SCHOOL OF INFORMATION CENTER FOR
SOCIAL MEDIA RESPONSIBILITY
UNIVERSITY OF MICHIGAN

Calibrate-Extrapolate: Rethinking Prevalence Estimation with Black Box Classifiers

Siqi Wu, Paul Resnick

ICWSM 2024
Buffalo, NY, USA

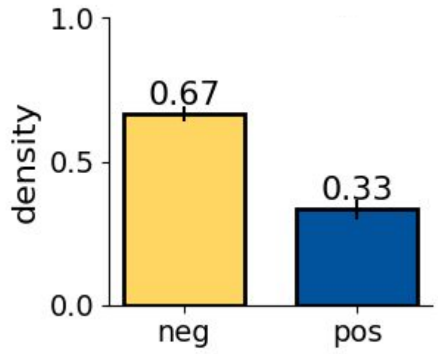
Tutorial: https://avalanchesiqi.github.io/prevalence-estimation-tutorial/

What is prevalence estimation?

Given an unlabeled dataset, count the frequency of each class in it

A core task in computational social science, to estimate the fraction of

- happy tweets in a day (Dodds et al. 2011)
- automated accounts on Twitter (Yang et al. 2020)
- cross-partisan discussion on YouTube (Wu and Resnick 2021)
- political discussion in non-political subreddits (Rajadesingan et al. 2021)
- anti-social posts on Reddit (Part et al. 2022)
- many many more...

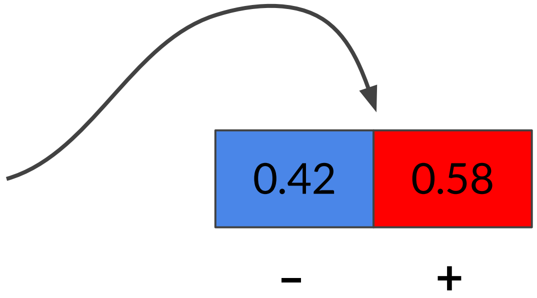


Common setup

Each item has a set of features **X**, and an unobserved ground truth label **GT**

X

	text	has_url	lang	GT
item 1	Lorem ipsum	yes	en	+
item 2	dolor sit amet	yes	fr	+
item 3	consequat vitae	no	en	–
item 4	…			–

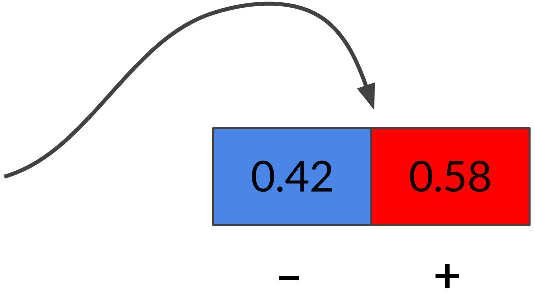


Why is prevalence estimation difficult in CSS?

- Social media data is often on a large scale
- Ground truth labels are difficult or expensive to obtain
- Obtained GT labels have noise

X

	text	has_url	lang	GT
item 1	Lorem ipsum	yes	en	+/?
item 2	dolor sit amet	yes	fr	?
item 3	consequat vitae	no	en	–
item 10000	...			?



(Pre-trained black box) machine learning classifiers can help

$$\textit{classifier: } \mathbf{X} \Rightarrow C(\mathbf{X}) \sim GT, \textit{ where } C(\mathbf{X}) \textit{ in } [0, 1]$$

VADER (sentiment), Perspective API (toxicity), ChatGPT (almost everything)…

X

	text	**has_url**	**lang**	**C(X)**	**GT**
item 1	Lorem ipsum	yes	en	0.87	+
item 2	dolor sit amet	yes	fr	0.72	?
item 3	consequat vitae	no	en	0.45	–
item 10000	…			0.18	?

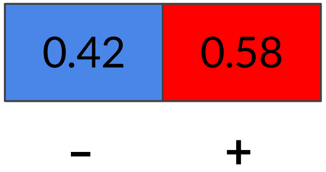
–	+
0.42	0.58

Why is prevalence estimation still difficult in CSS?

- What if we have a less accurate classifier?
- C(X) is a confidence score, but not a probability score
- How to make reliable estimates with fewer GT labels?

X

	text	has_url	lang	~~$C_1(X)$~~	$C_2(X)$	GT
item 1	Lorem ipsum	yes	en	~~0.87~~	0.55	+
item 2	dolor sit amet	yes	fr	~~0.72~~	0.48	?
item 3	consequat vitae	no	en	~~0.45~~	0.43	–
item 10000	…			~~0.18~~	0.32	?



Three commonly seen missteps

- *you want to estimate the prevalence of toxic comments on social media*
- *you have a very large dataset*
- *you hear good things about the Perspective API*

$$\overbrace{\hspace{6cm}}^{\textbf{X}}$$

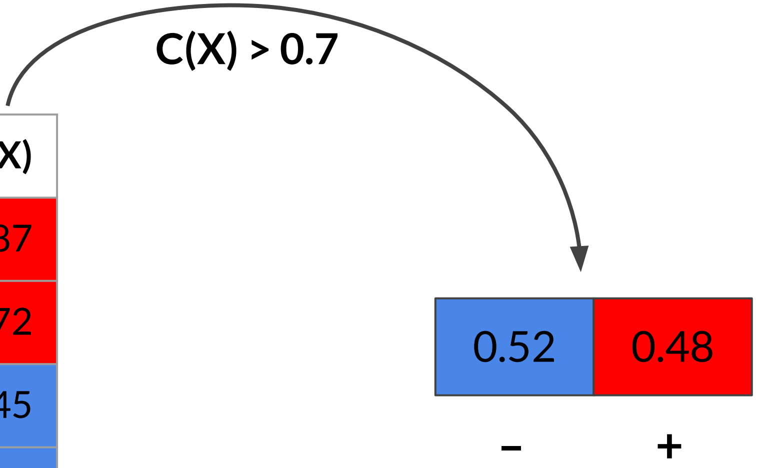
	text	**C(X)**
item 1	Lorem ipsum	0.87
item 2	dolor sit amet	0.72
item 3	consequat vitae	0.45
item 10000	…	0.18

