Estimating the Instantaneous Survival Rate of Digital Advertising and Marketing IDs: LIFESPAN by Cox-Proportional

ZEOTAP

NILAMADHABA MOHAPATRA
HUMEIL MAKHIJA
SWAPNA SARIT SAHU

AGENDA

- WHAT ARE A DIGITAL ADVERTISING ID OR DIGITAL MARKETING IDS
- PROBLEM STATEMENT (for maintaining the lifecycle of these IDs)
- CHALLENGES WITH TRADITIONAL TECHNOLOGIES
- EXPECTATION FROM THE NEW SOLUTION
- WHY LOOKING FOR A FEEDBACK BASED PARAMETRIC MODEL
- PROPOSED SOLUTION
- HOW TO FIND AND CHOOSE DURATION PARAMETER(IMPLICIT)
- COX PROPORTIONAL HAZARDS MODEL
- OBSERVATION
- COMPARISON AND VALIDATION
- CONCLUSION AND FUTURE WORK

ZEOTAP

WHAT ARE A DIGITAL ADVERTISING ID OR DIGITAL MARKETING IDS

- Digital Advertising ID or Digital Marketing IDs are called as generally called as Device IDs.
- The device ID is the currency of digital advertising and marketing ecosystem. It is used for identifying an user in the online space and enabling them with programating advertisement campaigns.
  - For example: Some Advertisement IDs are android ID, an apple ID, cookie ID etc
- One person can have many online identifier consisting of similar or different types of divide IDs as shown in the figure 1. below.
- These ids acts as a primary key with attributes like such is demographics, App install , Interest , Intent , etc linked to it. This helps us understand the user better so that targeting the user for a specific campaign become more effective.

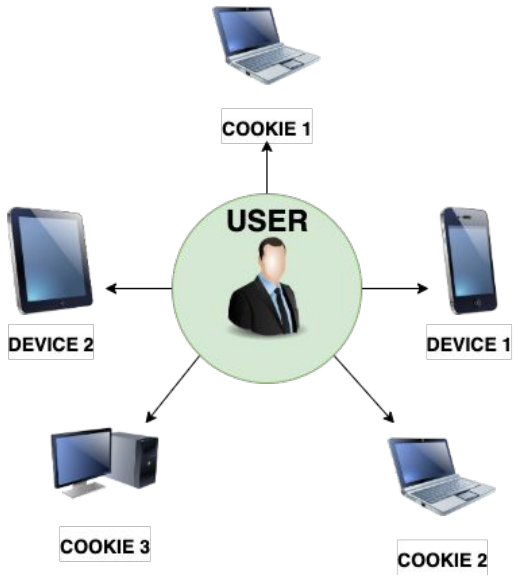


FIG 1.0

	ID	AGE	DEVICE_OS_VALUE	GENDER	FREQUENCY OF APP UPDATE	IAB	DP_CT
1	id1_abc_hash	30	ANDROID	MALE	1	{IAB12 : 213 , IAB14 : 201 , IAB11 : 203}	2
2	id2_abc_hash	20	IOS	FEMALE	1	{IAB12 : 213 }	5
3	id3_abc_hash	15	WINDOWS	UNKNOWN	3	{IAB22 : 213 , IAB54 : 205}	3
4	id4_abc_hash	25	IOS	MALE	2	{IAB22 : 213 , IAB54 : 205}	4
5	id5_abc_hash	30	LINUX	MALE	1	{IAB22 : 213 , IAB51 : 123}	2

FIG 2.0 Profile Store

ZEOTAP

PROBLEM STATEMENT (For Maintaining the Lifecycle of these IDs)

- These digital currencies are billions in number and is stored in a database (***Profile Store*** **refer Fig 2.0**) with numerous attributes linked to it.
- This optimisation problem comes into the picture because the ***Profile Store*** is ignorant of the existence of the ID (whether it is active or inactive).
- Keeping the IDs for a longer time will increase the load for the downstream pipelines that incur more storage and computation cost. This can also lead to digital campaigns(advertising or marketing) with low active users thus degrading the performance.
- Keeping it for less time, losses of ID prematurely can lead to multiple loss of information. This can affect the segment volume export for a campaign largely.
- Now the problem boils down to *finding an optimal time for each ID to keep it in the* ***Profile Store*** *as accurately as possible which will work as a proxy for the ID being active.*

CHALLENGES WITH TRADITIONAL TECHNOLOGIES

- Traditionally most of the non-feedback systems run on TTL based methods to purge the IDs and clean the database.
- A constant time to leave(TTL) such as 90/120/180 days are applied to the profile store which acts as a proxy of life.

