Incomplete Label Uncertainty Estimation for Petition Victory Prediction with Dynamic Features

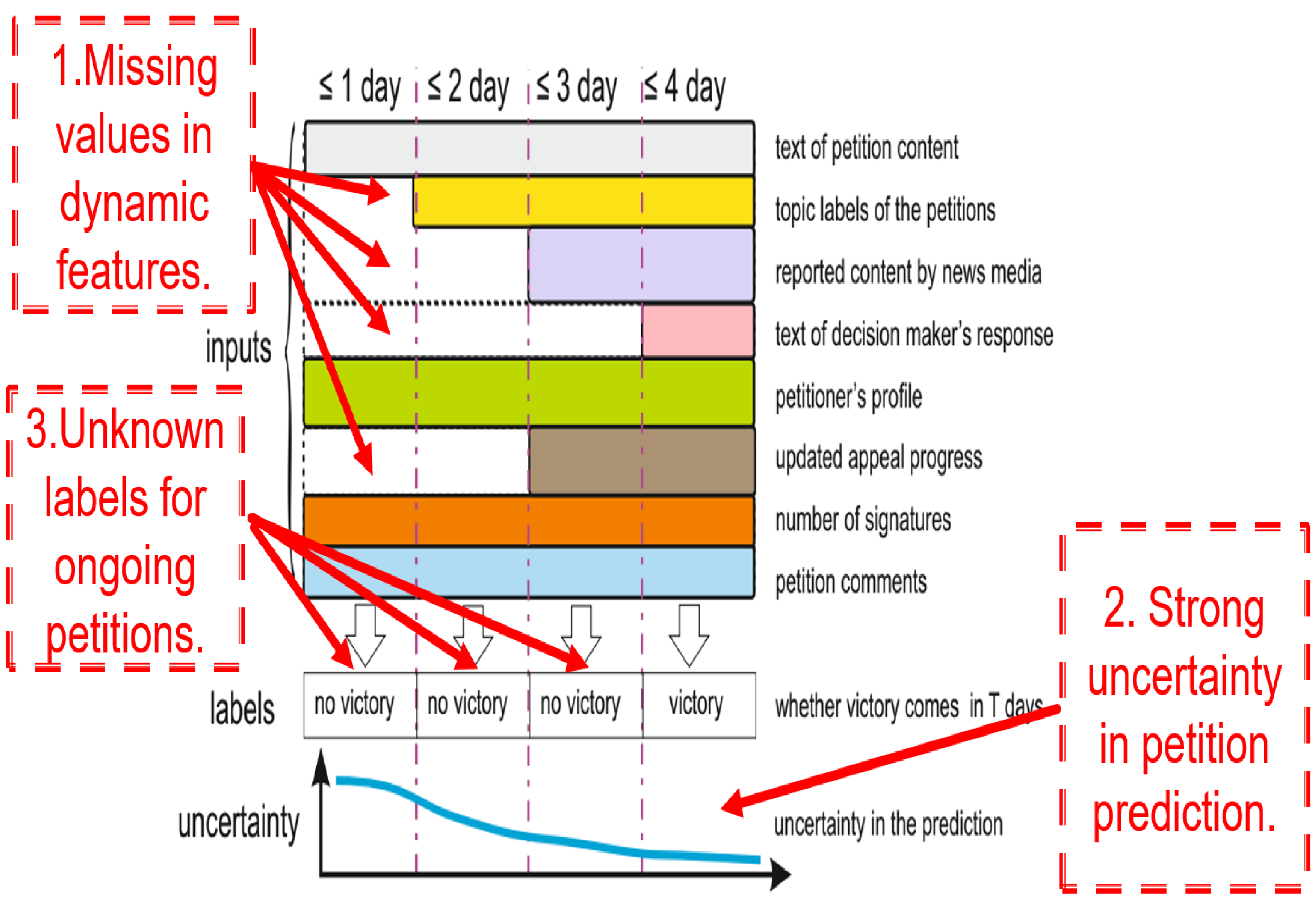
Junxiang Wang, Yuyang Gao, Andreas Zufle, Jingyuan Yang and Liang Zhao

An Example of Online Petitions

Petition Victory Prediction Problem

- A petition will be labeled as victorious if
  - (1). The required number of signatures is satisfied or
  - (2). The appeals of the petition launcher have been addressed by the decision-makers within a limited time interval.
- The petition victory prediction problem is formulated as
  - Given the petition vector $X_{i,t}$, the goal of this problem is to predict whether the i-th petition will succeed at time $t+\tau$ by learning the mapping $f: X_{i,t} \rightarrow Y_{i,t+\tau}$, where τ is the lead time.

