A Deep Learning Architecture for Egocentric Time-to-Saccade Prediction using Weibull Mixture-Models and Historic Priors

TIM ROLFF, Universität Hamburg, Germany
SUSANNE SCHMIDT, Universität Hamburg, Germany
FRANK STEINICKE, Universität Hamburg, Germany
SIMONE FRINTROP, Universität Hamburg, Germany

Real-time detection of saccades is of major interest for many applications in human-computer interaction and mixed reality. However, due to relatively low update rates and high latencies of current commercially available eye trackers, gaze events are typically detected after they occur with some delay. This limits interaction scenarios such as intent-based gaze interaction, redirected walking, or gaze forecasting.

In this paper, we propose a deep learning framework for time-to-event prediction of saccades. In contrast to previous approaches, we utilize past multimodal data captured from head-mounted displays. We combine the well-established transformer architecture with a Weibull Mixture Model. This also allows estimating the uncertainty of the prediction. Additionally, we propose a sampling strategy that differs from conventional approaches to better account for the temporal properties of gaze sequences. We demonstrate that our model achieves state-of-the-art performance by evaluating it on three datasets and performing multiple ablation studies.

CCS Concepts: • **Human-centered computing → HCI theory, concepts and models**; Virtual reality.

Additional Key Words and Phrases: time-to-saccade, time-to-event prediction, gaze event classification, virtual reality, deep learning

ACM Reference Format:
Tim Rolff, Susanne Schmidt, Frank Steinicke, and Simone Frintrop. 2023. A Deep Learning Architecture for Egocentric Time-to-Saccade Prediction using Weibull Mixture-Models and Historic Priors . In *2023 Symposium on Eye Tracking Research and Applications (ETRA '23), May 30-June 2, 2023, Tubingen, Germany*. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3588015.3588408

1 INTRODUCTION

When exploring a scene, we as humans constantly move our eyes to acquire information about the surrounding environment. One key reason for these constant eye movements is the limited area of sharp vision that our eyes possess, known as the fovea [Østerberg 1935]. The fovea is responsible for high visual acuity, but only covers a small portion (the central 2 degree) of the visual field [Davson 1990; De Valois and De Valois 1980], requiring us to constantly move our eyes to bring different parts of the scene into focus. These eye movements, including saccades, fixations, and smooth pursuits, are a crucial aspect of perception in both real-world and virtual environments (VEs), while their analyses provide valuable insights into different cognitive processes and the user's perception of the current scene. As each of these types has its inherent properties, it is possible to classify them into their respective category. This classification has been well-researched, and multiple algorithms have been proposed to identify saccades and fixations [Agtzidis et al. 2016; Andersson et al. 2017; Dar et al. 2021; Komogortsev and Karpov 2013; Salvucci and Goldberg 2000; Startsev et al. 2019; Zemblys et al. 2019]. However, a major restriction of these classification methods is that they

Manuscript submitted to ACM

can only label a gaze sample after it has been captured, inducing significant latency before the classification can be used for advanced human-computer interaction (HCI) techniques. Furthermore, they often have to rely on only a few samples to correctly classify eye movements, as these are of short duration and update rates of wearable eye trackers are comparably low with sampling rates around 100 Hz [Stein et al. 2021]. This particularly applies for saccades, which are fast eye movements with a duration of 30-80 ms [Holmqvist et al. 2011] and a peak velocity of up to $900°/s$ [Bahill et al. 1975]. Another limitation is the high latency of wearable eye trackers in head-mounted displays (HMDs), which can take up to 81 ms [Stein et al. 2021], further adding a delay prior a potential classification. Overall, the described aspects challenge the real-time utilization of gaze events and may result in an event being classified too lately for being used in real-time HCI applications. For instance, previous work has suggested inducing changes to the scene during a saccade or blink, when visual input is suppressed [Langbehn et al. 2018; Sun et al. 2018]. However, these techniques require fast and reliable detection of such fast eye movements and are rarely applicable due to the aforementioned delays. Previous approaches aim to mitigate this problem, either by relying on long saccade durations [Sun et al. 2018] or requiring intentional blinking [Langbehn et al. 2018], deviating from the pure observation of the entirety of natural eye movements.

A different approach for gaze classification was recently introduced by Rolff et al. [2022]. The authors redefined the problem of gaze classification as a *time-to-event prediction* problem of gaze events. They particularly focus on predicting saccades (i.e., time-to-saccade), but the flexibility of the proposed approach also allows the application to other gaze events, such as fixations or blinks. In contrast to the previously mentioned classical classification approaches, the remaining time until the next saccade is estimated for each input sample of an eye tracker. The information about the duration until a specific event occurs is beneficial, as it is often not essential if the class for each time step is known, but rather when its class will change [Langbehn et al. 2018; Sun et al. 2018]. Furthermore, knowledge about when the next saccade event will occur can be utilized for a wide variety of HCI applications, including scan path prediction [Yang et al. 2017] or to adaptively increase an eye tracker's sampling rate shortly before event occurrence [Leube et al. 2017]. Time-to-saccade prediction has enormous potential for virtual reality (VR) applications, for instance, to estimate gaze shifts in gaze forecasting [Hu et al. 2021a, 2020], blink or saccade detection for redirected walking [Langbehn et al. 2018; Sun et al. 2018], gaze contingent rendering [Arabadzhiyska et al. 2017], and intent-based gaze interaction [David-John et al. 2021]. Knowing the time to the next saccade could improve the above applications, for example, by forcing a gaze shift during the saccade in gaze forecasting or by rotating the VE from the exact beginning of the saccade in case of redirected walking.

