DEVIEW
2019

레이블링 조금 잘못돼도 괜찮아요!

Clova 가 레이블 노이즈 잡는 법

강재욱

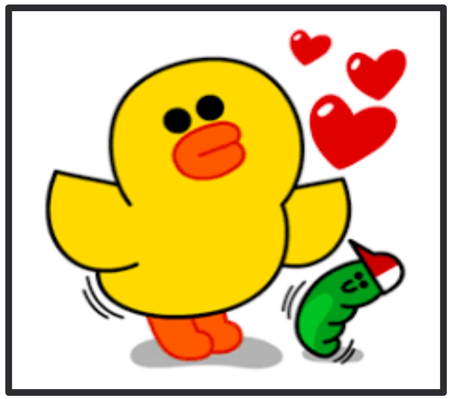
CLOVA Chatbot Model

DEVIEW
2019

레이블 노이즈란 무엇인가?

DEVIEW
2019

Sally

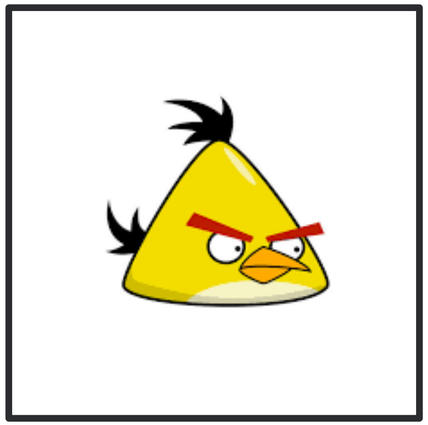


Sally

Sally

Sally ?

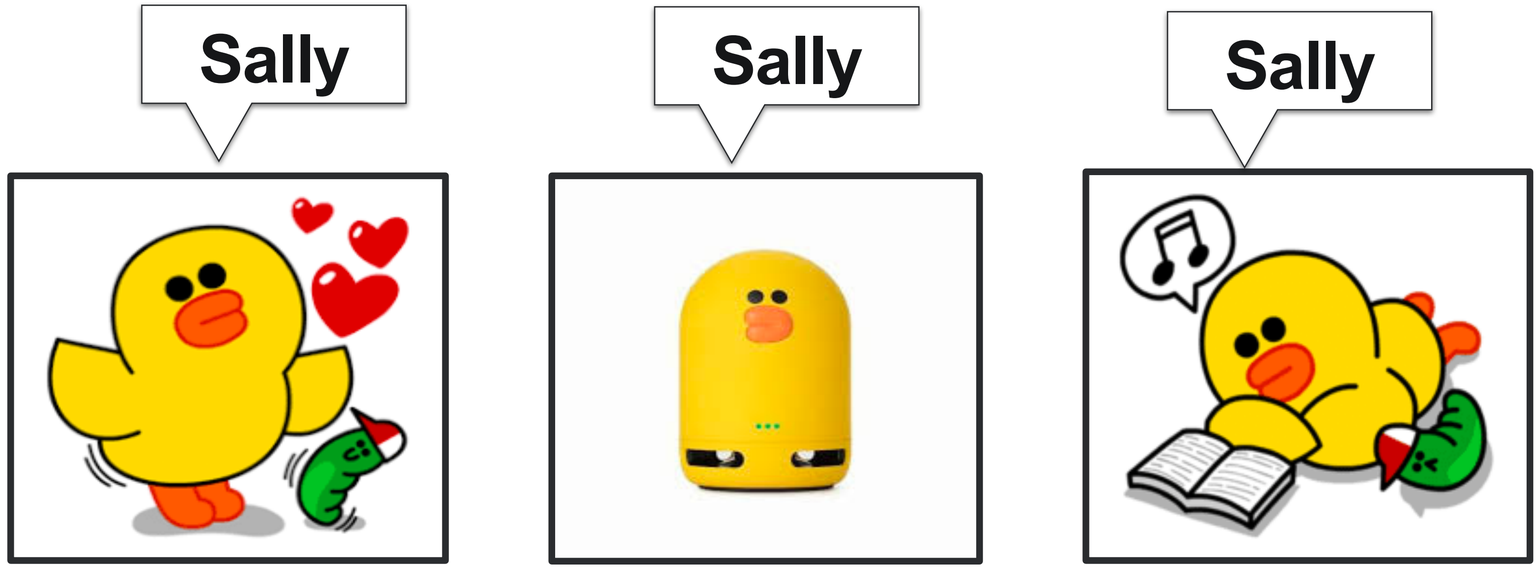
Sally ?



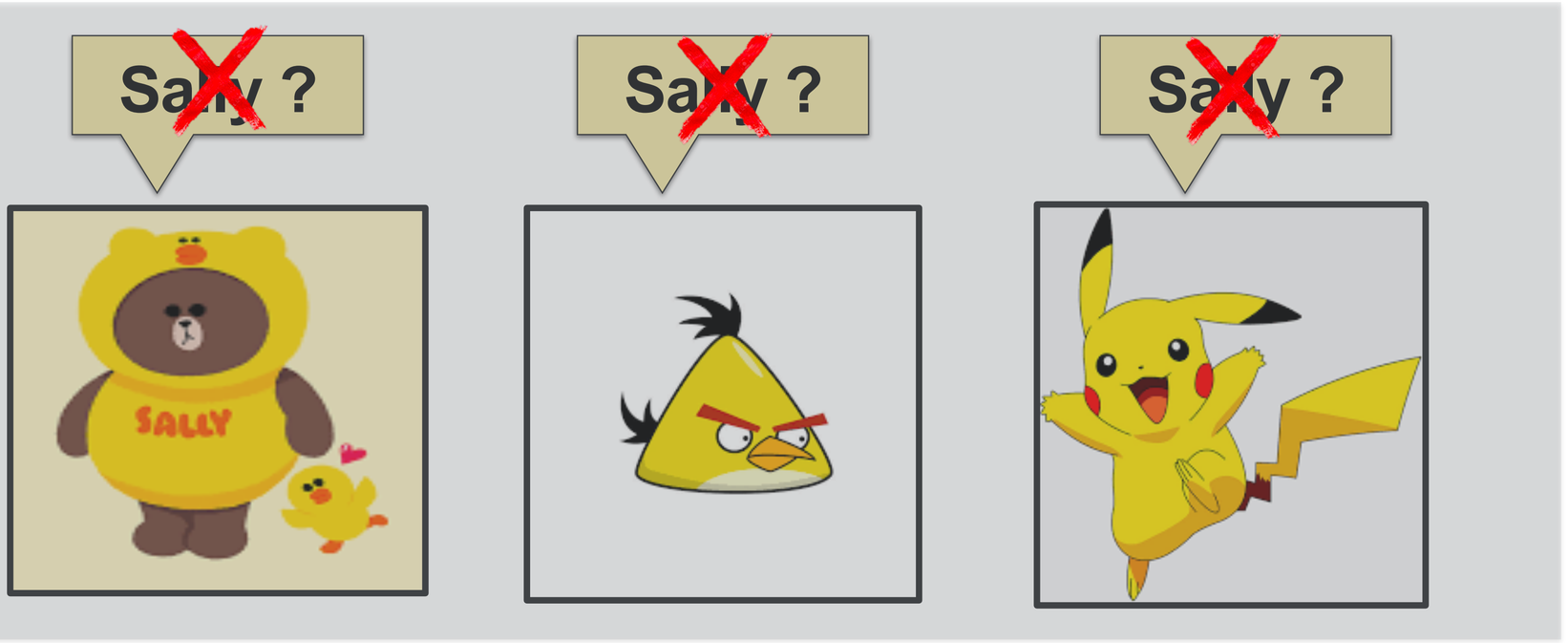
Sally ?

DEVIEW 2019

Clean Sally



Noisy Sally



같은 범주의 데이터를 잘못 설명하는

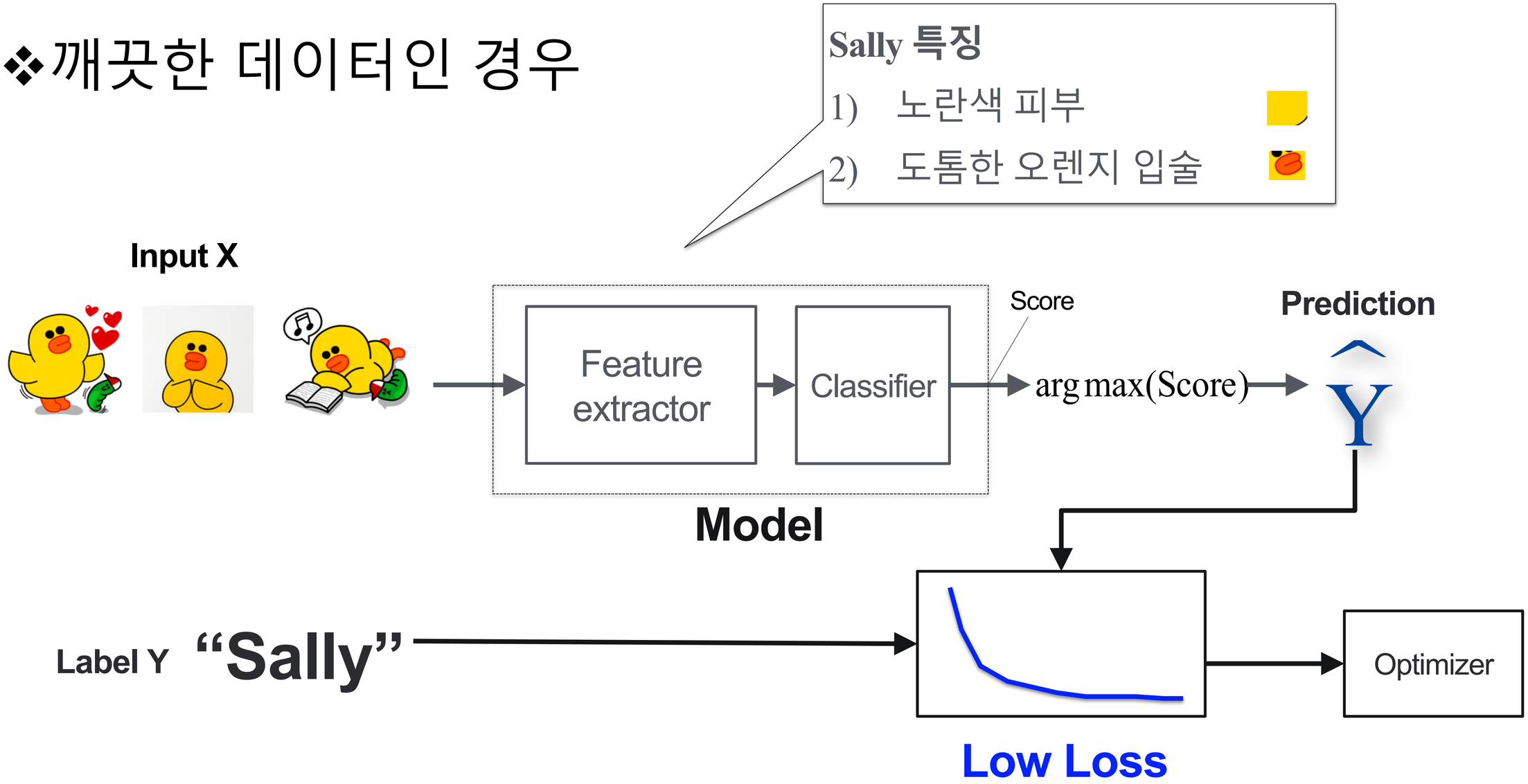
“의도되지 않은” Mislabeling

Label Noise is Everywhere !!

그럼 왜 문제인가?

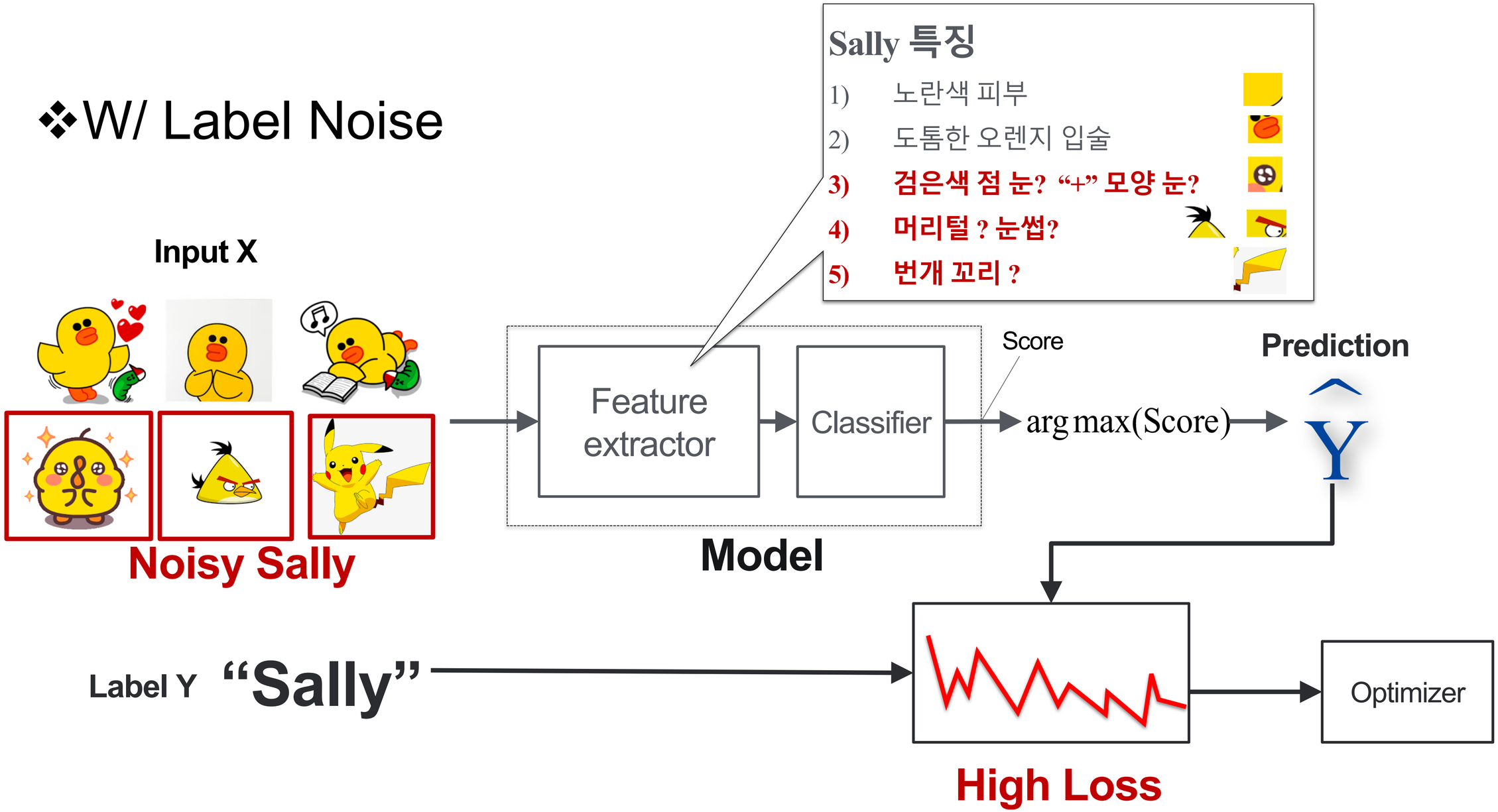
레이블 노이즈가 모델학습에 미치는 영향

❖깨끗한 데이터인 경우



레이블 노이즈가 모델학습에 미치는 영향

❖W/ Label Noise



레이블 노이즈는 모델의
Feature Extraction을 어렵게 한다

어려워진 Feature Extraction은

모델 성능을 떨어뜨린다

AI / 머신러닝/딥러닝 / 빅 데이터

2019.08.09

'바보야, 문제는 데이터야!' AI가 프로젝트가 폭망한 6가지 이유

Maria Korolov | CIO

AI 프로젝트가 기대에 미치지 못하는 주된 이유 중 하나는 데이터에 있다. 그러나 실수를 통해 배우고 장기적으로 노력할 수 있다면 AI 프로젝트가 더 나은 결과를 가져올 것이다.

http://www.ciokorea.com/news/127988

Think Data First before being AI First

발행한 날: 2019년 2월 22일

Srivatsan Srinivasan

Chief Data Scientist | Data (Big, Huge, Small, udent

글 23 + 팔로우

https://www.linkedin.com/pulse/think-data-first-before-being-ai-srivatsan-srinivasan/

https://www.trustinsights.ai/blog/2019/08/5-ways-your-ai-projects-fail-part-3-data-related-ai-failures/

그럼 레이블 노이즈!
어떻게 해결 할 것인가?