Research Challenges

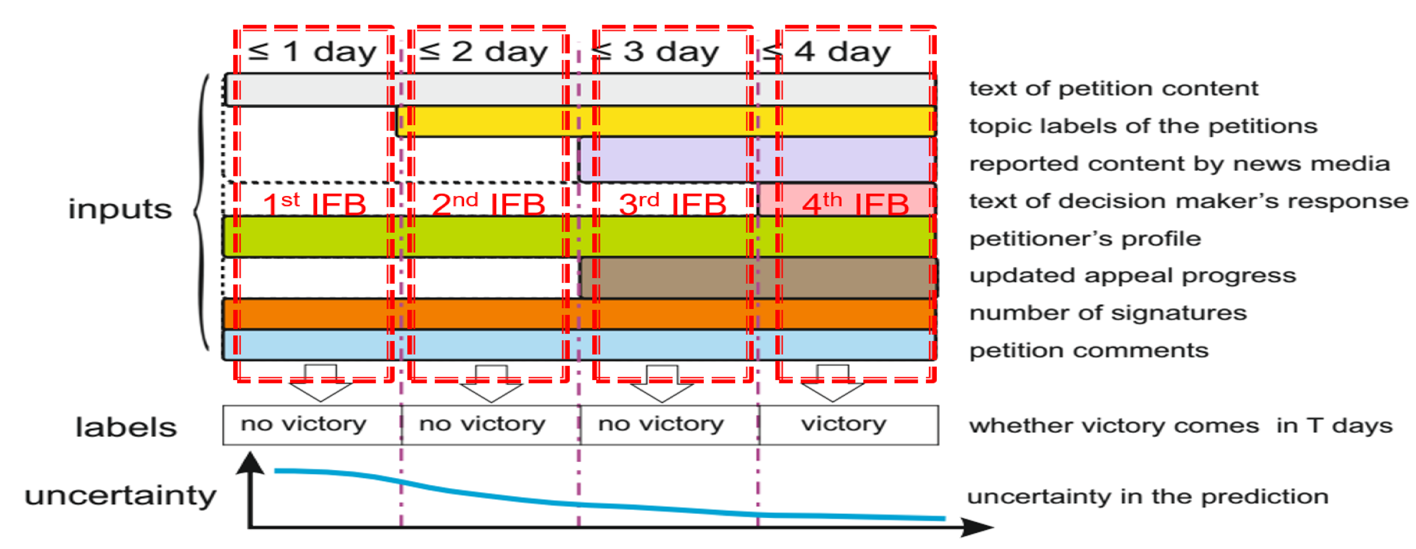


4. Scalability regarding increasing features and petitions.

- Computational scalability is challenging due to:
  - a fast growing number of online petitions;
  - an increasing number of features for each petition;
  - frequent updates of petition information.

Increasing Feature Block and Uncertainty Estimation

- Increasing Feature Block
  - An increasing feature block(IFB) is a block of petition sets that share the same available feature sets. All tasks are petitioned by



- Uncertainty Estimation
  - If the predicted label in IFB(j) is correct while that in IFB(j-1) is wrong, we **earn more certainty.**
  - If the predicted label in IFB(j) is wrong while that in IFB(j-1) is correct, we **lose more certainty**.

the accuracy earning of the classifier

$$earn(Y_{i,q}, Y_{i,p}, Y_{i,d_i}) = I(Y_{i,q} \neq Y_{i,d_i}) I(Y_{i,p} = Y_{i,d_i}) = (1 - Y_{i,q} Y_{i,d_i})(1 + Y_{i,p} Y_{i,d_i})/4$$

the accuracy losing of the classifier

$$lose(Y_{i,q}, Y_{i,p}, Y_{i,d_i}) = I(Y_{i,q} = Y_{i,d_i}) I(Y_{i,p} \neq Y_{i,d_i}) = (1 + Y_{i,q} Y_{i,d_i})(1 - Y_{i,p} Y_{i,d_i})/4$$

uncertainty function

$$R(Y_{i,q}, Y_{i,p}, Y_{i,d_i}) = lose(Y_{i,q}, Y_{i,p}, Y_{i,d_i}) - earn(Y_{i,q}, Y_{i,p}, Y_{i,d_i})$$

i: the i-th petition. d_i: the label time. $Y_{i,p} (p \in T_j)$: the predict label in IFB(j). $Y_{i,q} (q \in T_{j-1})$: the predict label in IFB(j-1). Y_{i,d_i}: the label of the i-th petition at time d_i.

