method

# LOPS: Learning Order Inspired Pseudo-Label Selection for Weakly Supervised Text Classification

issue

task

Advisor : Jia-Ling, Koh

Speaker : Ting-I, Weng

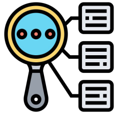
Source : ACL'22

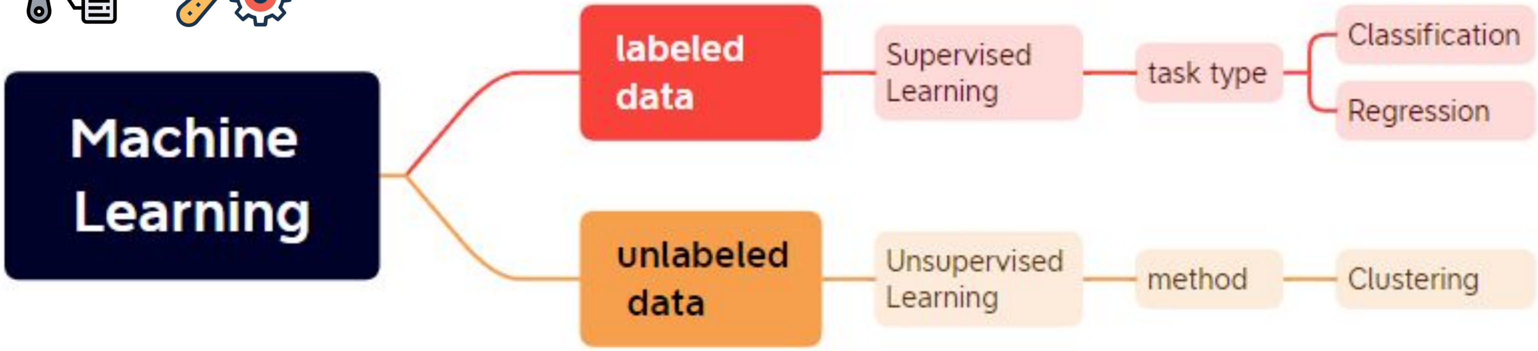
Date : 2023/10/17

# Outline

- Introduction
- Method
- Experiment
- Conclusion

# Machine Learning

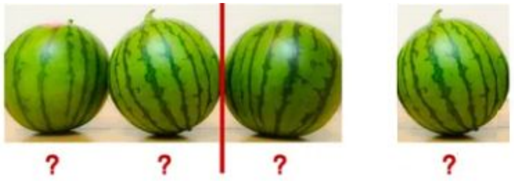


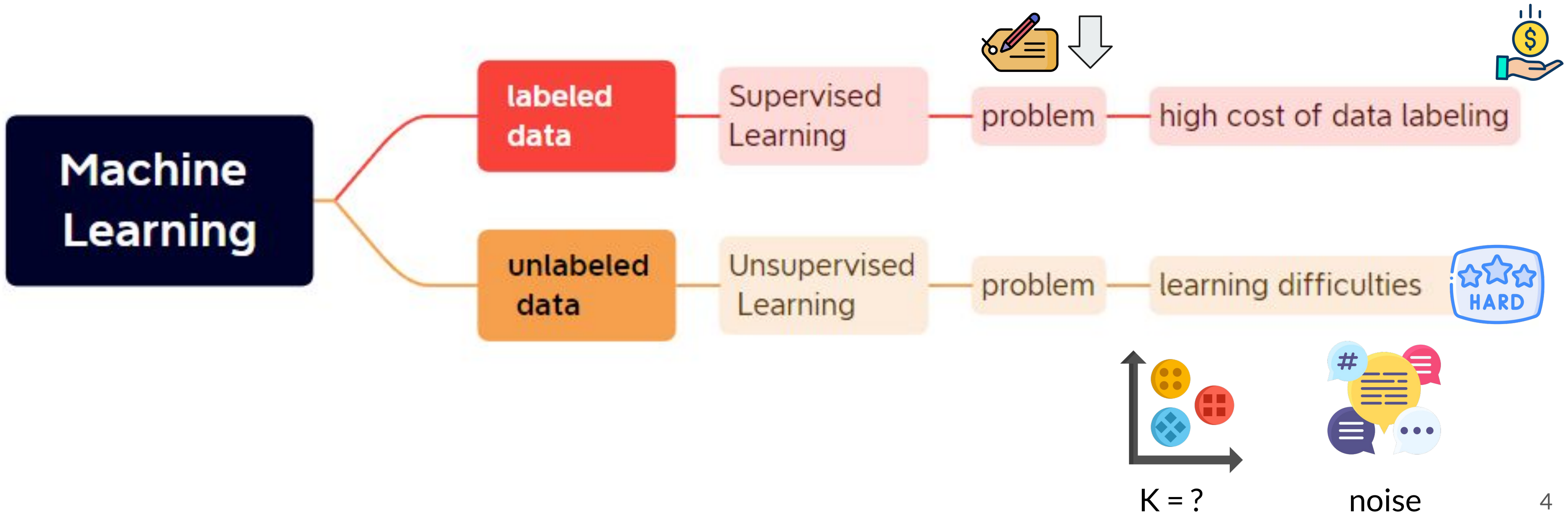

- **Classification problem**
  x : watermelons
  y : which class belongs to


- **Regression problem**
  x : watermelons
  y : real-value response

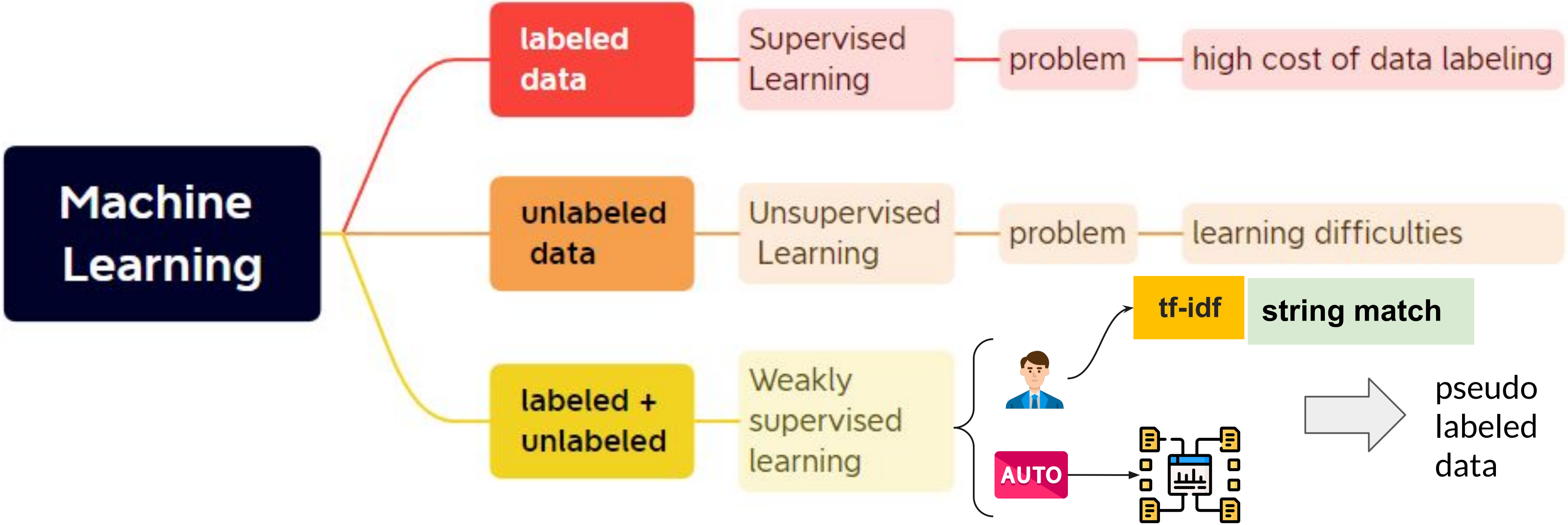

- **Clustering problem**
  x : watermelons
  y : ?