Misstep 1

➢ *"Perspective API suggests 0.7-0.9 as a threshold"*

X

	text	C(X)
item 1	Lorem ipsum	0.87
item 2	dolor sit amet	0.72
item 3	consequat vitae	0.45
item 10000	…	0.18



Misstep 1

➢ *"Perspective API suggests 0.7-0.9 as a threshold"*

❌ Dataset shift: Training and test datasets differ in important ways (Moreno-Torres et al. 2011)

Misstep 2

➢ *"Perspective API returns a probability score"*

$$\underbrace{\quad\quad\quad\quad}_{}X$$

	text	C(X)
item 1	Lorem ipsum	0.87
item 2	dolor sit amet	0.72
item 3	consequat vitae	0.45
item 10000	...	0.18

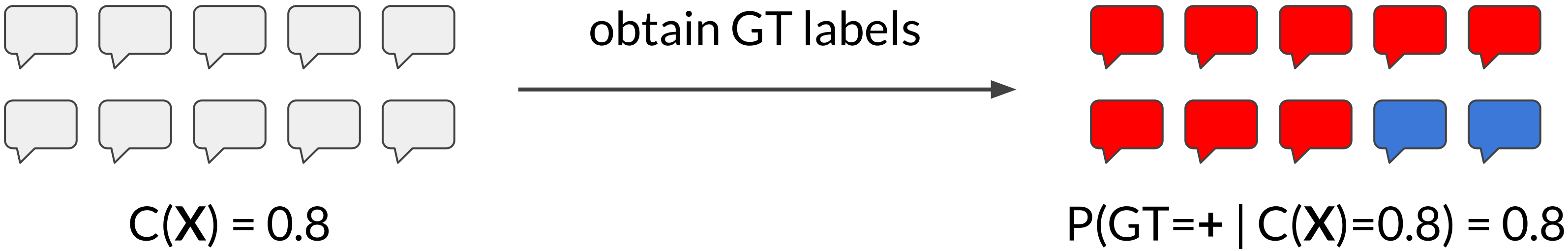
$\sum C(X) / |C(X)|$

–	+
0.55	0.45

Misstep 2

- ➢ *"Perspective API returns a probability score"*

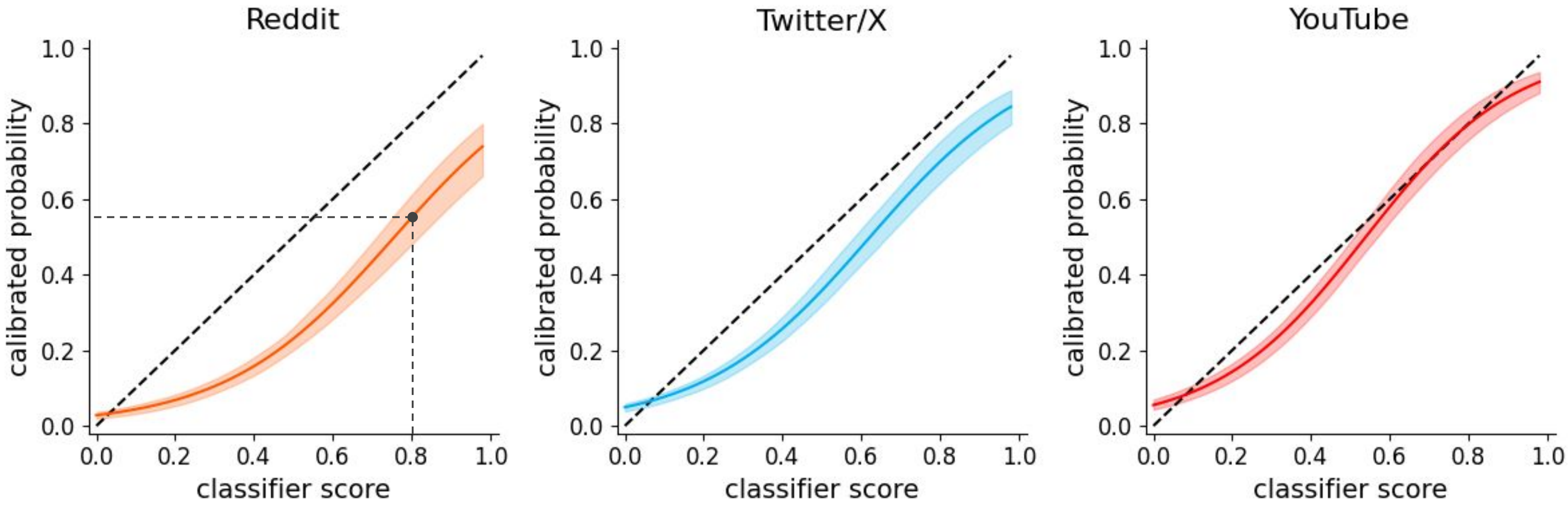
❌ Generally, one should not interpret classifier output as **calibrated** probability



Misstep 2

➢ *"Perspective API returns a probability score"*

❌ Generally, one should not interpret classifier output as **calibrated** probability

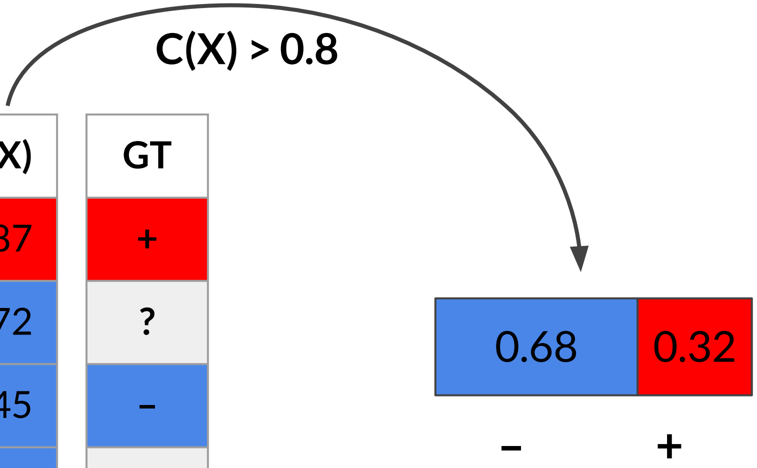


Misstep 3

➢ *you subsample the data, collect GT labels for the sample, find that Perspective API works well (e.g., F1=0.9) and the optimal threshold is 0.8*

