Survival text regression for time-to-event prediction in conversations

Christine De Kock
University of Cambridge
Department of Computer Science
cd700@cam.ac.uk

Andreas Vlachos
University of Cambridge
Department of Computer Science
av308@cam.ac.uk

Abstract

Time-to-event prediction tasks are common in conversation modelling, for applications such as predicting the length of a conversation or when a user will stop contributing to a platform. Despite the fact that it is natural to frame such predictions as regression tasks, recent work has modelled them as classification tasks, determining whether the time-to-event is greater than a pre-determined cut-off point. While this allows for the application of classification models which are well studied in NLP, it imposes a formulation that is contrived, as well as less informative. In this paper, we explore how to handle time-to-event forecasting in conversations as regression tasks. We focus on a family of regression techniques known as survival regression, which are commonly used in the context of healthcare and reliability engineering. We adapt these models to time-to-event prediction in conversations, using linguistic markers as features. On three datasets, we demonstrate that they outperform commonly considered text regression methods and comparable classification models.

1 Introduction

The task of predicting when an event will occur in a conversation frequently arises in NLP research. For instance, Backstrom et al. (2013) and Zhang et al. (2018b) predict when a conversation thread will terminate. Danescu-Niculescu-Mizil et al. (2013) define the task of forecasting when users will cease to interact on a social network based on their language use. Although these questions naturally lend themselves to regression, this presents some difficulties: datasets may be highly skewed towards shorter durations (Zhang et al., 2018b) and samples with a longer duration can contribute inordinately to error terms during training. Furthermore, classical regression models do not explicitly consider the effect of time as distinct from other features.

The abovementioned studies instead frame the time-to-event prediction as a classification task, predicting whether the current state will continue for a set number of additional timesteps. For instance, Backstrom et al. (2013) predict whether the number of responses in a thread will exceed 8, after seeing 5 utterances. This presents obvious limitations; such a setup would assign the same error for mistakenly classifying conversations of respectively 9 and 30 utterances as “short”. Additionally, its predictions are less informative: predicting that a conversation will be more than 8 utterances long is less telling than predicting whether it will be 9 or 30.

In this paper, we propose that *survival regression* is a more appropriate modelling framework for predicting when an event will occur in a conversation. Survival regression aims to predict the probability of an event of interest at different points in time, taking into account features of a subject as seen up to the prediction time. We apply survival models to two tasks: predicting conversation length, and predicting when conversations will get derail into personal attack. We report results for the conversation length prediction task on the datasets from Danescu-Niculescu-Mizil et al. (2012) and De Kock and Vlachos (2021), and evaluate the personal attack prediction task on the dataset of Zhang et al. (2018a). Our results illustrate that linear survival models outperform their linear regression counterparts, with an improvement in MAE of 1.22 utterances on the dataset of De Kock and Vlachos (2021). Further performance gains are made using neural network-based survival models. An analysis of the coefficients of our linear models indicates that survival models infer similar relationships as previous work on conversation length prediction, but that their predictions are more accurate than conventional regression and classification models due to their explicit accounting for the effect of time. On the personal attack prediction task,

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1219–1229
August 1–6, 2021. ©2021 Association for Computational Linguistics

the best survival model provides a 13% increase in ranking accuracy over linear regression models.

The remainder of this paper is structured as follows. In Section 2 we provide a description of key survival analysis concepts. In Section 3, we describe how we apply these concepts to conversations. Results are reported in Section 4.

2 Survival regression

Survival analysis is concerned with modelling time-to-event prediction, which often represents transitions between states throughout a subject's lifetime. In the general case, exactly one event of interest occurs per lifetime, after which the subject is permanently in the alternate state, often referred to as "death" in literature. In this section, we review some key concepts of survival analysis that are relevant to our work, however, we refer the interested reader to the exposition by Rodriquez (2007).

2.1 Definitions

Let T be a non-negative random variable representing the waiting time until the occurrence of an event. Given the cumulative distribution function $F(t)$ of the event time T, the *survival function* is defined as the probability of surviving beyond a certain point in time t:

$$\begin{aligned} S(t) &= P(T > t) \\ &= 1 - F(t). \end{aligned} \tag{1}$$

Per illustration, we consider the task of predicting conversation length using the dataset of disagreements of De Kock and Vlachos (2021). The event of interest is the end of a conversation, with time measured in utterances. We can estimate the survival function using Kaplan-Meier estimation (Jager et al., 2008) as follows:

$$S(t) = \prod_{t_i < t} \frac{R_i - d_i}{R_i}, \tag{2}$$

where d_i is the number of candidates who experience the event at time t_i, and R_i represents the so-called *risk set*, or candidates at risk of experiencing the event just prior to t_i. In Figure 1, the base function is the estimated survival probabilities over time for the full population. Only conversations of more than 5 utterances are considered; hence the survival probability is 1 for all curves up until $t = 5$. If we create subsets of the population by conditioning on the response time, the subset with a longer response time has a steeper decline, indicating that conversations where participants take longer to respond are more likely to end earlier. In survival regression, the aim is to learn survival regression functions based on such features, while the current time is modelled separately from them, unlike in standard regression models.