# Supervised & Unsupervised Learning Challenge

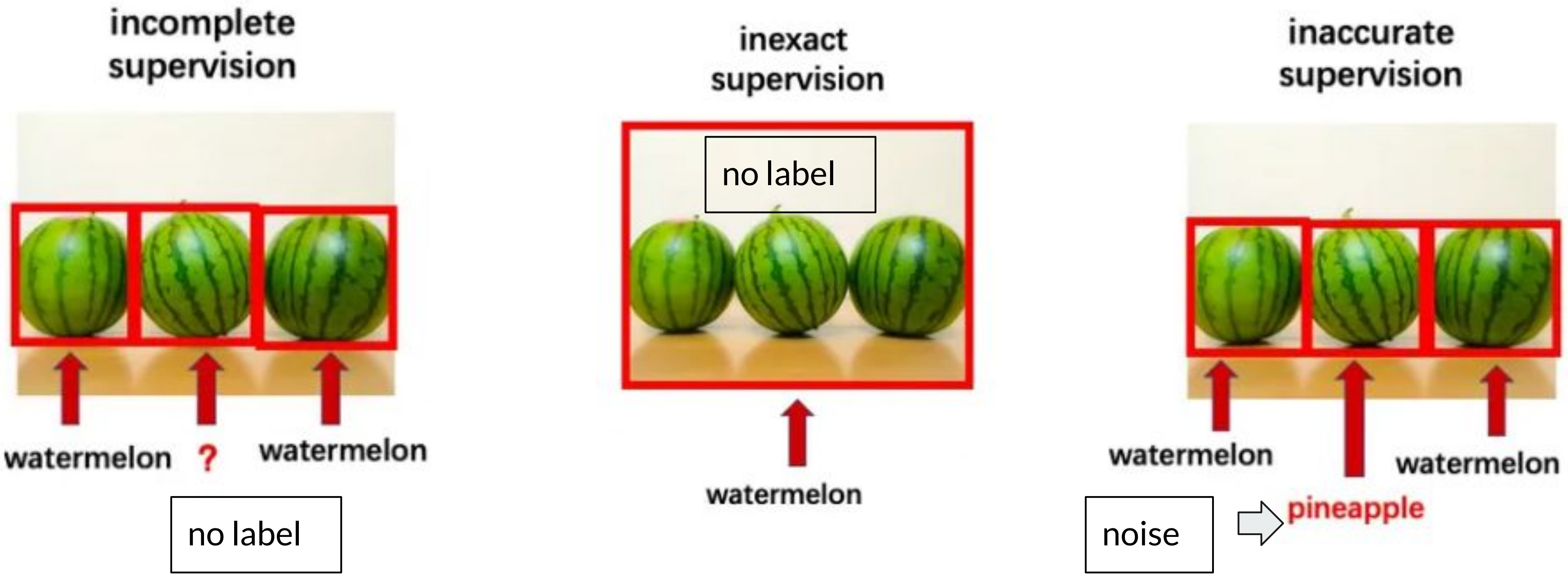


# Weakly supervised learning

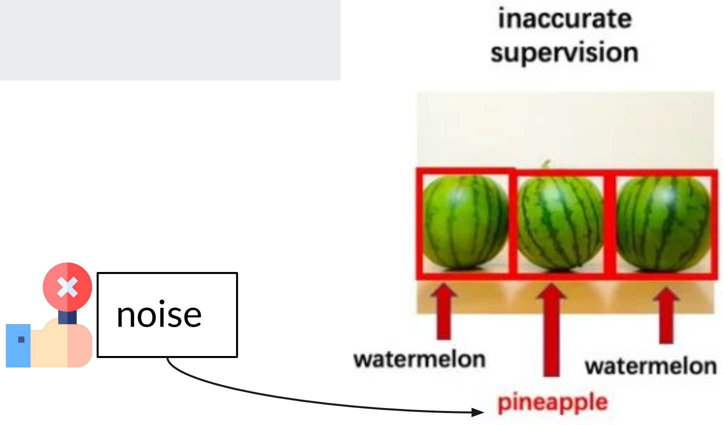


# Weakly supervised learning

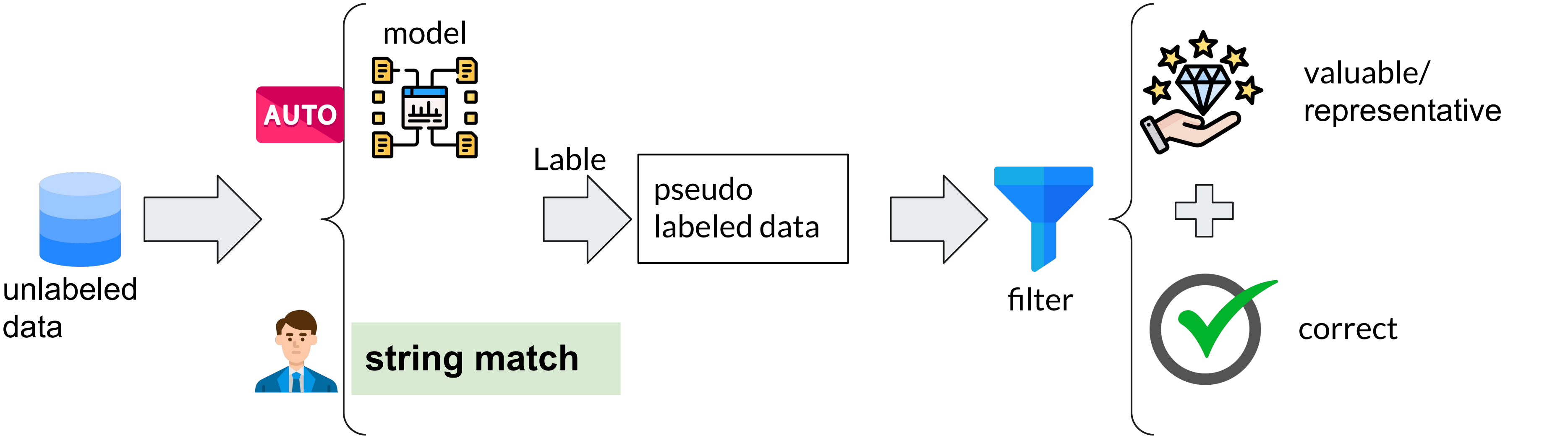
# Weakly supervised learning



# Task

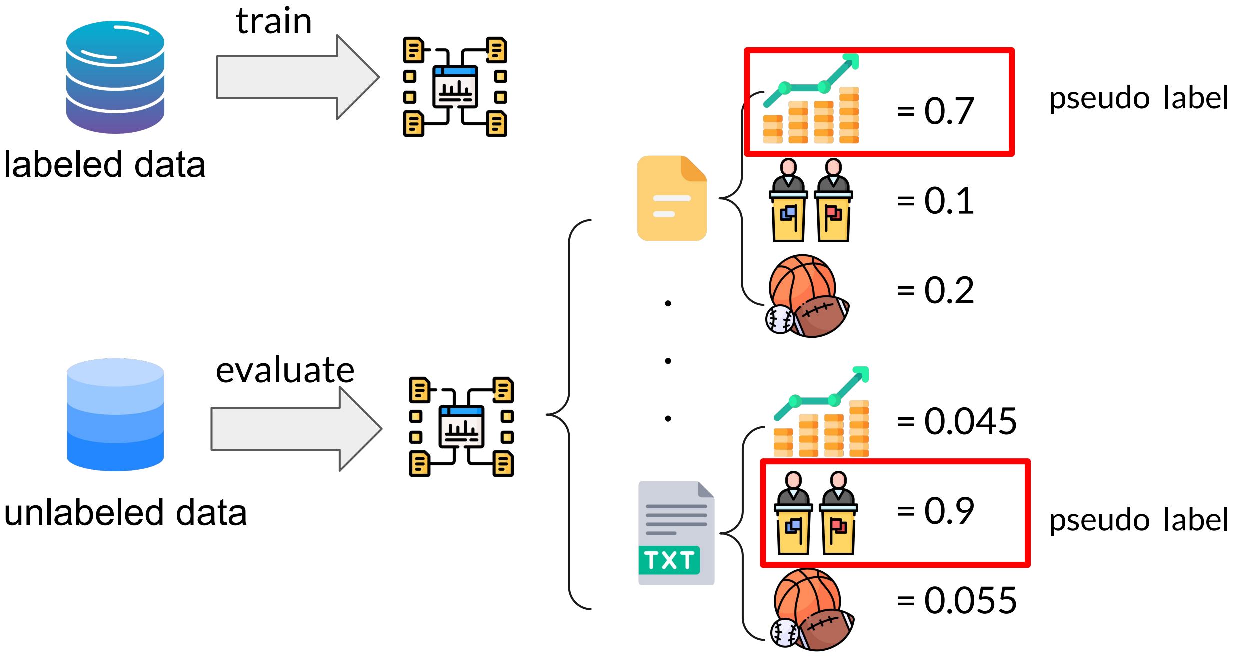


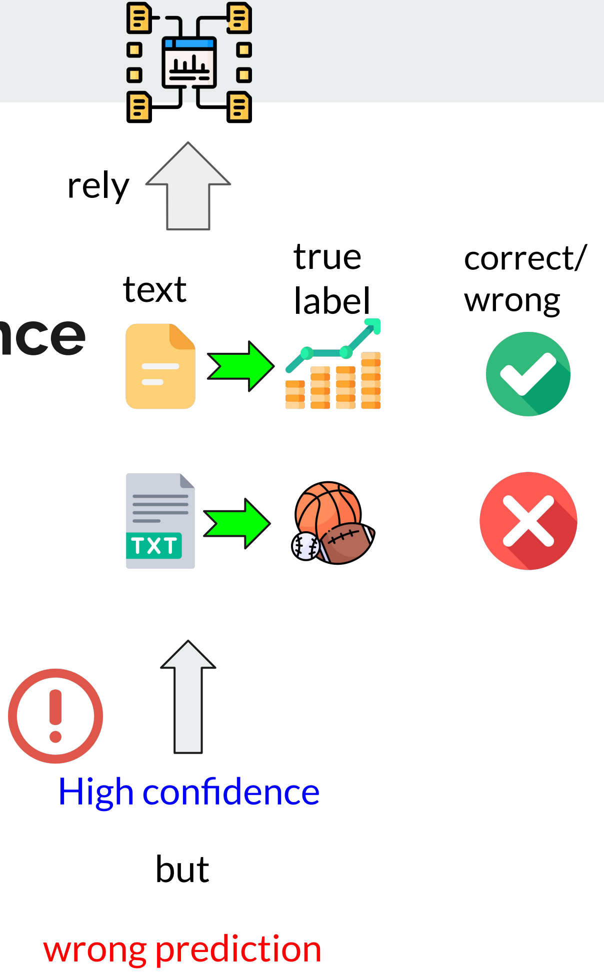
inaccurate
supervision
noise
watermelon
watermelon
pineapple

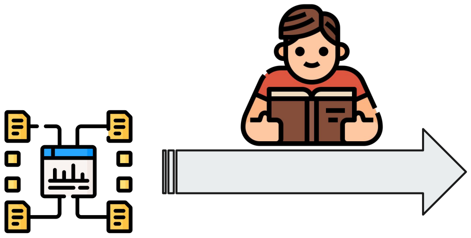