In this paper, we propose a deep learning architecture for time-to-saccade prediction. Our framework utilizes historic data to predict the probability of an event through a probabilistic recurrent Weibull mixture model [Nagpal et al. 2021; Weibull 1951]. The predicted probability enables to estimate the time-to-saccade while simultaneously expressing the uncertainty of the output. As fixation durations are highly variational [Nuthmann 2017], we propose the utilization of task features with other multimodal data samples of the HMDs or wearable eye trackers, such as last gaze positions and head accelerations. We also propose a sampling strategy for training and evaluation that allows to take temporal behavior of gaze sequences into account, when compared to previous work of Rolff et al. [2022]. To demonstrate the benefits of our approach, we will evaluate our proposed architecture against the current state-of-the-art algorithms and datasets, and perform ablation studies on our architecture regarding model parameters. For this validation, we use the DGaze [Hu et al. 2020], FixationNet [Hu et al. 2021a], and Ego4D [Grauman et al. 2022] datasets.

To summarize, our work combines the following contributions:

- We propose a multimodal deep learning architecture that outperforms the previous state-of-the-art approach for time-to-event prediction of saccades by utilizing historic multimodal data, such as gaze and task, to generate a prior distribution, which we utilize to modify a Weibull mixture model for the time-to-event.
- We propose a novel prediction strategy utilizing information from classical eye-movement classifiers.

2 RELATED WORK

2.1 Time-to-Event Prediction

Time-to-event prediction has been extensively studied, with multiple proposed approaches for non-recurrent predictions. As a result, multiple frameworks have been presented over the years, such as proportional hazards models like the Cox model [Cox 1972], non-parametric models like the Kaplan–Meier estimator [Kaplan and Meier 1958], survival forests [Ishwaran et al. 2008; Wright et al. 2017], parametric models using probability distributions [Ranganath et al. 2016], survival support vector machines [Pölsterl et al. 2015], or, more recently, deep learning-based approaches [Katzman et al. 2018; Kvamme and Borgan 2019; Lee et al. 2018]. In comparison, recurrent time-to-event prediction is still a recent research direction, with only a few proposed models so far. An early approach for recurrent time-to-event prediction using a parametric approach was suggested by Martinsson [2016], who proposes a recurrent neural network using the Weibull distribution for churn prediction, which was later extended by [Bennis et al. 2021] and [Nagpal et al. 2021]. In contrast to previously mentioned approaches, the method by Martinsson reevaluates the time-to-event of the same subject using the provided temporal data for each input sample. Similar approaches were later proposed by [Yang et al. 2017] and [Avati et al. 2020], who utilized a log-normal distribution as parametric model. Later, Soleimani et al. [2017] established the use of Gaussian processes for time-to-event prediction. More recently, Neumann et al. [2019] provided a method for future event prediction in the image domain using a Gaussian Mixture Model heatmap. Another notable method was recently proposed by Ren et al. [2019] and Hu et al. [2021b], who use a non-parametric approach by directly predicting the hazard function.

2.2 Time-to-Saccade Prediction & Fixation Duration Analysis

The concept of time-to-saccade prediction was introduced fairly recently, with the current state-of-the-art approach only utilizing classical machine learning algorithms [Rolff et al. 2022]. The authors found that a linear regressor with Nyström kernel approximation [Williams and Seeger 2001] performed the best on all evaluated datasets when trained through stochastic gradient descent (SGD) [Robbins and Monro 1951]. They estimate a good feature set by generating a total of 104 features calculated from gaze and inertial measurement unit (IMU) data. Furthermore, these features were ranked in their usefulness for the final prediction. To evaluate their approach, Rolff et al. utilized the mean absolute error (*mae*) on a set of randomly sampled time-to-event values. While the predictive model outperforms the baseline of the average duration length, the authors argue that the predictive error of the analyzed models may be still too high to be usable for VR or real-world applications.

Even though the problem statement of time-to-saccade prediction is different from fixation duration analysis, they closely relate to one another. The difference, however, is that most fixation duration studies analyze fixation durations on population data (see [Dorr et al. 2010; Kowler 2011; Nuthmann 2017; Nuthmann et al. 2010; Salthouse and Ellis 1980; Walshe and Nuthmann 2021]) rather than predicting singular gaze events of individuals, which is highly desired

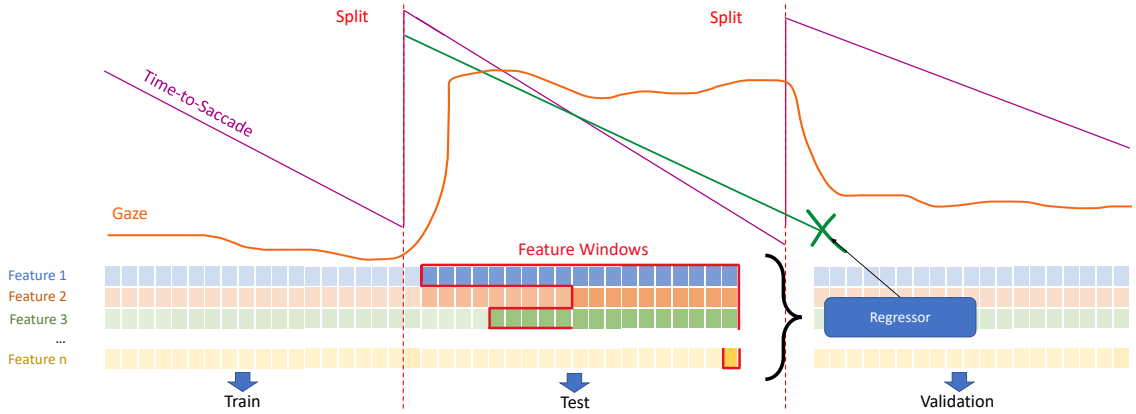

Fig. 1. Illustration of the sampling strategy: we split the features into multiple sequences at the occurrence of a saccade. Each sequence contains multiple samples and is placed into the respective dataset. A regression model (green line) is then used to predict the recurrent time-to-event of the sequence.

for the time-to-saccade prediction task. Nonetheless, these studies analyze the impact of specific data modalities on the overall fixation duration. For example, Nuthmann [2017] has shown that fixation durations are highly variable and depend on multiple factors like oculomotor movements and image stimuli, as well as other constraints, such as saccade suppression, amplitude of the next saccade, change in saccade direction, and viewing time [Dorr et al. 2010; Kowler 2011; Nuthmann 2017; Salthouse and Ellis 1980]. Some of the listed works already proposed statistical models for fixation durations based on population data [Nuthmann 2017; Nuthmann et al. 2010; Walshe and Nuthmann 2021].

