modes in a lumped-parameter manner, which relate to the cases when pedestrian opts for crossing and goes prior to the vehicle, and when he/she yields and goes after the vehicle. All prediction models considered are based on deep feedforward neural networks, whose structures are optimized in terms of number of hidden layers and number of neurons within layers to minimize a validation loss function. The models are trained and validated based on the data obtained from simulations of single-vehicle and single-pedestrian interactions, where the pedestrian is simulated according to a game theory-based model [17], while the vehicle is driven in an open-loop manner by following pre-defined control parameters [23]. The models elaboration is focused on the pedestrian entry time prediction only, as it is considered more difficult than the pedestrian exit time prediction [23].

The motivation for organizing prediction model in the hierarchical manner is threefold: (i) simplification of the prediction task by focusing on the entry time of pedestrian when he/she is to cross prior to vehicle, which is only relevant case from the perspective of vehicle

control; (ii) numerical efficiency while performing predictions; i.e., if the high-level model provides near-zero probability of pedestrian opting for cross decision, the low-level model prediction does not need to be executed, and (iii) enhancement of prediction interpretability by getting single-value probability at the high level (e.g., could be easily visualized and used for instance in driver-assistance systems). On the other hand, the need for training and validation of two submodels (i.e., high- and low-level ones) instead of a single one could be considered as a certain drawback of the hierarchical approach.

Apart from the novel hierarchical prediction model, the contribution of the paper is also in proposing the Gaussian entry time probability distribution parameter prediction, which is much simpler in terms of number of related prediction model outputs when compared to the discrete model [23]. Furthermore, the models predicting Gaussian probability distribution parameters does not have prediction horizon limitation, as the related distributions can capture entry time happening anywhere on the future time horizon. Finally, having Gaussian probability distribution prediction instead of the discrete one could enhance numerical efficiency of safe speed control, as certain scenarios of pedestrian entry time could be derived readily in a straightforward manner from predicted parameters (e.g., those happening at predefined distribution percentiles).

## Vehicle-pedestrian interaction model

### *Simulation scenario*

The illustration of considered single-vehicle single-pedestrian simulation framework is shown in Fig. 1. For the sake of simplicity, the vehicle is set to approach to the crosswalk longitudinally within its lane, while the pedestrian is set to approach laterally. Additional assumption is introduced: if the vehicle enters the crossing area first, the pedestrian automatically yields and waits at the edge of crosswalk until the vehicle exits the crosswalk. The vehicle and pedestrian states include their positions (relative to the crosswalk) and speeds, and related discrete-time state equations read [23]:

$$s_x(k+1) = s_x(k) + v_x(k)\Delta T + \frac{a_x(k)\Delta T^2}{2}, \quad (1)$$

$$v_x(k+1) = v_x(k) + a_x(k)\Delta T, \quad (2)$$

where $s_x(k)$ and $v_x(k)$ are position and speed ($x \in \{p, v\}$, where the index $p$ denotes the pedestrian, and $v$ the vehicle), $a_x$ is acceleration, $k$ is a discrete time step, and $\Delta T$ is the discretization time (set here to 0.1 s). The positions are negative for both agents when they are positioned prior to the crosswalk, i.e., while they are approaching the crosswalk, and become positive once when they step onto the crosswalk (see origins of local coordinate systems in Fig. 1).

The pedestrian is simulated according to an experimentally calibrated game theory-based model, previously adopted from [16] and extended in [17], which consists of: (i) perception, (ii) expectation, (iii) decision, and (iv) motion submodels (see Fig. 1b). Here, only the main model functionalities are briefly described, while details can be found in [17]. The perception model is aimed to emulate the way in which the pedestrian observes the surrounding environment, i.e., vehicle position/distance and speed in this case. The expectation model, from the perspective of pedestrian, provides the probability of vehicle opting for cross decision $p_{v,cross}$. The probability of yield decision is then simply calculated as: $p_{v,yield} = 1 - p_{v,cross}$. The decision model provides the pedestrian decision $D_p \in \{0,1\}$ in the current simulation time step, where $D_p = 0$ corresponds to the yield and $D_p = 1$ to the cross decision. The probabilities provided by the expectation model and pre-defined payoffs of certain hypothetical vehicle-pedestrian decision combinations are used as inputs of the decision model. The motion model then determines pedestrian acceleration $a_p$ according to the following rules: if the decision is to yield, the pedestrian slows down with a constant deceleration to stop right at the edge of crosswalk, and otherwise the pedestrian accelerates with a pre-determined maximum acceleration $a_{p,\max} = 2\ \mathrm{m/s^2}$ until he/she reaches the initial speed $v_{p,0}$:

$$a_p(k) = \begin{cases} -\dfrac{v_p(k)}{2|s_p(k)|}, & \text{if } D_p = 0, \\ \operatorname{sgn}\left(v_{p,0} - v_p(k)\right) a_{p,\max}, & \text{else if } D_p = 1. \end{cases} \quad (3)$$

The signum function sgn(.) is defined as:

$$\operatorname{sgn}(x) = \begin{cases} -1, & \text{if } x < 0, \\ 0, & \text{if } x = 0, \\ 1, & \text{if } x > 0. \end{cases}$$

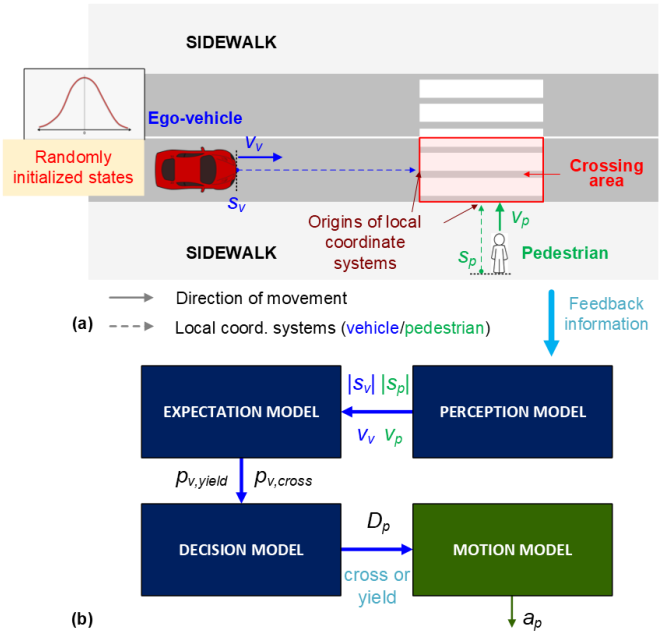


Figure 1. (a) Illustration of simulation framework involving single-vehicle and single-pedestrian, and (b) block diagram of pedestrian crossing model.

While the pedestrian behavior is simulated according to the presented game theory-based model, the vehicle is driven in an open-loop manner for the sake of data collection and thus prediction model training and testing purposes. Namely, at the beginning of each simulation, vehicle control parameters that determine the vehicle motion are prescribed, which include the reference acceleration $a_{v,R}$ and the target/reference speed $v_{v,R}$ (see illustration in Fig. 2). Note in Fig. 2 the vehicle keeps initial speed $v_{v,0}$ until the pedestrian appears, when the prescribed control parameters are started to be applied.

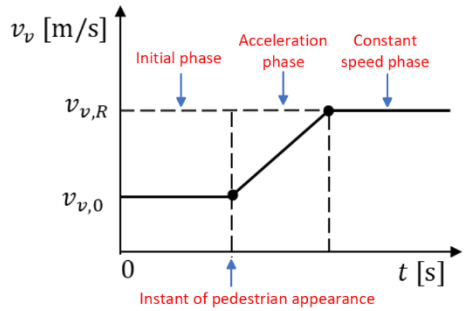


Figure 2. Illustration of vehicle speed trajectory determined by control parameters, i.e., reference speed $v_{v,R}$ and acceleration $a_{v,R}$ ($v_{v,0}$ is initial vehicle speed set at the beginning of simulation).