unlabeled
data
AUTO
model
string match
Lable
pseudo
labeled data
filter
valuable/
representative
correct

# straightforward solution:high confidence



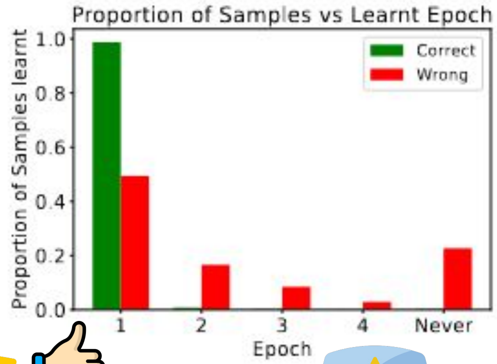



# related work



easy

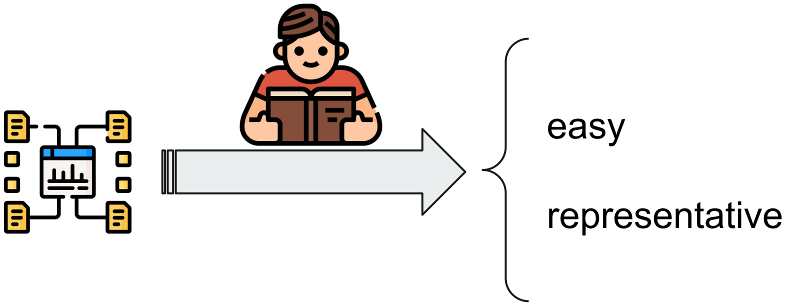
representative



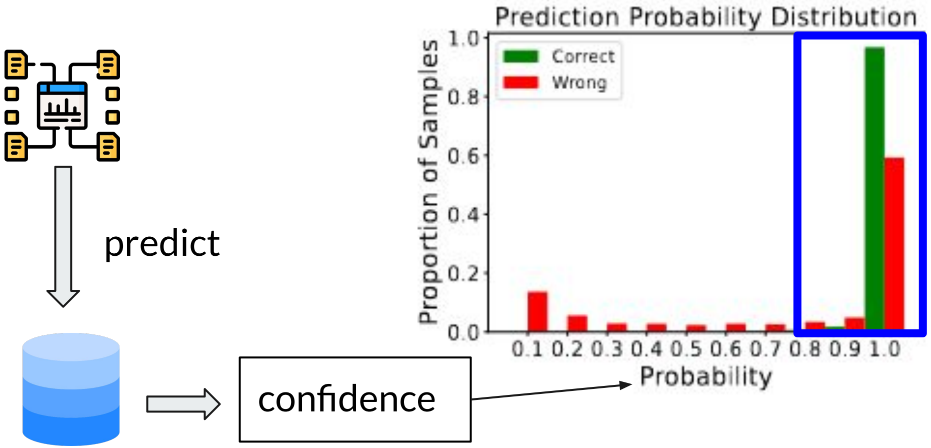


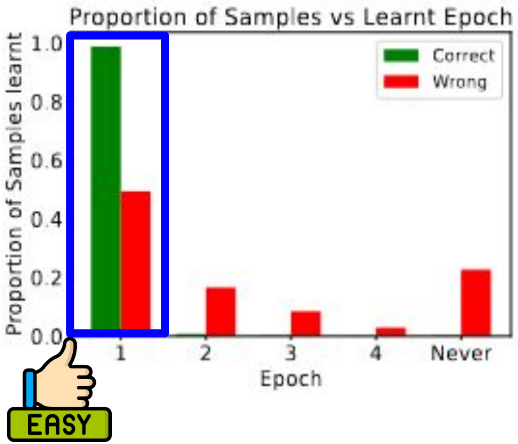

- epoch1
  - learn most of the representative instances
- other epoch
  - learn wrong instances