X

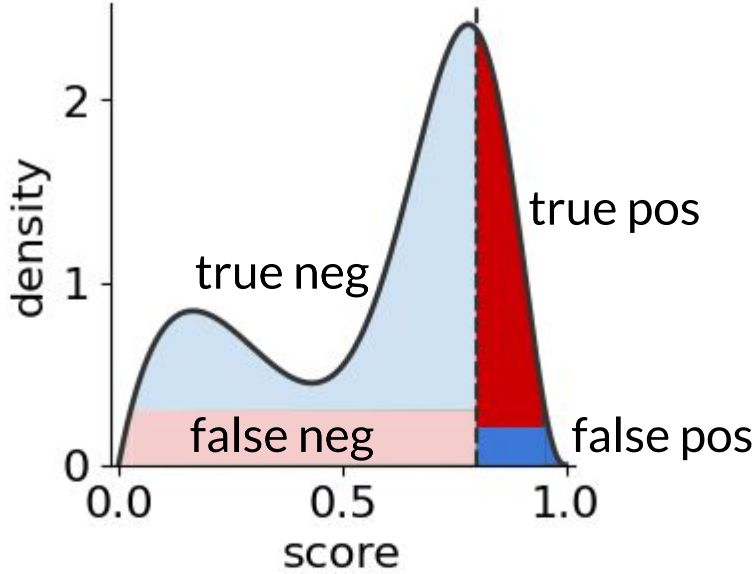
	text	C(X)	GT
item 1	Lorem ipsum	0.87	+
item 2	dolor sit amet	0.72	?
item 3	consequat vitae	0.45	–
item 10000	…	0.18	?



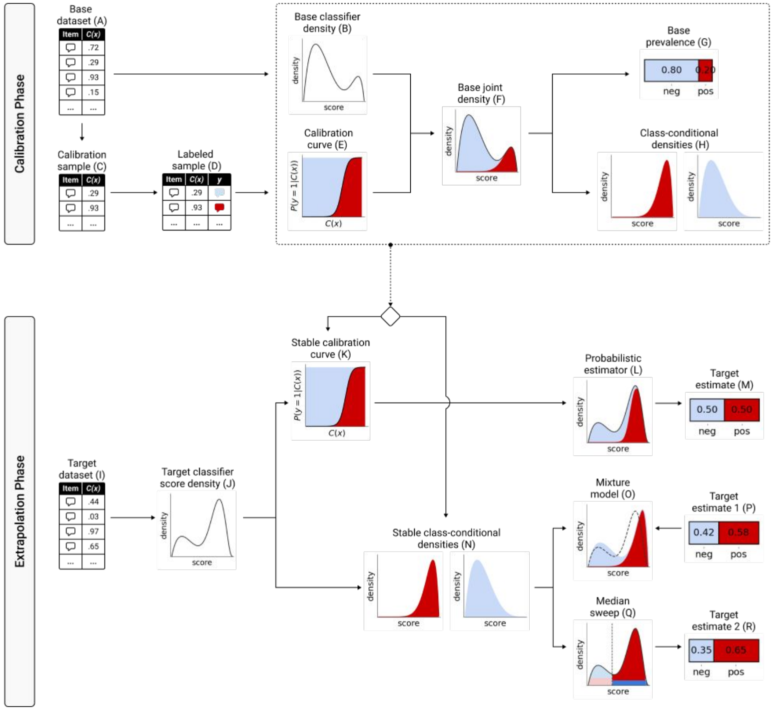
Misstep 3

➢ *you subsample the data, collect GT labels for the sample, find that Perspective API works well (e.g., F1=0.9) and the optimal threshold is 0.8*

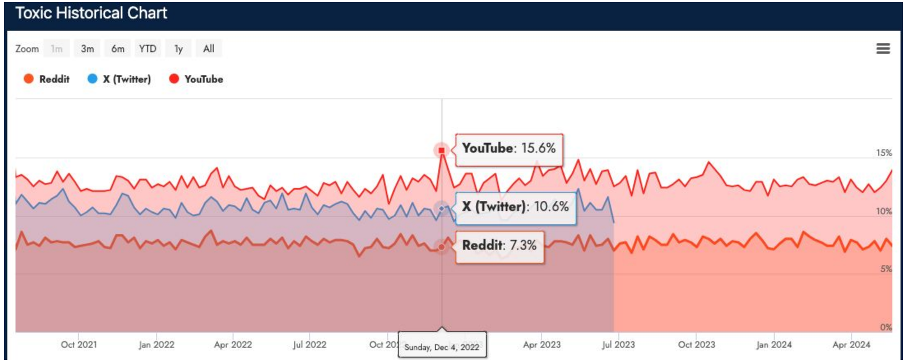
❌ Classifier errors are not accounted for