- There are certain problems with the TTL based system
  - Putting a TTL on the ***Profile Store*** level assumes all the IDs to have similar lifetime (the period in which they are active) which may not be true.
  - Putting a smaller TTL, losses of ID prematurely can lead to multiple loss of information. This can affect the segment volume export for a campaign largely.
  - Putting a higher TTL, can lead to the original problem of cost and computation.
  - Determining an optimal TTL is a tedious task as we are unaware about the creation time of the ID.

EXPECTATION FROM THE NEW SOLUTION

- The new solution should provide life information at granular level (Mechanism to treat each unique ID differently)
- Should be better than a TTL based system with any time window.
- Should save a lot of Computation and Storage cost without compromising volume and quality.
- Incremental and robust in nature.
  - Should not be losing value of an ID just because the algorithm flagged it in that way.
  - Score for incorporating active feedback which is coming continuously.

WHY LOOKING FOR A FEEDBACK BASED PARAMETRIC MODEL

- Problem 1: Lack of information regarding the exact time of ID creation makes TTL based methods inefficient. Being a DMP, we might have the time stamp at which the ID enters into our system which is very different from the actual ID creation time in the device.

- Problem 2 : Each ID with different attributes associated like OS, gender, age, device, bid stream frequency, etc has a different lifetime. In a feedback-based system, we have some uncensored IDs for which we have the lifetime information available. Using this information we can calculate the expected value of life and use it as a TTL. This method still works on the assumption that all the IDs have equal lifetime while the expected value of the uncensored IDs tries to infer the best TTL possible. In this approach, we still have some IDs whose real life is less than the TTL value and we are prolonging their deletion till it reaches the TTL and some IDs whose actual life is more than TTL value and we early delete it once it reaches the TTL. In both conditions, there is an opportunity to estimate lifetime value more efficiently.

PROPOSED SOLUTION

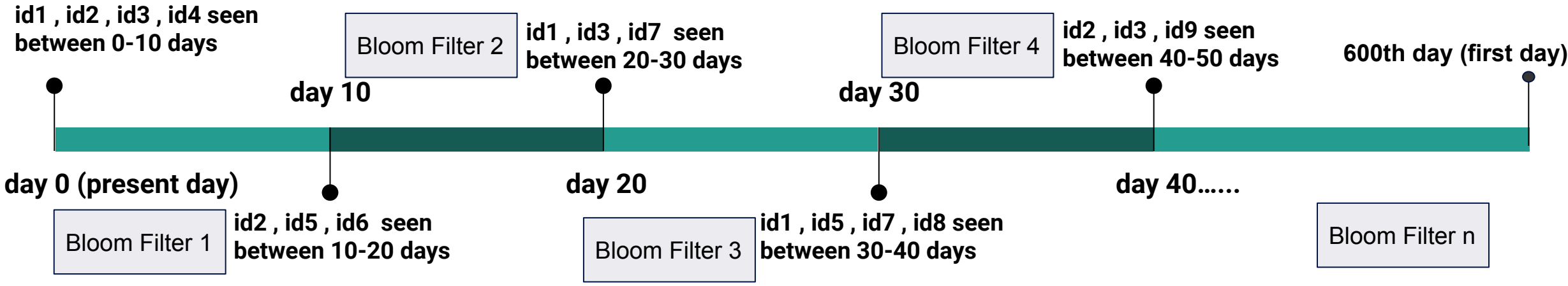
- Solution to the above problems comes as modeling it as a ***Survival Model***.
- Where at any time *t* given the covariates for a particular ID, we calculate the survival probability of it.