### *Simulation-generated dataset*

The dataset for prediction models is obtained from 10,000 vehicle-pedestrian interaction simulations. Each vehicle-pedestrian interaction is set to start with different combination of vehicle and pedestrian initial positions ($s_{v,0}$ and $s_{p,0}$), speeds ($v_{v,0}$ and $v_{p,0}$), and vehicle control parameters ($v_{v,R}$ and $a_{v,R}$). The initial vehicle position is set to $s_{v,0} = -100$ m, while the initial pedestrian position is sampled from the Gaussian distribution: $s_{p,0} \sim \mathcal{N}(-4, 0.8)$ m (the first argument is mean, and the second one is standard deviation). The pedestrian and vehicle speeds are also sampled from Gaussian distributions: $v_{p,0} \sim \mathcal{N}(1.38, 0.27)$ m/s and $v_{v,0} \sim \mathcal{N}(7.5, 2)$ m/s, respectively. The time instant of pedestrian appearance, i.e., beginning of the vehicle-pedestrian interaction is generated from an uniform distribution as $T_{p,0} \sim U(0, TTA_v)$ s, where $TTA_v = |s_{v,0}|/v_{v,0}$ is a predicted vehicle time-to-arrival (TTA) to the crosswalk under the constant speed assumption. The reference vehicle speed $v_{v,R}$ is generated from the same distribution as in the case of initial speed $v_{v,0}$, i.e., $v_{v,R} \sim \mathcal{N}(7.5, 2)$ m/s, while the reference acceleration is generated according to:

$$a_{v,R} = r \cdot a_{v,\max} \cdot \mathrm{sgn}\left(v_{v,R} - v_{v,0}\right), \qquad r \sim U(0,1), \tag{4a}$$

where $r$ is the random variable sampled from the unit uniform distribution $U(0,1)$, and $a_{v,\max}$ is predefined maximum (allowed) vehicle acceleration (here set to $a_{v,\max} = 2$ m/s$^2$).

In order to emulate uncertainty of pedestrian behavior, the expectation and decision model parameters are multiplied by the following random variable:

$$\varepsilon = 1 + \mathcal{U}(-1,1)\varepsilon_{\max}, \tag{4b}$$

where $\varepsilon_{\max}$ represents a parameter defining the maximum perturbation. Additionally, errors in observing the vehicle position and speed are emulated via perturbation of these quantities through sampling from the normal distributions $\hat{s}_v = \mathcal{N}(s_v,\ \varepsilon_{\max}|s_v|)$ and $\hat{v}_v = \max\left(v_v, \mathcal{N}(v_v,\ \varepsilon_{\max} v_v)\right)$, prior feeding them into the pedestrian expectation model.

The conducted interaction simulations and related data are divided into three subsets: train (70%), validation (15%) and test interactions (15%), which are used for finding the model parameters, tuning the model hyperparameters, and unbiased model evaluation, respectively. The datasets are denoted as $D_{train}$, $D_{val}$, and $D_{test}$; and are formed by selecting every fifth datapoint from each simulation (subsampling of the original data where $\Delta T = 0.1$ s). The subsampling is performed for the purpose of having reasonable size of individual datasets (see Table 1). Only datapoints for which entry time is lower or equal to 10 s is included within datasets, because a prediction time horizon of the discrete prediction model is set to 10 s (more details are given in the next section).

Table 1. Number of vehicle-pedestrian interactions and related datapoints within train, validation, and test datasets.

| Number of | Dataset | | | Total |
|---|---|---|---|---|
| | Train | Validation | Test | |
| **All interactions** | 7000 (70%) | 1500 (15%) | 1500 (15%) | 10,000 (100%) |
| **Datapoints** | 53,097 | 11,272 | 11,560 | 75,929 |
| **Pedestrian crosses prior to vehicle interactions** | 23,615 (44.5%) | 5069 (45.0%) | 5180 (44.8%) | 33,864 (44.6%) |

## Prediction models

### *Problem formulation*

The prediction models in this paper are conceived to provide a probability distribution at the output, rather than single-point predictions, to capture inherent uncertainty of pedestrian crossing behavior. All models analyzed are designed to have the same input features, i.e., the following variables from the current time step $k$: (i) vehicle position $s_{v,k}$, (ii) vehicle speed $v_{v,k}$, (iii) pedestrian position $s_{p,k}$, (iv) pedestrian speed $v_{p,k}$, (v) vehicle reference speed $v_{v,R,k}$, and (vi) vehicle reference acceleration $a_{v,R,k}$. The listed inputs are scaled to the interval $[-1, 1]$ by using the standard min-max normalization [18], prior feeding them to the prediction models, in order to annul the difference between different inputs ranges, thus facilitating the model training. The following training hyperparameter settings are used for all models considered: an optimization algorithm ADAM [19] with a learning rate $\alpha = 0.001$, a batch size 1000, the exponential linear unit (ELU) activation function in all hidden layers, and *sigmoid* or *softplus* activation function at the output (see Fig. 3, with more details given in the next subsections). The training is performed within Python by using Keras library [20] and TensorFlow [21] as a backend.

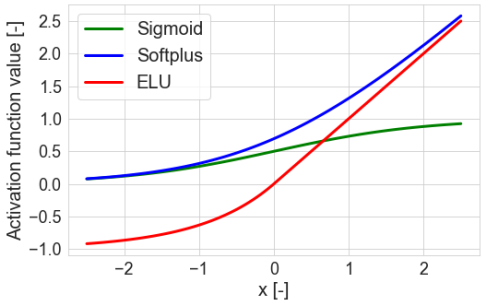


Figure 3. Activation functions used within prediction models (output of sigmoid function is typically interpreted as probability; *Softplus* function provide positive number and as such it is appropriate for prediction of entry time and Gaussian distribution standard deviation which are strictly positive values).

## *Discrete NN model*

The single-level model providing a discrete probability distribution is proposed in [23]. The model has been retrained on the newly generated dataset of vehicle-pedestrian interactions and used as a baseline model. The problem of pedestrian entry time prediction for this model is formulated as a classification task. The output of discrete model is sequence of probabilities $\hat{\boldsymbol{p}} = [\hat{p}_j]$, for $j = 1, 2, \dots, N$, which represents conditional probability distribution of discrete values of a pedestrian entry time over the prediction horizon $\tilde{\mathbf{t}} = [\tilde{t}_j = j\Delta T]$, where $N$ represents the prediction horizon length set to 100 (corresponding to 10 s for $\Delta T = 0.1$ s).

The output layer has *softmax* activation function since it transforms vector of numbers into a vector of probabilities. Thus, by applying *softmax* function it is ensured that the discrete model output satisfies the following property:

$$\sum_{j=1}^{N} \hat{p}_j = 1, \qquad \hat{p}_j \in [0,1]. \tag{5}$$

Supervised labels, i.e., the pedestrian entry time values, are formulated as one-hot vectors, $\mathbf{Y}_k \in \mathbb{R}^N$, where the value of $j^{\text{th}}$ element is set to one if the actual pedestrian entry time corresponds to the $j^{\text{th}}$ time step on the prediction horizon, and all other vector values are set to zero. A cross-entropy loss function is used for this multi-class classification problem:

$$L_d = \frac{1}{M} \sum_{i \in S} \sum_{k \in K_i} \left( - \sum_{j=1}^{N} Y_{k,j}^{i} \log\left(\hat{p}_{k,j}^{i}\right) \right), \tag{6}$$

where the index $i$ refers to $i^{\text{th}}$ simulation/interaction from the considered dataset $S$ (e.g., training dataset), $k$ to $k^{\text{th}}$ time step from $i^{\text{th}}$ simulation ($K_i$ is a set containing selected simulation steps from $i^{\text{th}}$ simulation), and $M$ is the total number of datapoints in the particular corresponding dataset, as given in Table 1. In the ideal perfect fit case, the cross-entropy loss is equal to zero. The optimal model architecture is obtained through a grid search of the number of hidden layers (1 to 8) and the number of neurons within each hidden layer (8, 16, 32, 64, 128), according to the minimal validation loss criterium. The final discrete model minimizing validation loss has six hidden layers with 16 neurons within each layer, and 3,172 parameters in total (see illustration in Fig. 4).