3 METHOD

In this section, we give an overview of our methodology. For the time-to-saccade prediction, we utilize historical gaze, IMU data, and task information through our proposed deep learning framework. The time-to-saccade redefinition is especially well-suited for deep learning, as it transforms a sparse classification problem of event changes into a dense regression problem, thus enabling the use of advanced concepts from recurrent time-to-event prediction. Additionally, we propose a different sampling technique that uses the information of classical gaze classification approaches. First, we will describe our sampling strategy, followed by a brief explanation of time-to-event prediction and Weibull mixture models. Lastly, we will detail our architecture and describe the loss function.

3.1 Sampling

Eye trackers often report gaze points along with additional information on gaze events like blinks, fixations, or saccades. With this information, we can start a new prediction directly after the end of the last event instead of predicting the time-to-saccade at an arbitrary location in the gaze stream. This avoids that the network has to learn to reset the time-to-saccade at the end of an event while simultaneously providing helpful properties for evaluation. Hence, we propose a sampling strategy that groups samples of the same time-to-saccade sequence into the same train, test, or validation dataset, as shown in Fig. 1. This is accomplished by splitting the gaze signal at the occurrence of an event, rather than selecting samples at random like Rolff et al. [2022], resulting in sequences comprised of multiple individual samples. This method also offers the benefit of strictly monotonically decreasing time-to-saccade values

within each sequence, with the rate dependent on the eye tracker's frequency. Using the update frequency f_i at step i, time-to-saccade values (tts) can be calculated through the equation $tts_{i+1} = tts_i - f_i^{-1}$. The first time-to-saccade value is equal to the total duration of the sequence, with subsequent values decreasing until the event. In contrast to random sampling, this provides the ability to take the temporal property of the gaze data into account, as otherwise the samples of the same time-to-saccade sequence might have been selected for different datasets. As a result, it would be impossible to evaluate the temporal behavior without the predictor having seen part of the data.

3.2 Survival Analysis & Weibull Mixture-Model

Before going into the details of our framework, we briefly explain the concepts of survival analysis and Weibull mixture models. First, we assume that gaze data is stateful, independently censored, and can be modeled as a time varying covariate. Statefulness implies that each step of the prediction is characterized by the preceding steps [Nagpal et al. 2021], whereas the assumption of independently censored data is the common assumption of random non-informative censoring in static survival analysis [Schober and Vetter 2018]. We argue that all assumptions are met for time-to-saccade prediction, as the remaining duration t of the to be predicted sequences is dependent on previous covariates, and close to none of the data sequences are censored, as only the time-to-event of the last sequence in a capture is unknown. As we follow the reinterpretation of gaze classification as a time-to-event problem, we can use the log-likelihood function $\mathcal{L}$ of typical survival models [Moore 2016], defined as the sum of log-likelihoods for uncensored sequences ($\mathcal{UC}$), where the exact time-to-saccade t_i^j is known, and right-censored sequences ($\mathcal{RC}$), where the time-to-saccade is known to be greater than t_i^j:

$$\mathcal{L}(\theta) = \sum_{j \in \mathcal{UC}} \sum_{i=1}^{n_j} \log P(T = t_i^j | \theta) + \sum_{j \in \mathcal{RC}} \sum_{i=1}^{n_j} \log P(T > t_i^j | \theta). \tag{1}$$

Here, n_j denotes the length of the sequence j. Due to the observation that there are less than 0.66% censored samples in our dataset, we simplify the log-likelihood by assuming that all samples are uncensored. Note here that $P(T = t_i^j | \theta)$ is not the probability of a saccade occurring at time t_i^j but its likelihood. Further, the likelihood function of a gaze sequence j is dependent on the set of n_j previously made recurrent observations $X^j = \{(x_i^j, t_i^j) | 1 \leq i \leq n_j\}$, with x_i^j being the corresponding observation at time step i, and t_i^j being the observed time-to-saccade. The simplification together with involvement of the recurrent observations lead to the following log-likelihood for all steps within a sequence j:

$$\mathcal{L}(\theta | X^j) = \sum_{i=1}^{n_j} \log P(T = t_i^j | x_i^j, \theta). \tag{2}$$

For a continuous probability density function (PDF) f, the likelihood $P(T = t)$ of a saccade happening at time t is equivalent to the value $f(t)$. When using a Weibull distribution, defined through shape β and scale η, as the PDF, the likelihood can be calculated through:

$$P(T = t_i^j | x_i^j, \theta) = f(t_i^j | x_i^j, \theta) = \frac{\beta_i(x_i^j, \theta)}{\eta_i(x_i^j, \theta)} \left(\frac{t_i^j}{\eta_i(x_i^j, \theta)} \right)^{\beta_i(x_i^j, \theta) - 1} \cdot e^{-\left(\frac{t_i^j}{\eta_i(x_i^j, \theta)} \right)^{\beta_i(x_i^j, \theta)}} \tag{3}$$