# hypothesize

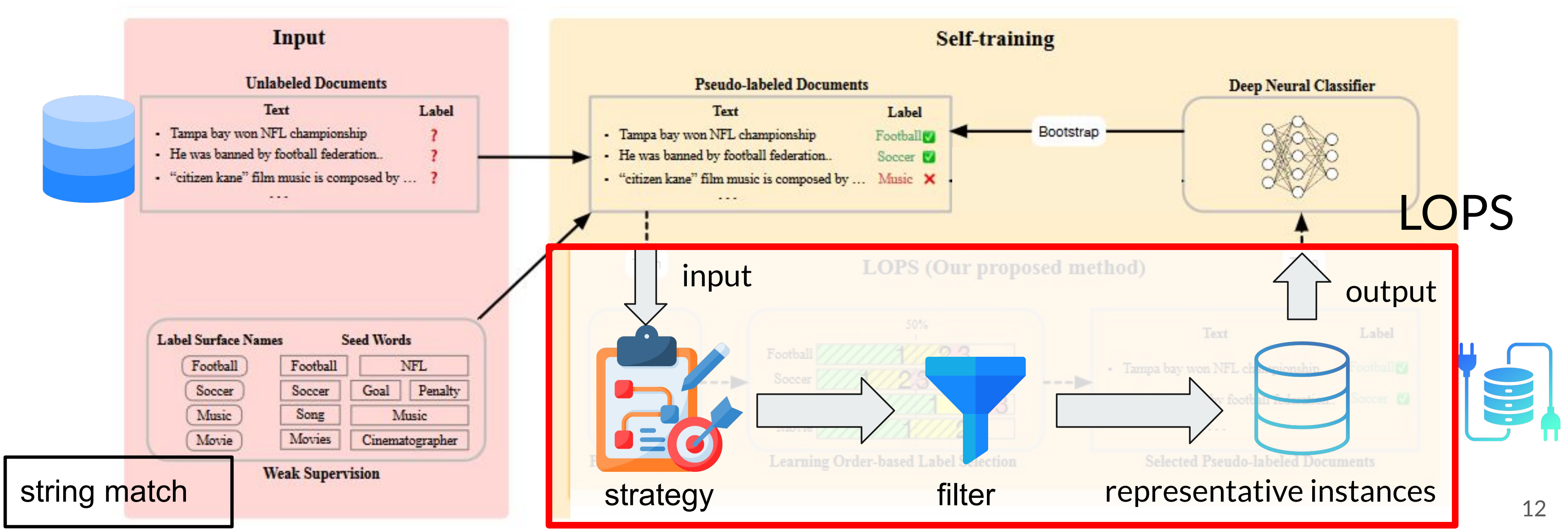



- learning order
  - be able to filter out most of wrongly labeled samples



# LOPS: Learning Order Inspired Pseudo-Label Selection



# Outline

- Introduction
- Method
- Experiment
- Conclusion

training phase

iteration=1(current)/10(total)

$$D = 1000 = D_{correct} + D_{wrong}$$

# Algorithm of LOPS

key word cannot be detected

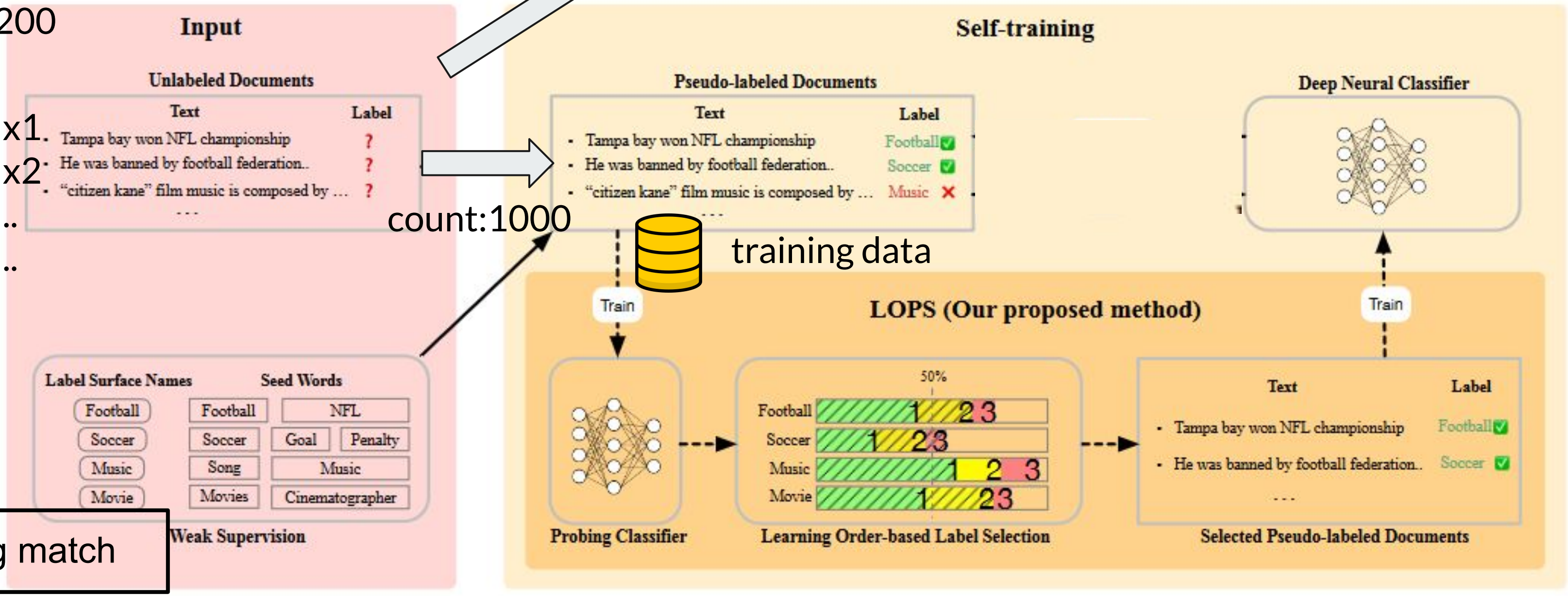
S{x}
count:1200

count:200

x1.
x2.
..
..

count:1000

training data




string match

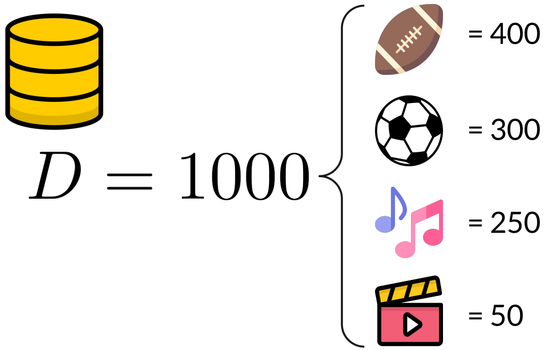
training phase

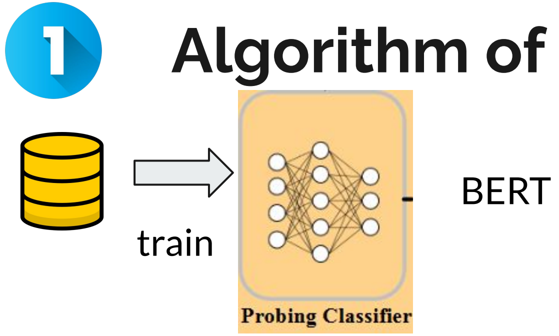
Epoch = 1(current)/10(total)