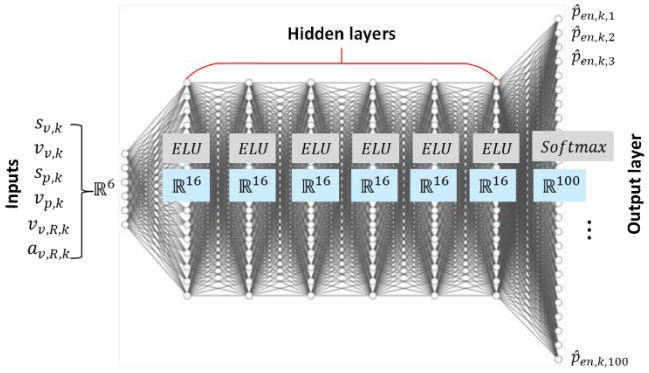


Figure 4. Optimal structure of discrete NN model.

## *Bimodal NN model*

Another baseline single-level model considered in this work is a feedforward NN model providing a bimodal Gaussian distribution parameters at the output (referred to as *bimodal model*). Here it is assumed that the pedestrian entry time can be successfully predicted with a simpler analytical probability distribution when compared to the output distribution of discrete model. The Gaussian distribution is used because of its following properties: (i) it is symmetrical around its mean value, so that the model uncertainty reflected in standard deviation is easy to interpret; and (ii) fitting the Gaussian distribution to data is simpler task than fitting the general distribution of the discrete model.

The bimodal NN model is designed to provide the following outputs: (i) the expected values of two Gaussian distributions ($\mu_1$ and $\mu_2$), (ii) the standard deviations of two Gaussian distributions ($\sigma_1$ and $\sigma_2$), and (iii) the weighting factor $\alpha \in [0,1]$ determining relative importance of the two modes. Additionally, the weighting factor ensures that a combination of two Gaussian distributions, $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$, is valid probability distribution (i.e., the integral of related PDF from $-\infty$ to $+\infty$ is to be equal to 1). Note that if $\alpha \approx 0$ or $\alpha \approx 1$, the model output distribution is effectively Gaussian distribution with a single distribution mode. Let $p_1(t|\,\mu_1, \sigma_1)$ and $p_2(t|\,\mu_2, \sigma_2)$ denote PDF of two Gaussian distributions $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$, respectively. The resulting bimodal distribution PDF is then defined as:

$$p_{bim}(t) = \alpha p_1(t|\,\mu_1, \sigma_1) + (1-\alpha) p_2(t|\,\mu_2, \sigma_2). \tag{7}$$

The learning problem is formulated as probability distribution estimation, and therefore the following standard negative log-likelihood is used as a loss function to be minimized:

$$L_{bim} = \frac{1}{M} \sum_{i \in S} \sum_{k \in K_i} -\log p_{bim}\left(t_{en,k}^{i} |\, \mu_{1,k}^{i}, \mu_{2,k}^{i}, \sigma_{1,k}^{i}, \sigma_{2,k}^{i}, \alpha_k^{i}\right), \tag{8}$$

where $M$ is the number of datapoints in the considered dataset $S$ (see Table 1), and $t_{en,k}^{i}$ is the actual/ground truth entry time from the perspective of $k^{\text{th}}$ time step from $i^{th}$ simulation. Note that the parameters $\mu_{1,k}^{i}, \mu_{2,k}^{i}, \sigma_{1,k}^{i}, \sigma_{2,k}^{i}, \alpha_k^{i}$ in (8) are provided by the prediction model depending on the model inputs from $k^{\text{th}}$ time step (see Fig. 5). The loss function (8) is derived from the maximum likelihood learning principle, and the minimization of $L_{bim}$ (due to negative sign) leads to the maximization of related likelihood [22].

There are two main reasons for introducing two Gaussian probability distribution modes instead of a single one: (i) a distribution with two modes has more flexibility and capacity in estimating true pedestrian entry time, and (ii) a distribution with two modes can naturally capture two possible outcomes of each interaction, i.e., pedestrian crosses prior to the vehicle and vice versa. The related model topology shown in Fig. 5 is somewhat more complex than a regular feedforward neural network architecture, as it has *base* layers at the first part of the model, which are followed by three branches, one for each target parameter: $\mu = (\mu_1, \mu_2)$, $\sigma = (\sigma_1, \sigma_2)$ and $\alpha$. These layers are referred to as *parameter layers* since they learn specific representations for each parameter separately. The base and parameter layers both represent hidden layers in the model. *Softplus* activation functions providing only positive values (see Fig. 3) are used within parameter layers that learn $\mu$ and $\sigma$, since both of these values should be positive. On the other hand, the sigmoid activation function is used within parameter layer that learns the weighting factor $\alpha$, since $\alpha$ can take values only in the interval [0, 1] (Fig. 3). The optimal NN model architecture in terms of minimal validation loss is obtained through a grid search of different numbers of base layers {1, 2, 3, 4}, parameters layers {1, 2, 3, 4}, and number of neurons {8, 16, 32, 64, 128} within hidden layers. The ultimately obtained optimal model has three base layers, one parameter layer, 64 neurons in each hidden layer, either base or parameter one, and 21,573 parameters in total (adjacent layers are mutually fully connected).

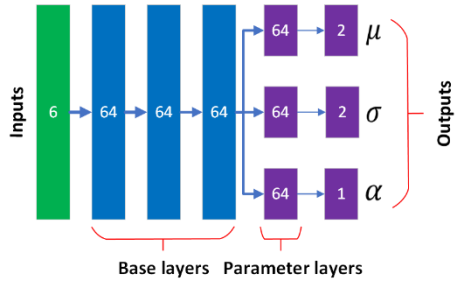


Figure 5. Illustration of bimodal neural network model.

## *Hierarchical NN model*

The *hierarchical model* consists of two distinctive submodels: (i) high-level one which is essentially binary classifier that predicts probability of pedestrian crossing prior to the vehicle, and (ii) low-level model that predicts pedestrian entry time via unimodal Gaussian distribution, whose parameters are provided in dependence on the model inputs. It should be noted that this entry time prediction is only for the hypothetical realization of pedestrian passing the crosswalk prior to the vehicle. As already stated in the introduction section, the learning problem is simplified in this way, since the low-level model is trained only on the data subset related to the pedestrian crossing prior to the vehicle outcomes. For this reason, the simpler unimodal (and not bimodal) Gaussian probability distribution can be employed for this task.

The following standard binary cross entropy is used as a loss function for binary classification in the high-level model [22]:

$$L_{h,high} = -\frac{1}{M}\sum_{i\in S}\sum_{k\in K_i} Y^i \log \hat{p}_k^i + \left(1-Y^i\right)\log\left(1-\hat{p}_k^i\right), \qquad (9)$$

where $Y^i$ denotes the binary variable which is set to 1 if in the related simulation the pedestrian crosses prior to the vehicle (otherwise takes value 0), and $\hat{p}_k^i$ is the probability of that event provided by the model based on inputs from $k^{\text{th}}$ time step from $i^{\text{th}}$ simulation.

The high-level model is trained on all datapoints from the dataset $D_{train}$ for both outcomes, i.e., when the pedestrian crosses prior to the vehicle and vice versa, while the low-level model is trained only on data where the pedestrian actually crossed prior to the vehicle. Similarly, as with the discrete and bimodal models, the optimal model architecture for the high-level model is obtained through grid search over different combinations of the number of hidden layers {1, 2, …, 8} and the number of neurons within them {8, 16, 32, 64, 128}. The resulting model has four hidden layers with 16 neurons within each as shown in Fig. 6, with 945 parameters in total.