Here, the parameters β and η are recurrently estimated for each step i through a neural network θ. For now, we assumed

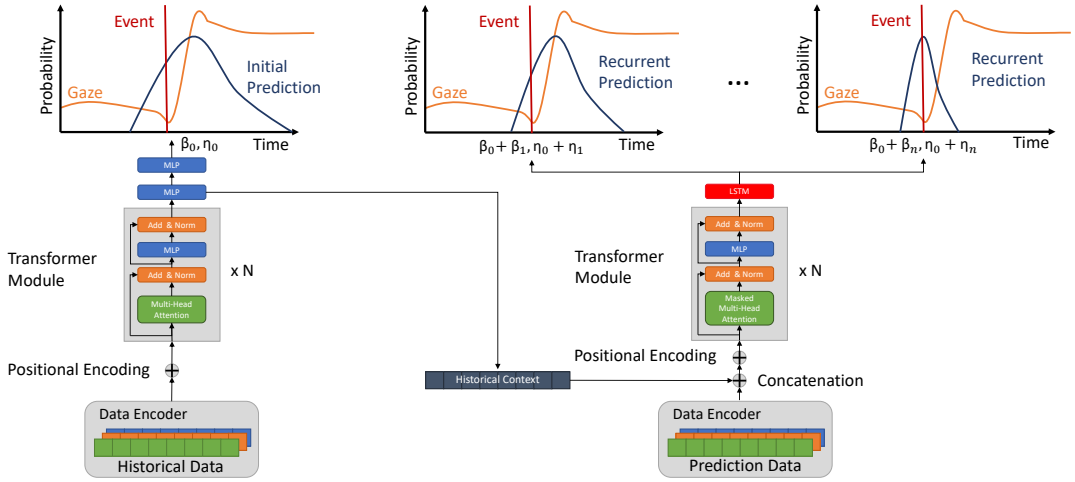

Fig. 2. Illustration of our proposed architecture for time-to-saccade prediction. We use two modules for the prediction of saccades, one utilizing historical data (left) and the other one using the historical context information along with newly sampled data for a recurrent time-to-event prediction (right). We describe our architecture in more detail in Sec. 3.3.

just a singular Weibull distribution; however, the probability distribution might be dependent on multiple factors shortening or lengthening the time-to-saccade. Therefore, we propose the utilization of a mixture model as the PDF $\hat{f}$ for time-to-saccade prediction:

$$\hat{P}(T = t_i^j | x_i^j, \theta) = \hat{f}(t_i^j | x_i^j, \theta) = \sum_c p(c | x_i^j, \theta) f(t_i^j | x_i^j, c, \theta), \tag{4}$$

with f denoting the probability distribution of the mixture coefficients, c being a latent variable, and $\sum_c p(c) = 1$. Substituting $\hat{P}$ for P into Equation 2 yields the following log-likelihood for a sequence j:

$$\mathcal{L}(\theta | X^j) = \sum_{i=1}^{n_j} \log \hat{f}(t_i^j | x_i^j, \theta). \tag{5}$$

3.3 Model

As gaze is captured over time, it inherits temporal properties which are captured as sequential data. Another observation is that changes in gaze events rarely occur when compared to the sample rate, with sometimes hundreds of samples being captured before the next event. Here, it is beneficial to redefine gaze event forecasting as a time-to-event instead of a classification problem when applying deep learning. This is due to the observation that directly classifying the gaze event would result in a sparse problem, whereas the reformulation results in a dense label for each sample containing the remaining time until the actual event will happen. Hence, we build upon the idea of recurrent estimation of the time-to-event as initially proposed by Martinsson [2016] and later extended by Nagpal et al. [2021].

Moreover, the sequential nature of the data allows us to utilize past gaze samples by employing the well-established transformer architecture [Vaswani et al. 2017], which we adapt with a temperature parameter for the self-attention layers as detailed by Lee et al. [2021]. We particularly choose a transformer-based architecture to be able to evaluate long sequence lengths but also because of its success on time-series data [Jiang et al. 2022; Tang and Matteson 2021; Zerveas et al. 2021; Zhou et al. 2021] and event forecasting [Yang et al. 2022; Zhang et al. 2020; Zuo et al. 2020]. Further, we split our architecture into two parts. First, the history encoder is used to predict an initial belief about the time-to-saccade t_0^j at the start of the sequence j using all recent historic samples x_{hist}^j of the data. This module predicts the shape β_0, location η_0, and mixture coefficients of our Weibull mixture model, estimating the initial probability distribution $\hat{f}(t_0^j|x_{\text{hist}}^j, \theta)$ of an event occurring at the time t_0^j given x_{hist}^j. As input, we use up to ten of the past seconds of multimodal data samples containing the last gaze positions, IMU data, and task. We base the inclusion of those features on the findings of Rolff et al. [2022], which measured the feature importance of different data modalities. Here, we opted to incorporate only the unprocessed data points into our analysis to avoid biasing the network with further extracted features and with the success of task data in gaze prediction [Hu et al. 2021a], we also choose to include it. As our second module, we use another transformer that we train through a generative pre-training (GPT) [Radford et al. 2018] approach, which masks the self-attention of the transformer model. Therefore, the prediction module has only access to previous samples, avoiding data leakage of the real time-to-saccade. As an additional input, we feed a generated context vector from the historic encoder along with the new samples for which we want to predict the time-to-saccade. As shown in Fig. 2, we add a final LSTM layer [Hochreiter and Schmidhuber 1997] to the time-to-saccade predictor. The inclusion of an LSTM in our transformer-based architecture follows the same idea as Lim et al. [2021], by using an LSTM layer for locality enhancement. We hypothesize that the locality helps in filtering the output to generate a more coherent time-to-saccade. We then use the output of the LSTM layer to update the initial shape and scale parameters of the probability distributions along with the mixture coefficients. To calculate the loss of our network, we use the log-likelihood function derived in Equation 5. Hence, we train our model by maximizing the probability of an event occurring at time t_i^j, which is equivalent to minimizing the negative log-likelihood. Furthermore, as we estimate two probability distributions, we must maximize the log-likelihood for the historic encoder θ_{hist}:

$$\mathcal{L}_{\text{hist}}(\theta_{\text{hist}}|\{x_{\text{hist}}^j, t_0^j\}) = \log \hat{f}(t_0^j|x_{\text{hist}}^j, \theta_{\text{hist}}), \tag{6}$$

and the log-likelihood of the predictor θ_{pred}:

$$\mathcal{L}_{\text{pred}}(\theta_{\text{pred}}|X^j) = \sum_{i=1}^{n_j} \log \hat{f}(t_i^j|x_i^j, x_{\text{hist}}^j, \theta_{\text{pred}}). \tag{7}$$

Here, the PDF for the time-to-event occurring at step t_i^j given the input x_i^j is defined through:

$$\hat{f}(t_i^j|x_i^j, x_{\text{hist}}^j, \theta_{\text{pred}}) = \frac{\phi(\beta_0)+\beta_i}{\phi(\eta_0)+\eta_i}\left(\frac{t_i^j}{\phi(\eta_0)+\eta_i}\right)^{\phi(\beta_0)+\beta_i-1} \cdot e^{-\left(\frac{t_i^j}{\phi(\eta_0)+\eta_i}\right)^{\phi(\beta_0)+\beta_i}}, \tag{8}$$

with ϕ being the stop gradient function masking the gradient on backpropagation. Note that we omit θ_{pred} and θ_{hist} as in Eq. 3 for the shape and location parameters for clarity. Finally, we define our weighted loss over all sequences as:

Table 1. Results on the listed DGaze, FixationNet and Ego4D datasets for the state of the art (SGD) by Rolff et al. [2022] and our proposed architecture. For comparison, we also evaluate a single-layer LSTM with the recurrent deep survival machine (RDSM) loss proposed by Nagpal et al. [2021]. We measure the *mean squared error* and *mean absolute error* with a lower error being preferred.

Metric \ Dataset	Avg. Time-to-Event			SGD [Rolff et al. 2022]			LSTM-RDSM [Nagpal et al. 2021]			Ours		
	DGaze	FixationNet	Ego4D	DGaze	FixationNet	Ego4D	DGaze	FixationNet	Ego4D	DGaze	FixationNet	Ego4D
mean squared error↓	0.1854s^2	0.2455s^2	0.0796s^2	0.1087s^2	0.1851s^2	0.0472s^2	**0.0699s^2**	0.1251s^2	**0.0305s^2**	0.0795s^2	**0.1184s^2**	0.0317s^2
mean absolute error↓	0.3039s	0.3391s	0.2000s	0.2369s	0.3079s	0.1559s	0.1976s	0.2289s	0.1276s	**0.1879s**	**0.2196s**	**0.1228s**

$$\mathcal{L} = -\sum_j n_j \cdot \mathcal{L}_{\text{hist}}(\theta_{\text{hist}} | \{x^j_{\text{hist}}, t^j_0\}) + \mathcal{L}_{\text{pred}}(\theta_{\text{pred}} | X^j). \tag{9}$$

As our model supports different data modalities, we encode discrete input using embeddings as Vaswani et al. [2017]. For the continuous values like gaze and IMU data, we first discretize the value, following the work of Reed et al. [2022]. Therefore, we first transform the continuous values using the μ-law and then bin the transformed samples into 256 different bins. We use 12 layers for both the time-to-saccade predictor and the history encoder modules with 12 heads and 16 units per head, a hidden dimension of 192, and the multi layer perception (MLP) dimension set to 384. We use 100 samples as input for the historic encoder and 32 mixture components for prediction. We would like to note that we did not perform hyperparameter optimization due to computational constraints. For training, we use an AdamW [Loshchilov and Hutter 2017] optimizer and train for 50 epochs. We schedule the learning rate as Vaswani et al. [2017] with a warm-up learning rate going from 1e−7 to the maximum learning rate of 3e−4 for 2500 steps, which we then reduce over 100,000 steps down to 1e−6. To reduce overfitting, we set the dropout to 0.5 and weight decay to 0.1. For comparison, we also train a single layer LSTM Recurrent Deep Survival Machine (LSTM-RDSM) [Nagpal et al. 2021] with a layer width of 192 neurons and 4 mixture components, which is equivalent to our proposed loss when setting the history length to zero and using a Weibull probability distribution.

4 EVALUATION

4.1 Datasets

For our evaluation, we use three different egocentric VR and real-world datasets: DGaze [Hu et al. 2020], FixationNet [Hu et al. 2021a], and Ego4D [Grauman et al. 2022]. DGaze and FixationNet provide video, gaze, and IMU data along with information on the nearest object captured from different VEs. DGaze includes over 20,000 gaze points per session from 43 participants in 86 sessions, captured using a 7invensun eye-tracker at 100 Hz. FixationNet includes 12,000 gaze points per session from 27 participants in 162 sessions, also captured at 100 Hz using a 7invensun eye-tracker. Ego4D captures egocentric videos of real scenes along with gaze data, covering a wide range of real-world scenarios. We only utilized data points with a specified eye-tracker, the Pupil Labs Invisible, resulting in a set of 27 sessions with 150,000 gaze points per session. As all datasets do not provide gaze labels, we use a similar labeling approach to Rolff et al. [2022]. In our case, we utilize the I-HMM algorithm, which has been shown to be more robust against noise [Salvucci and Goldberg 2000]. We provide details on the pre-processing and gaze classification in the supplementary materials.