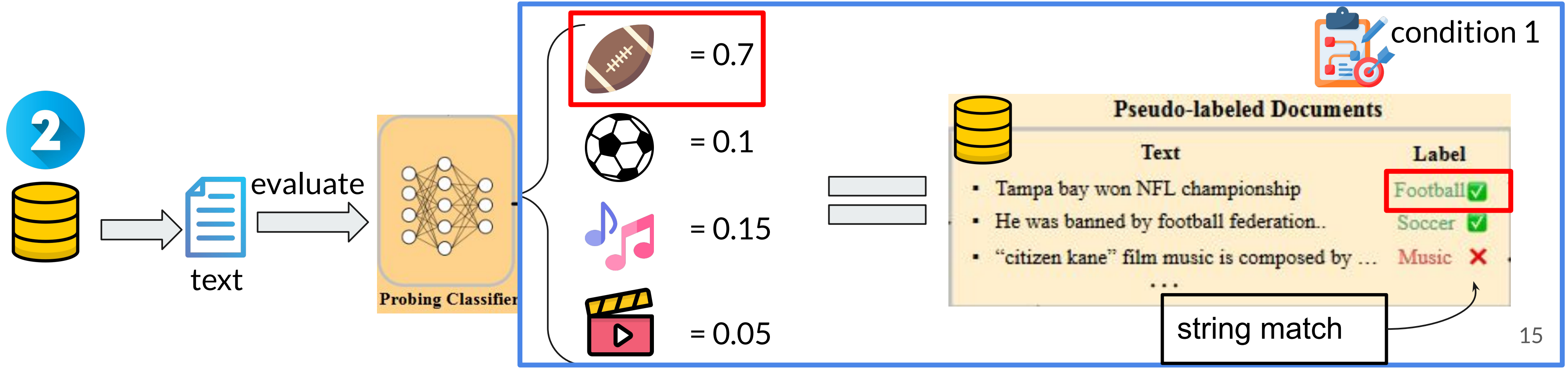
Tampa bay won NFL championship

text

# Algorithm of LOPS



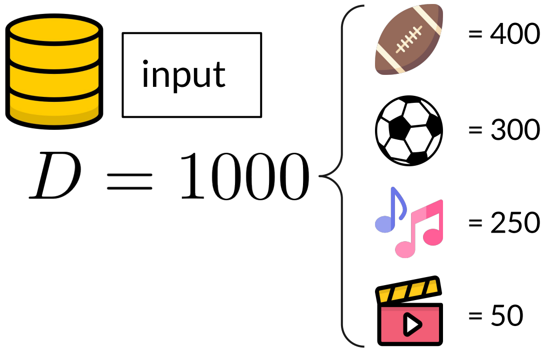



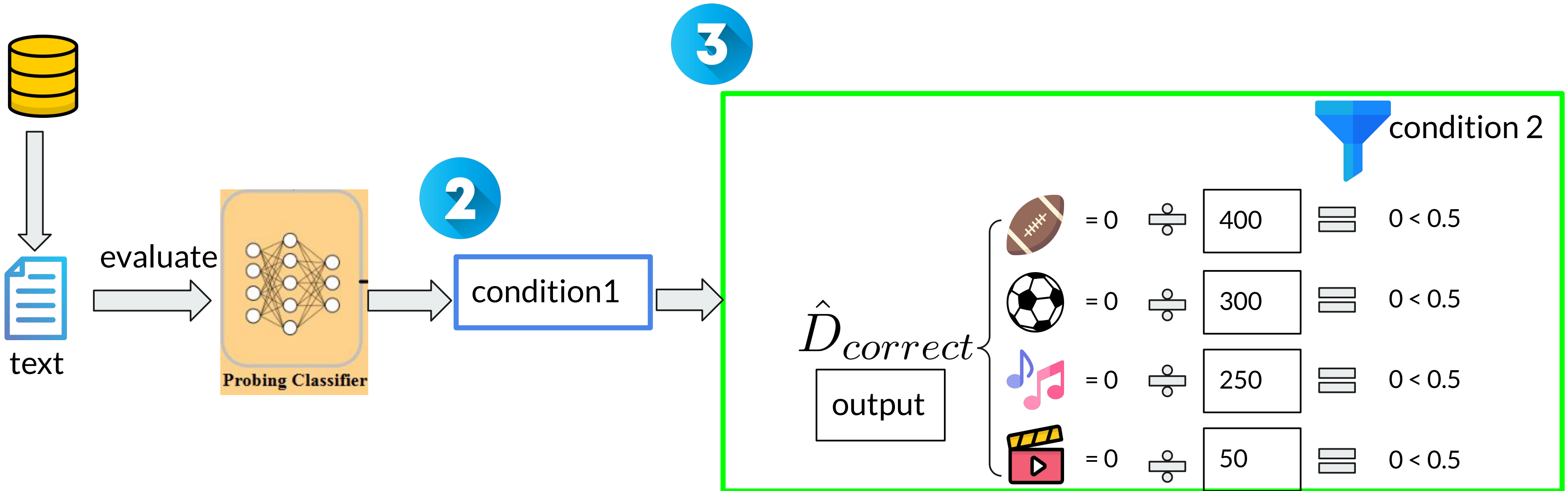



**training phase** $\tau = 50\%$

Epoch = 1(current)/10(total)

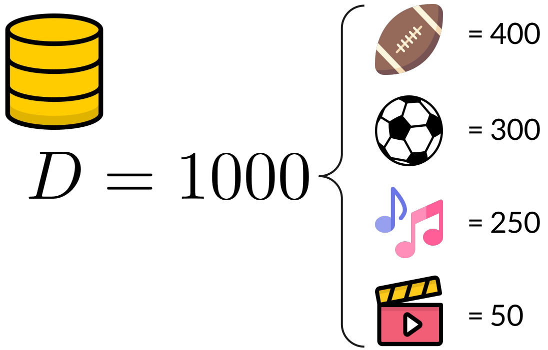
# Algorithm of LOPS

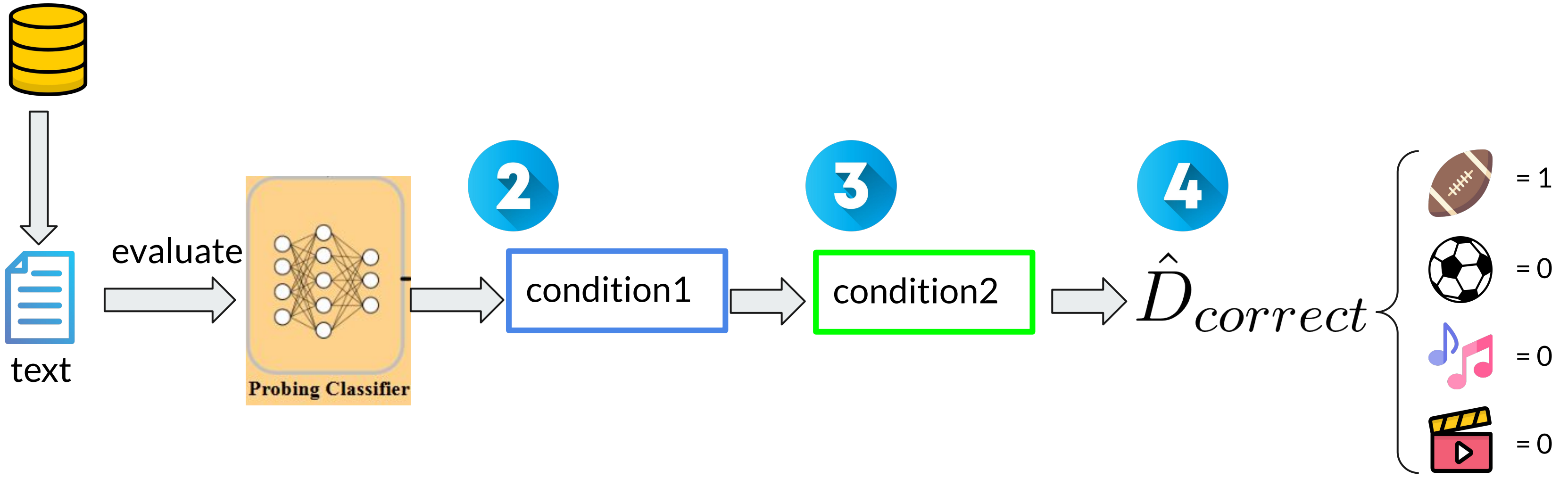


Epoch = 1(current)/10(total)