Multi-task Learning with Uncertainty Estimation(MLUE)

Empirical loss Regularization term Uncertainty function

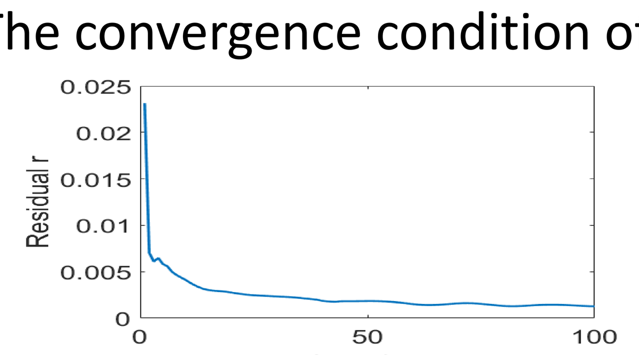
$$(Y, \beta^*, b^*) = \arg\min_{Y,\beta,b} \underbrace{Loss(Y;\beta,b)}_{\text{Empirical loss}} + \underbrace{\lambda_1 \Omega(\beta)}_{\text{Regularization term}} + \underbrace{\lambda_2 R(Y)}_{\text{Uncertainty function}}$$
$$\underbrace{s.t. \forall q \leq p, Y_{i,q} \leq Y_{i,p}}_{\text{Non-decreasing order}} \quad (1)$$

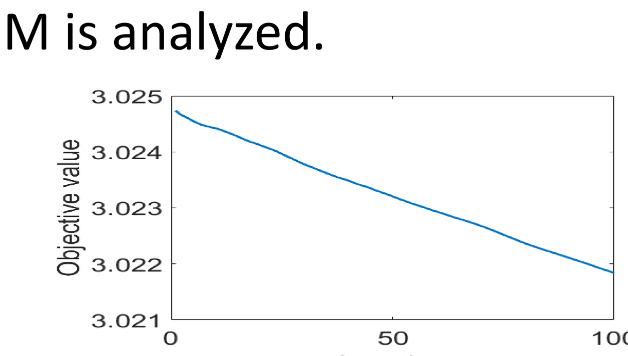
Non-decreasing order

β: the set of coefficients of all tasks.
b: the set of intercepts of all tasks.
Y: the set of all petition labels at any time(It contains known labels and unknown labels).

Nonconvex and discrete

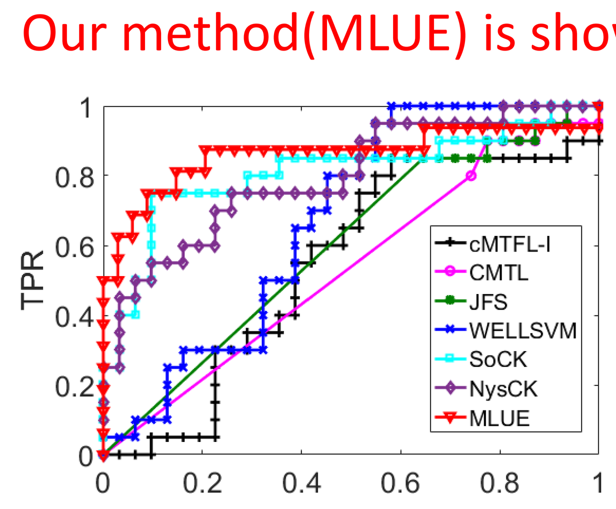
- we propose an Expectation-Maximization(EM)-like algorithm
  - E-step: update Y when fixing β and b.
  - M-step: update β and b when fixing Y.
- Updating Y: dynamic programming.
- Updating β and b: Alternating Direction Method of Multipliers(ADMM).
- The convergence condition of ADMM is analyzed.