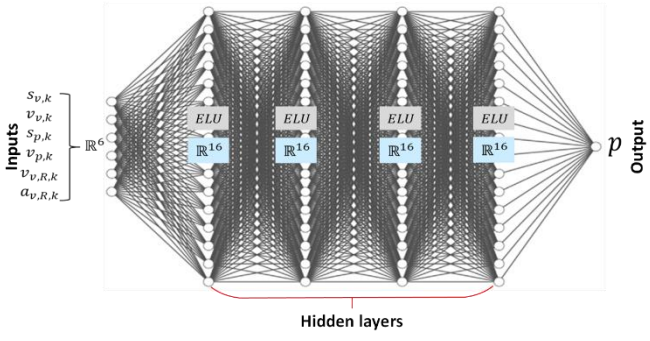


Figure 6. Illustration of high-level model architecture.

For the low-level model, the negative log-likelihood function is selected as a loss function [22]:

$$L_{h,low} = \frac{1}{M}\sum_{i\in S}\sum_{k\in K_i} -\log p_{low}\left(t_{en,k}^i \middle| \mu_k^i, \sigma_k^i\right), \qquad (10)$$

where $p_{low}$ is a PDF of unimodal Gaussian distribution defined by the mean value $\mu_k^i$ and the standard deviation $\sigma_k^i$ provided by the model based on the inputs from $k^{\text{th}}$ time step ($t_{en,k}^i$ is actual entry time relative to $k^{\text{th}}$ time step in $i^{\text{th}}$ simulation). The low-level model architecture optimization is performed over the same grid of number of hidden layers and neurons as in the case of bimodal model (see previous subsection). Finally, the resulting optimal architecture minimizing validation loss has two base layers, three parameter layers, 32 neurons in each hidden layer (both base and parameter; see Fig. 7), with 7,682 parameters in total.

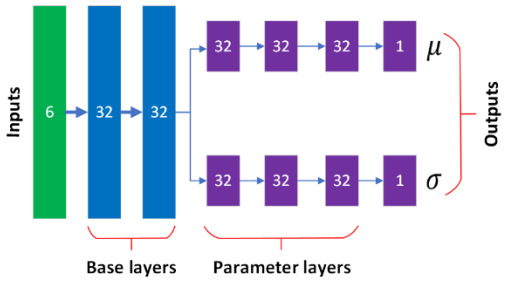


Figure 7. Optimal architecture of low-level model.

Fig. 8 provides example of high- and low-level model predictions, for one particular interaction from the test dataset $D_{test}$, in which the pedestrian crosses prior to vehicle. The model predictions are provided for several consecutive time steps, until the end of interaction when pedestrian enters the crosswalk. The results in Fig. 8 show that the high-level model provides relatively very high probabilities for the actual pedestrian crossing first outcome (>0.87), while the low-level model consistently sets predicted probability distribution very close to the actual pedestrian entry time.

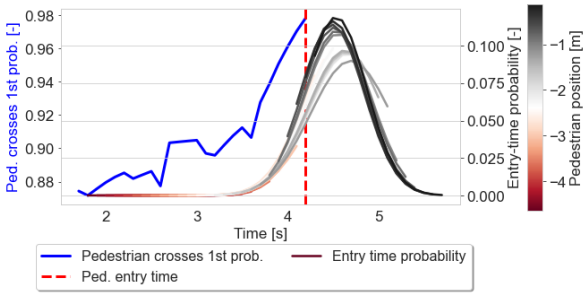


Figure 8. Illustration of predictions of high- (blue) and low-level models' predictions (red-grey) for whole interaction of vehicle and pedestrian with following initial conditions: $s_{p,0} = -4.6$ m, $v_{p,0} = 1.8$ m/s, $s_{v,0} = -86.8$ m, $v_{v,0} = 8.7$ m/s, $v_{v,R} = 10.6$ m/s, $a_{v,R} = 0.7$ m/s$^2$.

The proposed hierarchical model is aimed to be potentially applied within an AV safe speed control strategy in the following manner: if the high-level model probability is lower than some predefined low (close-to-zero) threshold, the possibility of pedestrian going prior to the vehicle is neglected and the low-level model prediction is not performed in this case. Otherwise, the low-level model prediction is performed and resulted entry time distribution is

provided along with the high-level probability to the safe speed control strategy.

## Results and models validation

In this section, the proposed hierarchical model is assessed and compared against the single-level baseline models based on the following metrics calculated on the test dataset $D_{test}$: predicted probability that the pedestrian crosses prior to the vehicle, probability of the actual pedestrian entry time values provided by the models, statistics of residuals related to the expected entry time under model distributions and real/actual pedestrian entry times. The prediction models are additionally validated through comparison of their probability distributions with ground truth ones, obtained by repetitive simulations of vehicle-pedestrian interactions for a wide range of simulation initial conditions.

### *Comparative assessment with respect to test dataset*

As it can be observed in Fig. 9, the models that have cross-entropy (CE) as the loss function yield less oscillations in the validation loss when compared to the models trained to minimize negative log-likelihood (NLL) loss. Table 2 gives the train, validation, and test loss values for each prediction model, which correspond to the epoch characterized by the minimum validation loss (vertical blue lines in Fig. 9). Note that only loss function values of bimodal and low-level models could be directly compared, as these models have the same loss definition and predict the same values. Somewhat lower loss, i.e., better accuracy in the case of bimodal model when compared to the low-level one could be attributed to the higher bimodal model flexibility (cf. model outputs in Figs. 5 and 7). Slight differences between the test/validation loss and the train loss values could be a sign of minor model overfitting.

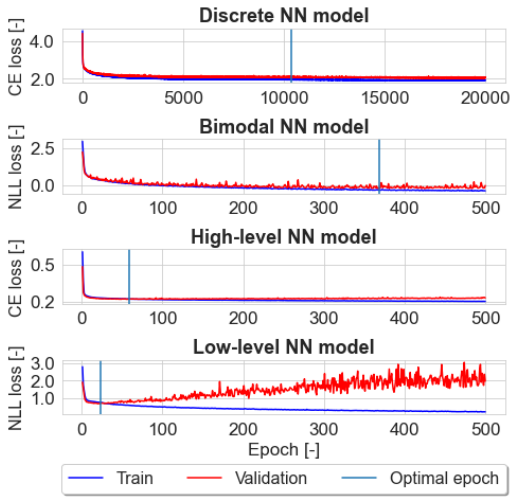


Figure 9. Training and validation learning curves of selected prediction model architectures, obtained by using training and validation datasets.

Table 2. Train, validation, and test loss values corresponding to minimum validation loss epochs given by vertical blue lines in Fig. 9.

| | Prediction NN-based model | | | |
|---|---|---|---|---|
| | **Discrete** | **Bimodal** | **High-level** | **Low-level** |
| **Train** | 1.90 | −0.39 | 0.25 | 0.20 |
| **Validation** | 2.02 | −0.23 | 0.26 | 0.67 |
| **Test loss** | 2.08 | −0.21 | 0.28 | 0.77 |
| **Optimal epoch** | 10355 | 368 | 59 | 23 |
| **Loss function** | Cross-entropy (CE) | Neg. log-likelihood (NLL) | Cross-entropy (CE) | Neg. log-likelihood (NLL) |

An example of model predictions for a particular datapoint from the test dataset is shown in Fig. 10. In this example, the pedestrian crosses prior to the vehicle and all three models put majority of their probability distributions around the actual pedestrian entry time designated by the vertical red line. Both discrete and bimodal models put secondary distribution mode right after the vehicle exit time, as there are also datapoints within dataset with similar inputs which result in pedestrian crossing after the vehicle.