# Algorithm of LOPS

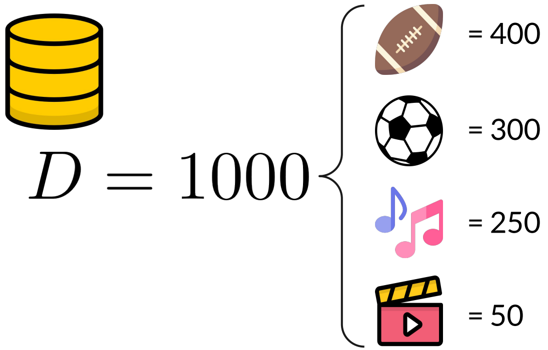


**training phase**

$\tau = 50\%$

Epoch = 3(current)/10(total)

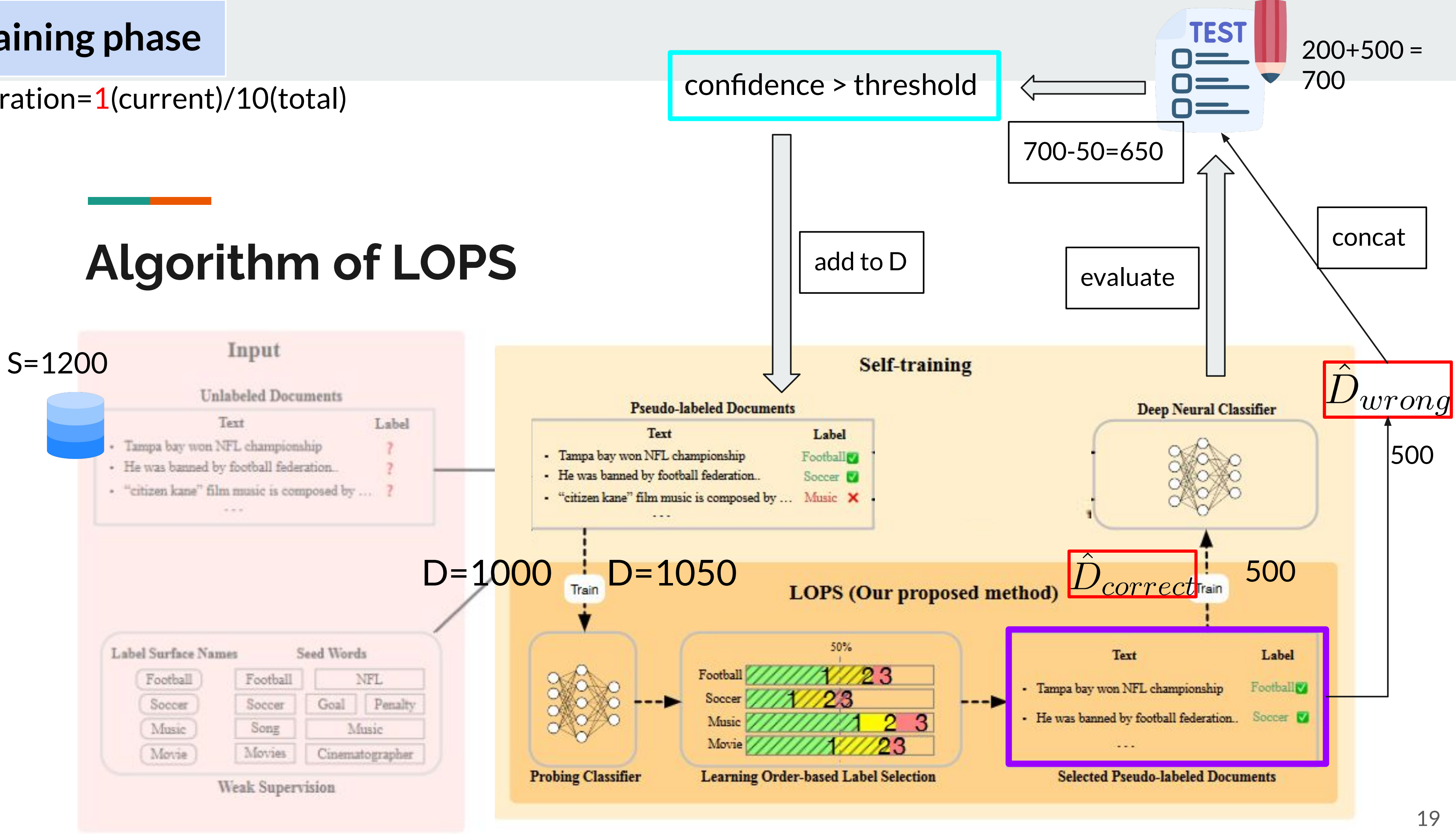



# Algorithm of LOPS

# training phase

iteration=1(current)/10(total)

## Algorithm of LOPS



Epoch = 1(current)/10(total)

text → string match = pseudo labeled data

w(x)

# learning order

T = 10

t = 2 $\eta = 0.8$

t = 3 $\eta = 0.7$

t = 4 $\eta = 0.6$

The higher the better

learning order

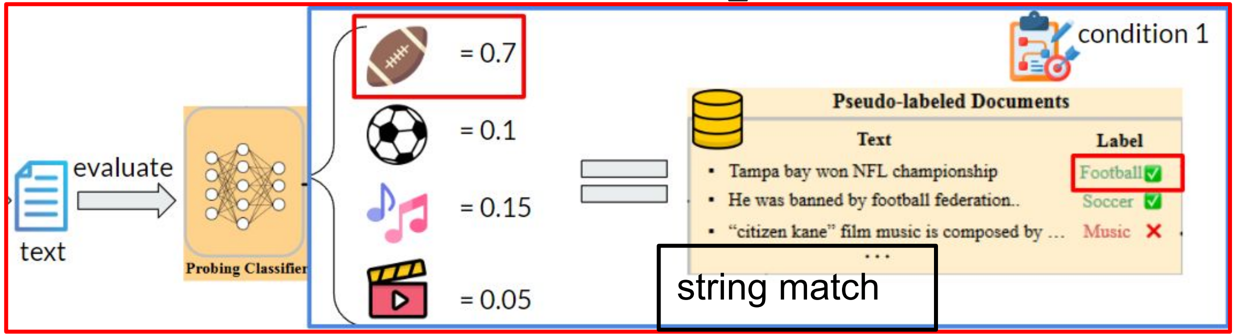
pseudo label

current epoch

$$\eta(x, w(x)) = 1 - \frac{1}{T}\min\{t \mid \boxed{\arg\max_j \mathbf{f}^t(x)[j] = w(x)}\},$$

total epochs

text



# Outline

- Introduction
- Method
- Experiment
- Conclusion

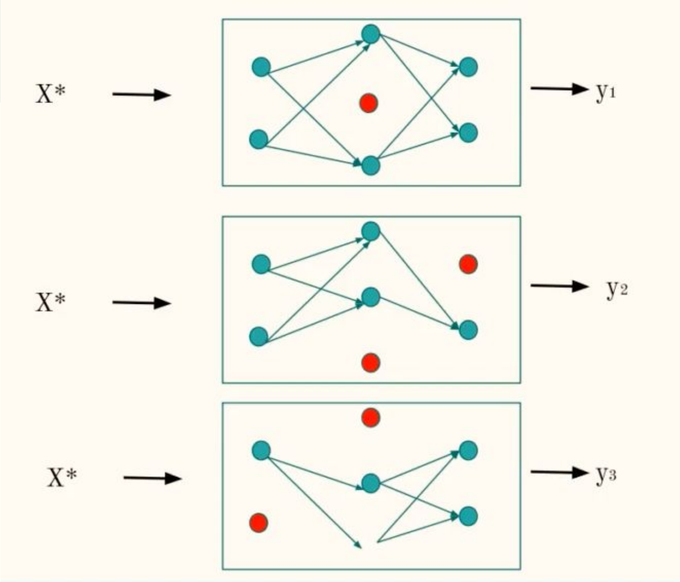
# Dataset

string match

initial pseudo labels

- New York Times
  - science、sports、music..
- 20Newsgroups
  - computers、baseball...
- AGNews
  - business、sports...