Contributions: Conceptual framework + real world application

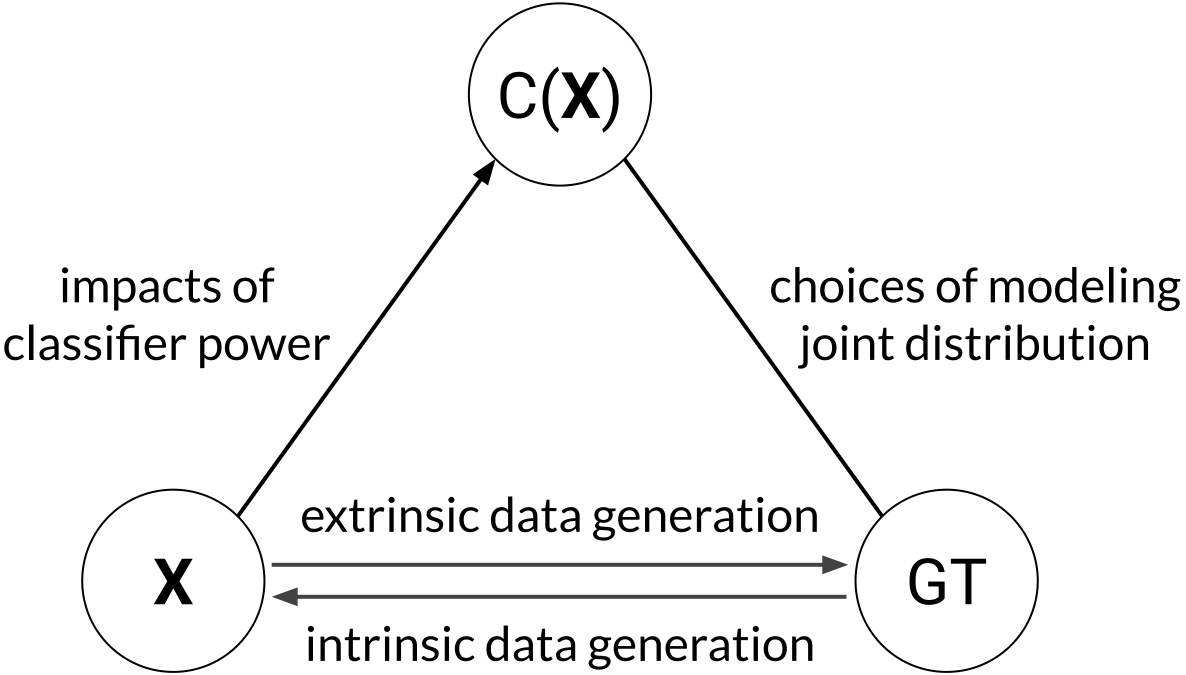


Calibrate-Extrapolate framework



How many **H|O|T** (hateful, offensive, toxic) comments are posted on social media every day?

Contributions: The interplay between **X**, C(**X**), and GT



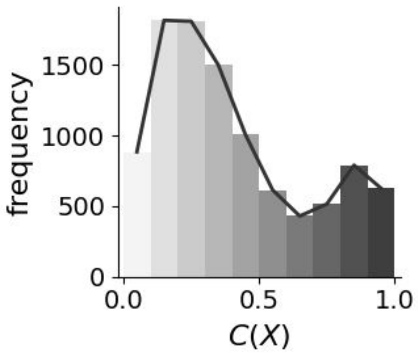
Outline

1. Introduction
2. **How to do prevalence estimation? Calibrate & Extrapolate**
3. Application: Estimating the fraction of H|O|T comments on news articles
4. Practical advice for prevalence estimation

The classifier score density for unlabeled datasets

	C(X)
item 1	0.87
item 2	0.72
item 3	0.45
item 10000	...

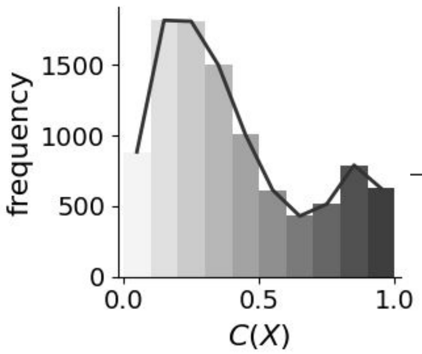
classifier score density



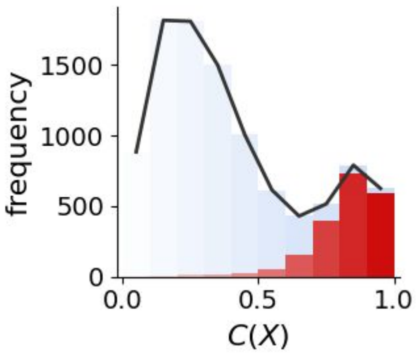
The (classifier output, ground truth) joint distribution

	C(X)
item 1	0.87
item 2	0.72
item 3	0.45
item 10000	…

classifier score density



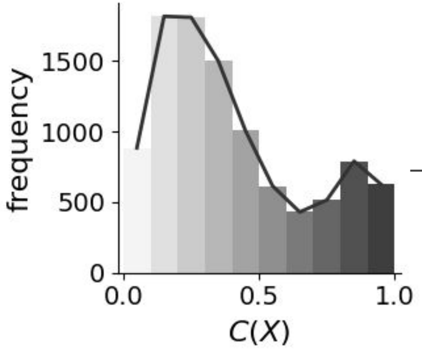
joint distribution



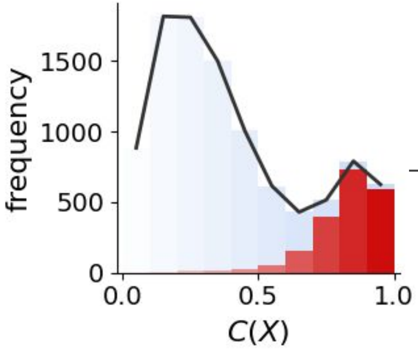
Deriving label density

	C(X)
item 1	0.87
item 2	0.72
item 3	0.45
item 10000	...