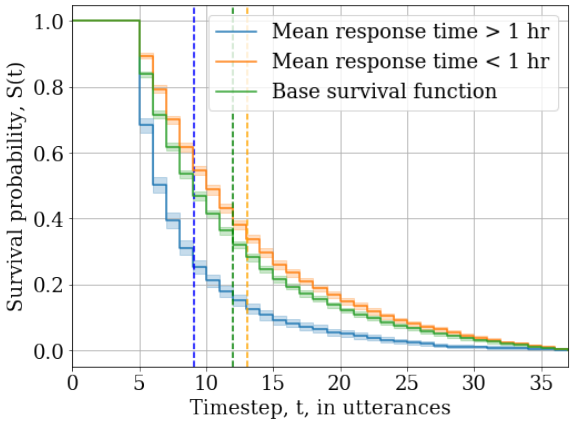


Figure 1: Survival functions for the conversation length prediction task, for the full population (orange) and subsets conditioning on the response time. Dashed lines indicate the expected event time per population.

To estimate the expected event time given a survival function, one can find the expected value of the survival function as follows:

$$\hat{T} = E[S] = \int_0^\infty S(t)dt. \tag{3}$$

These values are indicated with dashed lines in Figure 1, denoting the average conversation length on the full population and the two subsets based on the response time. The instantaneous risk of the event occurring at a point in time, i.e. the probability of the event time T being in a small interval around t, is defined by the *hazard function*:

$$h(t) = \lim_{dt \to 0} \frac{P(t \leq T < t + dt | T \geq t)}{dt}. \tag{4}$$

The cumulative hazard is given by $H(t) = \int_0^t h(t)dt$ and is related to the survival function according to $S(t) = e^{-H(t)}$.

Parametric survival regression models (described in more detail in Section 2.3 and 2.4) are often optimised to predict either the survival or the hazard function, given that it is always possible to convert between them. Such models can include feature representations (such as the response time in Figure 1) to obtain individualised predictions.

2.2 Censoring

A common consideration in survival studies is the presence of censoring, where a participant leaves a study before the end of the observation period, or they do not experience the event of interest within this period. Under censoring, each subject i has an associated potential censoring time C_i and a potential lifetime T_i. We observe $Y_i = min\{T_i, C_i\}$, i.e. the minimum of the censoring and lifetimes, and an indicator variable δ_i for whether the observation ended with death or censoring.

Consider the task of predicting personal attacks (described in more detail in Section 3). Conversations that end without a personal attack occurring can be considered analogous to patients dropping out of a study before the end of the observation period. The duration of the observation can then be taken as the censoring time.

Different survival models account for censoring in different ways. For instance, for a survival curve estimated with the Kaplan-Meier method (Equation 2), censored individuals are removed from the set of candidates at risk (R_i) at the censoring time, without having experienced the event of interest.

2.3 Cox Proportional Hazards

The Cox Proportional Hazards (Cox-PH) model (Cox, 1972) is the most widely used model for survival analysis (Rodriquez, 2007; Kvamme et al., 2019). It has the following hazard function:

$$h(t|\mathbf{x};\theta) = h_0(t) \cdot exp(g(\mathbf{x};\theta)). \qquad (5)$$

$h_0(t)$ represents the baseline hazard for the population at each timestep, such as the base survival function in Figure 1. The $g(\mathbf{x};\theta)$ term is often referred to as the *risk function* and specifies how the feature vector $\mathbf{x}$ of a sample is taken into account using parameters θ. In our experiments, we consider two variations of this approach:

Linear Cox The traditional Cox-PH model (Cox, 1972) uses a linear weighting of a feature vector to calculate the risk function as follows:

$$g(\mathbf{x};\theta) = \theta^T\mathbf{x}. \qquad (6)$$

This model is still widely used in survival analysis research, e.g. Suchting et al. (2019); Zhang et al. (2018c).

DeepSurv DeepSurv (Katzman et al., 2018) uses a neural network to compute the risk function $g(\mathbf{x};\theta)$, where θ represents the weights of the network. The advantage of this is that the neural network can learn nonlinear features from the training data, which often improves predictive accuracy.

During training, the parameters θ are optimised with maximum likelihood estimation for both models. Given individuals i with event time T_i in dataset D, let R_i denote the risk set at T_i and δ_i the censoring indicator. Then, the likelihood of the data is given by:

$$\begin{aligned} L_{cox}(\theta, D) &= \prod_{i \in D} \Big(\frac{h_0(t)\dot{e}xp(g(\mathbf{x_i};\theta))}{\sum_{j \in R_i} h_0(t)\dot{e}xp(g(\mathbf{x_j};\theta))}\Big)^{\delta_i} \\ &= \prod_{i \in D} \Big(\frac{exp(g(\mathbf{x_i};\theta))}{\sum_{j \in R_i} exp(g(\mathbf{x}_j;\theta))}\Big)^{\delta_i}. \qquad (7) \end{aligned}$$

Intuitively, we aim to maximise the risk of i experiencing an event, over all other candidates at risk at time T_i. In the context of predicting conversation lengths, this means that at time T_i, any conversation that has not yet ended could end, but we want to maximise the probability of the candidate that had indeed ended then over the rest. This is referred to as a *partial likelihood*, in reference to the fact that the effect of the features can be estimated without the need to model the change of the hazard over time. The indicator term δ expresses that only non-censored samples contribute terms that impact the likelihood (since the contribution of censored samples would be 1); however, the censored samples would be included in the risk set R_i up until their respective censoring times.