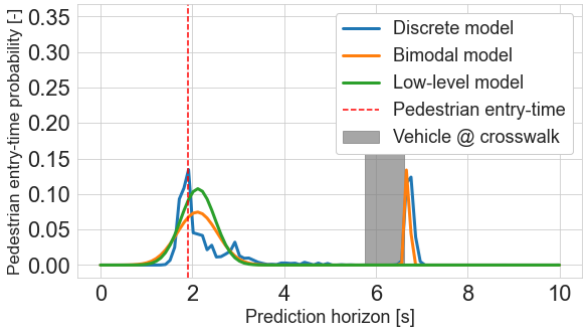


Figure 10. Illustration of model predictions for following datapoint from test set: $s_{p,0} = -3.3$ m, $v_{p,0} = 1.8$ m/s, $s_{v,0} = -55$ m, $v_{v,0} = 6.2$ m/s, $a_{v,R} = 1.8$ m/s$^2$, and $v_{v,R} = 10.4$ m/s.

First, the following probabilities which individual models put on particular simulation outcomes are compared: the pedestrian crossing prior to the vehicle ($P_{p,prior}$), and the pedestrian crossing after the vehicle ($P_{p,after}$). The hierarchical model directly provides this probability at the high level, while for the discrete and bimodal models this value is implicitly contained within predicted probability distributions and should be calculated. Since the vehicle is driven in an open-loop manner by applying predefined control parameters (see Fig. 2), it is possible to project the vehicle to the crosswalk and calculate its entry time to the crosswalk $t'_{v,en}$. For the discrete model, $P_{p,prior}$ is calculated as a sum of probabilities for all time steps $j$ on the horizon for which $0 < j\Delta T < t'_{v,en}$ holds:

$$P_{p,prior} = \sum_j \hat{p}_{en,j}\,. \tag{11a}$$

For the bimodal model, this probability is calculated by integrating the related PDF $p_{bim}$ defined in (7):

$$P_{p,prior} = \int_0^{t'_{v,en}} p_{bim}(t)dt\,. \tag{11b}$$

Note that the probability of pedestrian crossing after the vehicle can be then calculated as: $P_{p,after} = 1 - P_{p,prior}$.

Comparative analysis of different models in terms of $P_{p,prior}$ is illustrated in Fig. 11. These results show that $P_{p,prior}$ and $P_{p,after}$ distributions are very similar for all three prediction models, with most of $P_{p,prior}$ values being very close to the ideal value of 1 when the pedestrian actually crossed prior to the vehicle and very close to the ideal value of 0 when the pedestrian actually crossed after the vehicle. It should be noted that the high-level model has somewhat

higher number of probabilities $P_{p,prior}$ closer to the ideal value of one when compared to other models.

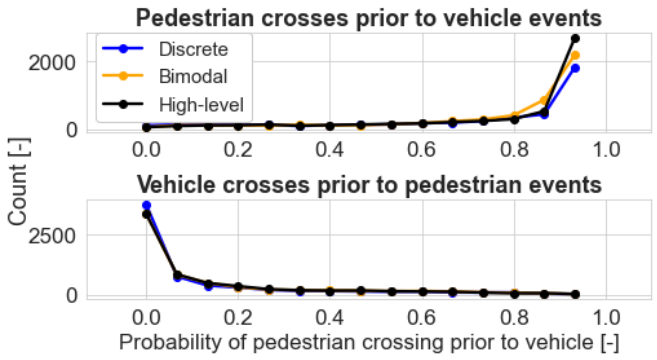


Figure 11. Distribution of predicted probabilities that pedestrian crosses prior to vehicle $P_{p,prior}$ for different prediction models and two simulation outcomes: (i) pedestrian crosses first, and (ii) vehicle crosses first.

Next, all the models are compared in terms of number of misclassifications of the right simulation outcome. For this purpose, a probability threshold is introduced $p_{thresh}$. If $P_{p,prior} < p_{thresh}$ in the case of pedestrian crossing first event, the corresponding classification is labeled as incorrect, while it is otherwise correct. The results are shown in Table 3 for the test dataset and three values of the threshold $p_{thresh}$ (0.01, 0.1, and 0.2). This metric could be especially important from the perspective of AV driving safety, because neglecting the possibility of pedestrian going prior to the vehicle could lead to safety critical situations. According to the results, the high-level model has the lowest number of misclassifications, which could be expected since it is directly trained to predict the probability of pedestrian crossing prior to vehicle.

Table 3. Number of misclassifications for which $P_{p,prior} < p_{thresh}$ holds among 5180 test data points with outcome of pedestrian going prior to ~~the~~ vehicle.

| **No. of misclassifications for ped. going prior to vehicle outcomes** | | | |
|---|---|---|---|
| **Model** | $p_{thresh} = 0.01$ | $p_{thresh} = 0.1$ | $p_{thresh} = 0.2$ |
| **Discrete** | 11 (0.21%) | 176 (3.39%) | 378 (7.29%) |
| **Bimodal** | 3 (0.05%) | 116 (2.23%) | 288 (5.55%) |
| **High-level** | 3 (0.05%) | 103 (1.98%) | 267 (5.15%) |

The probability threshold $p_{thresh}$ should be selected for the purpose of performing low-level model prediction. Namely, the low-level model prediction should be performed only when the high-level model predicts significant probability of pedestrian crossing prior to the vehicle (i.e., when the related probability $p$ is larger than the selected threshold, $p > p_{thresh}$). Setting larger threshold $p_{thresh}$ would lead to lower number of low-level model evaluations, while on the other hand potentially neglecting actual pedestrian crossing prior to vehicle events, and vice versa. Thus, an additional analysis is performed to detect satisfactory probability threshold candidates. The results given in Table 4 suggest that the satisfactory candidates would be 0.001 or 0.01, where the number of low-level model evaluations would be reduced to around 10% (i.e., 901 and 1710 among 11560 points), while none or negligible number (0.1%) of pedestrian crossing first outcomes would be neglected for these thresholds.

Table 4. High-level model prediction statistics for different probability threshold values.

| **Predictions for test dataset (11,560 datapoints in total)** | | | | |
|---|---|---|---|---|
| **Probability threshold ($p_{thresh}$)** | 0.0001 | 0.001 | 0.01 | 0.1 |
| **No. of predicted probabilities lower than $p_{thresh}$** | 363 | 901 | 1710 | 3946 |
| **No. of pedestrian crossing first outcomes** | 0 (0%) | 0 (0%) | 3 (0.1%) | 103 (2.6%) |

Another relevant aspect of prediction models is amount of probability which models put on the actual pedestrian entry time value on the prediction horizon (denoted also as actual entry time probability; see red line probabilities in Fig. 10). By definition, the probability of a single-point actual pedestrian entry time $t_{p,en}$ under any continuous probability distribution is zero. To overcome this issue, the continuous-time Gaussian probability distributions of bimodal and hierarchical low-level models are discretized to obtain related probability as:

$$p(t_{p,en}) = \int_{t_{p,en-\Delta T/2}}^{t_{p,en+\Delta T/2}} f(t)dt\,, \qquad (12)$$

where $f(t)$ denotes a PDF of considered continuous-time distribution and $\Delta T$ = 0.1 s is the discretization step of simulation models. The results shown in Fig. 12 point out that for both possible outcomes, the discrete model puts slightly higher probability values on the actual entry time. This can be explained by the fact that the discrete model by its structure has higher flexibility in the output than the bimodal and low-level models, as it predicts actual values of PDF rather than "lumped" parameters ($\mu$ and $\sigma$) of prescribed Gaussian distributions. For the outcomes when the vehicle crosses prior to pedestrian, the low-level model probabilities are not shown since this model is only trained on the subset of data where the pedestrian crosses prior to vehicle.

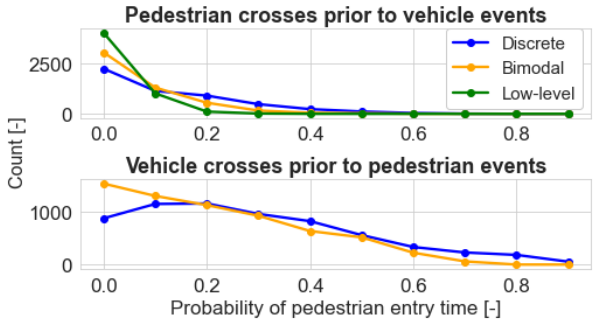


Figure 12. Distributions of probabilities that different models put on actual/ground truth entry time values from test dataset.