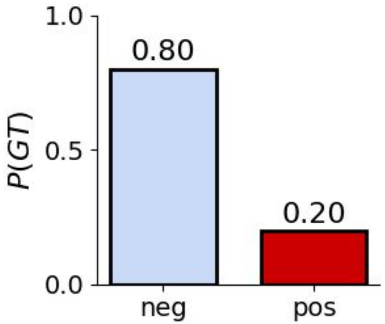
classifier score density



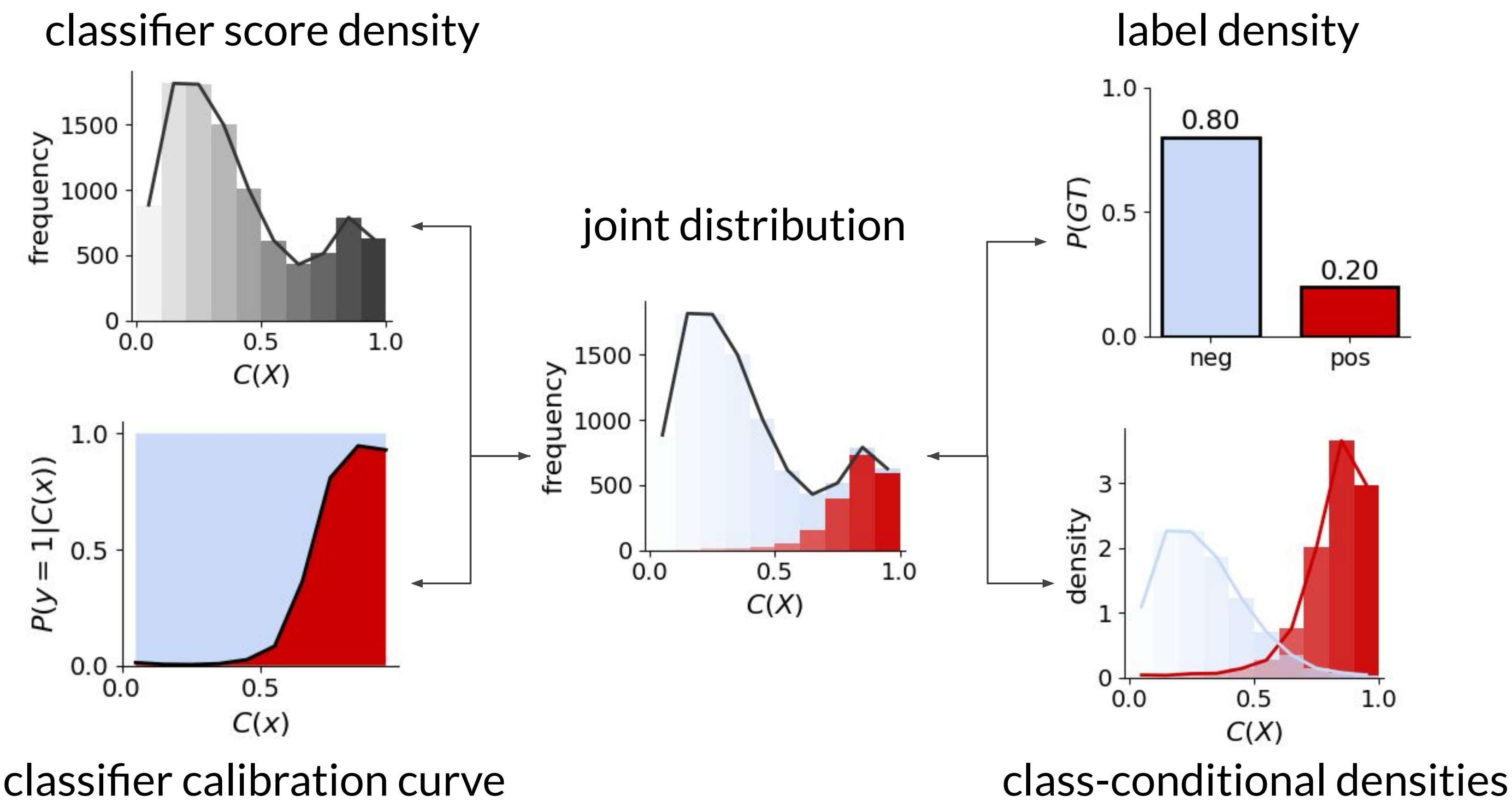
joint distribution



label density



Rethinking prevalence estimation as modeling joint distribution of C(X) and GT

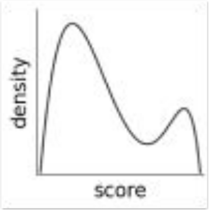


Calibration phase: A one-time prevalence estimate is needed for a single dataset

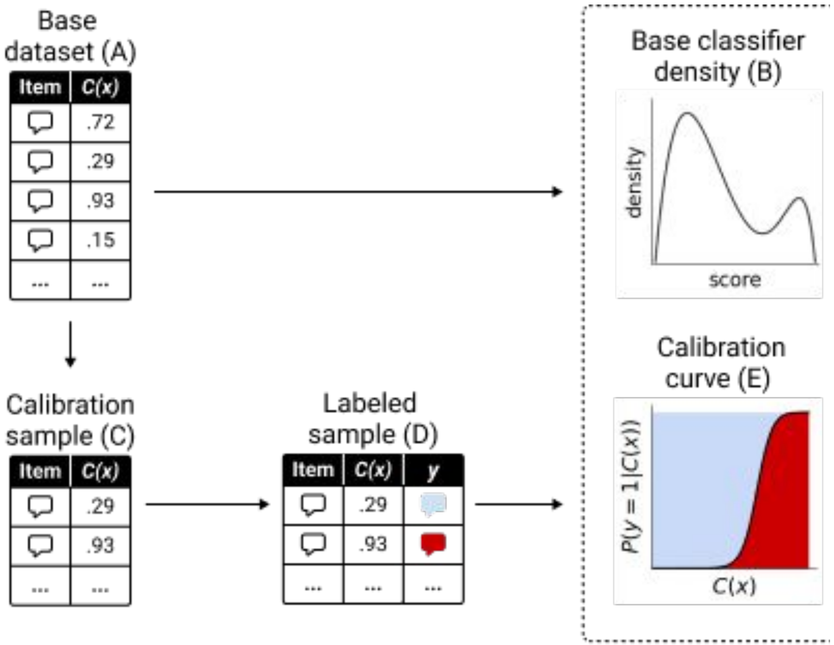
Base dataset (A)

Item	$C(x)$
	.72
	.29
	.93
	.15
...	...