- We have used a semiparametric cox proportional hazard model do so.
- The cox-proportional hazard model expect 2 things
  - *Set of Covariates*
  - *Duration Parameter*

- Where duration parameter acts as a proxy of life.
- Challenge in our use case: ***Explicit duration parameter was not available in our data set.***

HOW TO FIND DURATION PARAMETER

- As a DMP we lack of information regarding the exact ID creation time.
- All we have is the time stamp at which the ID enters into our system which is very different from the actual ID creation time.
- To get a proxy of ID life we took bid requests from DSPs as implicit feedback which act as a proxy of device IDs life.
- The requests are collected for **N** = 2 years timeframe and merged **d** days(timestep) into 1 bin forming 'n' groups :
  - **n = N/d**
- Then we built our bloom filter on each group which is used to check whether a particular ID is seen by the DSP in that particular time interval or not.



ZEOTAP

CONTINUE

In our analysis we took 600 days interval and run over bloom filter over groups of 10 days .

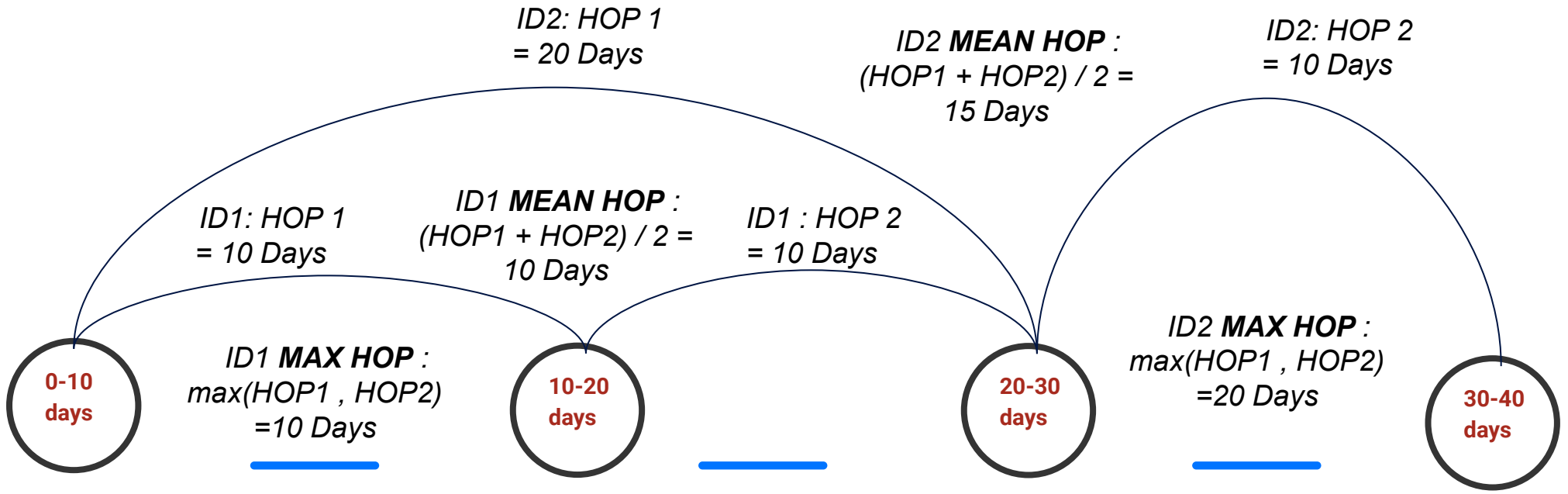
- **Training set** : 600 - 200 days [Initial day to 400 days]
- **Test set** : 200 - 0 days [till present day with ground truth duration parameter value for future timestamps prediction]

We defined certain terminologies using bloom filter data for each single Device ID:

- ***ACTIVE LIST*** : It is defined as a list containing all the timestamps at which that device ID was active.
- ***ACTIVE MIN :*** It is defined as the latest occurrence timestamp of the device ID in ***ACTIVE LIST*** (i.e min ***ACTIVE LIST***)
- ***ACTIVE MAX :*** It is defined as the latest occurrence timestamp of the device ID in ***ACTIVE LIST*** (i.e max ***ACTIVE LIST***)
- ***LIFE :*** It defined as the difference between ***ACTIVE MAX*** and ***ACTIVE MIN***.
- ***MAX HOP :*** It is defined as maximum timestamp difference in between any two consecutive timestamp of ***ACTIVE LIST***.
- ***MEAN HOP :*** It is defined as the mean of the timestamp difference between any two consecutive timestamps of ***ACTIVE LIST***.