2.4 Survival regression as classification

A different approach to survival regression is to use classification to predict the timestep when an event will occur. DeepHit (Lee et al., 2018) is a neural network model that predicts a distribution over timesteps in this fashion. This provides more modelling flexibility compared to the Cox-PH models, where features are incorporated through the risk function and combined with a baseline hazard.

The model can incorporate multiple competing risks with distinct events of interest, and models censoring as a special type of risk. The output of the network is a vector representing the joint probability that the subject will experience each non-censoring event for every timestep t in the observation period. Censoring is assumed to take place at random and is therefore not included in the prediction. In the case of a single risk, it therefore predicts a vector: $\hat{y}_i = [\hat{y}_{t=0}, ..., \hat{y}_{t=t_{max}}]$, where each output element $\hat{y}_t$ represents the estimated probability $\hat{P}(t|\mathbf{x}, \theta)$ that a subject with feature vector $\mathbf{x}$ will experience the event at time t under the model parameters θ. Instead of a survival function, DeepHit defines a risk-specific *cumulative incidence function* (CIF) which expresses the probability that the event occurs before a time t^*, conditioned on features $\mathbf{x}^*$:

$$\begin{aligned} F(t^*|\mathbf{x}^*) &= \hat{P}(T \leq t^*|\mathbf{x}^*, \theta) \\ &= \sum_{t=0}^{t^*} \hat{P}(t|\mathbf{x}^*, \theta) = \sum_{t=0}^{t^*} \hat{y}_t^* \end{aligned} \tag{8}$$

The loss function for training DeepHit has two components: an event time likelihood and a ranking loss. The ranking loss ensures that earlier events are predicted to happen before later events based on their CIF, but does not penalise models for mispredicting the times in absolute terms. The event time likelihood maximises the probability of the event occurring at the right time ($y^{(i)}_{T^{(i)}}$), or, in the case of censoring, it maximises the probability of the event not happening before the censoring time $(1 - F(T^{(i)}|\mathbf{x}^{(i)}, \theta)$.

2.5 Previous applications in NLP

A small number of NLP studies have employed techniques from survival analysis for time-dependent tasks. Navaki Arefi et al. (2019) use survival regression to investigate factors that result in posts being censored on a Chinese social media platform, finding that negative sentiment is associated with shorter lifetimes. Stewart and Eisenstein (2018) use a linear Cox model to infer factors that are predictive of non-standard words falling out of use in online discourse, finding that words that appear in more linguistic contexts survive longer. Other applications include modelling fixation times in reading (Nilsson and Nivre, 2011) and evaluating dialogue systems (Deriu et al., 2020). However, none of these studies considered time-to-event prediction tasks based on conversations.

3 Survival regression in conversations

We evaluate survival models on two tasks, predicting conversation length and predicting when personal attacks will occur, where each conversation is a subject and the time is measured in utterances.[1]

Dataset	**# Convs.**	**Median time to event**	**Task**
Talk	16 896	6	**1**
Dispute	8 554	9	**1**
Attack	3 466	7	**2**

Table 1: Characteristics of datasets used in this paper. All three datasets originate from Wikipedia Talk pages.

Task 1: Predicting conversation length Having seen t utterances, predict the remaining conversation length in utterances. We use the dataset of Wikipedia Talk page discussions by Danescu-Niculescu-Mizil et al. (2012) (hereafter referred to as **Talk**) and the dataset of disagreements on Wikipedia Talk pages by De Kock and Vlachos (2021) (referred to as **Dispute**) for this task. The Talk dataset was also used to perform the thresholded classification version of this task in Backstrom et al. (2013), mentioned in Section 1.

Task 2: Predicting personal attacks Having seen t utterances, predict the number of utterances until a personal attack occurs. Conversations where no personal attack occurs are censored during training, and the conversation length is used as the observation time. Just less than half of the conversations contain personal attacks (1 569 out of 3 466). This is a novel task; previous work has only addressed predicting whether conversations will derail into personal attack, without attempting to predict when in a conversation this may occur (Zhang et al., 2018a; Chang and Danescu-Niculescu-Mizil, 2019). The motivation cited in both of the abovementioned studies is to prioritise conversations at risk of derailing for preemptive moderation. Survival models can give a more informative answer that takes into account the time until the attack, and therefore which conversations pose the most immediate risk. We use the dataset of Zhang et al. (2018a) for our experiments on this task.