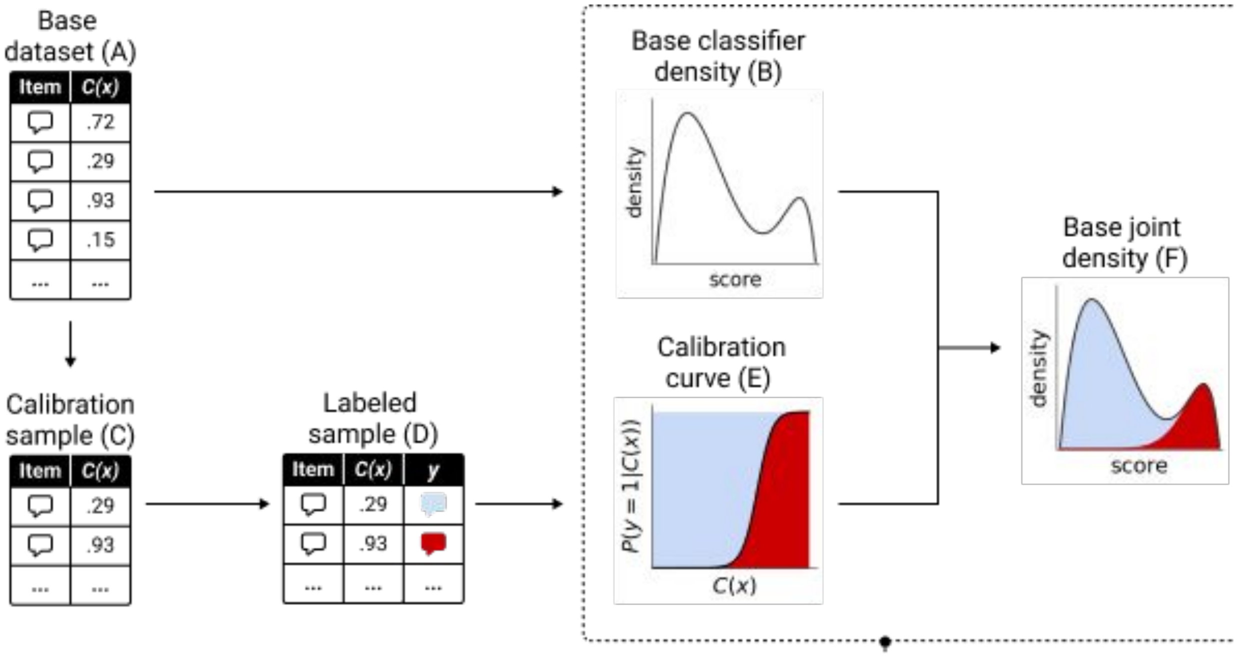
Base classifier density (B)



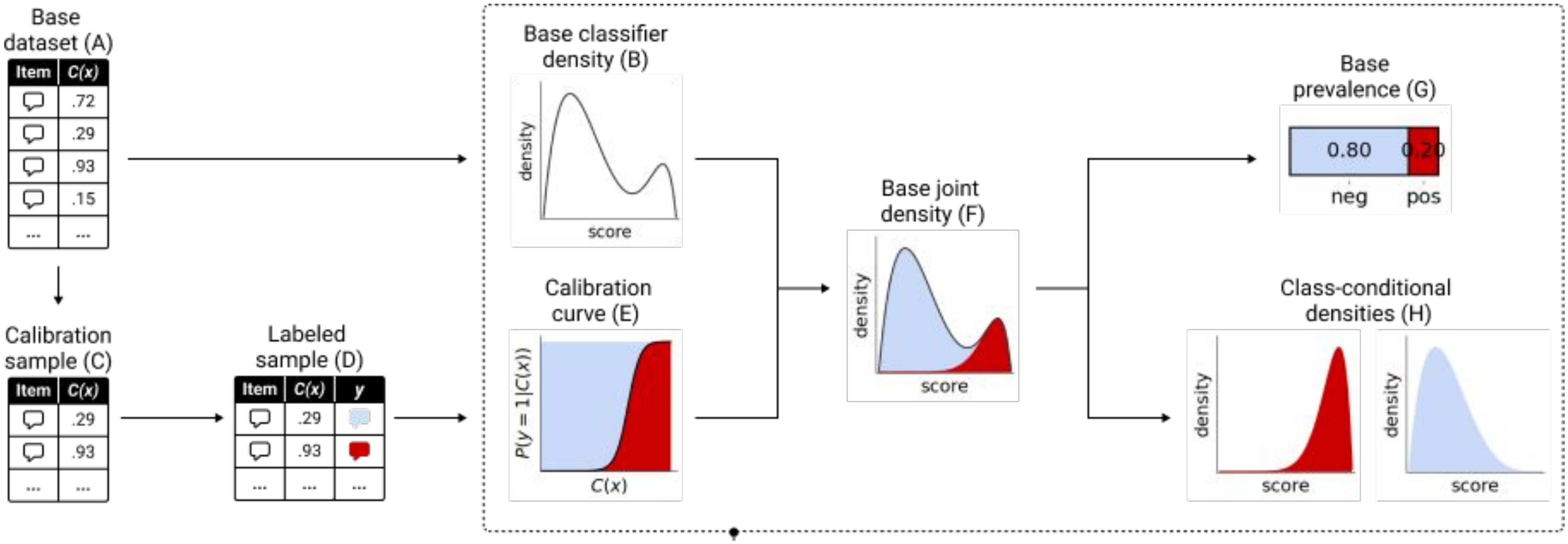
Calibration phase: A one-time prevalence estimate is needed for a single dataset



Calibration phase: A one-time prevalence estimate is needed for a single dataset



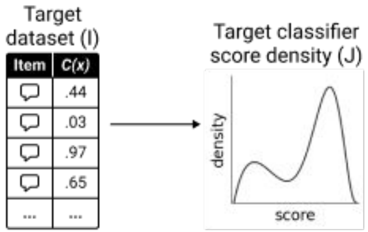
Calibration phase: A one-time prevalence estimate is needed for a single dataset