훈련 모델 =

훈련방법(데이터, 모델 구조)

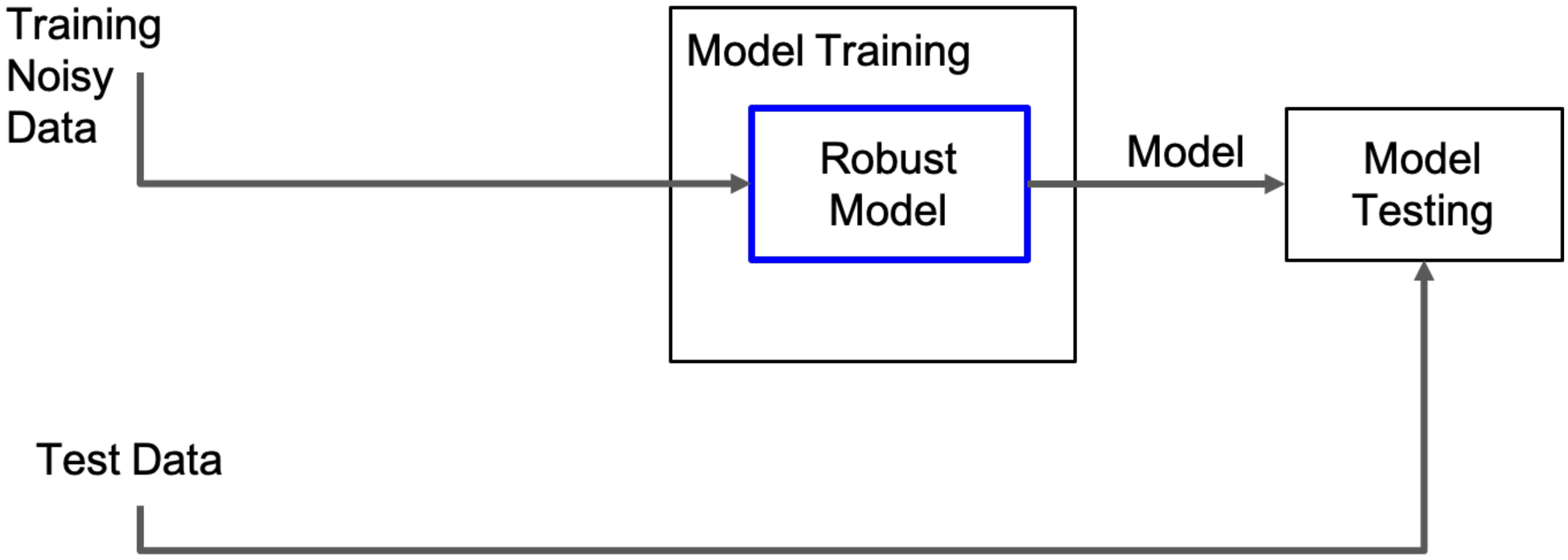
Approach 1: 모델 구조

DEVIEW 2019

❖ 복잡한 패턴도 잘 인식하는 **모델 구조**을 쓴다

Approach 1: 모델 구조

❖ Label Noise Robust Model



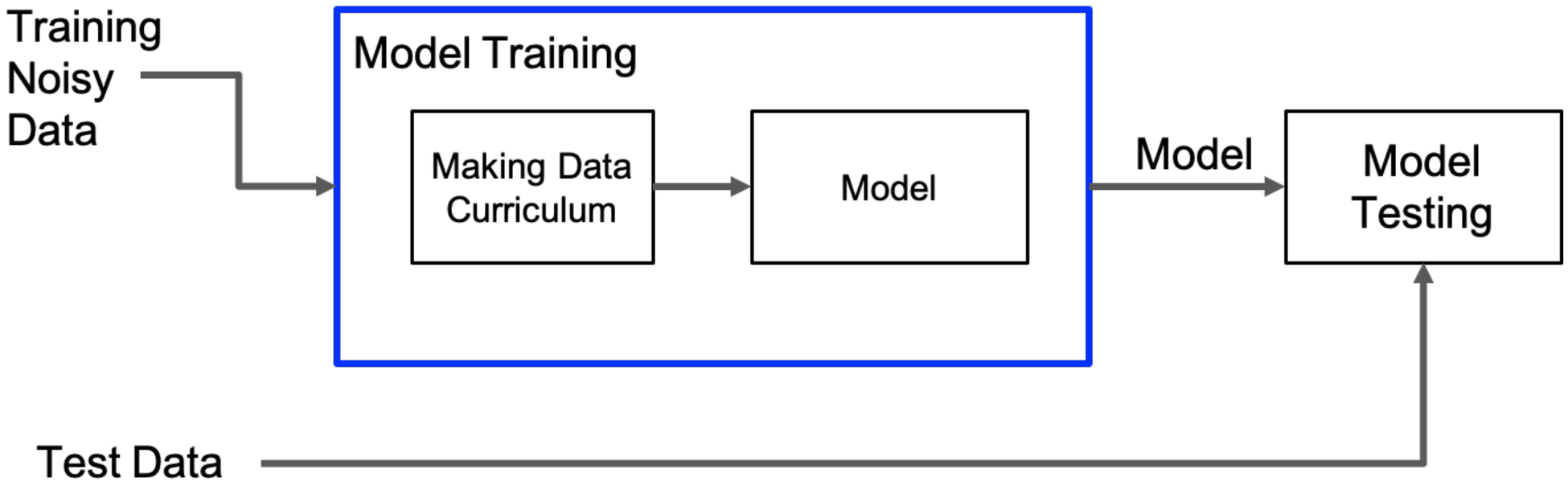
무거운 모델 == 서빙+훈련 계산량 증가...

Approach 2: 훈련 방법

❖ **커리큘럼**을 만들어서 **학습**시킨다

Approach 2-1:Curriculum Learning

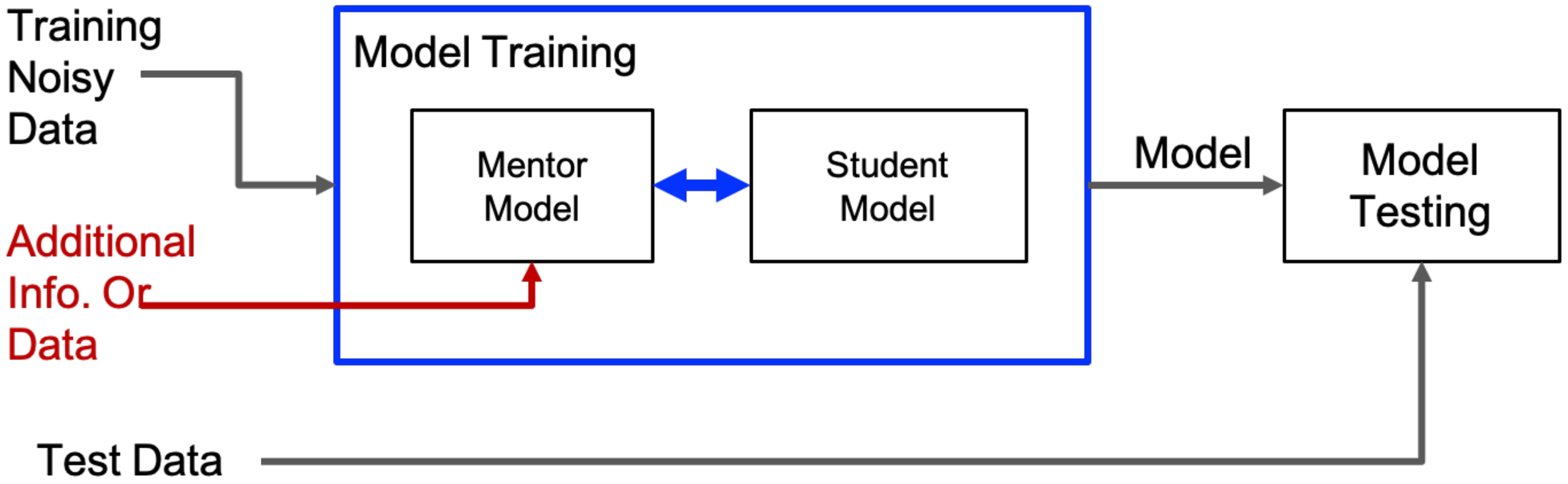
[Y. Bengio: 2009]



Approach 2-2: MentorNet

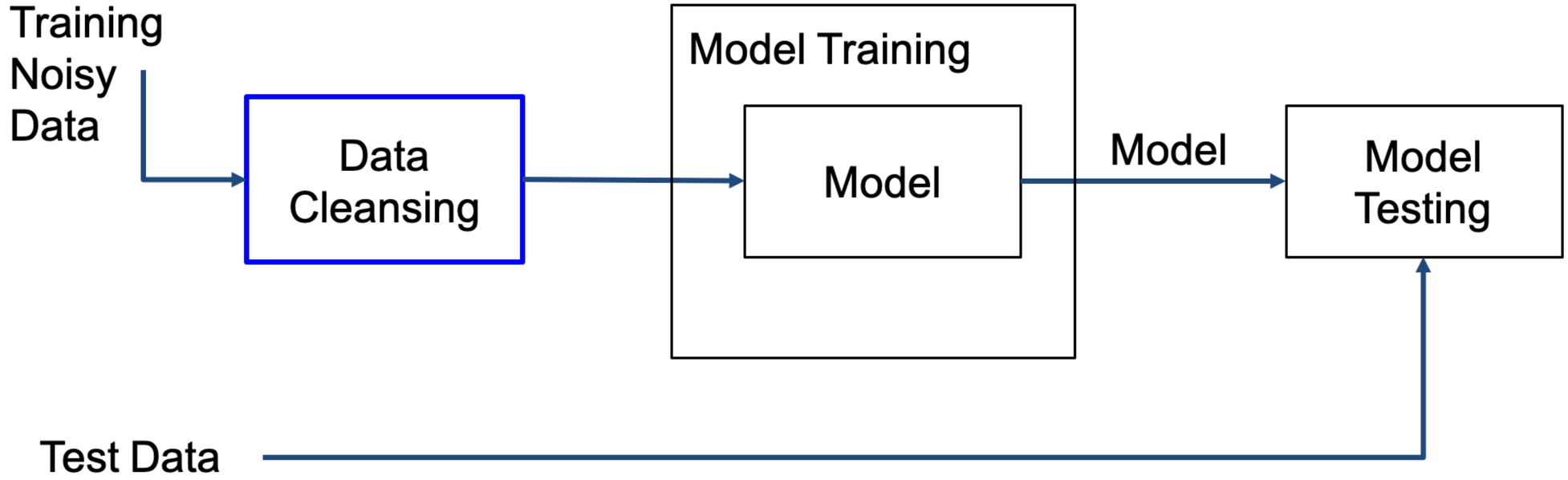
[Lu Jiang, et al.:2018],[Bo Han, et al.: 2019]

DEVIEW 2019



훈련 계산량 증가 + 추가 데이터 필요

Approach 3: Data Cleaning Method



Human Data Cleaning

DEVIEW 2019

❖ **사람이 수동으로** 작업....

Same or different person?

Same Different Not sure

Same or different person?

Same Different Not sure

Same or different person?

Same Different Not sure

Human Data Cleaning

❖ “돈”줘서 다른 **사람이 수동으로** 작업….

각각의 파일을 참고하시어 스크립트 정비작업 부탁드립니다.

1) ambiguous한 쿼리 대상들이 모두 동일한 의도인 경우 통합 진행

ex) 배고파 - 배고프다/출출하네

출출하네 - 출출해

-> 배고파- 배고프다/출출하네/출출해

2) ambiguous한 쿼리 대상들이 모두 동일한 의도를 가지지 않은 경우 택1하여 하나의 쿼리만 유지하는 작업 진행

ex) 슬퍼 - 슬프다/슬퍼요/우울하다

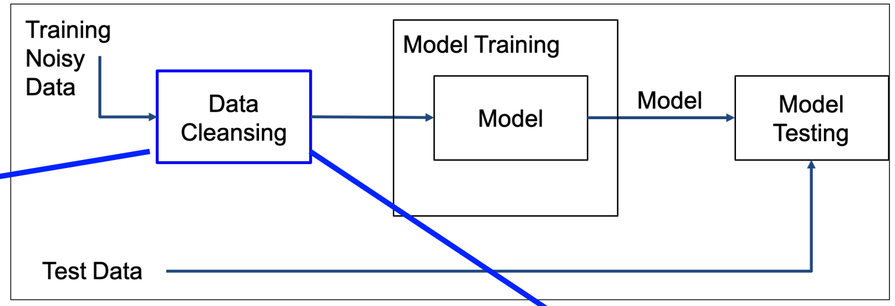
우울해 - 우울하다/우울하네

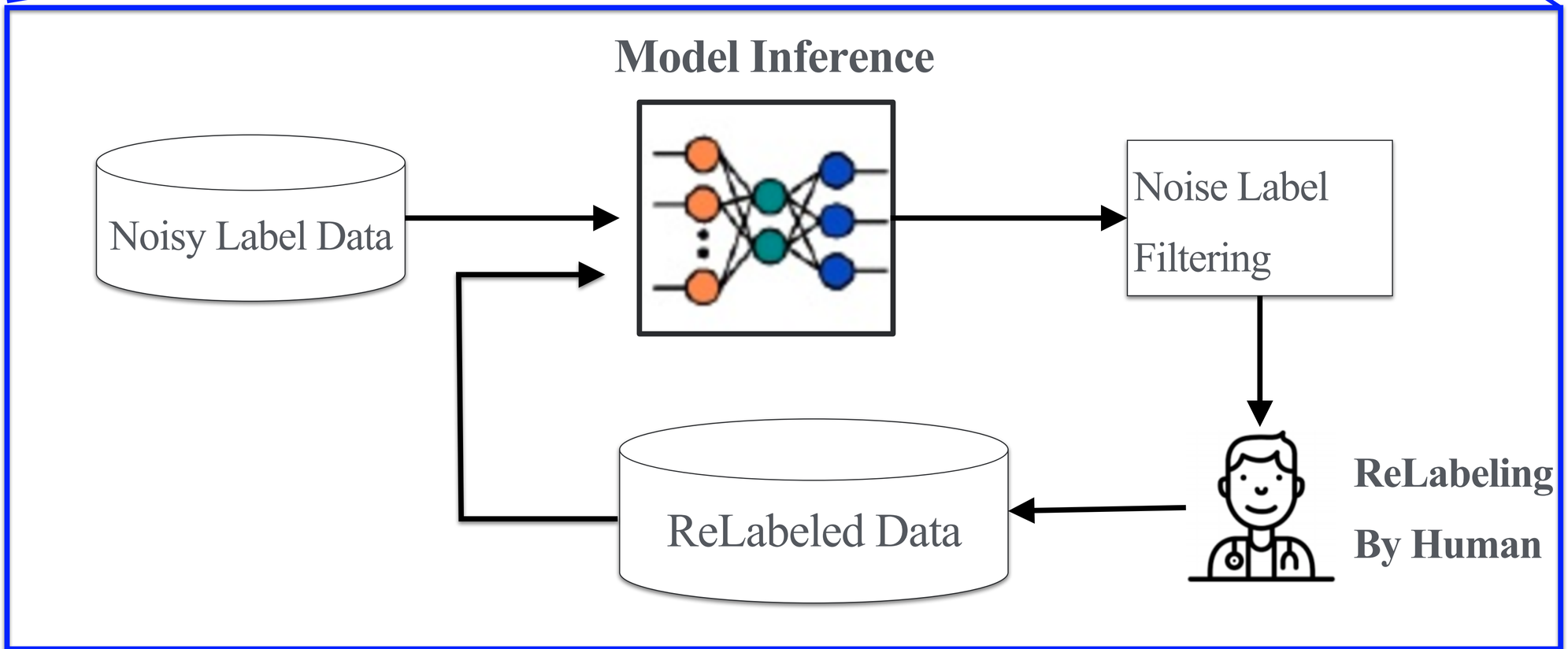
-> 슬퍼 - 슬프다/슬퍼요/~~우울하다~~

우울해 - 우울하다/우울하네

Active Learning for Label Cleaning

[Konyushkova'2017],[Yoo'2019]