Out of all the above variables , ***MEAN HOP*** , ***MAX HOP*** and ***LIFE*** are selected as duration parameters.

EXPLAINING DURATION PARAMETERS



ID1 : tp=0
ID2 : tp=0

ID1 and ID2 first seen between 0-10 days

ACTIVE LIST ID1 : [0,10,20]
LIFE: 20 DAYS
MAX HOP ID1: 10 DAYS
MEAN HOP ID1: 10 DAYS

ID1 : tp=10

ID1 was active between 10-20 days

ACTIVE LIST ID2 : [0,20,30]
LIFE: 30 DAYS
MAX HOP ID2: 20 DAys
MEAN HOP ID2: 15 DAYS

ID1 : tp=20
ID2 : tp=20

ID1 and ID2 were active between 20-30 days

ACTIVE LIST ID3 : [30]
LIFE:0 DAYS
MAX HOP ID3: 0 DAYS
MEan HOP ID3: 0 DAYS

ID2 : tp=30
ID3 : tp=30

ID2 and ID3 were active between 30-40 days

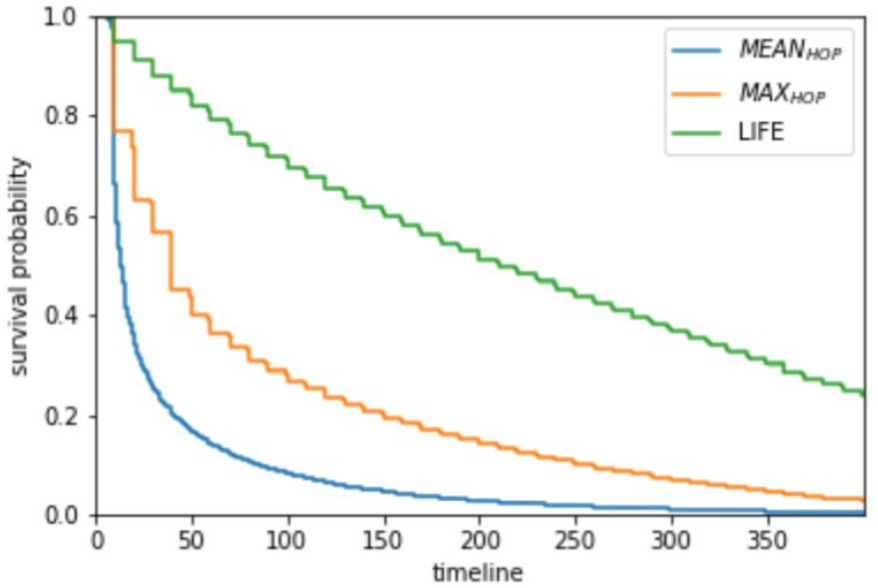
ZEOTAP

CHOOSING DURATION PARAMETER

We analysed all 3 duration parameters and concluded that :

1. ***LIFE*** as duration parameter is not suitable for survival analysis due to its linear decay .

2. In the case where an ID is not observed by the bloom filter for a long time and suddenly it's observed by the filter thus increasing its ***MAX HOP*** value but it is not the expected(mean) time an ID will take to be seen hence ***MEAN HOP*** will not change drastically so ***MEAN HOP*** is taken as duration parameter but not the ***MAX HOP***.