Characteristics of the datasets we use are shown in Table 1. We use only conversations where the event of interest occurs after the fifth utterance, and we remove conversations longer than the 95th percentile as these are often flame wars which may have confounding impacts. Data is split into training, development and test sets with ratios 75:10:15.

3.1 Metrics

Two metrics are calculated to evaluate model performance: mean absolute error and concordance index. The mean absolute error (MAE) for a dataset

[1] The task of predicting when users would cease to use a platform would also have been an interesting case for this study; however, the datasets of Danescu-Niculescu-Mizil et al. (2013) are no longer available.

of n test samples is defined as

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}. \quad (9)$$

This metric provides an easily interpretable score, and it is commonly used in evaluating regression models, e.g. Bitvai and Cohn (2015). However, MAE is not robust to outliers; large errors on a few values can outweigh many correct predictions.[2] MAE is also ill-defined in the presence of censoring as there is no event time to compare against, and it cannot be used to compare model performance between different datasets. For these reasons, we also include the concordance index (Harrell Jr et al., 1996), which is concerned with ordering rather than absolute values. A pair of observations i, j is considered concordant if the prediction and the ground truth have the same inequality relation, i.e. ($y_i > y_j$, $\hat{y}_i > \hat{y}_j$) or ($y_i < y_j$, $\hat{y}_i < \hat{y}_j$). The concordance index (**CI**) is the fraction of concordant pairs. A random model, or a model that predicts the same value for every sample, would yield a score of 0.5. A perfect score is 1. In the presence of censoring, censored samples are only compared with uncensored samples of a smaller event time, since it is known in that case that the uncensored sample should be assigned a later event time.

A disadvantage of the CI score is that it does not reflect how accurate the predictions are in absolute terms, meaning that good CI scores can be achieved with predictions in the wrong range. The two scores thus provide complementing views on model performance.

3.2 Features

The features we consider are based on previous work on conversation length prediction and predicting personal attacks. These are:

- Politeness (**POL**): The politeness strategies from Zhang et al. (2018a) as implemented in Convokit (Chang et al., 2020), which capture greetings, apologies, and saying "please", etc.
- Arrival sequences (**ARR**): The order in which speakers partake in the first 5 utterances, defined by Backstrom et al. (2013).
- Hypergraph (**HYP**): Conversation structure features based on the reply tree, proposed by Zhang et al. (2018b) and implemented in Convokit (Chang et al., 2020). These features capture dynamics between participants, such as engagement and reciprocity.
- Sentiment (**SENT**): Positive and negative sentiment word counts, as per the lexicon of Liu et al. (2005), also implemented in Convokit.
- Time features (**TIME**): Log mean time between utterances and time between last two utterances, inspired by Backstrom et al. (2013).
- Utterance lengths (**LEN**): Log mean utterance length features, measured in tokens.
- Number of participants (**PART**): Also used in Backstrom et al. (2013) and Zhang et al. (2018b).
- Turn-taking features (**TURNS**): The fraction of turns and tokens contributed by the top user, inspired by Niculae et al. (2015).

For the POL, SENT, TIME and LEN features, we include both the mean value throughout the conversation and the gradient of a straight-line fit to capture how the feature changes throughout it. All features are calculated up to the point of prediction, and not for the full conversation.

4 Results

4.1 Experimental setup

We use partly conditional training (Zheng and Heagerty, 2005) to account for using features that change over time, such as politeness, in contrast with static features like the arrival sequence. Under partly conditional training, a feature measured at time t predicts the risk of the occurrence of an event at a future time T. In our case, each individual is a conversation and features are measured after every utterance. Each measurement t of a conversation i is recorded as an individual entry in the dataset, with event time $T_{i,t} = T_i - t$.

This construction is illustrated in Table 2 for the Talk dataset. There are 307 conversations that contain 12 utterances, but 0 samples of length 12 in the training data, since all conversations of this length have 0 utterances remaining and regression is therefore unnecessary. However, we include the first 11 utterances of the length-12 conversations in the training set at $t = 11$, since the remaining length here could be either 0 or 1. As such, there are (642+307=) 769 samples of length 11. We use

[2] This issue is even more pronounced in the root mean-squared-error, which is another popular metric for regression (Hyndman and Koehler, 2006).

t	# Convs. of length t	# Samples of length t
5	6 596	16 896
6	4 010	10 300
7	2 358	6 290
8	1 539	3 932
9	984	2 393
10	640	1 409
11	462	769
12	307	0
Total	16 896	41 989

Table 2: Training set configuration for the Talk dataset. For every conversation, we add a snapshot of its feature values at every timestep to the training data.

a minimum value of $t = 5$ to ensure there is sufficient information from which to make a prediction. Details for the other datasets are in Appendix A.