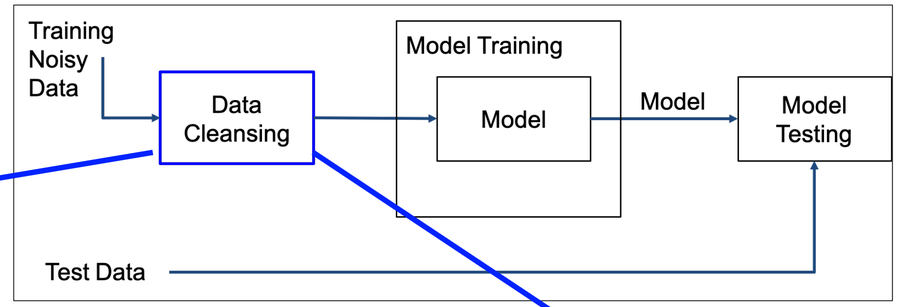


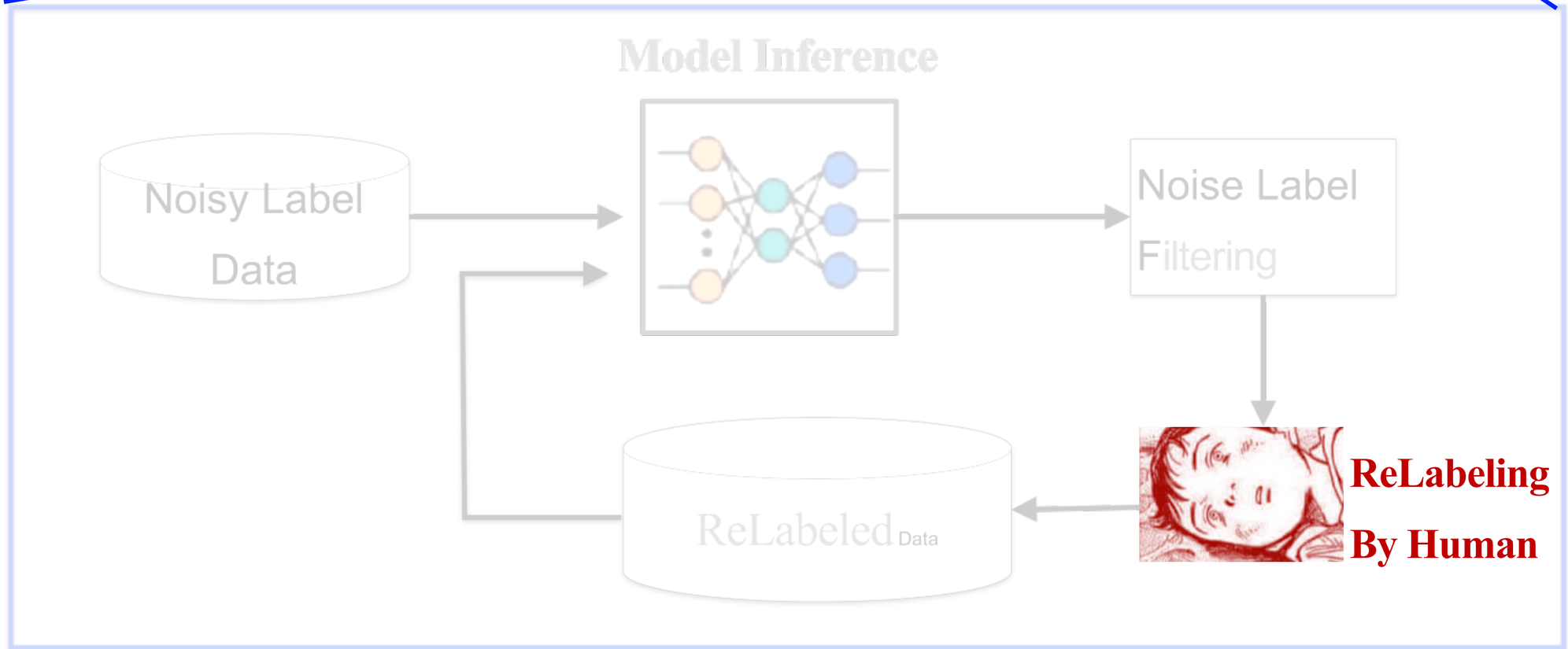


Active Learning for Label Cleaning

[Konyushkova'2017],[Yoo'2019]

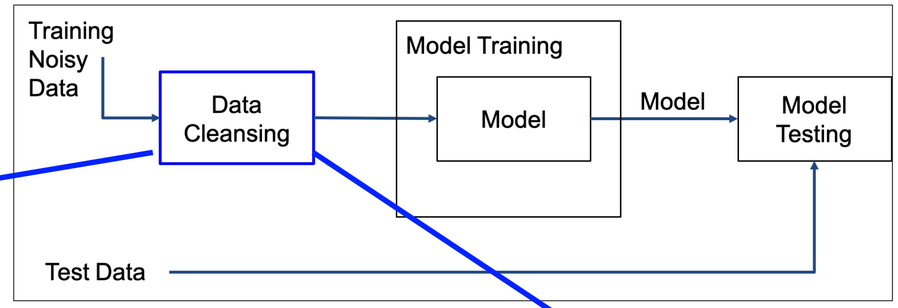
DEVIEW 2019

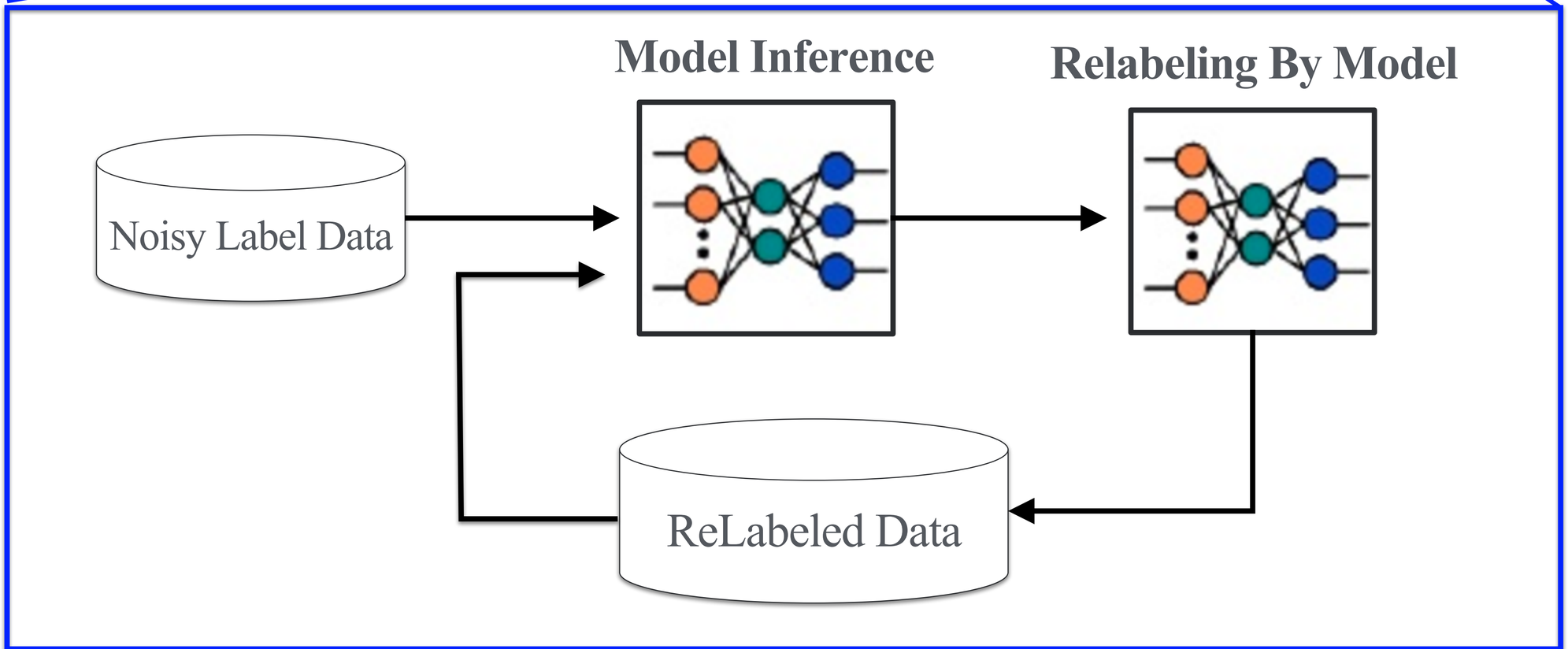




사람의 도움 없이
레이블 노이즈를 제거 할 수 없을까?

Our Plan


Training
Noisy
Data
Data
Cleansing
Model Training
Model
Model
Model
Testing
Test Data


Model Inference
Relabeling By Model
Noisy Label Data
ReLabeled Data

오늘의 이야기

DEVIEW
2019

Clova 가 레이블 노이즈 잡는법!

- **레이블링 바로잡는 AutoML**
- **FAQ/Chat 데이터 셋에 적용해 보기**
- **구현 삽질기 및 향후 해결 과제**

Project Khan

DEVIEW
2019

인공지능이
인공지능 챗봇을 만든다.

DEVIEW
2018

이 재 원

Company.AI

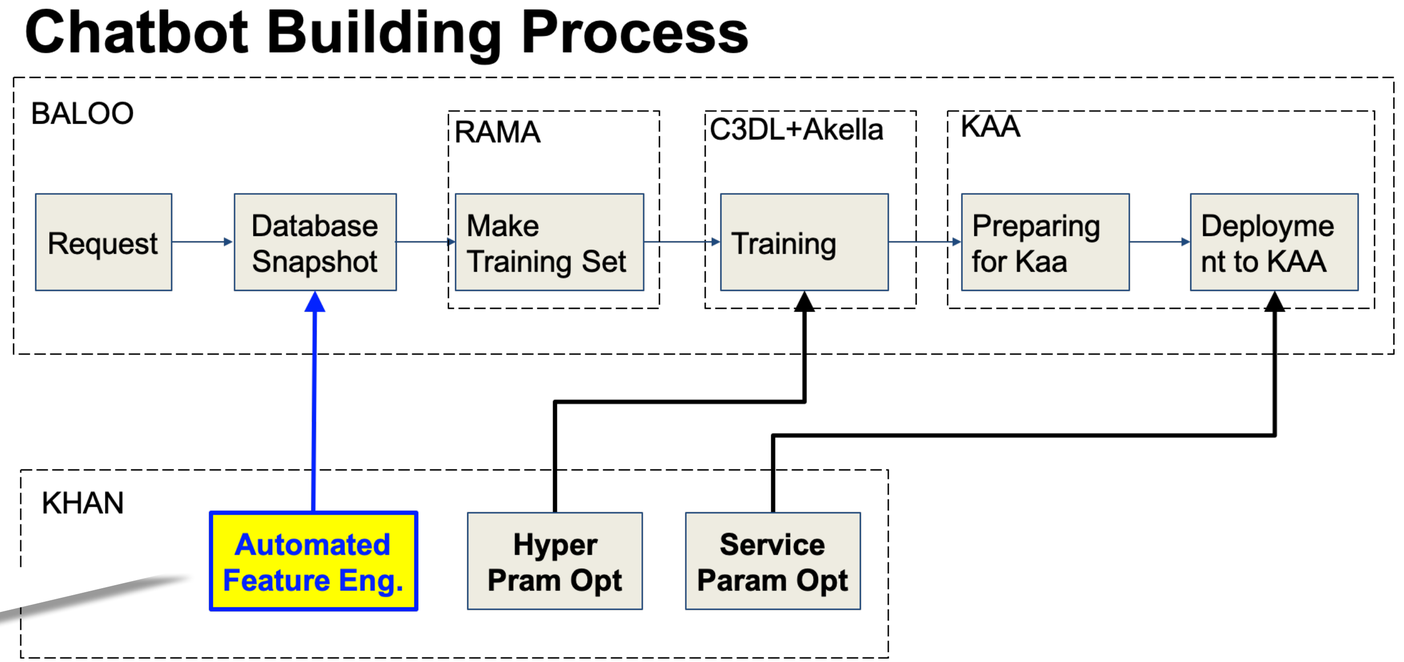
NAVER

작년에 이은 2번째 AutoML 발표!

Project Khan

❖ AutoML Project for Chatbot AI Builder

- Automated feature engineering
- Automated model validation
- Optimize hyper-parameters
- Optimize Service-parameters



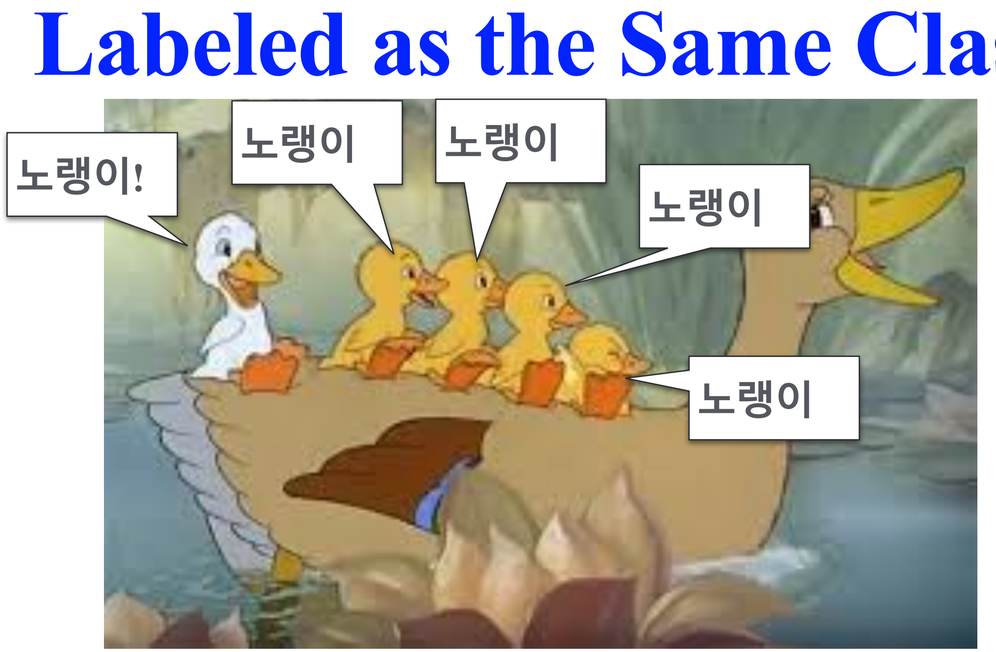
오늘 레이블 노이즈
이야기는 여기에 해당!