Kaplan–Meier curve for duration parameter importance

ZEOTAP

COX PROPORTIONAL HAZARDS MODEL

- **Covariates:**
  - AGE:Numeric value range between 18 to 100 defined by Demographic age of the device ID.
  - DP_CT:Numeric value defined by number of data partner contributed to the profile information of the device ID.
  - FREQUENCY_OF_APP_UPDATE:Numeric attribute defined by frequency (in no. of days) of application data being updated in the device ID profile store.
  - GENDER:Categorical value defined by demographic gender of the device ID. It can take a value either MALE or FEMALE.
  - DEVICE OS:Categorical value defined by the device IDs
    OPERATING SYSTEM. It can take a value of ANDROID, IOS or others (which includes windows, linux etc).
- **Duration Parameter**
  - Mean hop

EXPERIMENTS

Keeping ***MEAN HOP*** as duration parameter multiple Cox proportional hazard models were built with different sets of profile attributes as covariates and finally achieved a concordance score of 0.9 with the covariates **(AGE, DP_,CT, FREQUENCY_OF_APP_UPDATE , GENDER (MALE, FEMALE) , DEVICE_OS(ANDROID, IOS, OTHERS))**

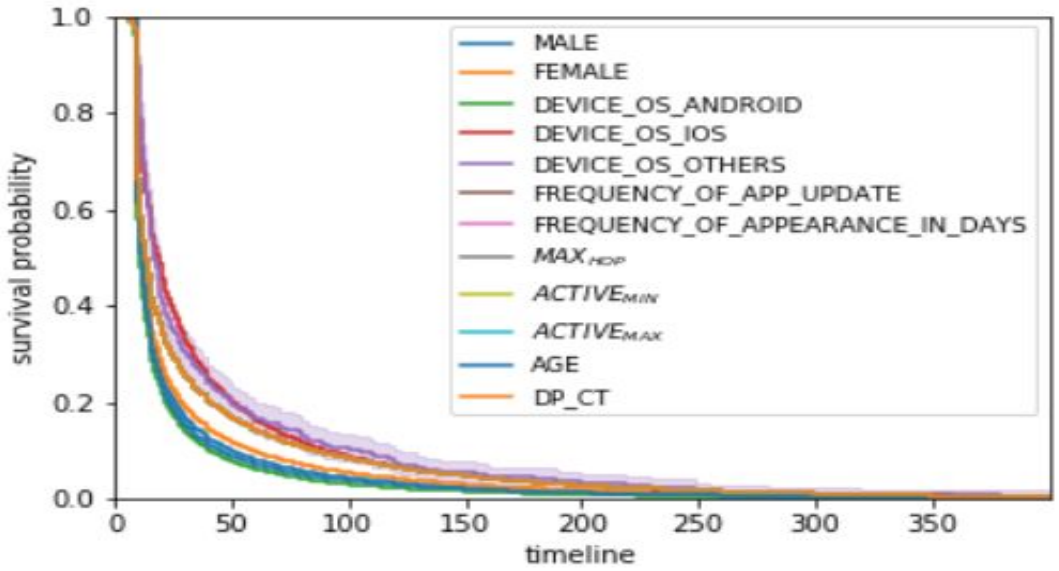


Figure 2: Effect of all the Covariates on $MEAN_{HOP}$

OBSERVATION FROM MODEL

- F**requency_of_app_update** and **Age** has a little effect on the hazard rate.
- IDs with gender as **Male** or **Female** tend to suffer from lower survival rates with a higher risk of 18% and 22% respectively. 3. device IDs with the operating system as **Android** or **IOS** have a little but negative impact on the hazard rate but the device IDs having operating systems like **Blackberry**, **Windows**, **Meego**, **Linux**, etc (categorized as others) are associated with a very low hazard rate of -22% thus lives longer.
- IDs with high value for **DP_CT** found to increase the hazard by 10%. The reason could be more the value of DP_CT more the confidence we have about the covariates of the device IDs. We also observed that the last active timestamp tends to be associated with a 1.7% higher hazard rate .