Our **baseline** model is a univariate Kaplan-Meier estimator (Jager et al., 2008), which predicts the same event time for all samples without taking features into account. For this model and the linear Cox-PH model, we use the implementations in `lifelines`[3]. We use grid search on the validation set for each model to determine hyperparameter values, experimenting with regularisation values in $[0, 0.01, 0.1, 0.5]$, L1 ratios in $[0, 0.1, 0.5, 1]$ and learning rates in $[0.01, 0.1, 0.5, 1]$. We also compare to a **linear regression** model, implemented in `scikit-learn`[4] and using the same features. For the linear regression model, we truncate predictions at 0 since negative times are invalid. Finally, to compare to previous work on **threshold classification**, we implement a logistic regression classifier, using the median of each training set as the cut-off point. For these models, the upper and lower quartiles are used to compute the MAE. For instance, for the Dispute dataset, the threshold value is 9. To calculate MAE, we use an event time of 5 if the model predicts the shorter class and 12 for the longer class.

For the neural models (DeepSurv and DeepHit), we use the implementations in `PyCox`[5] by Kvamme et al. (2019). For both we use two hidden layers with $[128, 64]$ nodes, dropout ($p = 0.3$), batch normalisation, and the Adam optimiser (Kingma and Ba, 2014) with learning rate 0.01.

4.2 Task 1: Predicting conversation length

Results for the conversation length prediction task are shown in Table 3 for the Dispute and Talk datasets (left and middle columns respectively). MAE scores should not be compared between datasets since the datasets have different length distributions, with the Dispute dataset having conversations of up to 37 utterances, compared to a maximum of 12 in the Talk dataset.

For both datasets, all survival models outperform the linear regression and threshold classification models on the MAE metric. The survival baseline uses only population-level knowledge of the event time distribution, and predicts the same event time for all samples, whereas the other baselines take into account information from the features and can therefore tailor predictions per sample. While this results in the survival baseline having the worst CI (0.5), it is still better than linear reg and threshold in terms of MAE, illustrating the importance of separating the effect of time from the other features; time alone can be highly predictive.

The DeepHit model performs the best on the MAE metric on both datasets, with a statistically significant difference from the Linear Cox model at the P=0.01 level using the sign test. The latter performs better than DeepHit on the CI metric for the Dispute dataset, however, this difference is not statistically significant (P=0.869, using a randomised permutation test).

Coefficient analysis Since the linear Cox model performed well on the Dispute dataset and is more interpretable than its deep counterparts, we show its 10 largest coefficients in absolute value in Table 4. Positive weights are associated with larger risk function values, and therefore a shorter conversation. Time between utterances is the most predictive feature, with a longer time between utterances correlating positively with shorter conversations (also observed in Figure 1). This corroborates the findings of Backstrom et al. (2013) and Zhang et al. (2018b). Having more participants is also correlated with shorter conversations. This suggests that the conversations in the Dispute dataset are less prone to long expansionary-style threads (as defined by Backstrom et al. (2013)), where many participants each contribute one utterance. Given the dataset consists of disagreements, it is not surprising that there would rather be focused discussions between a small number of participants.