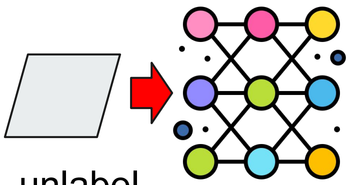
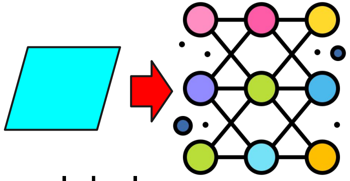
| Dataset | # Docs | # labels | Noise Ratio(%) |
|---|---|---|---|
| NYT-Coarse | 13,081 | 5 | 11.47 |
| NYT-Fine | 13,081 | 26 | 31.80 |
| 20News-Coarse | 17,871 | 5 | 12.50 |
| 20News-Fine | 17,871 | 17 | 25.67 |
| AGNews | 120,000 | 4 | 16.26 |
| Books | 33,594 | 8 | 37.32 |



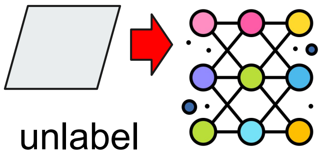
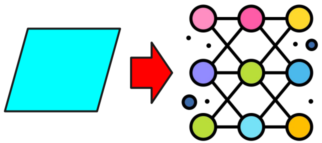

# baseline - label selection methods

- Entropy
  - uses entropy to compute uncertainty scores
- Probability
  - use the prediction probabilities corresponding to pseudo-labels in descending orde and select the number of samples
- Random
  - is similar to Probability, however use random select the samples
- Monte-Carlo Dropout (MC-Dropout)
  - Uncertainty estimates for probability score calculations

# Query-Strategy - Least Confident

| | $\hat{y} = argmax_y P_\theta(y\|x)$ | $x^*_{LC} = \underset{x}{argmax}\ 1 - P_\theta(\hat{y}\|x)$ |
|---|---|---|
|  unlabel; class A: 0.93<br>class B: 0.05<br>class C: 0.02 | $\hat{y} = 0.93$ | $x^*_{LC} = 1 - 0.93 = 0.07$ |
|  unlabel; class A: 0.55<br>class B: 0.35<br>class C: 0.1 | $\hat{y} = 0.55$<br>Confident ⇩ uncertain ⇧ | $\boxed{x^*_{LC} = 1 - 0.55 = 0.45}$ |

# Query-Strategy - Entropy

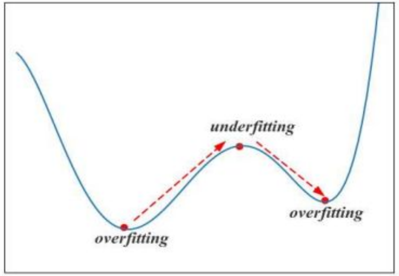
| | | $\log P_\theta(y_i \mid x)$ | $P_\theta(y_i \mid x) \log P_\theta(y_i \mid x)$ | $x_E^* = \underset{x}{\arg\max} - \sum_i P_\theta(y_i \mid x) \log P_\theta(y_i \mid x)$ |
|---|---|---|---|---|
|  unlabel | class A: 0.93<br>class B: 0.05<br>class C: 0.02 | class A: -0.104<br>class B: -4.321<br>class C: -5.6438 | class A: -0.09672<br>class B: -0.21605<br>class C: -0.11287 | -(0.09672+0.21605+ 0.11287) = -0.4256<br>$x_E^* = -(-0.4256) = 0.4256$ |
|  unlabel | class A: 0.55<br>class B: 0.35<br>class C: 0.1 | class A: -0.8624<br>class B: -1.5145<br>class C: -3.3219 | class A: -0.47432<br>class B: -0.53007<br>class C: -0.33219 | -(0.47432+0.53007+ 0.33219) = -1.33658<br>$\boxed{x_E^* = -(-1.33658) = 1.33658}$ |

entropy ↑ uncertain ↑

https://ieeexplore.ieee.org/document/9008796

first learn easy sample,

then learn difficult sample




# baseline - overfitting to underfitting (O2U Net)

1. pre-training

$$D = 1000 = D_{correct} + D_{wrong}$$

train

fixed learning rate

2. cyclical training

train

- compute loss of every sample
- update learning rate
- remove topK% samples with large loss from D

3. training on clean data

D’=500

train

model

# baseline - Learning Stability

Learning Stability

model

current epoch

perturbation

$$s(x; f, \omega) = \frac{1}{T}\left|\{t \mid f(t, x+\delta) = y(x),\ t \in \{1, \cdots, T\}\}\right|,$$

text

optimizer

total epochs

true label

T = 10

t = 2 $\Rightarrow$ s = 0.2

t = 5 $\Rightarrow$ s = 0.5

t = 10 $\Rightarrow$ s = 1

HIGH QUALITY

- metric
  - Micro-F1
  - Macro-F1

$$D = 1000 = D_{correct} + D_{wrong}$$

# Experiment

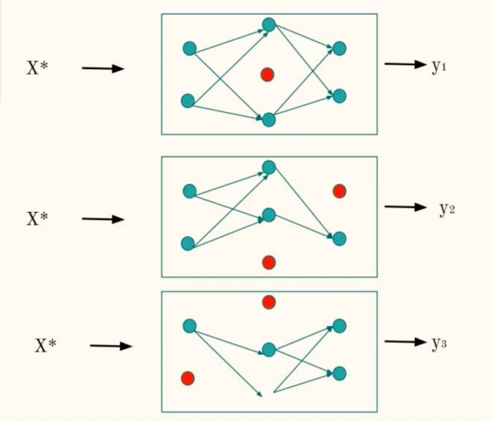
| Classifier | Method | Coarse-grained Datasets | | | | | | | | Fine-grained Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NYT-Coarse | | 20News-Coarse | | AGNews | | Books | | NYT-Fine | | 20News-Fine | |
| | | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 |
| BERT | Standard | 90.1(0.17) | 80.3(0.91) | 77.3(0.27) | 76.4(0.76) | 75.4(0.64) | 75.4(0.47) | 55.7(0.54) | 57.9(0.82) | 77.2(0.36) | 71.6(0.43) | 70.0(0.30) | 69.6(0.25) |
| | LOPS | **94.6(0.36)** | **88.4(0.50)** | **81.7(1.00)** | **80.7(0.43)** | **79.5(0.86)** | **79.5(0.58)** | **57.7(0.87)** | **59.5(0.46)** | **84.3(0.54)** | **81.6(0.34)** | **73.8(0.61)** | **72.7(1.00)** |

- standard : all data with noisy
- LOPS : strategically select representative samples

# Experiment

| | | Coarse-grained Datasets | | | | | | | | Fine-grained Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NYT-Coarse | | 20News-Coarse | | AGNews | | Books | | NYT-Fine | | 20News-Fine | |
| Classifier | Method | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 |
| BERT | LOPS | **94.6(0.36)** | **88.4(0.50)** | **81.7(1.00)** | **80.7(0.43)** | **79.5(0.86)** | **79.5(0.58)** | **57.7(0.87)** | **59.5(0.46)** | **84.3(0.54)** | **81.6(0.34)** | **73.8(0.61)** | **72.7(1.00)** |
| | Random | 90.3(0.47) | 80.9(0.47) | 79.0(1.00) | 76.8(1.50) | 76.3(0.35) | 76.3(0.65) | 56.1(0.18) | 58.2(0.35) | 78.4(0.94) | 71.7(0.47) | 71.4(0.50) | 70.6(1.00) |