Table 1: Effect of covariates on Hazard Rate

COVARIATES	AGE	MALE	FEMALE	DP_CT	DEVICE_ OS_ ANDROID	DEVICE_ OS_ IOS	DEVICE_ OS_ OTHERS	FREQUENCY_ OF_ APP_ UPDATE	ACTIVE MIN	ACTIVE MAX
COEFFICIENT VALUES	-0.00	0.17	0.20	0.10	0.04	0.02	-0.25	-0.01	0.02	-0.02
HAZARD RATIO	0.99	1.18	1.22	1.10	1.04	1.01	0.78	0.99	1.017	0.980

COMPARISON AND VALIDATION

We predicted the survival probability for each of the device IDs for the future timesteps. A threshold 0.07 or less is used to flag a device ID as dead at any given time step and deleted from the storage. We ran the model and compared our results with industry-standard TTLs(of 90, 120, or 180 days) along with the Mean of the LIFE (ground truth). We performed our analysis on these 11 attributes to compare our results :

1. **MODEL(Hazard) LIFETIME (α)** : Then model predicted lifetime for a device ID.(in number of days) . Ti is the time-step at which the predicted survival probability of an ID drops below the threshold.

$$\alpha = T_i^{id} - ACTIVE\ MAX^{id}$$

2. **ACTUAL LIFETIME (γ)**: The actual lifetime observed for a device ID.(in number of days)

$$\gamma = ACTIVE\ MAX^{id} - ACTIVE\ MIN^{id}$$

3. **PROBABILISTIC MODEL LIFETIME (ω)**: The Lifetime observed for a device ID by probabilistic models .(in number of days) . $Tprob_i$ is the time-step at which the predicted probability by the logistic regression and naive bayes models of an ID drops below the threshold.

$$\omega = Tprob_i^{id} - ACTIVE\ MIN^{id}$$

4. **MODEL LIFETIME MEAN (α_μ)**: It is defined as the mean of the lifetime distribution predicted by the survival model.
5. **PROBABILISTIC MODEL MEAN (ω_μ):** It is defined as the mean of the lifetime distribution predicted using probabilistic model or the expected value of ω. (Naive bayes and logistic regression models)

CONTINUE

6. **ACTUAL LIFETIME MEAN (γ_μ)** : It is defined as the mean of the lifetime distribution (Using Ground Truth values) or the expected value of γ.

7. **MEAN ABSOLUTE ERROR($MAE_{MODEL\_BASED(hazard)}$) ($MAE_{\alpha-\gamma}$)**: It is defined as the expected value of $|\alpha - \gamma|$

$$MAE_{\alpha-\gamma} = \mathbb{E}|\alpha - \gamma|$$

8. **MEAN ABSOLUTE ERROR($MAE_{PROBABILISTIC\_MODEL\_BASED}$) ($MAE_{\omega-\gamma}$)**: It is defined as the expected value of $|\omega - \gamma|$

$$MAE_{\omega-\gamma} = \mathbb{E}|\omega - \gamma|$$

9. **MEAN ABSOLUTE ERROR (MAE_{TTL_BASED}) ($MAE_{\gamma\mu-\gamma}$)**: It is defined as the expected value of $|\gamma_\mu - \gamma|$

$$MAE_{\gamma\mu-\gamma} = \mathbb{E}|\gamma_\mu - \gamma|$$

10. **CI : Confidence interval (For model based MAE)**

11. **MEAN%_ERROR_REDUCTION_PER_ID $\Downarrow\epsilon$ MAE_{TTL}**: It is defined as % error reduced by predicting the life of an ID using a model-based approach compared to any feedback or non-feedback based TTL approach.

$$\frac{|MAE_{TTL\_BASED} - MAE_{MODEL\_BASED}| \times 100}{MAE_{TTL\_BASED}}$$

ZEOTAP

OBSERVATION: ROBUSTNESS OF THE MODEL

Table 2: Model stability and robustness check w.r.t sample size

EXP	MODEL AND TTL BASED FEATURES				
	α_μ	γ_μ	$MAE_{\alpha-\gamma}$	99% CI of $MAE_{\alpha-\gamma}$	SAMPLE SIZE
EXP-1	284.15	301.12	49.11	[48.69-49.54]	33,207
EXP-2	287.78	303.46	51.71	[51.52-51.99]	76,855
EXP-3	287.37	303.38	50.75	[50.54-51.81]	316,641
EXP-4	298.96	309.90	51.59	[51.50-51.68]	811,447
EXP-5	299.23	309.96	51.55	[51.50-51.61]	1,894,083