DEVIEW
2019

레이블링 바로잡는 AutoML

아이디어

- **문제:** 흰오리 한마리가 mislabeled 되어 있다. 주변을 보고 고칠 수 있는가?

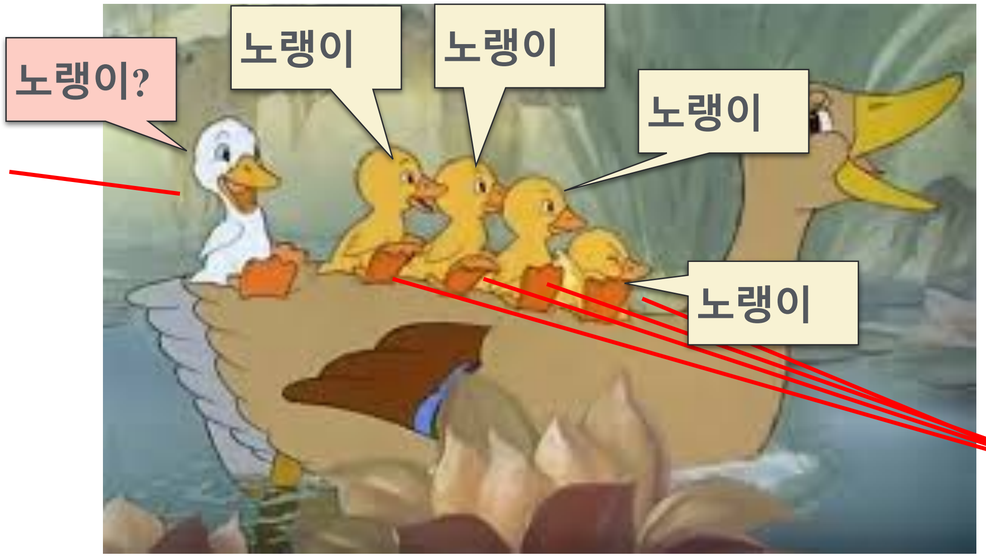


아이디어

- **문제:** 흰오리 한마리가 mislabeled 되어 있다. 주변을 보고 고칠 수 있는가?

Labeled as the Same Class

The Mislabeled (Label Noise)



The Well-labeled

DEVIEW
2019

여러분은 딱 보면 아실텐데

DEVIEW
2019

어떻게 하면 AI한테
똑같은 일을 시킬 수 있을까?

DEVIEW
2019

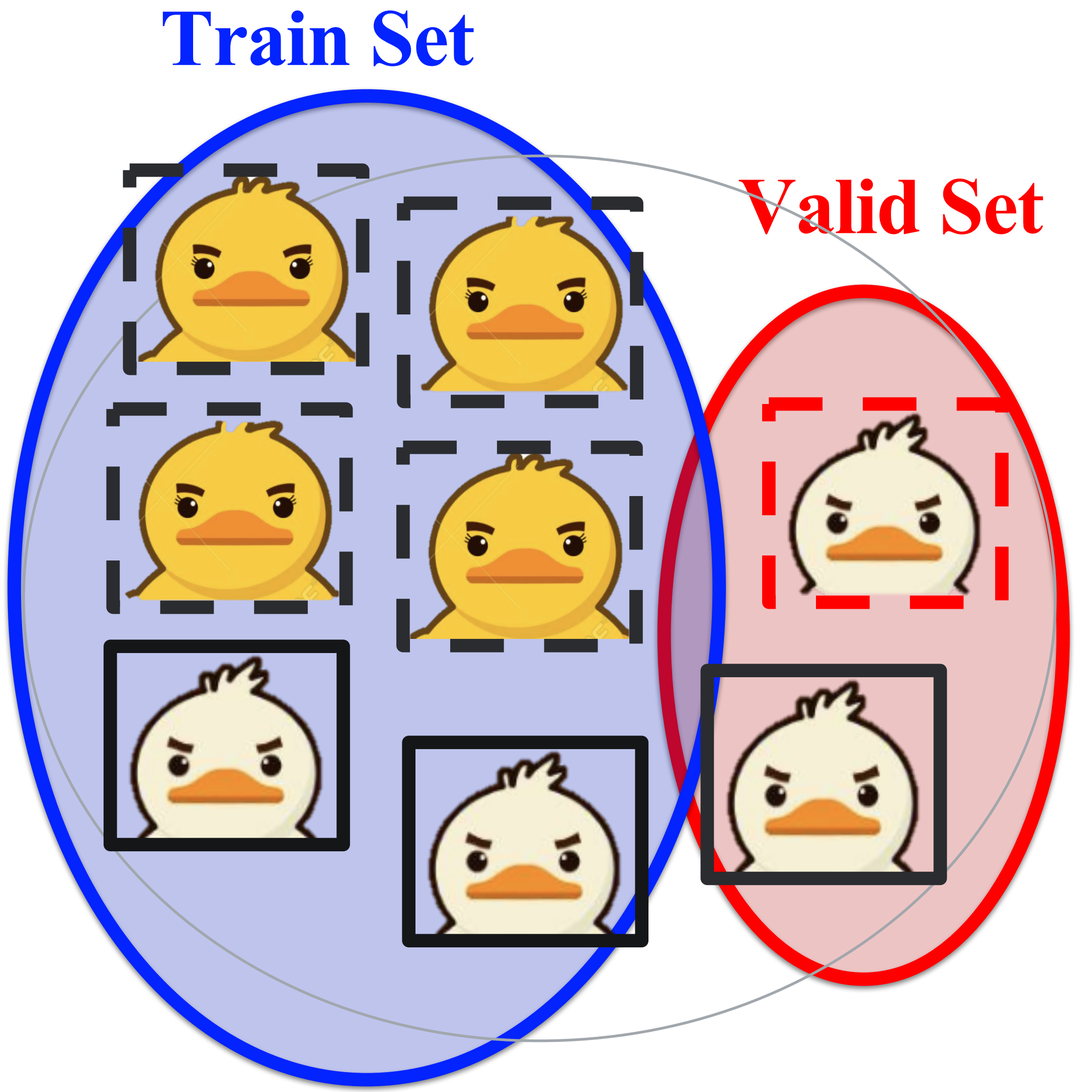
Split – Train – Check

알고리즘

Split - Train - Check 알고리즘

Split:

- 전체 dataset을 train /valid set으로 분할

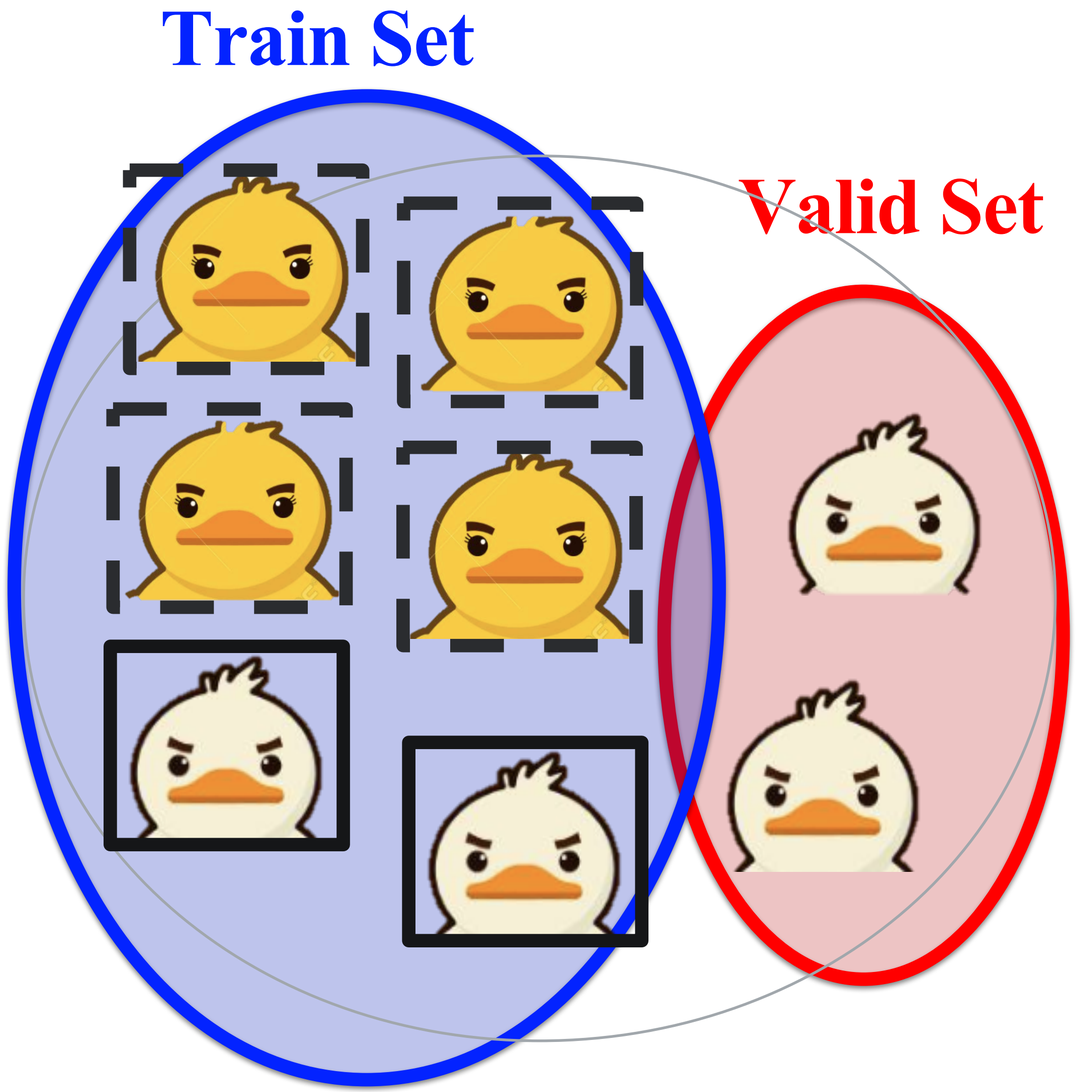


Split - Train - Check 알고리즘

Split:

- 전체 dataset을 train /valid set으로 분할

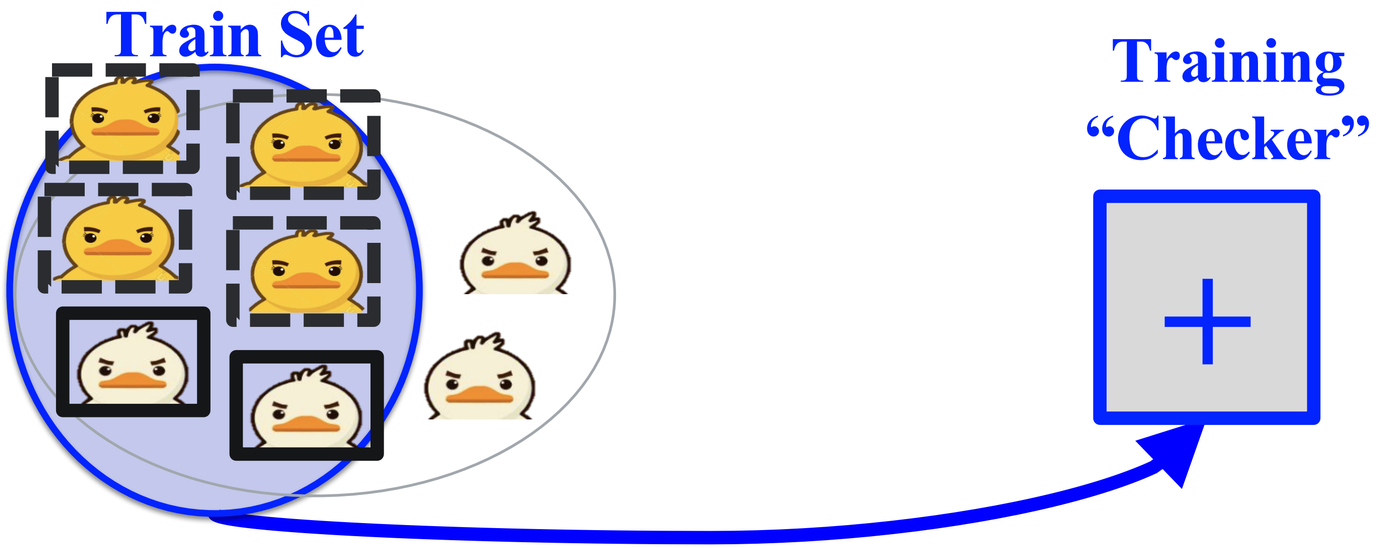
- **Train set**: correction을 위한 “기준 데이터”
- **Valid set**: 검사 대상
  (검사를 위해 Valid set에서 label 제거)



Split - Train - Check 알고리즘

Train:

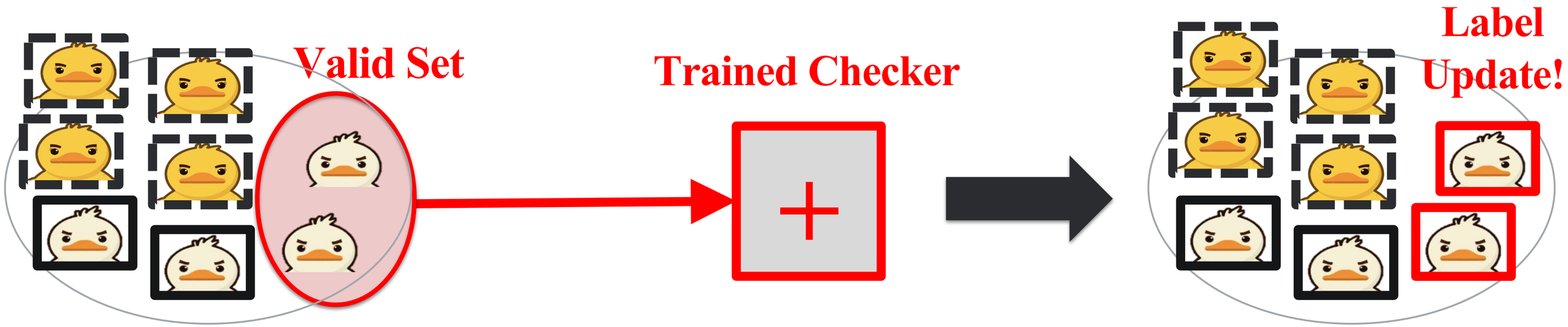
- **Checker** : 레이블 Correction용도로 훈련한 모델
- train set으로 checker를 훈련



Split - Train - Check 알고리즘

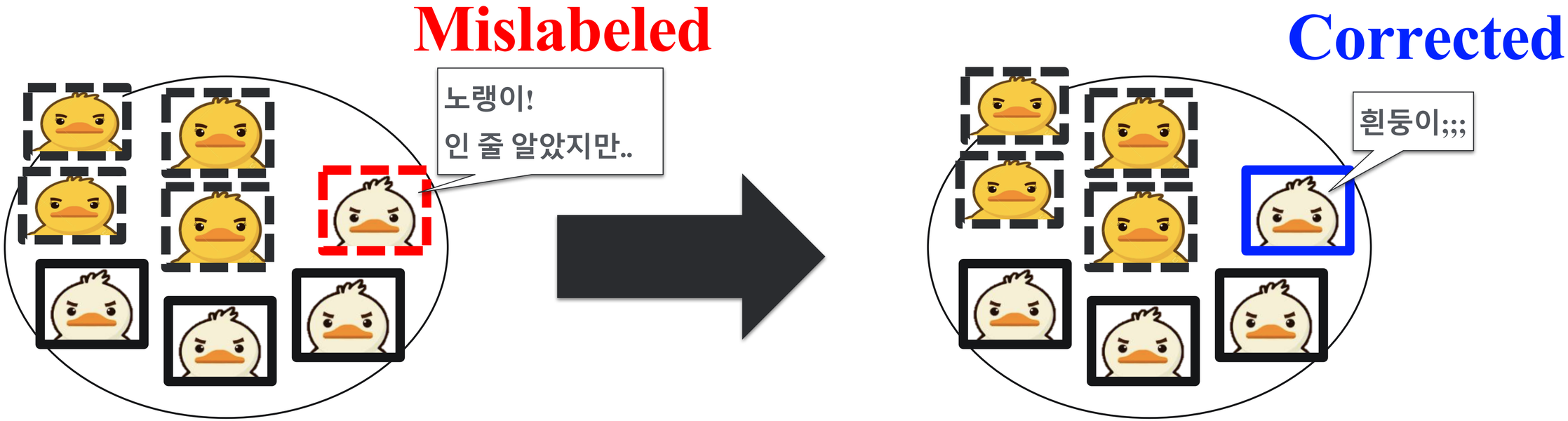
Check:

- valid set를 훈련된 checker에 입력하여 labeling 검사!