- LOPS are strategic




# Experiment

| Classifier | Method | Coarse-grained Datasets | | | | | | | | Fine-grained Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NYT-Coarse | | 20News-Coarse | | AGNews | | Books | | NYT-Fine | | 20News-Fine | |
| | | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 |
| BERT | LOPS | **94.6(0.36)** | **88.4(0.50)** | **81.7(1.00)** | **80.7(0.43)** | **79.5(0.86)** | **79.5(0.58)** | **57.7(0.87)** | **59.5(0.46)** | **84.3(0.54)** | **81.6(0.34)** | **73.8(0.61)** | **72.7(1.00)** |
| | MC-Dropout | 89.3(0.41) | 79.3(0.45) | 80.7(0.17) | 77.7(0.24) | 75.8(0.34) | 75.0(0.41) | 55.1(0.15) | 56.7(0.61) | 72.1(0.74) | 69.0(0.41) | 68.0(0.21) | 68.7(0.26) |

- MC-dropout : probability score
- LOPS : learning order

- high standard deviations are highlighted in blue
- low performances are highlighted in red

# Experiment

| Classifier | Method | NYT-Coarse Mi-F1 | NYT-Coarse Ma-F1 | 20News-Coarse Mi-F1 | 20News-Coarse Ma-F1 | AGNews Mi-F1 | AGNews Ma-F1 | Books Mi-F1 | Books Ma-F1 | NYT-Fine Mi-F1 | NYT-Fine Ma-F1 | 20News-Fine Mi-F1 | 20News-Fine Ma-F1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Coarse-grained Datasets | | | | | | | | Fine-grained Datasets | | | |
| BERT | Standard | 90.1(0.17) | 80.3(0.91) | 77.3(0.27) | 76.4(0.76) | 75.4(0.64) | 75.4(0.47) | 55.7(0.54) | 57.9(0.82) | 77.2(0.36) | 71.6(0.43) | 70.0(0.30) | 69.6(0.25) |
| | LOPS | **94.6(0.36)** | **88.4(0.50)** | **81.7(1.00)** | **80.7(0.43)** | **79.5(0.86)** | **79.5(0.58)** | **57.7(0.87)** | **59.5(0.46)** | **84.3(0.54)** | **81.6(0.34)** | **73.8(0.61)** | **72.7(1.00)** |
| | MC-Dropout | 89.3(0.41) | 79.3(0.45) | 80.7(0.17) | 77.7(0.24) | 75.8(0.34) | 75.0(0.41) | 55.1(0.15) | 56.7(0.61) | 72.1(0.74) | 69.0(0.41) | 68.0(0.21) | 68.7(0.26) |
| | Entropy | 91.2(0.41) | 83.1(0.47) | 80.4(0.23) | 78.0(0.54) | **80.4(0.47)** | **80.0(0.42)** | 55.2(0.74) | 56.7(0.42) | 43.4(9.84) | 18.1(6.98) | 64.3(0.74) | 63.6(0.83) |
| | O2U-Net | 92.9(0.41) | 85.9(0.69) | 80.9(0.28) | 78.5(0.19) | 79.8(0.47) | 79.8(0.53) | 55.8(0.27) | 56.8(0.36) | 14.7(10.24) | 8.70(7.31) | 71.1(0.36) | 71.2(0.75) |
| | Random | 90.3(0.47) | 80.9(0.47) | 79.0(1.00) | 76.8(1.50) | 76.3(0.35) | 76.3(0.65) | 56.1(0.18) | 58.2(0.35) | 78.4(0.94) | 71.7(0.47) | 71.4(0.50) | 70.6(1.00) |
| | Probability | 92.3(1.50) | 85.1(2.00) | 78.6(2.50) | 77.5(3.00) | 77.4(1.25) | 77.6(1.34) | 54.3(1.12) | 56.5(1.43) | 46.6(2.50) | 22.3(0.50) | 47.8(23.50) | 47.9(23.50) |
| | Stability | 93.3(0.50) | 86.5(0.50) | 76.7(5.00) | 75.4(5.00) | 79.3(0.75) | 79.5(0.35) | 55.0(0.43) | 57.0(0.19) | 48.1(29.50) | 35.5(33.50) | 73.5(0.50) | 72.5(1.00) |

- LOPS has stability

# Experiment

| Classifier | Method | Coarse-grained Datasets | | | | | | | | Fine-grained Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NYT-Coarse | | 20News-Coarse | | AGNews | | Books | | NYT-Fine | | 20News-Fine | |
| | | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 | Mi-F1 | Ma-F1 |
| BERT | Standard | 90.1(0.17) | 80.3(0.91) | 77.3(0.27) | 76.4(0.76) | 75.4(0.64) | 75.4(0.47) | 55.7(0.54) | 57.9(0.82) | 77.2(0.36) | 71.6(0.43) | 70.0(0.30) | 69.6(0.25) |
| | LOPS | **94.6(0.36)** | **88.4(0.50)** | **81.7(1.00)** | **80.7(0.43)** | **79.5(0.86)** | **79.5(0.58)** | **57.7(0.87)** | **59.5(0.46)** | **84.3(0.54)** | **81.6(0.34)** | **73.8(0.61)** | **72.7(1.00)** |
| | MC-Dropout | 89.3(0.41) | 79.3(0.45) | 80.7(0.17) | 77.7(0.24) | 75.8(0.34) | 75.0(0.41) | 55.1(0.15) | 56.7(0.61) | 72.1(0.74) | 69.0(0.41) | 68.0(0.21) | 68.7(0.26) |
| | Entropy | 91.2(0.41) | 83.1(0.47) | 80.4(0.23) | 78.0(0.54) | **80.4(0.47)** | **80.0(0.42)** | 55.2(0.74) | 56.7(0.42) | 43.4(9.84) | 18.1(6.98) | 64.3(0.74) | 63.6(0.83) |
| | O2U-Net | 92.9(0.41) | 85.9(0.69) | 80.9(0.28) | 78.5(0.19) | 79.8(0.47) | 79.8(0.53) | 55.8(0.27) | 56.8(0.36) | 14.7(10.24) | 8.70(7.31) | 71.1(0.36) | 71.2(0.75) |
| | Random | 90.3(0.47) | 80.9(0.47) | 79.0(1.00) | 76.8(1.50) | 76.3(0.35) | 76.3(0.65) | 56.1(0.18) | 58.2(0.35) | 78.4(0.94) | 71.7(0.47) | 71.4(0.50) | 70.6(1.00) |
| | Probability | 92.3(1.50) | 85.1(2.00) | 78.6(2.50) | 77.5(3.00) | 77.4(1.25) | 77.6(1.34) | 54.3(1.12) | 56.5(1.43) | 46.6(2.50) | 22.3(0.50) | 47.8(23.50) | 47.9(23.50) |
| | Stability | 93.3(0.50) | 86.5(0.50) | 76.7(5.00) | 75.4(5.00) | 79.3(0.75) | 79.5(0.35) | 55.0(0.43) | 57.0(0.19) | 48.1(29.50) | 35.5(33.50) | 73.5(0.50) | 72.5(1.00) |
| | OptimalFilter | 98.3(0.27) | 96.4(0.37) | 94.7(0.37) | 94.9(0.61) | 89.4(0.46) | 89.3(0.76) | 76.2(0.21) | 76.7(0.19) | 97.4(0.71) | 92.2(0.62) | 87.6(0.37) | 86.5(0.36) |

OptimalFilter : remove all the wrongly annotated samples

# Outline

- Introduction
- Method
- Experiment
- Conclusion

# Conclusion

- Propose a method that considers learning order, LOPS, which can be used as a plug-in for text classifiers and weak supervision

- Learning sequence-based methods are more stable and effective than probability-based methods