Features 4, 6, 7 and 8 are from the hypergraph

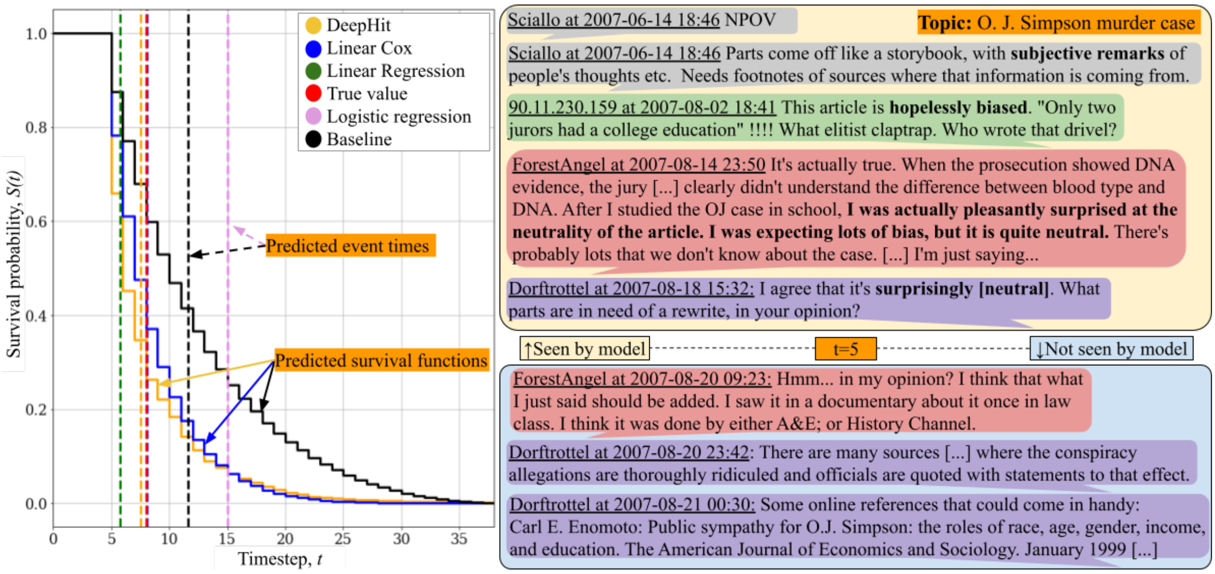
[3] `lifelines.readthedocs.io`

[4] `scikit-learn.org`

[5] `github.com/havakv/pycox/tree/master/pycox`

Model	**TASK 1**				**TASK 2**	
	Dispute		**Talk**		**Attack**	
	MAE ↓	**CI ↑**	**MAE ↓**	**CI ↑**	**MAE ↓**	**CI ↑**
Linear regression	6.213	0.560	1.329	0.542	1.566	0.497
Threshold classification	6.652	0.545	1.505	0.462	1.521	0.489
Survival baseline	5.186	0.500	1.312	0.500	1.585	0.500
Linear Cox	4.995	**0.581**	1.282	0.573	1.481	0.605
DeepSurv	5.014	0.567	1.276	0.575	1.487	0.601
DeepHit	**4.926**	0.578	**1.189**	**0.584**	**1.403**	**0.627**

Table 3: MAE and CI for Task 1 (predicting conversation length) and Task 2 (predicting personal attacks), using the Dispute, Talk and Attack datasets. For MAE, lower values are preferred; for CI, higher. The bold values indicate the best model per metric, for each dataset. Statistical significance is discussed in the text.



Sciallo at 2007-06-14 18:46 NPOV

Topic: O. J. Simpson murder case

Sciallo at 2007-06-14 18:46 Parts come off like a storybook, with **subjective remarks** of people's thoughts etc. Needs footnotes of sources where that information is coming from.

90.11.230.159 at 2007-08-02 18:41 This article is **hopelessly biased**. "Only two jurors had a college education" !!!! What elitist claptrap. Who wrote that drivel?

ForestAngel at 2007-08-14 23:50 It's actually true. When the prosecution showed DNA evidence, the jury [...] clearly didn't understand the difference between blood type and DNA. After I studied the OJ case in school, **I was actually pleasantly surprised at the neutrality of the article. I was expecting lots of bias, but it is quite neutral.** There's probably lots that we don't know about the case. [...] I'm just saying...

Dorftrottel at 2007-08-18 15:32: I agree that it's **surprisingly [neutral]**. What parts are in need of a rewrite, in your opinion?

ForestAngel at 2007-08-20 09:23: Hmm... in my opinion? I think that what I just said should be added. I saw it in a documentary about it once in law class. I think it was done by either A&E; or History Channel.

Dorftrottel at 2007-08-20 23:42: There are many sources [...] where the conspiracy allegations are thoroughly ridiculed and officials are quoted with statements to that effect.

Dorftrottel at 2007-08-21 00:30: Some online references that could come in handy: Carl E. Enomoto: Public sympathy for O.J. Simpson: the roles of race, age, gender, income, and education. The American Journal of Economics and Sociology. January 1999 [...]

Figure 2: Example of a conversation in the Dispute dataset and the predictions of different models at $t = 5$. For this sample, the prediction from the linear Cox model matches the true value.

Feature	**Type**	**Coef.**
1. Time between last two utterances	TIME	0.161
2. Mean time between utterances	TIME	0.121
3. # participants	PART	0.045
4. Mean # replies received per post	HYP	0.042
5. # first person pronouns, mean	POL	0.037
6. Mean of non-zero # repliers per user	HYP	-0.035
7. Fraction of users with more than 1 replier	HYP	-0.035
8. Triadic replies to commenters in mid-thread	HYP	0.034
9. Length of last utterance	LEN	-0.031
10. Arrival sequence 00110	ARR	-0.026

Table 4: Coefficients of the linear Cox model for Task 1, using the Dispute dataset. Positive weights are associated with shorter time-to-event values.

feature set, which describes the structure of the reply-tree. Feature 4 indicates that a thread which forms a shallow tree, with posts receiving many direct responses, is likely to terminate soon. Features 6 and 7 indicate that interactions with multiple users is likely to extend the conversation. The length of the last utterance before the prediction is made (feature 9) is negatively associated with a short time-to-event, indicating that a long last utterance means that the conversation is still ongoing. Finally, the arrival sequence 00110 (feature 10) encodes the order in which two participants (indexes 0 and 1) contributed the first 5 utterances.

The only language-related feature in the 10 most predictive features is feature 5, the number of first person pronouns, which is positively correlated with a shorter time-to-event. Seven of the ten features on this list are from previous work on predicting conversation length (TIME, ARR and HYP), although this was for the thresholded classification version of the task. This indicates that the survival models are inferring similar relationships as the classification variant, while providing a more informative prediction and better performance on the MAE and CI metrics.

Results per timestep To gain an understanding for how our models perform at different timesteps in the conversation, we also evaluate the predictive accuracy at every utterance index. The intuition here is illustrated in Figure 2 for $t = 5$, with the task being: having seen 5 utterances, predict the conversation length. We can see that the actual conversation length is 8, as there are 3 more utterances after the prediction, which are not seen by the model. The linear Cox model predicts the right value in this case. We also show the survival functions predicted by DeepHit, the baseline and the linear Cox model. As explained in Section 2 (Equation 3), the predictions are the expected values of the survival function. We depict predictions in relation to the prediction time; for instance, the linear Cox model predicts a time-to-event of 3 utterances at timestep 5, meaning that the event time will be timestep 8.