4.2 Metrics

As suggested by Rolff et al. [2022], we do not use common metrics for quantitative evaluation of time-to-event data, such as the concordance index [Harrell Jr et al. 1996], cumulative dynamic AUC [Hung and Chiang 2010], or the brier

Table 2. Ablations on DGaze [Hu et al. 2020], FixationNet [Hu et al. 2021a], and Ego4D [Grauman et al. 2022] datasets with (*W/ LSTM*) and without (*W/o LSTM*) final LSTM layer using the parameters described in Sec. 3.3. The errors shown here are the *mean squared error* and *mean absolute error*, with a lower value being preferred in both cases.

Model	MSE			MAE		
	DGaze	FixationNet	Ego4D	DGaze	FixationNet	Ego4D
W/ LSTM	$0.0795s^2$	$\underline{\mathbf{0.1184}s^2}$	$\underline{\mathbf{0.0317}s^2}$	$\underline{\mathbf{0.1879}s}$	$\underline{\mathbf{0.2196}s}$	$\underline{\mathbf{0.1228}s}$
W/o LSTM	$\underline{\mathbf{0.0709}s^2}$	$0.1212s^2$	$0.0323s^2$	$0.2075s$	$0.2798s$	$0.1451s$

score [Graf et al. 1999]. Instead, we also compute the **mean absolute error** (*mae*) and the **mean squared error** (*mse*) between the time-to-event and the prediction and use our earlier introduced sampling strategy (cf. Sec. 3.1) to evaluate on sequences rather than individual random samples, contrasting previous work by Rolff et al. [2022].

4.3 Results

Table 1 shows our quantitative results, comparing our approach against the state of the art. Our proposed architecture reliably outperforms the state-of-the-art approach on all metrics by an average of 27.72%. For the *mse*, we were able to increase the performance by 31.91% and for the *mae* by 23.53%. On average, our model has a *mae* of 0.1768s and an average *mse* of $0.0765s^2$, whereas the state-of-the-art model has an average *mae* of $0.2336s^2$ and an average *mse* of 0.1137s.

Table 2, shows the difference in performance without a final LSTM layer. With the exception of DGaze, we found that the model with a final LSTM layer outperforms the model without an LSTM on all datasets and metrics. On average, the model without final LSTM layer achieves a *mse* of $0.0748s^2$ and a *mae* of 0.2108s. Additionally, we performed several ablation studies on our architecture with different lengths of utilized historic samples and mixture components, which can be found in the supplementary materials.

5 CONCLUSION AND DISCUSSION

In this paper, we proposed a transformer-based architecture for time-to-saccade prediction of gaze events. Our architecture builds upon the idea of utilizing historic data to predict an initial prior distribution. We employ another network that uses a generated context vector for the prediction of the time-to-saccade using a Weibull mixture model. We showed that our model outperforms the state-of-the-art method for time-to-saccade prediction by Rolff et al. [2022] on all evaluated datasets, achieving an average improvement of 27.72%. In comparison to the single-layer LSTM with RDSM loss that Nagpal et al. [2021] proposed for general time-to-event prediction, we only achieved a small improvement of the mean square error on the FixationNet dataset and of the mean absolute error across all datasets. Based on a qualitative inspection, we believe this is due to a more coherent output of the LSTM, whereas our network often predicts steps, thus, requiring more research to smooth out the generated time-to-event. However, despite the reported 27% improvement, we would still like to note that we believe that the predicted error is still too large for accurate utilization. As [Celikcan et al. 2020; Drewes et al. 2021] found differences in gaze behavior, we believe more research on differences in prediction on real-world and VR datasets is required. This also includes possible data modalities that might be available depending on the application. Furthermore, more research on optimal neural network architectures need to be made that fit the task at hand. Another future direction of research is the utilization of visual data for time-to-saccade prediction, for example, by using information about the environment or video data. Another addition might be a

user-specific calibration to further increase performance. Moreover, we would like to emphasize that more work needs to be done on the collection of big datasets for robust pretraining of the networks, especially with a focus on the annotation of events. Their current unavailability limits further research on deep learning methods for time-to-saccade prediction, with data quality depending on automated event annotation.

REFERENCES

Ioannis Agtzidis, Mikhail Startsev, and Michael Dorr. 2016. Smooth Pursuit Detection Based on Multiple Observers. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications* (Charleston, South Carolina) *(ETRA '16)*. Association for Computing Machinery, New York, NY, USA, 303–306.

Richard Andersson, Linnea Larsson, Kenneth Holmqvist, Martin Stridh, and Marcus Nyström. 2017. One algorithm to rule them all? An evaluation and discussion of ten eye movement event-detection algorithms. *Behavior research methods* 49, 2 (2017), 616–637.

Elena Arabadzhiyska, Okan Tarhan Tursun, Karol Myszkowski, Hans-Peter Seidel, and Piotr Didyk. 2017. Saccade landing position prediction for gaze-contingent rendering. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–12.

Anand Avati, Tony Duan, Sharon Zhou, Kenneth Jung, Nigam H. Shah, and Andrew Y. Ng. 2020. Countdown Regression: Sharp and Calibrated Survival Predictions. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference (Proceedings of Machine Learning Research, Vol. 115)*. PMLR, 145–155.

A Terry Bahill, Michael R Clark, and Lawrence Stark. 1975. The main sequence, a tool for studying human eye movements. *Mathematical biosciences* 24, 3-4 (1975), 191–204.

Achraf Bennis, Sandrine Mouysset, and Mathieu Serrurier. 2021. DPWTE: A Deep Learning Approach to Survival Analysis Using a Parsimonious Mixture of Weibull Distributions. (2021), 185–196 pages.