Finally, the actual pedestrian entry time is compared with the expected pedestrian entry time calculated from the probability distributions provided by the models. In the case of discrete model, the expected entry time is calculated in the following way: if the final outcome is pedestrian crossing prior to vehicle, the expected entry time is calculated from the first probability distribution mode (i.e., time steps < $t'_{v,en}$), while the second probability distribution mode is

used, otherwise (i.e., time steps $\geq t'_{v,en}$; see [23]). For the hierarchical low-level model, the pedestrian entry time expectation is readily available as the model output ($\mu$ value; see Fig. 7). For the bimodal model, it is taken from the mode whose expectation is closer to the actual pedestrian entry time.

The difference of actual and expected entry time values, denoted also as residuals, are given in Fig. 13, while the corresponding basic statistics is provided in Table 5. In the pedestrian crosses prior to vehicle outcome, all the three models have very similar, near-zero mean value of residuals and comparable standard deviations of residuals, being around 0.6 s. The low-level model on average slightly overestimates the pedestrian entry time which is reflected in negative mean residual value. In the case of vehicle crosses prior to pedestrian outcome, the discrete model has somewhat better results than the bimodal model in terms of lower standard deviation of residuals.

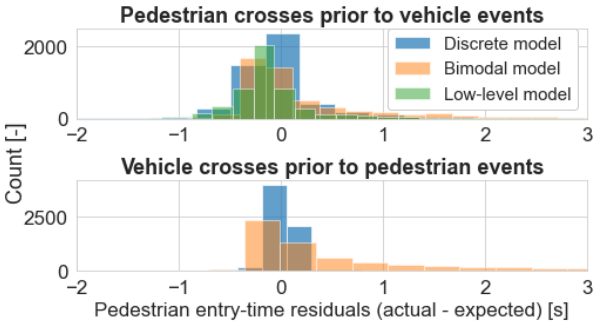


Figure 13. Distributions of pedestrian entry time residuals for two possible outcomes.

Table 5. Summary statistics of pedestrian entry time residuals, given for test dataset (percentages given in table denote related distributions' percentiles).

| **Pedestrian crosses prior to vehicle event** | | | | | |
|---|---|---|---|---|---|
| **Model** | Mean [s] | Std. [s] | 25% | 50% | 75% |
| **Discrete** | 0.00 | 0.66 | −0.22 | −0.08 | 0.04 |
| **Bimodal** | 0.01 | 0.52 | −0.18 | −0.06 | 0.05 |
| **Low-level** | −0.03 | 0.58 | −0.28 | −0.14 | 0.00 |
| **Vehicle crosses prior to pedestrian event** | | | | | |
| **Model** | Mean [s] | Std. [s] | 25% | 50% | 75% |
| **Discrete** | 0.02 | 0.18 | −0.01 | 0.03 | 0.07 |
| **Bimodal** | 0.00 | 1.14 | −0.04 | −0.01 | 0.02 |

## *Comparative analysis with respect to ground truth pedestrian entry time probability distributions*

For the purpose of additional model validation, ground truth entry time probability distributions are obtained for 500 different vehicle and pedestrian initial conditions, and 100 repetitive simulations are conducted for each of 500 initial conditions. A pre-analysis conducted over several different input points has shown that 100 repetitive simulations are enough for related entry time distribution to converge (see probability distributions for different number of repetitive simulations in Fig. 14 for a single initial condition). Now the probability distributions provided by the models could be directly compared against these empirical ground truth distributions, thus providing a basis for direct model validation.

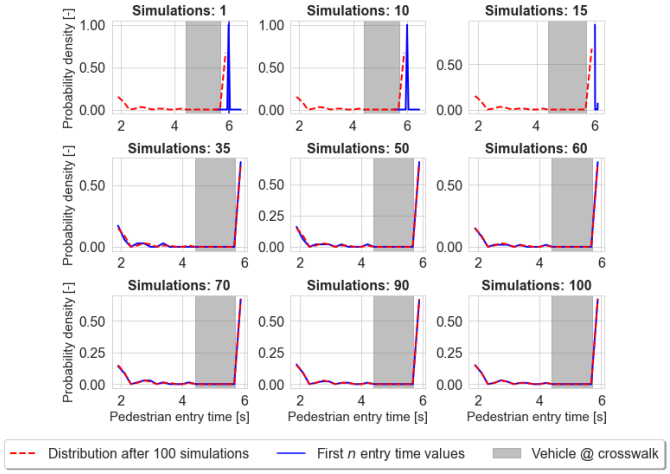


Figure 14. Illustration of convergence of empirical entry time distribution towards ground truth distribution. Initial conditions of this illustration: $s_{P,0}$ = −2.4 m, $v_{p,0}$ = 1.3 m/s, $s_{v,0}$ = −33.3 m, $v_{v,0}$ = 8.3 m/s, $a_{v,R}$ = −1.0 m/s$^2$, $v_{v,R}$ = 6.9 m/s.

For this purpose, Kullback-Leibler divergence (KL; also called relative entropy) and Jensen-Shannon divergence (JS) metrics are used (calculated by using Python Tensorflow module). For two discrete probability distributions $P$ and $Q$ defined on the same probability space $X$, the KL divergence is defined as [22]:

$$D_{KL}(P \,||\, Q) = \sum_{x \in X} P(x) \ln\left(\frac{P(x)}{Q(x)}\right). \tag{13}$$

It actually represents the expected value of logarithmic difference between $P$ and $Q$ distributions, where the expected value is taken with respect to the distribution $P$. It is not symmetrical, as $D_{KL}(P \,||\, Q) \neq D_{KL}(Q \,||\, P)$ holds in general case. The intuition behind KL divergence is that when the probability for a certain event under $P$ is large, but the probability for the same event under $Q$ is small, KL score is large. KL divergence can take values in the interval $[0, +\infty]$. The JS divergence is defined by using the KL divergence as [22]:

$$D_{JS}(P \,||\, Q) = \frac{1}{2} D_{KL}(P \,||\, M) + \frac{1}{2} D_{KL}(Q \,||\, M), \tag{14}$$

where $M = \frac{1}{2}P + \frac{1}{2}Q$. Unlike the KL divergence, the JS divergence is symmetrical, and its values fall in the range $[0, 1]$. For two discrete probability distributions, $D_{KL}(P \,||\, Q) = 0$ or $D_{JS}(P \,||\, Q) = 0$ suggests that two distributions are identical.

To validate the probability distributions of different models against the ground truth ones (one per each interaction) according to the KL and JS metrics, they should be first discretized if being continuous-time originally (those of bimodal and hierarchical low-level models). This is conducted by using (12) and by normalizing the obtained discrete-time probabilities to satisfy the condition on the sum of probabilities over the prediction horizon to be equal one. The resulting KL and JS distributions, shown in Fig. 15, indicate that the discrete model slightly outperforms other models for both divergence

measures, as the majority of its KL and JS values are close to the ideal value of zero. On the other hand, the hierarchical low-level model has somewhat worse divergence results since it is trained only to learn the outcome when pedestrian crosses prior to vehicle and has less flexibility in the output.

Apart from calculating the KL and JS metrics, correlations of the model distributions with ground truth ones are calculated and shown in Fig. 16 (by using Matlab function *corr*(.)). These results essentially confirm the results from Fig. 15, i.e., the discrete model slightly outperforms the bimodal model and to somewhat larger extent the low-level model, as more of its correlations with ground truth distribution are closer to the ideal value of 1. Again, these results could be explained by the highest flexibility of discrete model, i.e., the richer formulation of its output.