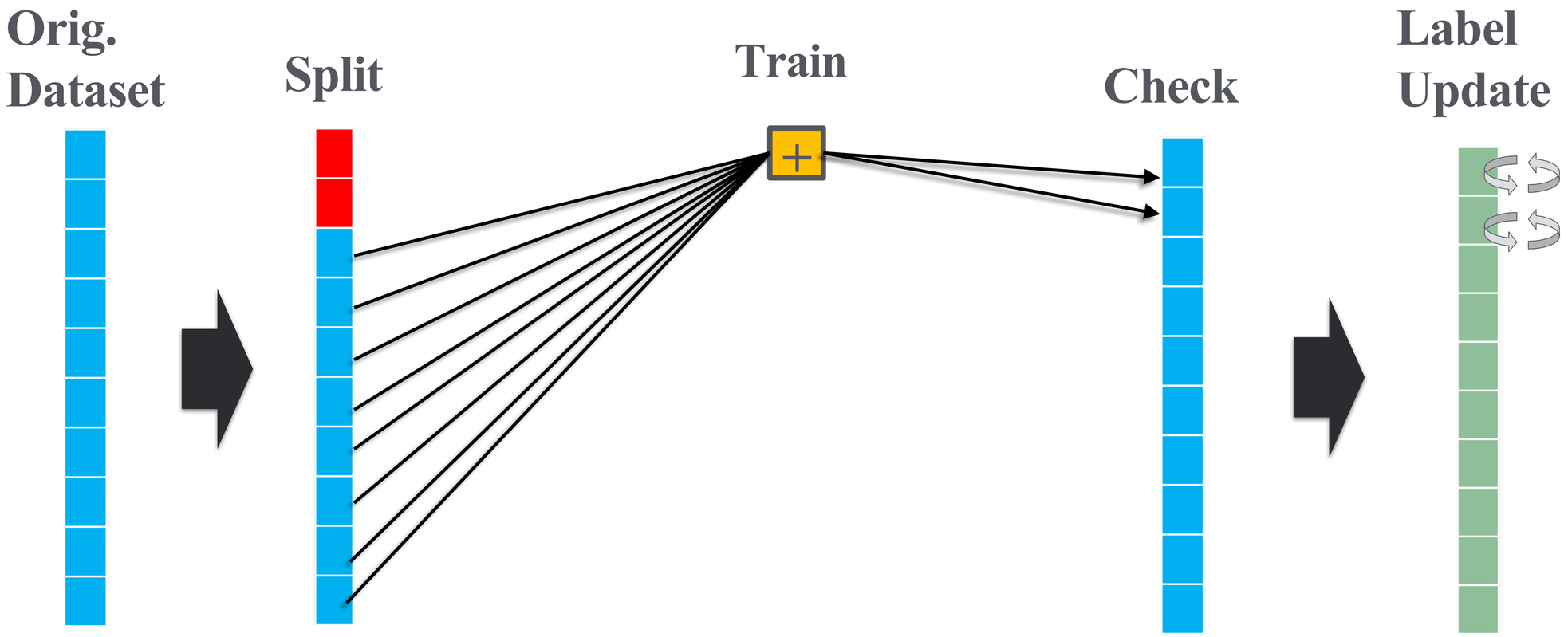


Split - Train - Check 알고리즘

Train set보고 valid set의 Mislabeling을 correction 가능!

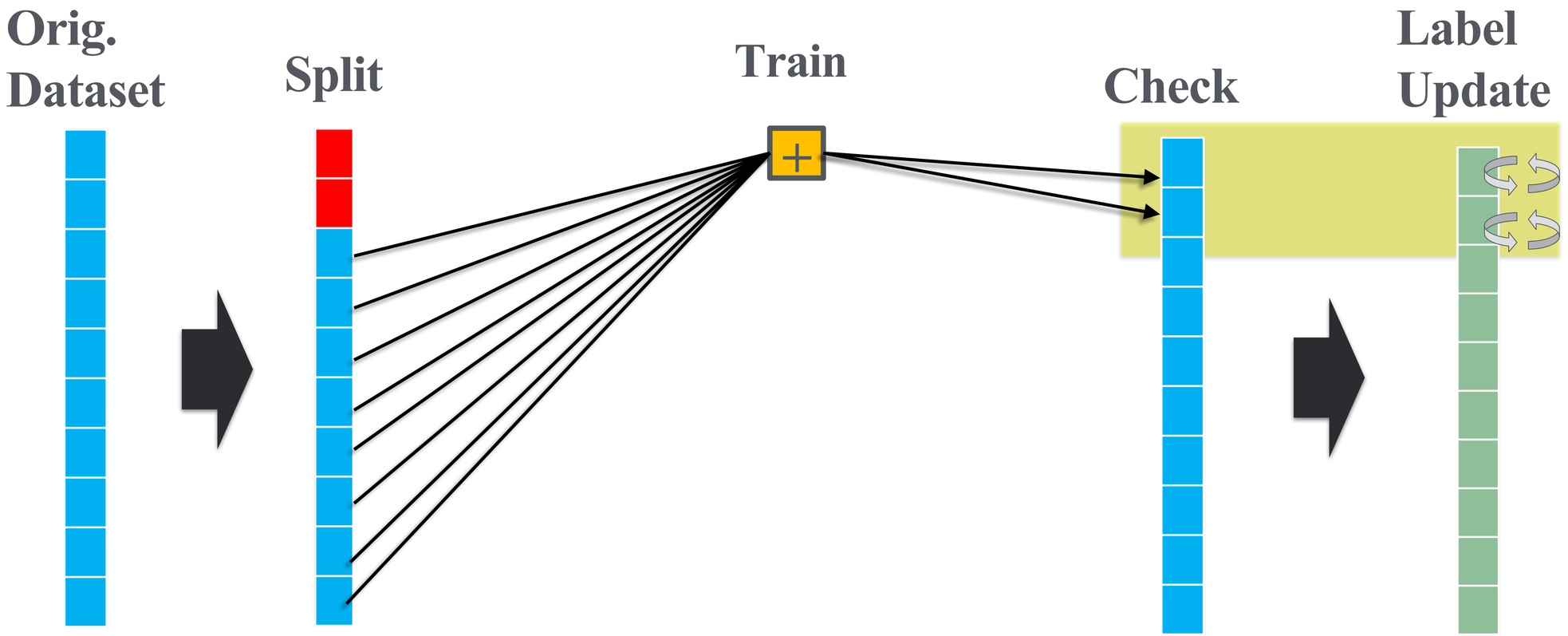


Split - Train - Check 알고리즘



현재 방식은 모든 데이터를 검사 할 수 없음

Split - Train - Check 알고리즘

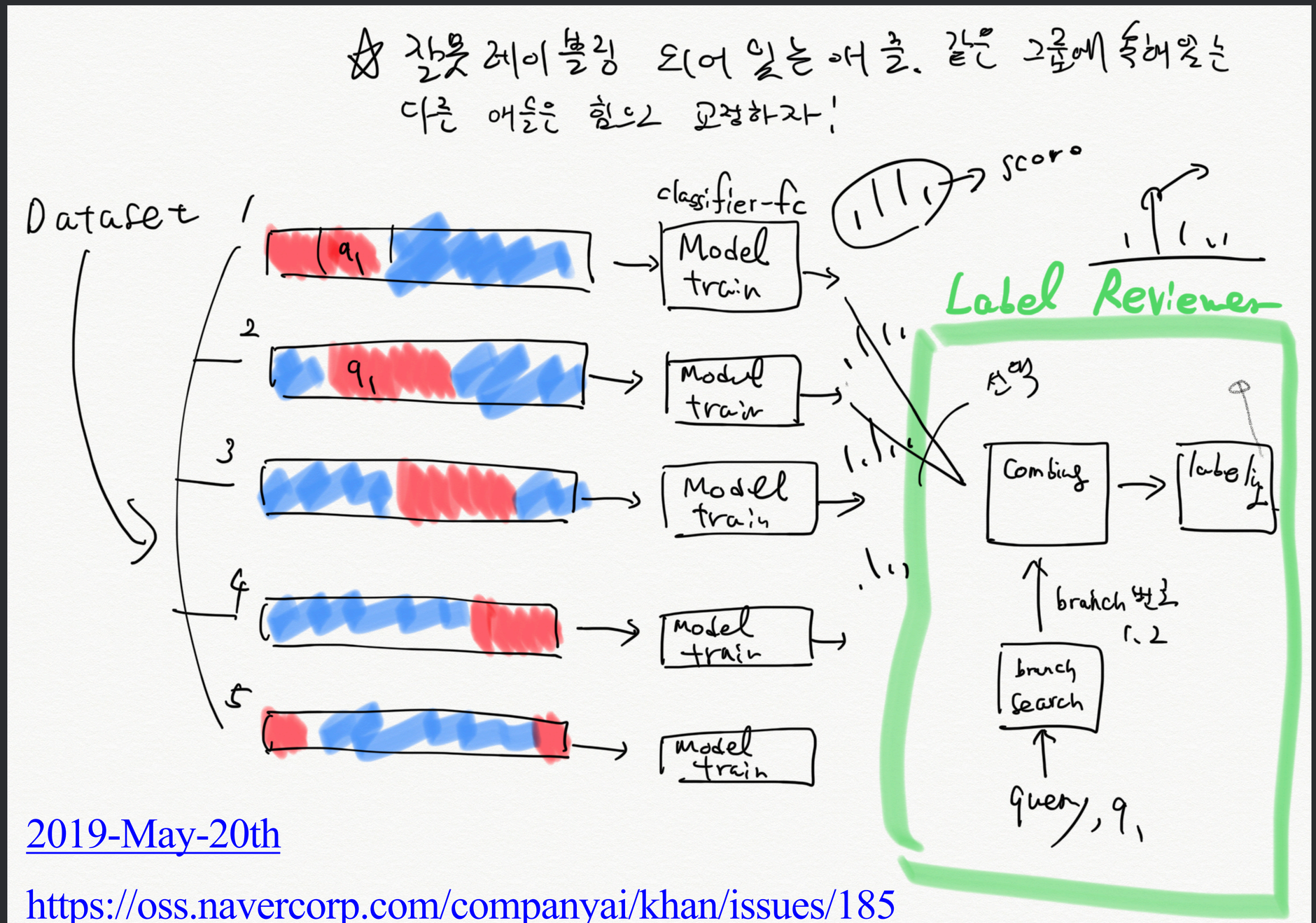


전체 데이터를 빠짐없이 check할 수 있는 방법이 있을까?

MultiSplit – Train – Check - Vote

MultiSplit: 여러버전의 Split branch를 구성

Vote: Label update를 위해서 각 branch의 “Split-Train-Check” 결과를 결합



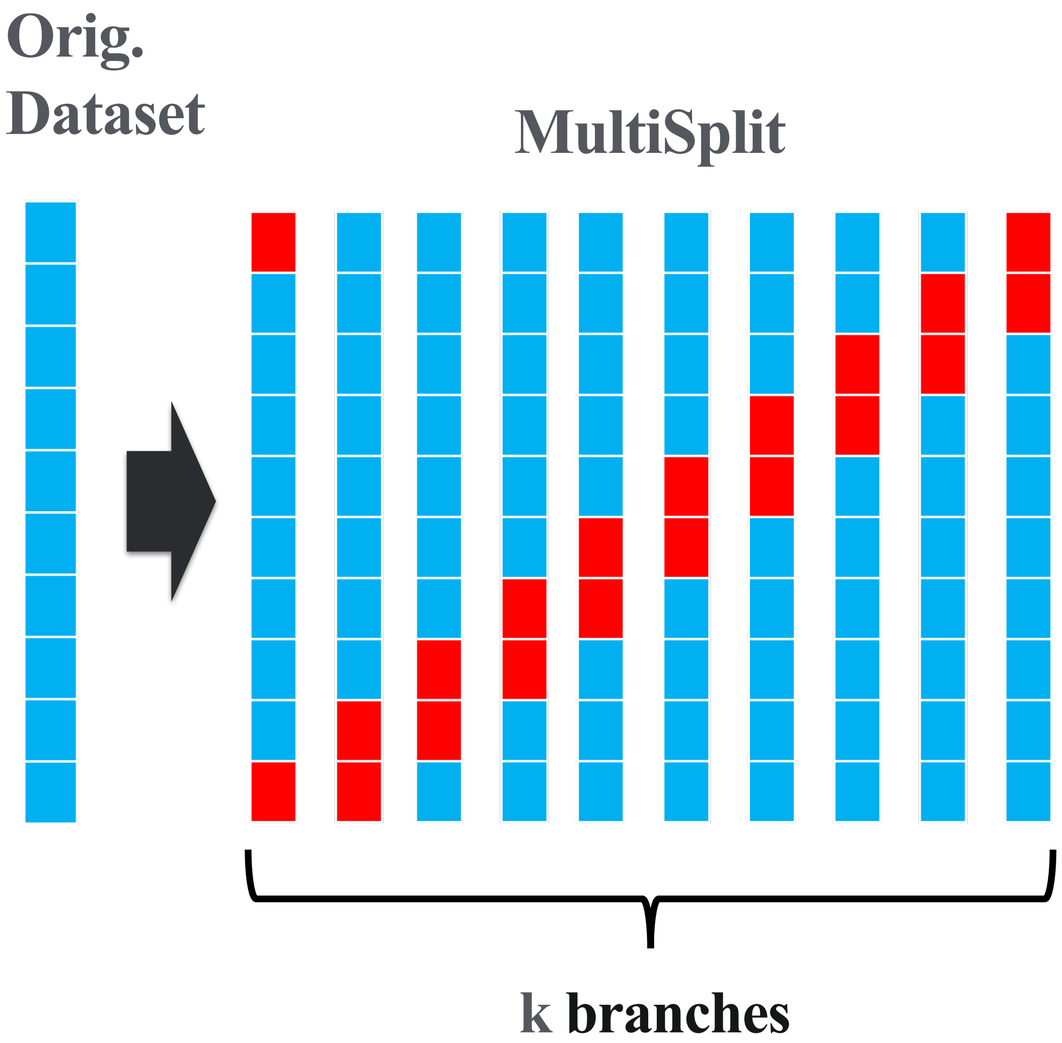
[2019-May-20th]