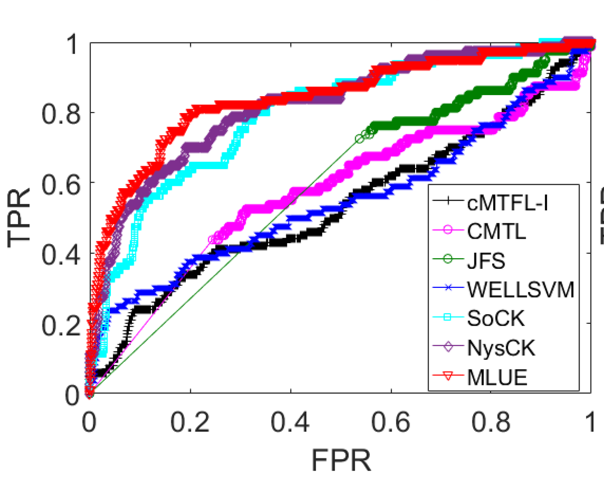


Experiments Results

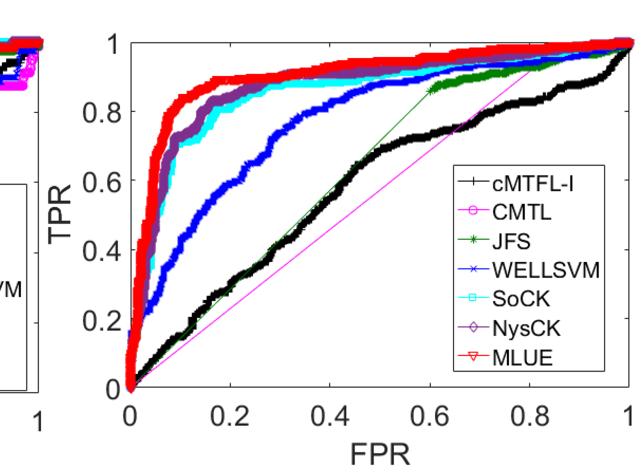
Our method(MLUE) is shown in the red curve, which covered other baselines.



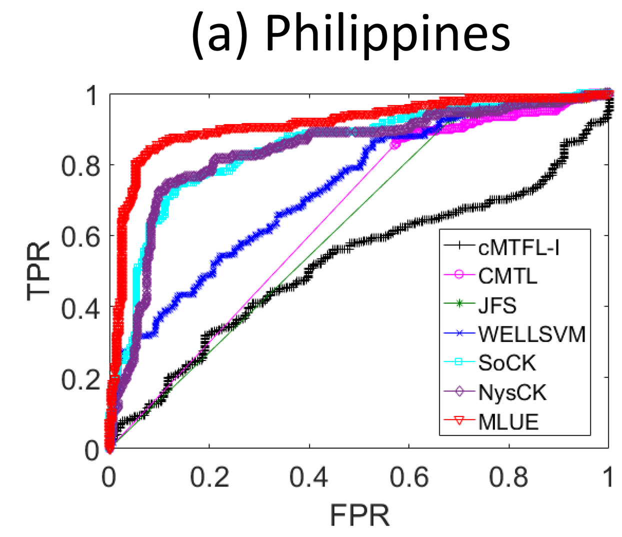
(a) Philippines



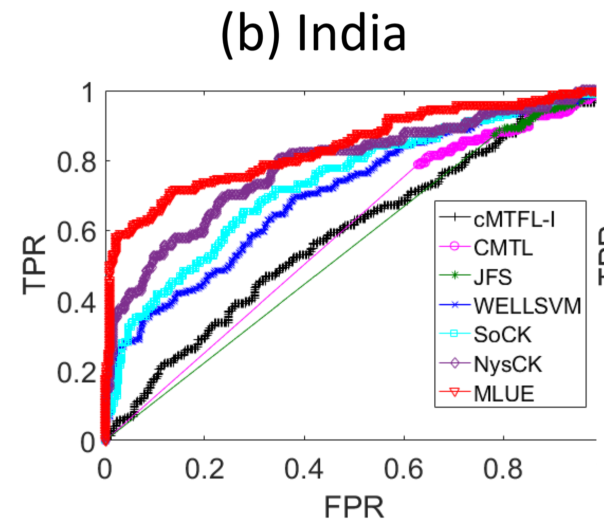
(b) India



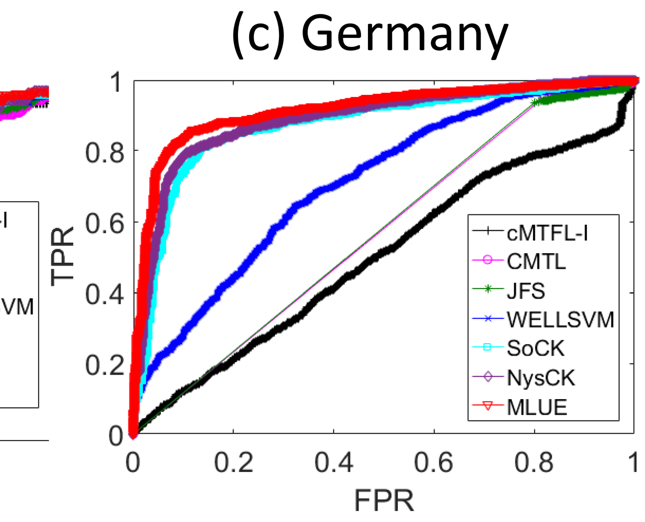
(c) Germany



(d) Australia



(e) Canada



(f) United States

The training time of our method(MLUE) increases linearly with number of features and petitions.