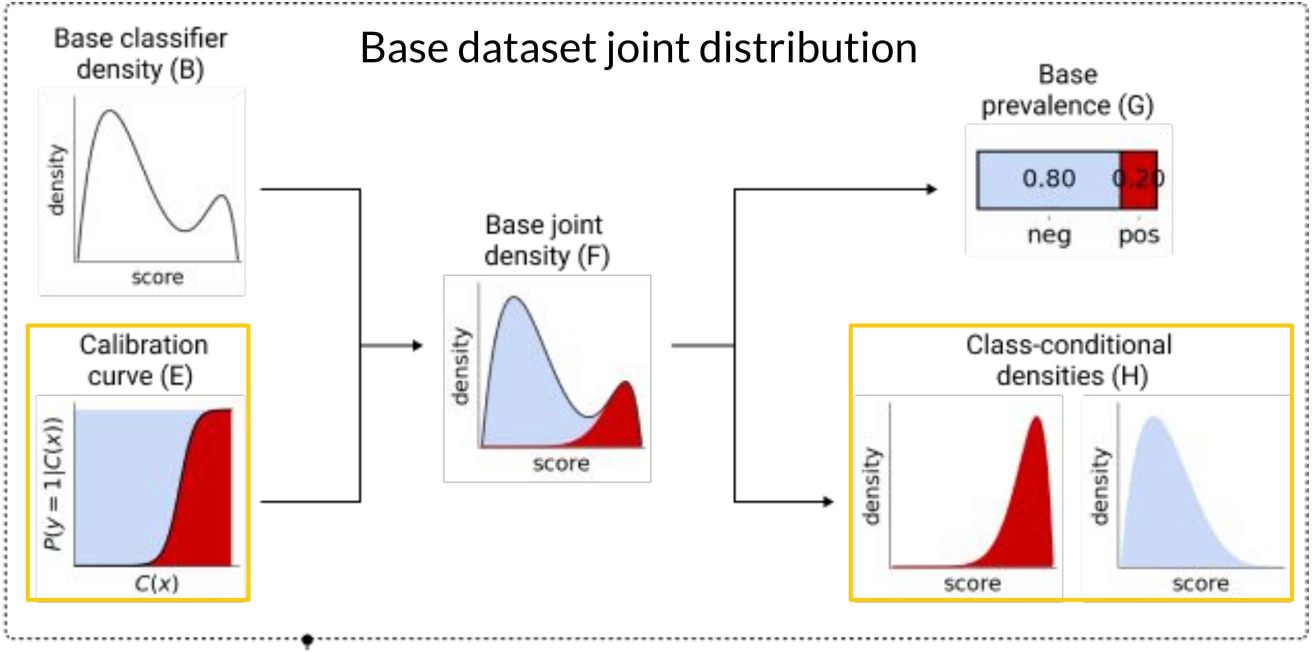
Notes and observations for the calibration phase

- Base dataset ⇔ Calibration sample, always assume stable calibration curve
- Use purposive sampling to increase the number of potential minority class
- What if we have a weak classifier? Unbiased estimate if repeated many times, but the CI will be wider

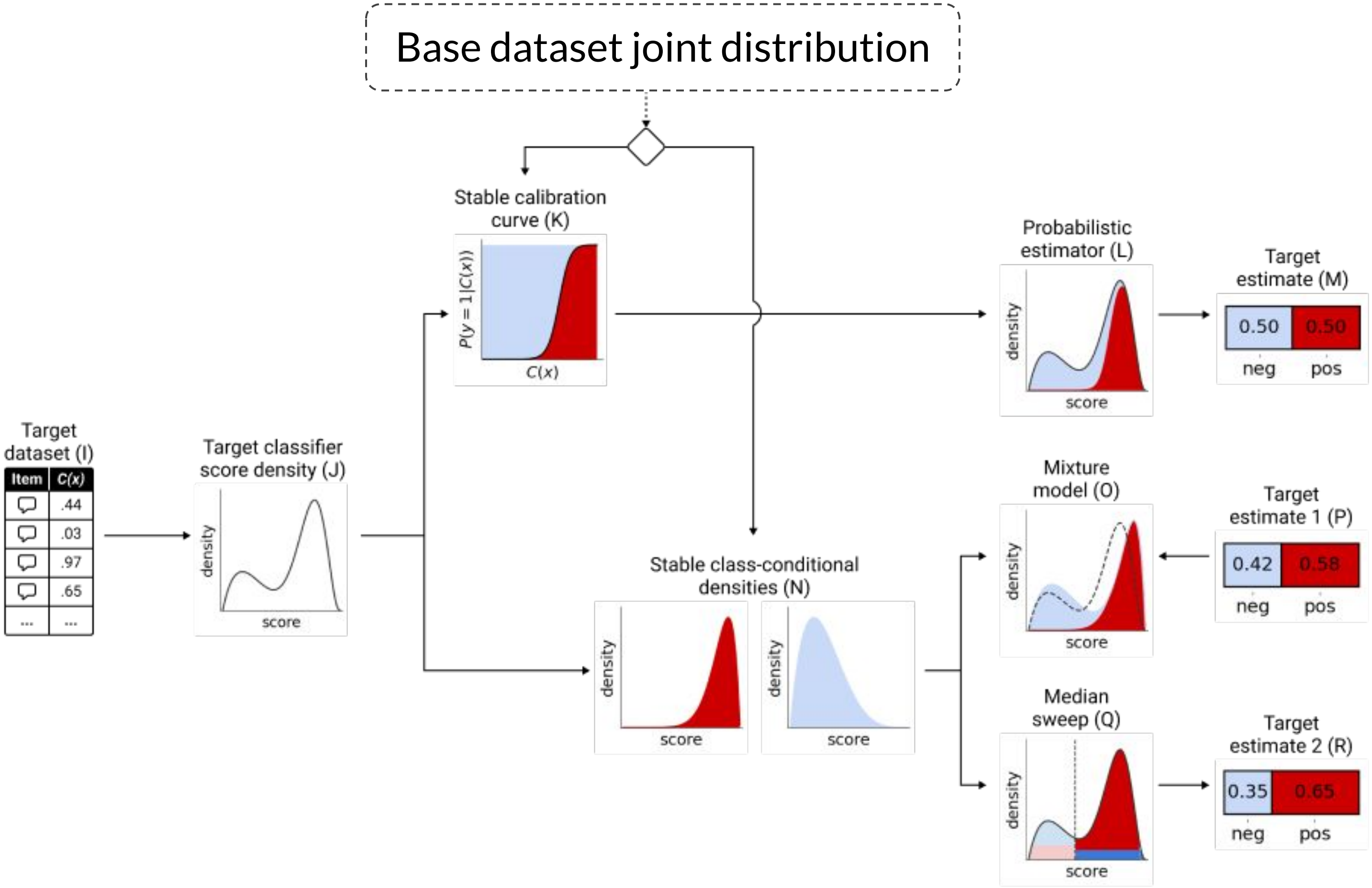
Extrapolation phase: Multiple estimates are needed for *related* datasets