https://oss.navercorp.com/companyai/khan/issues/185

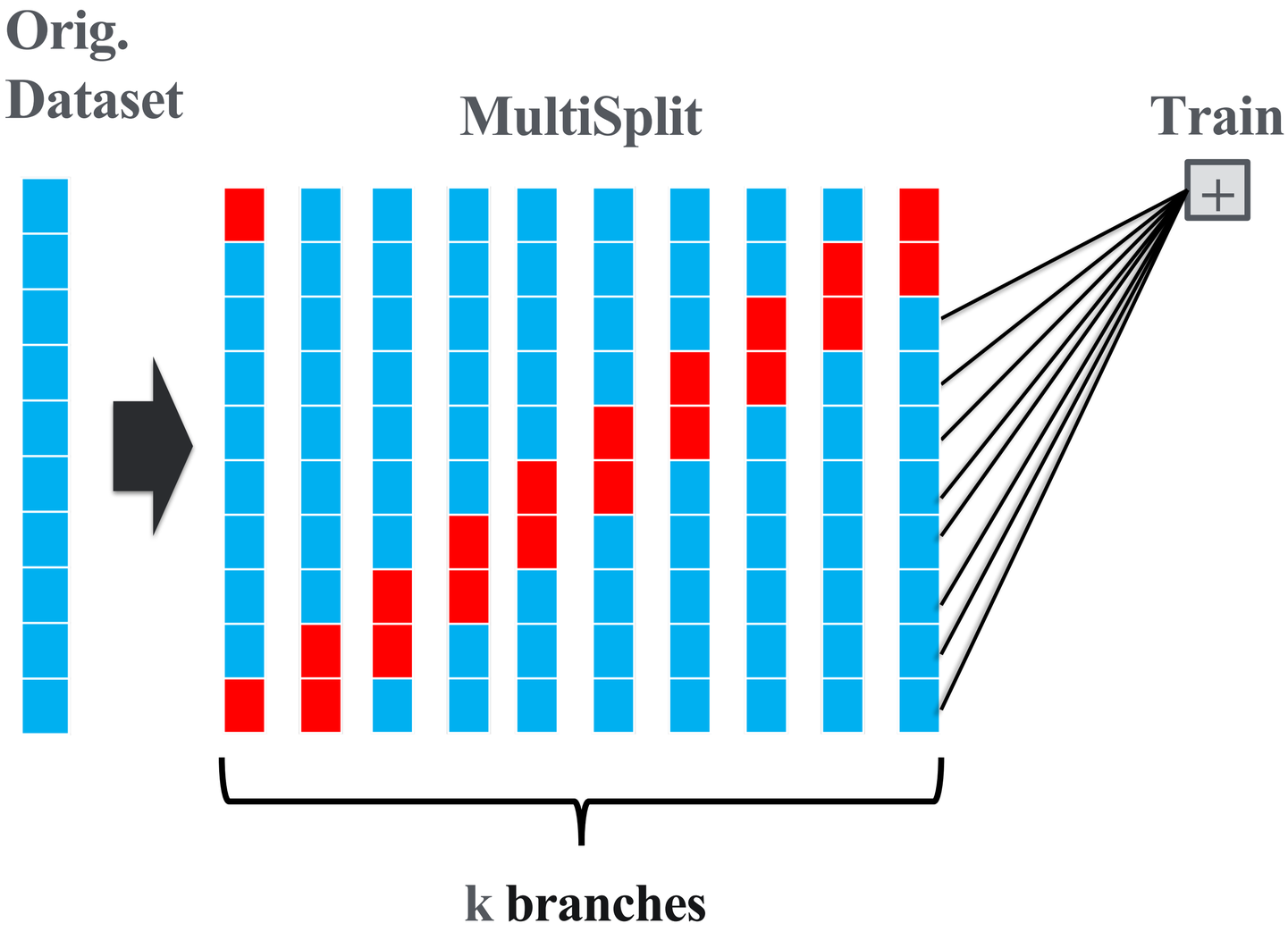
MultiSplit - Train – Check -Vote 알고리즘

Orig.
Dataset

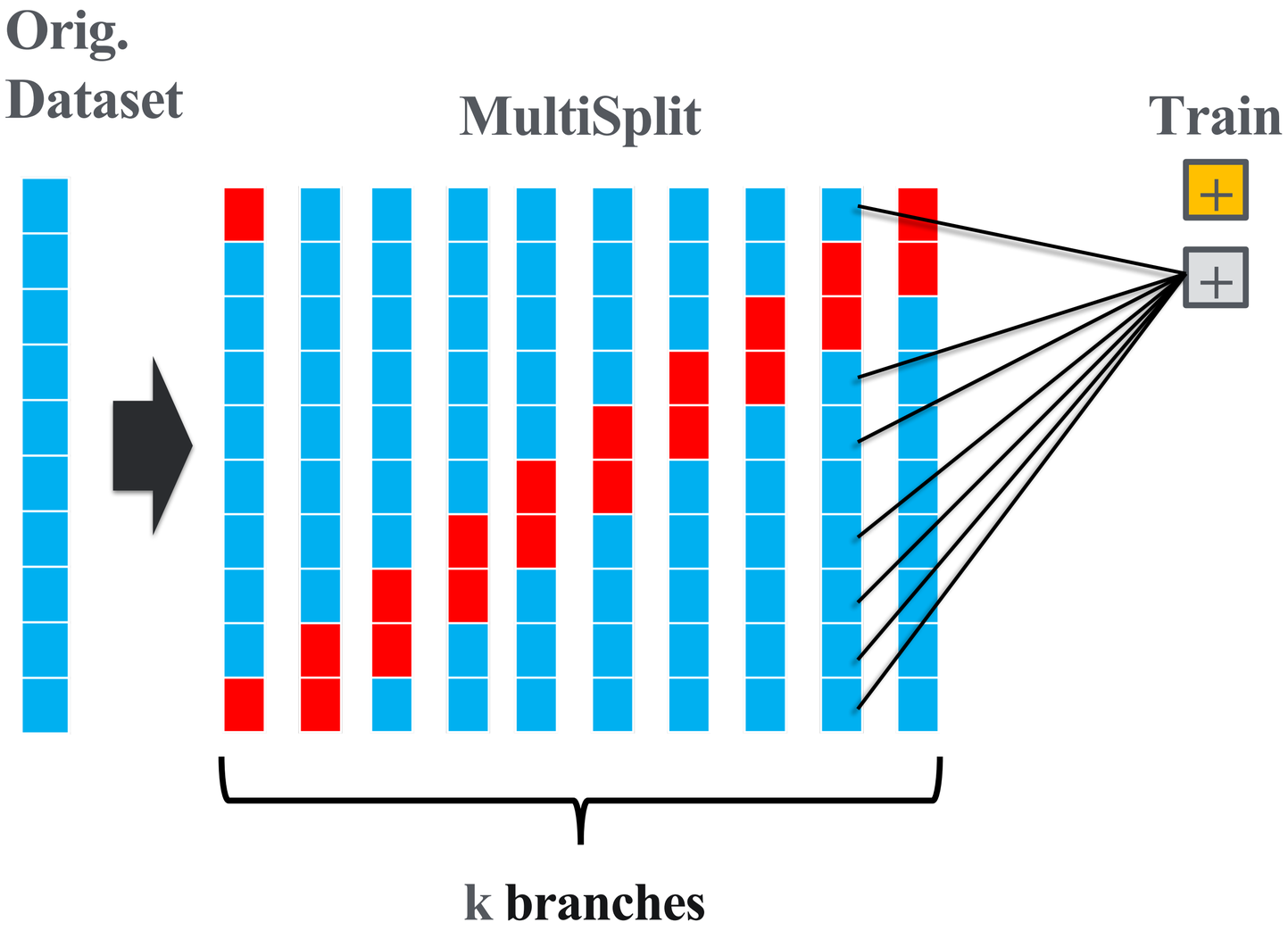
MultiSplit - Train – Check -Vote 알고리즘



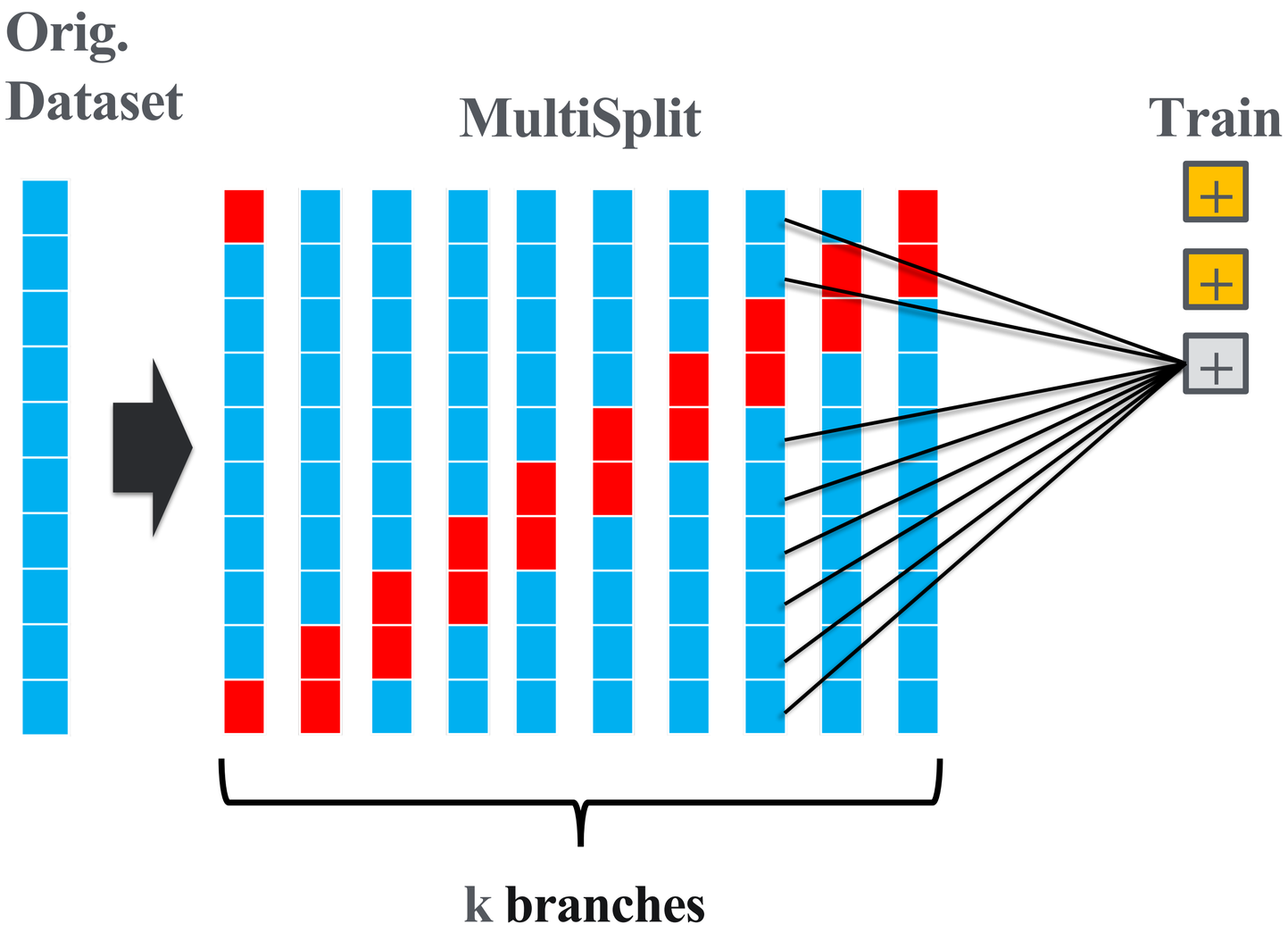
MultiSplit - Train – Check -Vote 알고리즘



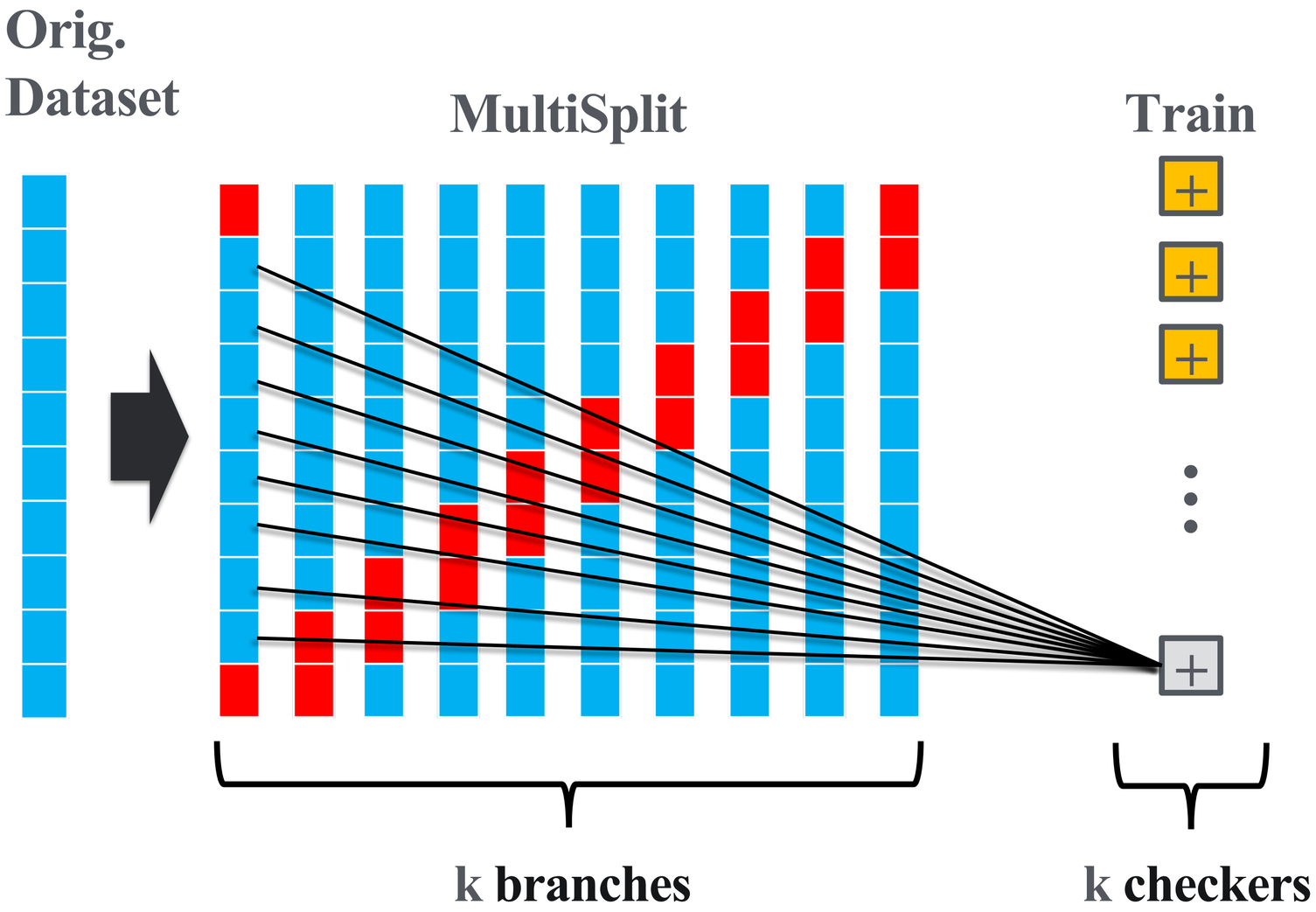
MultiSplit - Train – Check -Vote 알고리즘



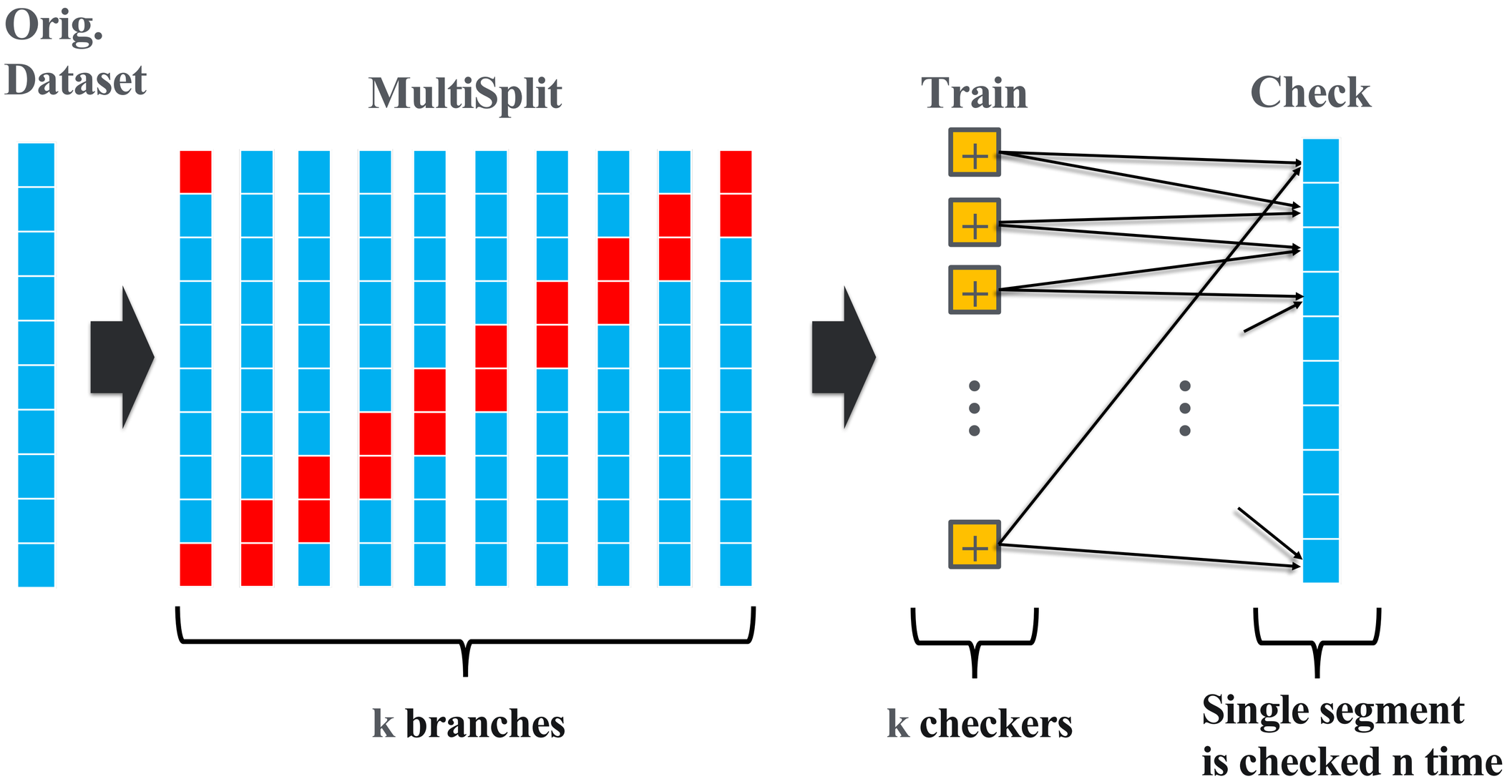
MultiSplit - Train – Check -Vote 알고리즘



MultiSplit - Train – Check -Vote 알고리즘

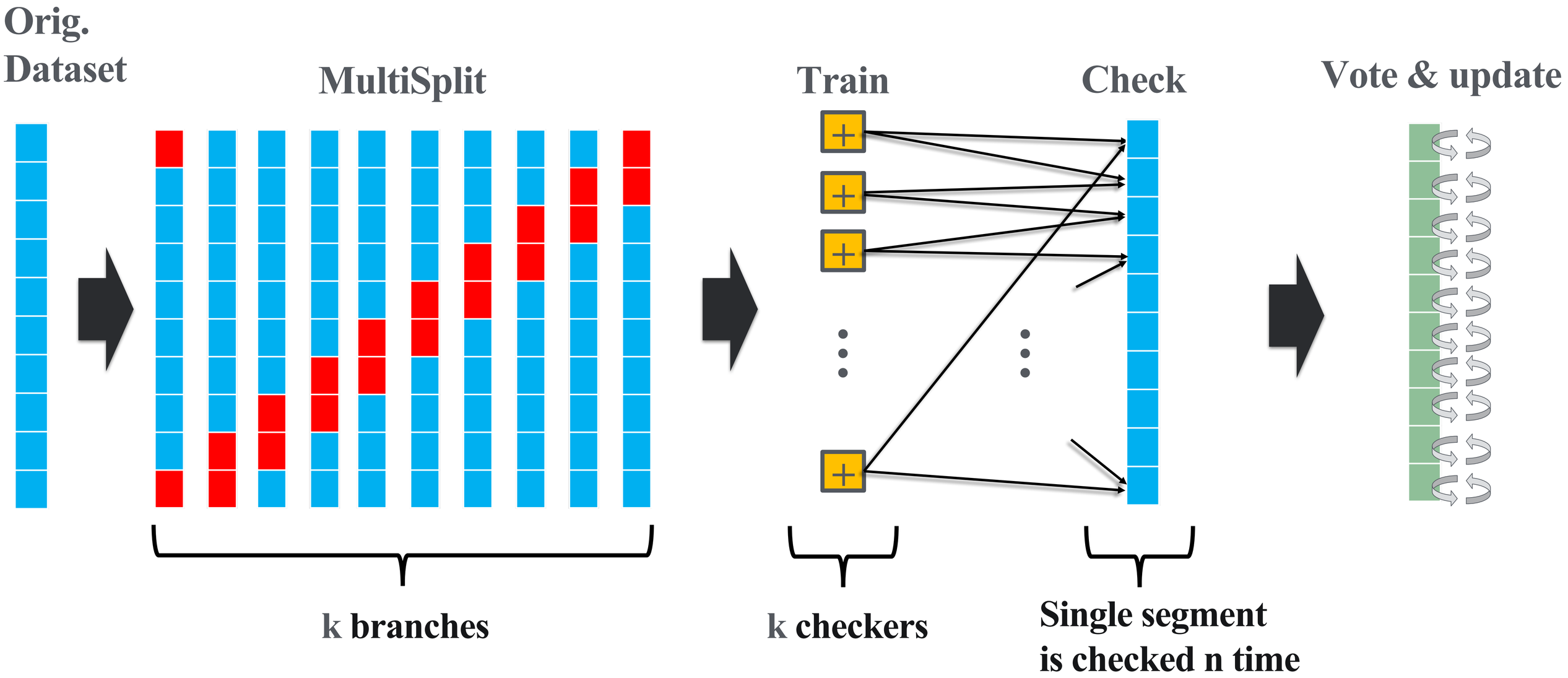