Figure 3 shows the MAE and CI scores, aggregated per timestep, for the Talk dataset. Lower MAE scores are observed for later timesteps. However, this does not mean that these models are necessarily better; the possible range of error is smaller later in a conversation. An interesting deviation here is the linear regression model, which performs the best at the last timestep. Upon inspection, we note that this is because the model predicts small values (with median values in range 0.02-0.09) at every timestep. This strategy is likely the result of the dataset being biased towards shorter conversations. At smaller values of t, there is a portion of long conversations which would contribute large errors to drive up the MAE, but this is not present at larger t, hence the discrepancy.

CI allows for more direct comparison of models at different t, since it measures ranking accuracy. On this metric, DeepHit performs better than the linear regression model at all but the last timestep. Both models perform slightly worse at the last two timesteps. A reason for this may be that there are fewer training samples available at larger t, as illustrated in Table 2. Similar trends are observed in the Dispute dataset.

4.3 Task 2: Predicting personal attacks

Results for the personal attack prediction task are shown in the right column of Table 3. Compared to Task 1, higher values are observed on the CI. A reason for this may be that conversation length prediction has to rely on more subtle cues that indicate a conversation has run its course (e.g. users signing off) which are not captured by our features.

We observe again that the DeepHit model performs the best, and that the survival models outperform the three baselines. Due to censoring, the MAE score here is calculated using only uncensored samples; i.e. samples where a personal attack does occur. The censored samples are accounted for in the CI metric, as explained in Section 3.1.

A key question in this task is whether the survival models manage to prioritise samples where a personal attack occurs over censored examples. This means that when comparing a pair consisting of a censored and an uncensored example, we would like for the predicted time to event of the censored example to be higher. We can calculate how often this is true for the DeepHit model by calculating the concordance index of the predicted event times and the inverse of the censoring indicator. For instance, given a pair of samples $s = [censored, uncensored]$, the indicator function is [0,1] and the inverse therefore [1,0]. The ordering of the inverse ($s_0 > s_1$) should be concordant with the predictions. Using the DeepHit model, we find that the model ranks samples where an attack occurs over censored examples in 57.75% of cases, compared to 51.92% with the linear regression model, and 50% for a random baseline. The best model of Zhang et al. (2018a) has a predictive accuracy of 64.9% for classifying which conversations will derail into personal attacks, but does not predict when this will occur.

5 Conclusions

In this paper, we proposed that survival analysis is a useful but hitherto ignored framework for time-to-event prediction tasks in conversations, which are frequently framed as classification tasks. We provided evidence to this by showing that survival models outperform both linear regression and logistic regression models on two tasks and three datasets. The survival regression models explored can be useful in other tasks, for instance, predicting escalation in customer service conversations.

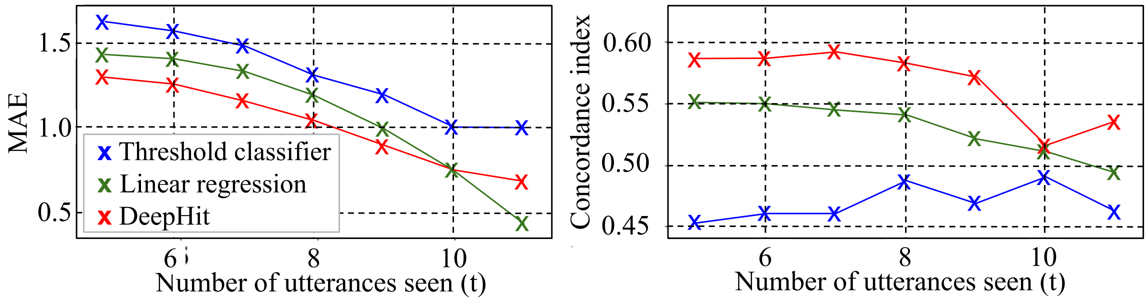


Figure 3: MAE and CI calculated per timestep, t, for the Talk dataset on Task 1. For MAE, lower values are preferred; for CI, higher.

Acknowledgements

Christine De Kock is supported by scholarships from Huawei and the Oppenheimer Memorial Trust. Andreas Vlachos is supported the EPSRC grant no. EP/T023414/1: Opening Up Minds.

References

Lars Backstrom, Jon Kleinberg, Lillian Lee, and Cristian Danescu-Niculescu-Mizil. 2013. Characterizing and curating conversation threads: expansion, focus, volume, re-entry. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 13–22.

Zsolt Bitvai and Trevor Cohn. 2015. Non-linear text regression with a deep convolutional neural network. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 180–185, Beijing, China. Association for Computational Linguistics.

Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. ConvoKit: A toolkit for the analysis of conversations. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 57–60, 1st virtual meeting. Association for Computational Linguistics.

Jonathan P. Chang and Cristian Danescu-Niculescu-Mizil. 2019. Trouble on the horizon: Forecasting the derailment of online conversations as they develop. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4743–4754.