Ufuk Celikcan, Mehmet Bahadir Askin, Dilara Albayrak, and Tolga K Capin. 2020. Deep into visual saliency for immersive VR environments rendered in real-time. *Computers & Graphics* 88 (2020), 70–82.

David R Cox. 1972. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)* 34, 2 (1972), 187–202.

Asim H Dar, Adina S Wagner, and Michael Hanke. 2021. REMoDNaV: robust eye-movement classification for dynamic stimulation. *Behavior research methods* 53, 1 (2021), 399–414.

Brendan David-John, Candace Peacock, Ting Zhang, T. Scott Murdison, Hrvoje Benko, and Tanya R. Jonker. 2021. Towards Gaze-Based Prediction of the Intent to Interact in Virtual Reality. In *ACM Symposium on Eye Tracking Research and Applications* (Virtual Event, Germany) *(ETRA '21 Short Papers)*. Association for Computing Machinery, New York, NY, USA, Article 2, 7 pages. https://doi.org/10.1145/3448018.3458008

Hugh Davson. 1990. *Physiology of the Eye*. Bloomsbury Publishing.

Russell L De Valois and Karen K De Valois. 1980. Spatial vision. *Annual review of psychology* 31, 1 (1980), 309–341.

Michael Dorr, Thomas Martinetz, Karl R Gegenfurtner, and Erhardt Barth. 2010. Variability of eye movements when viewing dynamic natural scenes. *Journal of vision* 10, 10 (2010), 28–28.

Jan Drewes, Sascha Feder, and Wolfgang Einhäuser. 2021. Gaze during locomotion in virtual reality and the real world. *Frontiers in Neuroscience* 15 (2021), 656913.

Erika Graf, Claudia Schmoor, Willi Sauerbrei, and Martin Schumacher. 1999. Assessment and comparison of prognostic classification schemes for survival data. *Statistics in medicine* 18, 17-18 (1999), 2529–2545.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Abrham Gebreselasie, Cristina González, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolář, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Ziwei Zhao, Yunyi Zhu, Pablo Arbeláez, David Crandall, Dima Damen, Giovanni Maria Farinella, Christian Fuegen, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. 2022. Ego4D: Around the World in 3,000 Hours of Egocentric Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 18995–19012.

Frank E Harrell Jr, Kerry L Lee, and Daniel B Mark. 1996. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in medicine* 15, 4 (1996), 361–387.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. 2011. *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford, Oxford, England.

Shi Hu, Egill Fridgeirsson, Guido van Wingen, and Max Welling. 2021b. Transformer-based deep survival analysis. In *Survival Prediction-Algorithms, Challenges and Applications*. PMLR, 132–148.

Zhiming Hu, Andreas Bulling, Sheng Li, and Guoping Wang. 2021a. Fixationnet: Forecasting eye fixations in task-oriented virtual environments. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2681–2690.

Zhiming Hu, Sheng Li, Congyi Zhang, Kangrui Yi, Guoping Wang, and Dinesh Manocha. 2020. DGaze: CNN-based gaze prediction in dynamic scenes. *IEEE transactions on visualization and computer graphics* 26, 5 (2020), 1902–1911.

Hung Hung and Chin-Tsang Chiang. 2010. Estimation methods for time-dependent AUC models with survival data. *Canadian Journal of Statistics* 38, 1 (2010), 8–26.

Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. 2008. Random survival forests. *The annals of applied statistics* 2, 3 (2008), 841–860.

Hao Jiang, Lianguang Liu, and Cheng Lian. 2022. Multi-Modal Fusion Transformer for Multivariate Time Series Classification. In *2022 14th International Conference on Advanced Computational Intelligence (ICACI)*. IEEE, 284–288. https://doi.org/10.1109/ICACI55529.2022.9837525

Edward L Kaplan and Paul Meier. 1958. Nonparametric estimation from incomplete observations. *Journal of the American statistical association* 53, 282 (1958), 457–481.

Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. 2018. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC medical research methodology* 18, 1 (2018), 1–12.

Oleg V Komogortsev and Alex Karpov. 2013. Automated classification and scoring of smooth pursuit eye movements in the presence of fixations and saccades. *Behavior research methods* 45, 1 (2013), 203–215.

Eileen Kowler. 2011. Eye movements: The past 25 years. *Vision research* 51, 13 (2011), 1457–1483.

Håvard Kvamme and Ørnulf Borgan. 2019. Continuous and discrete-time survival prediction with neural networks.

Eike Langbehn, Frank Steinicke, Markus Lappe, Gregory F Welch, and Gerd Bruder. 2018. In the blink of an eye: leveraging blink-induced suppression for imperceptible position and orientation redirection in virtual reality. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–11.

Changhee Lee, William Zame, Jinsung Yoon, and Mihaela Van Der Schaar. 2018. Deephit: A deep learning approach to survival analysis with competing risks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32. AAAI, 8. Issue 1.

Seung Hoon Lee, Seunghyun Lee, and Byung Cheol Song. 2021. Vision Transformer for Small-Size Datasets.

Alexander Leube, Katharina Rifai, and Katharina Rifai. 2017. Sampling rate influences saccade detection in mobile eye tracking of a reading task. *Journal of Eye Movement Research* 10, 3 (2017), 11.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. 2021. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting* 37, 4 (2021), 1748–1764.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled Weight Decay Regularization.

Egil Martinsson. 2016. *Wtte-rnn: Weibull time to event recurrent neural network*. Master's thesis. Chalmers University of Technology & University of Gothenburg.

Dirk F Moore. 2016. *Applied Survival Analysis Using R*. Vol. 473. Springer.