MultiSplit - Train – Check -Vote 알고리즘



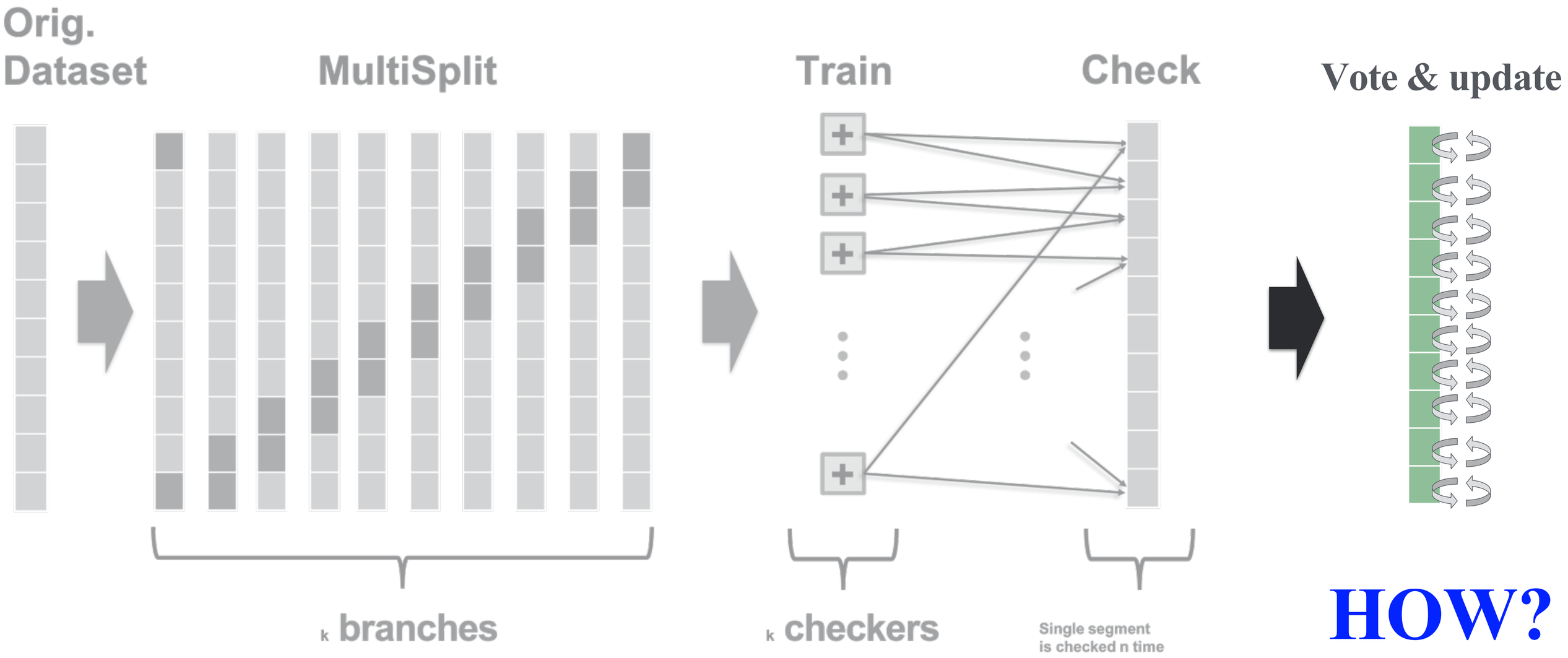
MultiSplit - Train – Check -Vote 알고리즘

DEVIEW 2019



MultiSplit - Train – Check -Vote 알고리즘

DEVIEW 2019



HOW?

DEVIEW
2019

어떻게 Vote하면 좋을까?

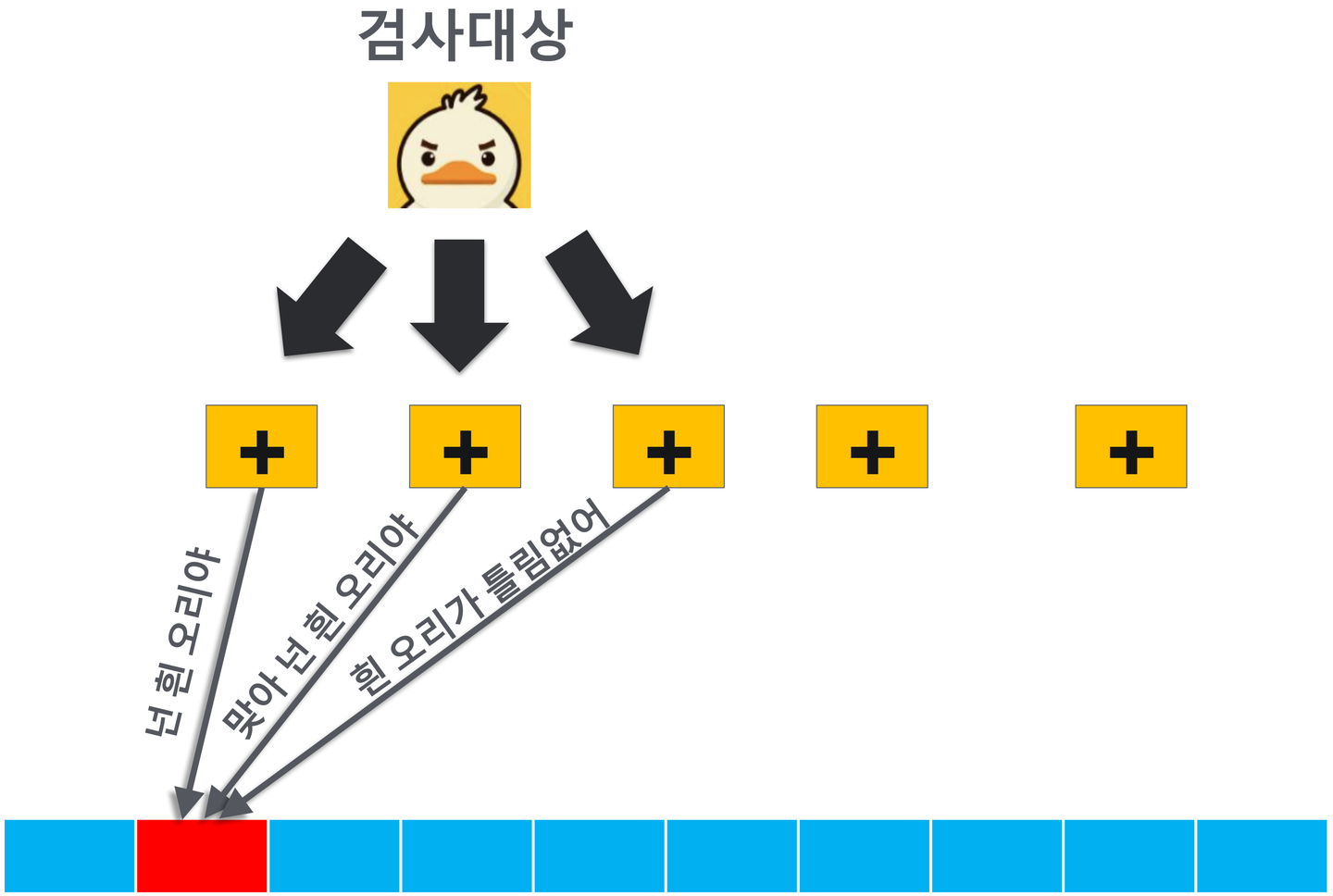
MultiSplit - Train – Check -Vote 알고리즘

Majority Vote:

MultiSplit - Train – Check -Vote 알고리즘

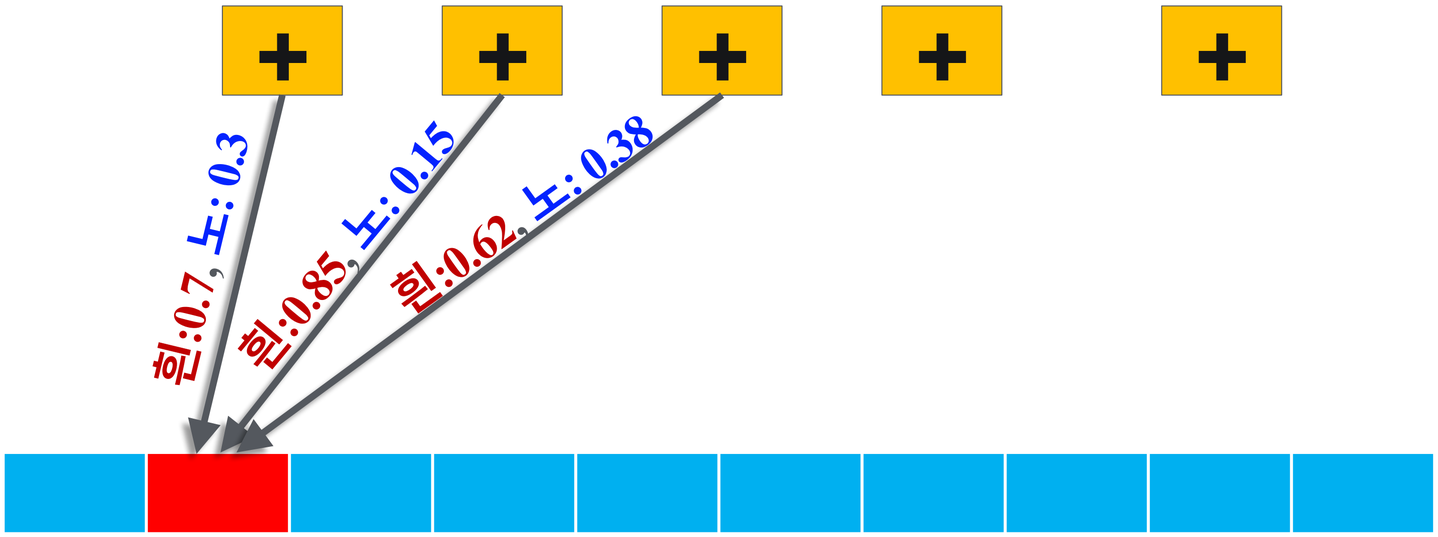
Majority Vote:

- 가장 단순한 방법
- n개 checker 들의 다수결
- 흰둥이: 노랭이 = 3 : 0



Checker의 결과가 확률 분포 (soft-value)인 것을 활용할 수 있을까?