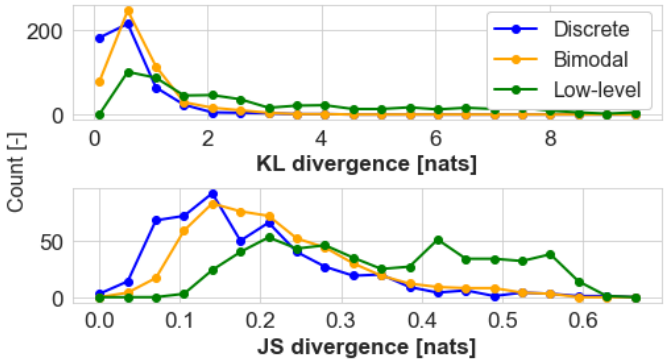


Figure 15. Distributions of KL and JS divergence metrics reflecting difference between model and ground truth distributions for 500 different initial conditions.

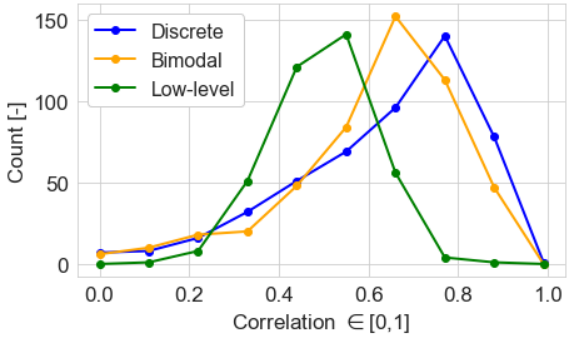


Figure 16. Distributions of correlation values for model and ground truth distributions for 500 different initial conditions.

Fig. 17 illustrates that the low-level model can successfully predict the first mode for different randomly selected initial conditions, although its KL and JS scores are relatively large (cf. Fig. 15). The second probability mode related to pedestrian surpassing the crosswalk after the vehicle is mostly dominant in these examples, and the low-level model is still successful to capture the entry time values that belong to the relevant first probability mode related to the pedestrian crosses first outcome. The only exception relates to the case where only pedestrian crossing after the vehicle outcomes has actually occurred (see ground truth distribution in the second row and the first column subfigure). However, it is shown here that the low-level model still succeeds to capture the actual pedestrian entry time although being trained only for pedestrian crossing first events, which points to good model extrapolation capabilities.

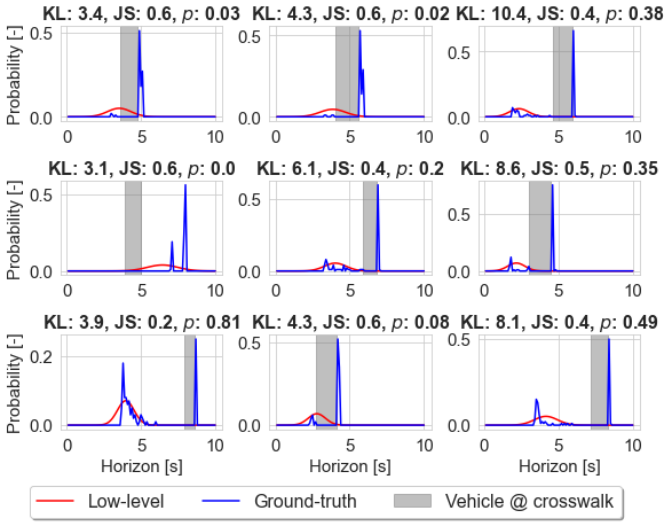


Figure 17. Low-level model predictions along with ground truth entry time probability distributions for nine randomly selected initial conditions (out of 500; note that high-level model output probability $p$ value is given in figure titles).

Furthermore, the share of pedestrian crossing prior to vehicle events is shown and analyzed in Fig. 18. Approximately 25% of all initial conditions have nearly balanced two probability modes related to two possible simulation outcomes, which are found to be more demanding for prediction. The ground truth ratios for each of 500 interactions from Fig. 18 are shown against the probabilities provided by the high-level model in Fig. 19. Well alignment of high-level model predictions with the actual ratios (i.e., points located around ideal unit) point out to the good model accuracy.

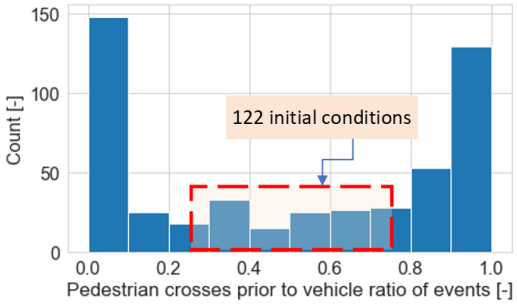


Figure 18. Distribution of pedestrian crosses 1st events share among 100 simulations for each of 500 initial conditions (for 122 initial conditions shares of pedestrian going first are in range $[0.25, 0.75]$, revealing that nearly 25% have more or less balanced probability modes of two possible simulation outcomes).

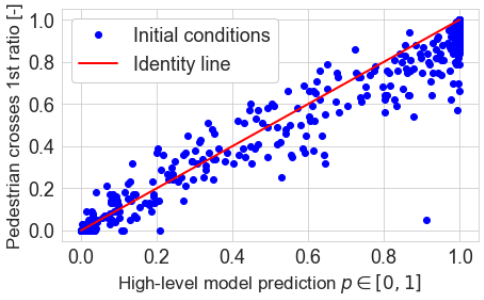

Figure 19. High-level model probability predictions vs. actual shares of pedestrian crossing first, obtained on additional dataset related to 500 initial conditions and 100 repetitive simulations for each initial condition.

Additionally, the average probability that each model puts on the actual pedestrian crossing first events among 100 simulations for each initial condition is analyzed. For some $j^{\text{th}}$ initial condition, the average probability is calculated as

$$p_{avg} = \frac{1}{m_j}\sum_{i=1}^{m_j} p_{model}(t_{p,en,j,i}), \quad (15)$$

where $p_{model}$ is the probability that $model \in$ {discrete, bimodal, low level} puts on the actual ground truth pedestrian entry time value $t_{p,en,j,i}$, where the subscript $i$ denotes $i^{\text{th}}$ simulation among 100 for which pedestrian crossed first, and $m_j$ is the number of outcomes for $j^{\text{th}}$ initial condition for which the pedestrian crossed first (note that $p_{model}$ is calculated by applying Eq. (12) for time-continuous distributions). According to Fig. 20, for initial conditions that result in high number of events where the pedestrian crosses prior to the vehicle, the bimodal and discrete models have similar $p_{avg}$ values. On the other hand, for initial conditions with relatively small number of these events (i.e., less than 40), the low-level model outperforms both bimodal and discrete model since it puts higher probabilities on average on these events. In other words, the low-level model is more successful in capturing rare events when pedestrian crosses prior to the vehicle than other models, which is at the expense of lower probabilities for more predictable outcomes (i.e., those with a high share of pedestrian going prior to the vehicle, e.g., for ≥80).

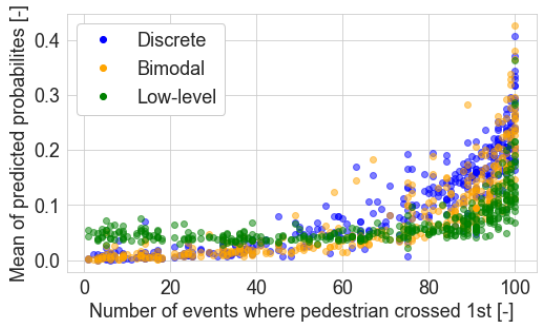


Figure 20. Analysis of average predicted probabilities of actual entry time values for different models and case of pedestrian crossing first, obtained for 500 additional conditions based on 100 simulation per each initial condition.