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

Cristian Danescu-Niculescu-Mizil, Lillian Lee, Bo Pang, and Jon Kleinberg. 2012. Echoes of power: Language effects and power differences in social interaction. In *Proceedings of the 21st international conference on World Wide Web*, pages 699–708.

Cristian Danescu-Niculescu-Mizil, Robert West, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. No country for old members: User lifecycle and linguistic change in online communities. In *Proceedings of the 22nd international conference on World Wide Web*, pages 307–318.

Christine De Kock and Andreas Vlachos. 2021. I beg to differ: A study of constructive disagreement in online conversations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2017–2027, Online. Association for Computational Linguistics.

Jan Deriu, Don Tuggener, Pius von Däniken, Jon Ander Campos, Alvaro Rodrigo, Thiziri Belkacem, Aitor Soroa, Eneko Agirre, and Mark Cieliebak. 2020. Spot the bot: A robust and efficient framework for the evaluation of conversational dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3971–3984, Online. Association for Computational Linguistics.

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):361–387.

Rob J Hyndman and Anne B Koehler. 2006. Another look at measures of forecast accuracy. *International Journal of Forecasting*, pages 679–688.

Kitty J Jager, Paul C Van Dijk, Carmine Zoccali, and Friedo W Dekker. 2008. The analysis of survival data: the kaplan–meier method. *Kidney international*, 74(5):560–565.

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).

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Håvard Kvamme, Ørnulf Borgan, and Ida Scheel. 2019. Time-to-event prediction with neural networks and cox regression. *Journal of Machine Learning Research*, 20(129):1–30.

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*, volume 32.

Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th international conference on World Wide Web*, pages 342–351.

Meisam Navaki Arefi, Rajkumar Pandi, Michael Carl Tschantz, Jedidiah R. Crandall, King-wa Fu, Dahlia Qiu Shi, and Miao Sha. 2019. Assessing post deletion in Sina Weibo: Multi-modal classification of hot topics. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 1–9, Hong Kong, China. Association for Computational Linguistics.

Vlad Niculae, Srijan Kumar, Jordan Boyd-Graber, and Cristian Danescu-Niculescu-Mizil. 2015. Linguistic harbingers of betrayal: A case study on an online strategy game. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1650–1659, Beijing, China. Association for Computational Linguistics.

Mattias Nilsson and Joakim Nivre. 2011. A survival analysis of fixation times in reading. In *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, pages 107–115, Portland, Oregon, USA. Association for Computational Linguistics.

G. Rodriquez. 2007. Lecture notes on generalized linear models. `https://data.princeton.edu/wws509/notes/`.

Ian Stewart and Jacob Eisenstein. 2018. Making "fetch" happen: The influence of social and linguistic context on nonstandard word growth and decline. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4360–4370, Brussels, Belgium. Association for Computational Linguistics.

Robert Suchting, Emily T Hebert, Ping Ma, Darla E Kendzor, and Michael S Businelle. 2019. Using elastic net penalized cox proportional hazards regression to identify predictors of imminent smoking lapse. *Nicotine and Tobacco Research*, 21(2):173–179.

Justine Zhang, Jonathan P Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Nithum Tahin, and Dario Taraborelli. 2018a. Conversations gone awry: Detecting early signs of conversational failure. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics.*, volume 1.

Justine Zhang, Cristian Danescu-Niculescu-Mizil, Christina Sauper, and Sean J Taylor. 2018b. Characterizing online public discussions through patterns of participant interactions. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):1–27.

Zhongheng Zhang, Jaakko Reinikainen, Kazeem Adedayo Adeleke, Marcel E Pieterse, and Catharina GM Groothuis-Oudshoorn. 2018c. Time-varying covariates and coefficients in cox regression models. *Annals of translational medicine*, 6(7).

Yingye Zheng and Patrick J Heagerty. 2005. Partly conditional survival models for longitudinal data. *Biometrics*, 61(2):379–391.

A Characteristics of the training data

Details of the Dispute and Attack datasets are shown in Tables 6 and 5.

t	**# Convs. of length *t***	**# Samples of length t**
5	480	3 466
6	959	2 986
7	699	2 027
8	473	1 328
9	371	855
10	312	484
11	172	0
Total	**3 466**	**11 146**

Table 5: Training set configuration for the Attack dataset.

t	**# Convs. of length t**	**# Samples of length t**
5	1 374	9 928
6	1 044	8 554
7	833	7 180
8	710	6 136
9	571	5 303
10	467	4 593
11	427	4 022
12	372	3 128
13	328	2 756
14	315	2 428
15	260	2 113
16	181	1 853
17	185	1 672
18	141	1 487
19	153	1 346
20	144	1 193
21	100	1 049
22	112	949
23	106	837
24	90	731
25	68	641
26	70	573
27	69	503
28	60	434
29	60	374
30	48	314
31	43	266
32	54	223
33	31	169
34	38	138
35	40	100
36	30	60
37	30	0
Total	**8 554**	**74 608**

Table 6: Training set configuration for the Dispute dataset.