Chirag Nagpal, Vincent Jeanselme, and Artur Dubrawski. 2021. Deep Parametric Time-to-Event Regression with Time-Varying Covariates. In *Survival Prediction-Algorithms, Challenges and Applications*. PMLR, 184–193.

Lukas Neumann, Andrew Zisserman, and Andrea Vedaldi. 2019. Future event prediction: If and when. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.

Antje Nuthmann. 2017. Fixation durations in scene viewing: Modeling the effects of local image features, oculomotor parameters, and task. *Psychonomic bulletin & review* 24, 2 (2017), 370–392.

Antje Nuthmann, Tim J Smith, Ralf Engbert, and John M Henderson. 2010. CRISP: a computational model of fixation durations in scene viewing. *Psychological review* 117, 2 (2010), 382.

G. Østerberg. 1935. *Topography of the Layer of Rods and Cones in the Human Retina*. A. Busck.

Sebastian Pölsterl, Nassir Navab, and Amin Katouzian. 2015. Fast Training of Support Vector Machines for Survival Analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Vol. 9285. Springer, 243–259.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Rajesh Ranganath, Adler Perotte, Noémie Elhadad, and David Blei. 2016. Deep survival analysis. In *Machine Learning for Healthcare Conference*. PMLR, 101–114.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. 2022. A Generalist Agent.

Kan Ren, Jiarui Qin, Lei Zheng, Zhengyu Yang, Weinan Zhang, Lin Qiu, and Yong Yu. 2019. Deep recurrent survival analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. AAAI, 4798–4805. Issue 1.

Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The annals of mathematical statistics* 22, 3 (1951), 400–407.

Tim Rolff, Frank Steinicke, and Simone Frintrop. 2022. When do Saccades begin? Prediction of Saccades as a Time-to-Event Problem. In *ACM Symposium on Eye Tracking Research & Applications*, Vol. 1. ACM, New York, USA.

Timothy A Salthouse and Cecil L Ellis. 1980. Determinants of eye-fixation duration. *The American journal of psychology* 93, 2 (1980), 207–234.

Dario D. Salvucci and Joseph H. Goldberg. 2000. Identifying Fixations and Saccades in Eye-Tracking Protocols. In *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications* (Palm Beach Gardens, Florida, USA) *(ETRA '00)*. Association for Computing Machinery, New York, NY, USA, 71–78. https://doi.org/10.1145/355017.355028

Patrick Schober and Thomas R Vetter. 2018. Survival analysis and interpretation of time-to-event data: the tortoise and the hare. *Anesthesia and analgesia* 127, 3 (2018), 792.

Hossein Soleimani, James Hensman, and Suchi Saria. 2017. Scalable joint models for reliable uncertainty-aware event prediction. *IEEE transactions on pattern analysis and machine intelligence* 40, 8 (2017), 1948–1963.

Mikhail Startsev, Ioannis Agtzidis, and Michael Dorr. 2019. 1D CNN with BLSTM for automated classification of fixations, saccades, and smooth pursuits. *Behavior Research Methods* 51, 2 (2019), 556–572.

Niklas Stein, Diederick C Niehorster, Tamara Watson, Frank Steinicke, Katharina Rifai, Siegfried Wahl, and Markus Lappe. 2021. A comparison of eye tracking latencies among several commercial head-mounted displays. *i-Perception* 12, 1 (2021), 2041669520983338.

Qi Sun, Anjul Patney, Li-Yi Wei, Omer Shapira, Jingwan Lu, Paul Asente, Suwen Zhu, Morgan McGuire, David Luebke, and Arie Kaufman. 2018. Towards virtual reality infinite walking: dynamic saccadic redirection. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–13.

Binh Tang and David S Matteson. 2021. Probabilistic transformer for time series analysis. *Advances in Neural Information Processing Systems* 34 (2021), 23592–23608.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017), 11.

R Calen Walshe and Antje Nuthmann. 2021. A Computational Dual-Process Model of Fixation-Duration Control in Natural Scene Viewing. *Computational Brain & Behavior* 4, 4 (2021), 463–484.

Waloddi Weibull. 1951. A Statistical Distribution Function of Wide Applicability. *Journal of applied mechanics* (1951), 5.

Christopher Williams and Matthias Seeger. 2001. Using the Nyström Method to Speed Up Kernel Machines. In *Advances in Neural Information Processing Systems*, T. Leen, T. Dietterich, and V. Tresp (Eds.), Vol. 13. MIT Press, Boston, USA. https://proceedings.neurips.cc/paper/2000/file/19de10adbaa1b2ee13f77f679fa1483a-Paper.pdf

Marvin N Wright, Theresa Dankowski, and Andreas Ziegler. 2017. Unbiased split variable selection for random survival forests using maximally selected rank statistics. *Statistics in medicine* 36, 8 (2017), 1272–1284.

Chenghao Yang, Hongyuan Mei, and Jason Eisner. 2022. Transformer Embeddings of Irregularly Spaced Events and Their Participants. In *International Conference on Learning Representations (ICLR)*. 21.

Yinchong Yang, Peter A Fasching, and Volker Tresp. 2017. Modeling progression free survival in breast cancer with tensorized recurrent neural networks and accelerated failure time models. In *Machine Learning for Healthcare Conference*. PMLR, 164–176.

Raimondas Zemblys, Diederick C Niehorster, and Kenneth Holmqvist. 2019. gazeNet: End-to-end eye-movement event detection with deep neural networks. *Behavior research methods* 51, 2 (2019), 840–864.

George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. 2021. A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2114–2124.

Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. 2020. Self-attentive Hawkes process. In *International Conference on Machine Learning*. PMLR, 11183–11193.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11106–11115. Issue 12.

Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. 2020. Transformer Hawkes Process. In *International Conference on Machine Learning*. PMLR, 11692–11702.