## Conclusion

This paper has proposed a novel hierarchical neural network (NN)-based model, which consists of high-level and low-level submodels, and is aimed for prediction of pedestrian entry time at unsignalized crosswalks while interacting with an autonomous vehicle. The models were verified on synthetic large-scale dataset obtained by simulations of a game theory-based pedestrian model interacting with the vehicle controlled in an open-loop manner. The model performance was analyzed in terms of model entry time residuals (i.e., difference between actual and model expected pedestrian entry times), the amount of probability that models put on actual outcomes (i.e., pedestrian crosses prior or after the vehicle), and the probability put on actual pedestrian entry time values. Additionally, the considered prediction models were validated against the ground truth probability distributions obtained by repetitive simulations for fixed initial conditions (500 initial conditions and 100 simulations per one initial condition).

The high-level submodel of the hierarchical model slightly outperforms other models in terms of predicting the ~~of~~ final interaction outcome. On the other hand, the discrete model somewhat outperforms other models with respect to the amount of probability that it puts on the actual pedestrian entry time values, due to its relatively high output layer flexibility. Significant practical advantage of the models that provide Gaussian distributions at the output (i.e., the single-level bimodal and the low-level submodel of the hierarchical model), is the direct interpretability of entry time uncertainty through the predicted standard deviation. The discrete model slightly outperforms other models also in terms of KL and JS divergence between model probability distributions and the ground truth distribution obtained through repetitive simulations for each initial condition. In addition, the high-level model predictions for additional performance evaluation set are close to the actual shares of pedestrian crossing prior to the vehicle, which justifies hierarchical modelling approach.

In general, the low-level model is not restricted to have just one distribution mode. It could be designed to have bimodal distribution or to have even higher flexibility. However, even with one distribution mode, the low-level model has very competitive results compared to much more flexible bimodal and discrete models. The models presented in this work could be used as a benchmark for more simpler models (e.g., logistic regression ones), with significantly less parameters, which would be preferable from the standpoint of model online adaptation. The future work will be directed towards analysis of potential application of NN models specialized for sequence modelling and prediction, such as recurrent NN models, which would take history of pedestrian (and vehicle) movement into account for more accurate predictions.

## References


1. J. C. Gerdes and S. M. Thornton, “Implementable Ethics for Autonomous Vehicles”, *Autonomous Driving*, (Berlin, Heidelberg: Springer Berlin Heidelberg, 2016), 87–102, doi: 10.1007/978-3-662-45854-9_5.
2. I. Wolf, “The Interaction Between Humans and Autonomous Agents”, *Autonomous Driving*, (Berlin, Heidelberg: Springer Berlin Heidelberg, 2016), 103–124, doi: 10.1007/978-3-662-48847-8_6.
3. A. Rasouli and J. K. Tsotsos, “Autonomous Vehicles That Interact With Pedestrians: A Survey of Theory and Practice,” *IEEE Trans. Intell. Transp. Syst.*, 21(3): 900–918, 2020, doi:10.1109/TITS.2019.2901817.
4. B. Volz, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, “Inferring Pedestrian Motions at Urban Crosswalks,” *IEEE Trans. Intell. Transp. Syst.*, 20(2):544–555, 2019, doi:10.1109/TITS.2018.2827956.
5. J. Eilbrecht, M. Bieshaar, S. Zernetsch, K. Doll, B. Sick, and O. Stursberg, “Model-predictive planning for autonomous vehicles anticipating intentions of vulnerable road users by artificial neural networks,” presented at 2017 IEEE Symposium Series on Computational Intelligence (SSCI), USA, Nov. 27 - Dec. 1, 2017, pp. 1–8, doi: 10.1109/SSCI.2017.8285249.
6. R. Elvik, “A review of game-theoretic models of road user behaviour” *Accid. Anal. Prev.*, 62: 388–396, 2014, doi:

10.1016/j.aap.2013.06.016.

7. B. Volz, K. Behrendt, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, “A data-driven approach for pedestrian intention estimation” presented at 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Nov. 1-4, 2016, pp. 2607–2612, doi: 10.1109/ITSC.2016.7795975.

8. A. Rasouli, I. Kotseruba, and J. K. Tsotsos, “Understanding Pedestrian Behavior in Complex Traffic Scenes” *IEEE Trans. Intell. Veh.*, 3(1):61–70, 2018, doi: 10.1109/TIV.2017.2788193.

9. A. Rasouli, I. Kotseruba, T. Kunic, and J. Tsotsos, “PIE: A Large-Scale Dataset and Models for Pedestrian Intention Estimation and Trajectory Prediction” presented at 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, Oct. 27 - Nov. 2, 2019, pp. 6261–6270, doi: 10.1109/ICCV.2019.00636.

10. S. Shalev-Shwartz, S. Shammah, and A. Shashua, “On a Formal Model of Safe and Scalable Self-driving Cars”, arXiv:1808.06887v5 [cs], Aug. 2017, doi.org/10.48550/arXiv.1708.06374.

11. F. Schneemann and P. Heinemann, “Context-based detection of pedestrian crossing intention for autonomous driving in urban environments,” presented at 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, Oct. 9-14, 2016, pp. 2243–2248, doi: 10.1109/IROS.2016.7759351.

12. E. Rehder and H. Kloeden, “Goal-Directed Pedestrian Prediction,” presented at 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile, Dec. 7-13, 2015, pp. 139–147, doi: 10.1109/ICCVW.2015.28.

13. N. Radwan, W. Burgard, and A. Valada, “Multimodal Interaction-aware Motion Prediction for Autonomous Street Crossing”, *The International Journal of Robotics Research*, 39(13):1567-1598, 2020, doi:10.1177/0278364920961809.

14. S. Zamboni, Z. T. Kefato, S. Girdzijauskas, N. Christoffer, and L. D. Col, “Pedestrian Trajectory Prediction with Convolutional Neural Networks”, *Pattern Recognition*, 121, 2022, 108252, ISSN 0031-3203.

15. T. Su, Y. Meng, and Y. Xu, “Pedestrian Trajectory Prediction via Spatial Interaction Transformer Network”, presented at 2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops), July 11-17, 2021, pp. 154-159, doi:10.1109/IVWorkshops54471.2021.9669249.

16. P. Chen, C. Wu, and S. Zhu, “Interaction between vehicles and pedestrians at uncontrolled mid-block crosswalks” *Saf. Sci.*, 82:68–76, 2016, doi:10.1016/j.ssci.2015.09.016.

17. B. Škugor, J. Topić, J. Deur, , V. Ivanović, and H. E. Tseng, “Analysis of a Game Theory-Based Model of Vehicle-Pedestrian Interaction at Uncontrolled Crosswalks” 4th International Conference on Smart Systems and Technologies, Osijek, Croatia, 14-16 October, pp. 73-81, 2020.

18. L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, “Normalization Techniques in Training DNNs: Methodology, Analysis and Application” Sep. 2020, https://doi.org/10.48550/arXiv.2009.12836.

19. D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization”, presented at 3rd International Conference for Learning Representations, San Diego, USA, May 7-9, 2015, doi.org/10.48550/arXiv.1412.6980

20. Keras, Computer software, Chollet, F., & others, 2015.

21. TensorFlow: A System for Large-Scale Machine Learning, Computer software, M. Abadi, P. Barham, J. Chen, et.al., 2015.

22. Q. Wang, Y. Ma, K. Zhao, and Y. Tian, “A Comprehensive Survey of Loss Functions in Machine Learning” *Ann. Data Sci.*, 9(2):187–212, 2022, doi:10.1007/s40745-020-00253-5.

23. J. Topic, B. Skugor, J. Deur, V. Ivanovic, H.E. Tseng, " Neural Network-based Prediction of Pedestrian Crossing Behavior at Unsignalized Crosswalks", presented at 5th International Conference on Smart Systems and Technologies, Osijek, Croatia, Oct. 19-21, 2022.

## Acknowledgments

It is gratefully acknowledged that this work has been supported by Ford Motor Company. The work of second and third author has also been supported through scientific-technological Croatian-Hungarian cooperation project “Design of Automated Driving Systems Based on Estimation of Wheel-road Contact Features for Handling Emergency Situations (AVEST)”.