OBSERVATION: EFFICIENCY

Table 3: Model accuracy comparisons w.r.t TTL based approach

EXP	MODEL V/S TTL BASED APPROACH							
	MAE_{TTL_BASED} (TTL = γ_μ)	$\Downarrow \epsilon_{MAE_{TTL=\gamma_\mu}}$	MAE_{TTL_BASED} (TTL = 90)	$\Downarrow \epsilon_{MAE_{TTL=90}}$	MAE_{TTL_BASED} (TTL = 120)	$\Downarrow \epsilon_{MAE_{TTL=120}}$	MAE_{TTL_BASED} (TTL = 180)	$\Downarrow \epsilon_{MAE_{TTL=180}}$
EXP-1	107.33	54%	219.71	78%	193.24	75%	150.17	67%
EXP-2	107.40	52%	222.553	77%	196.02	74%	152.34	66%
EXP-3	107.484	53%	222.42	77%	195.88	74%	152.23	67%
EXP-4	107.68	52%	222.106	77%	195.58	74%	152.03	66%
EXP-5	106.56	52%	222.160	77%	195.64	74%	152.09	66%

1. As we observed that for any industry-standard TTL of (90,120 or 180) days(non-feedback based) our model reduces down the error by approximately **66% to 77%** days per device ID.
2. In a Feedback based systems where the life distribution is known if we use the expected lifetime value as a TTL, then also our model reduces the error up to **52% to 54%** days per device ID.

COMPARISON OF DIFFERENT METHODS

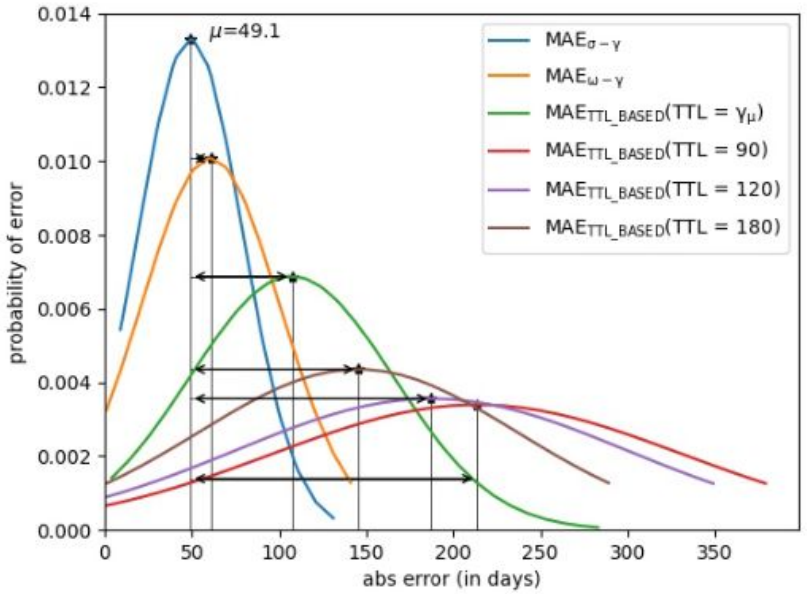


Figure 4: Distribution of MAE_{MODEL_BASED} along with MAE_{TTL_BASED} and $MAE_{PROBABILISTIC\_MODEL\_BASED}$

CONCLUSION AND FUTURE WORK

- The mean deviation of predicted lifetime of an ID using our model from actual lifetime comes out to be around **50 days** from the earlier error rate of **108 days** after using the ground truth mean value as TTL which is **52%** more efficient .
- For this kind of task hazard model is preferred as we observed it outperforms all classification based systems.
- Our model reduces **storage and computation cost** by **10% to 16%** per run with a frequency of 2 months run.
- This also provide us a lever to decide how long to keep a device ID in our profile store thus optimizing the computation and processing cost as well as the volume.
- In future work we would like to automate the threshold picking criteria which is selected experimentally in this work.

THANK YOU

ZEOTAP

Q & A

ZEOTAP