Extrapolation phase: Borrowing from base dataset joint distribution



Extrapolation phase: Multiple estimates are needed for *related* datasets



Data generation, stability assumptions, and prevalence estimation techniques

Stability assumption	Stable attribute	Data generation	Causal chain	Prevalence estimation technique
Stable calibration curve	$P(GT\|C(X))$	Extrinsic	$GT \leftarrow X \rightarrow C(X)$	Probabilistic Classify and Count
Stable class-conditional densities	$P(C(X)\|GT)$	Intrinsic	$GT \rightarrow X \rightarrow C(X)$	Mixture model, Median sweep

Dallas Card, and Noah A. Smith. "The importance of calibration for estimating proportions from annotations." In *NAACL*. 2018.
Zhijing Jin, et al. "Causal Direction of Data Collection Matters: Implications of Causal and Anticausal Learning for NLP." In *EMNLP*. 2021.

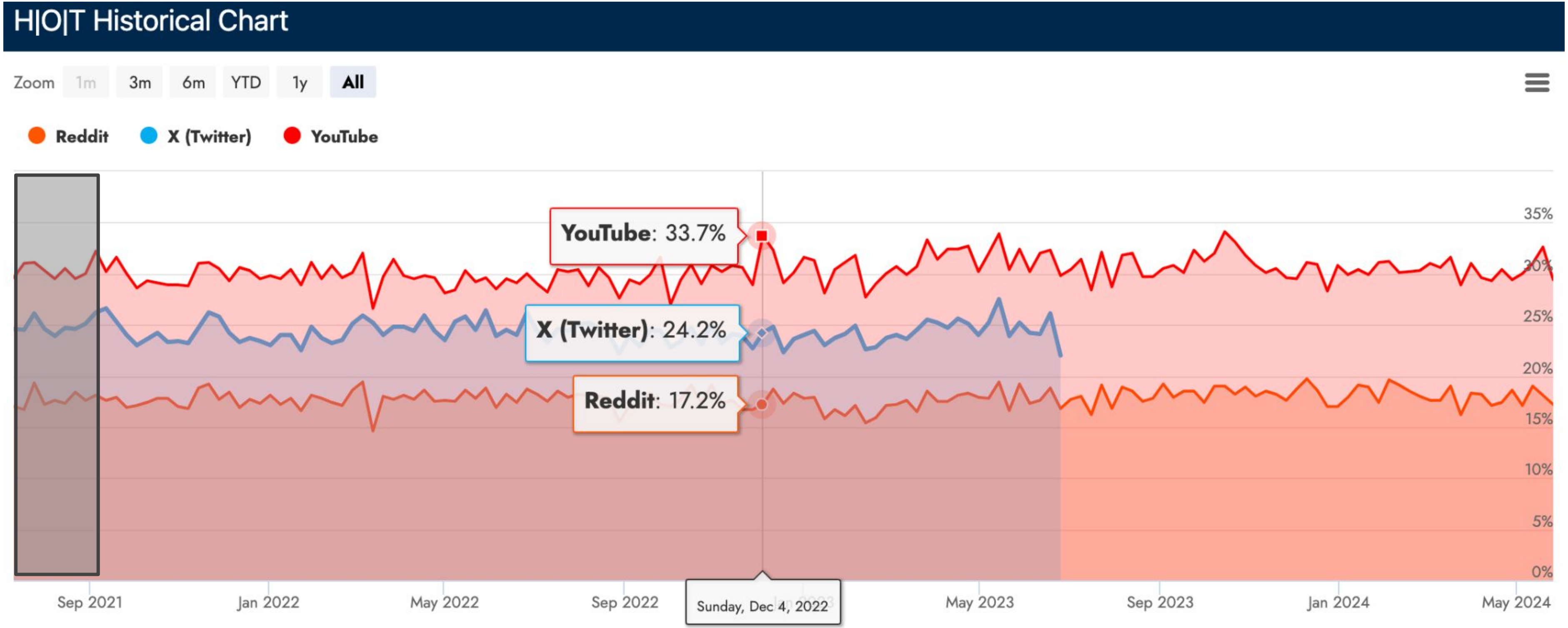
Notes and observations for the extrapolation phase

- Base dataset ⇒ Target dataset, choose stable calibration curve or stable class-conditional densities based on the data generation process
- What if we have a weak classifier? If we pick the correct stability assumption, the estimate will be fine. But a stronger classifier makes it more robust to wrong stability assumption.

Outline

1. Introduction
2. How to do prevalence estimation? Calibrate & Extrapolate
3. **Application: Estimating the fraction of H|O|T comments on news articles**
4. Practical advice for prevalence estimation

H|O|T speech: Fraction of hateful, offensive, or toxic comments on news articles



Project webpage: https://csmr.umich.edu/projects/hot-speech/

Practical advice for prevalence estimation

- Never safe to make a prevalence estimate based on a classifier trained on different datasets, without gathering human labels for calibration
- If a prevalence estimate is needed for a single dataset,
  - Balanced dataset → Random sample to annotate
  - Imbalanced dataset → Purposive sample to produce a calibration sample with more balanced labels
- If prevalence estimates are needed for multiple related datasets,
  - First estimate the joint distribution of a base dataset
  - Then borrow properties from base dataset joint distribution by making stability assumption based on the data generation process