PICO: Probabilistic Iterative COrrection



PICO 핵심 3가지

DEVIEW 2019

Checker결과 베이지안 **확률 결합**

레이블링의 **Iterative Probabilistic correction**

레이블링 히스토리의

Hidden Markov Modeling 통한 **반영**

Thomas Bayes
1701-1761

Robert G. Gallager
1931-present

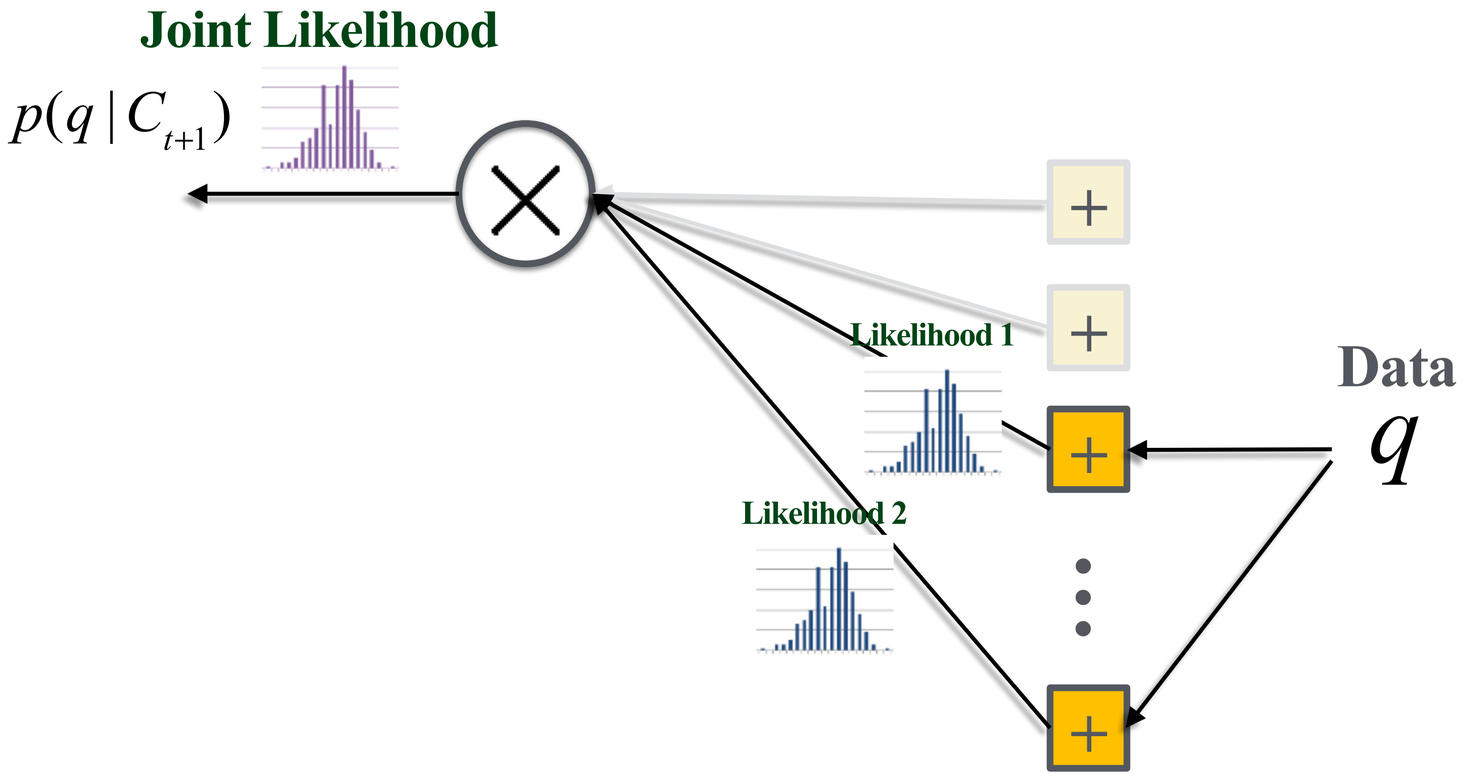
Andrey A. Markov
1856–1922

0) PICO Vote 시작!

DEVIEW 2019

Trained
k Checkers

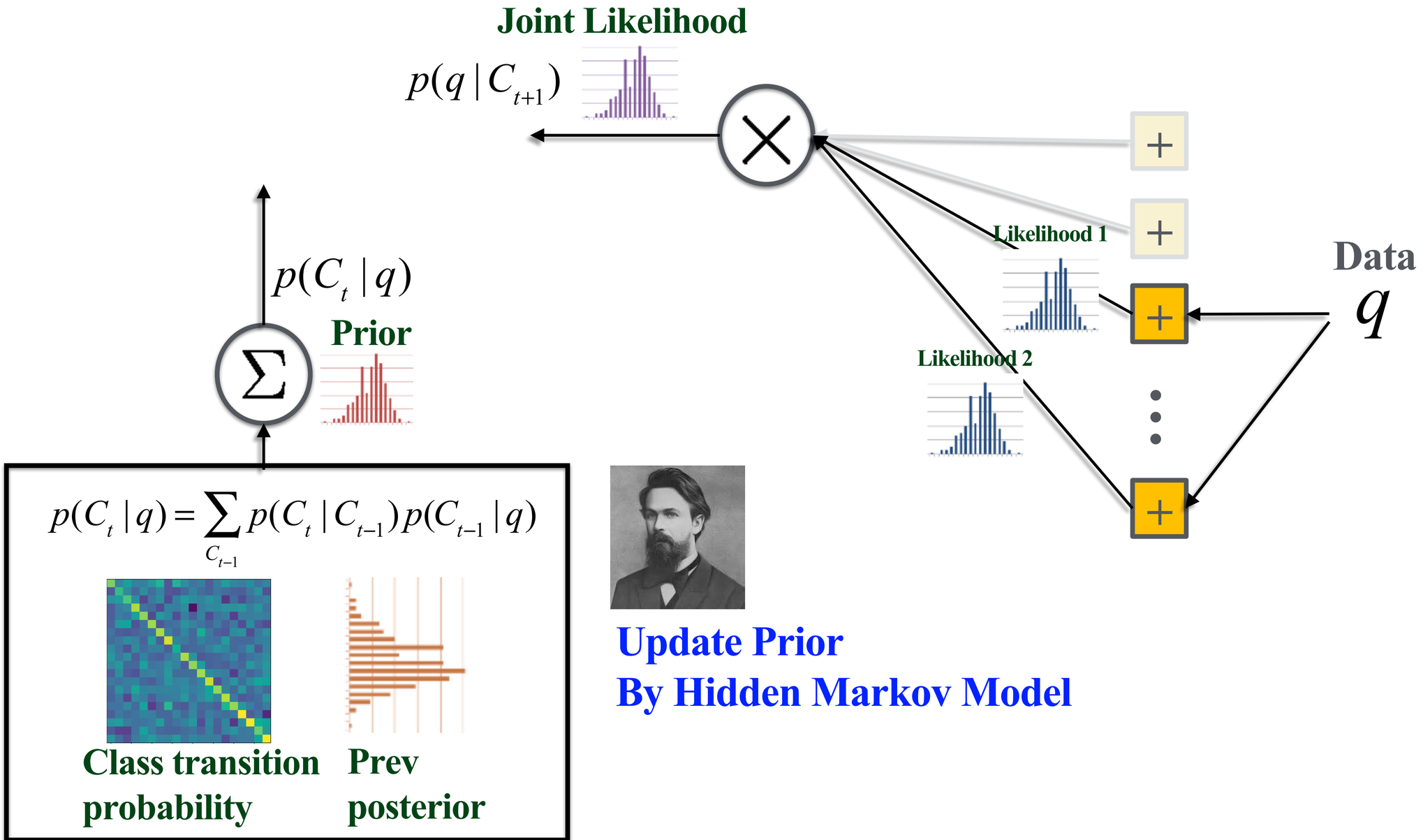
1) Checker결과 확률적 Vote



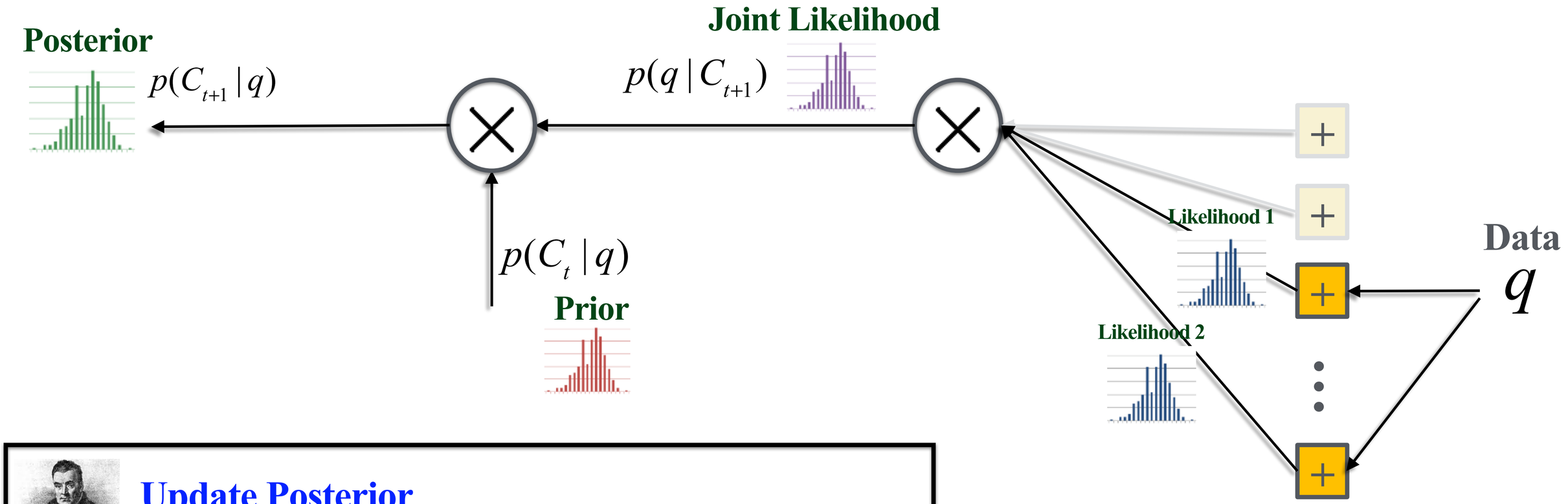
Take n related check results from k

$$p(q \mid C_{t+1}) \propto \prod_{k \in ne(q)} check_k(q)$$

2) 레이블링 사전 히스토리 업데이트



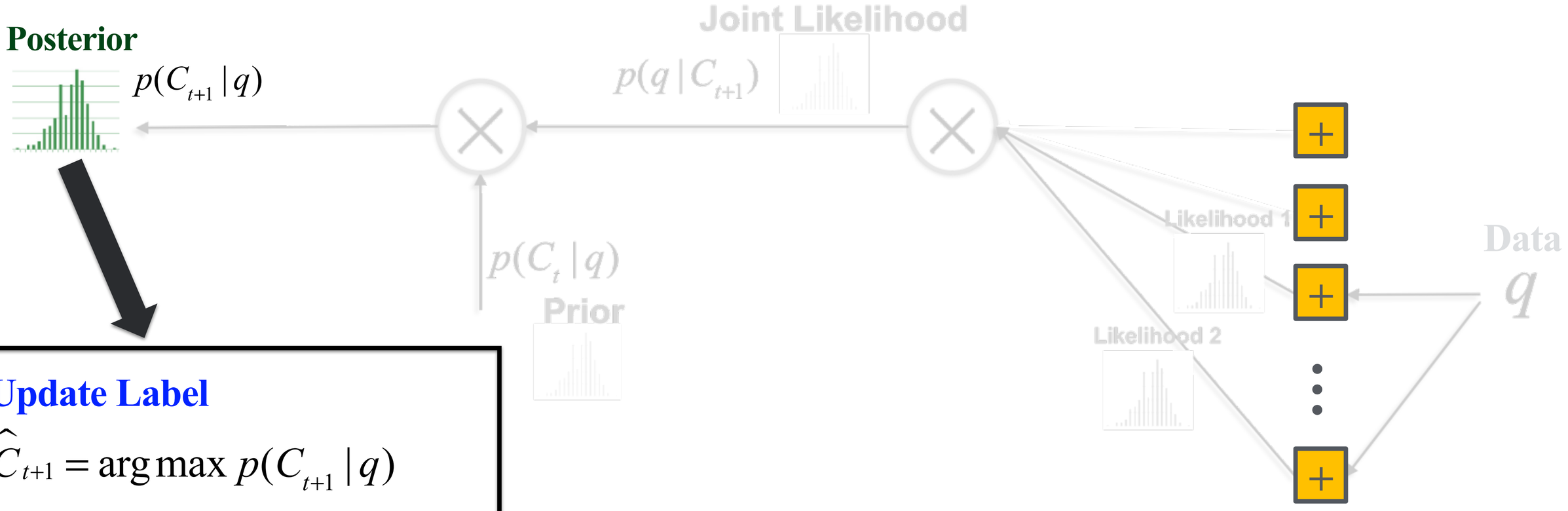
3) 베이즈 규칙으로 사후확률 계산



Update Posterior
By Bayes Rule

$$p(C_{t+1} \mid q) = \frac{1}{Z} p(C_t \mid q) \times \prod_{k \in ne(q)} check_k(q)$$

4) 레이블 업데이트!



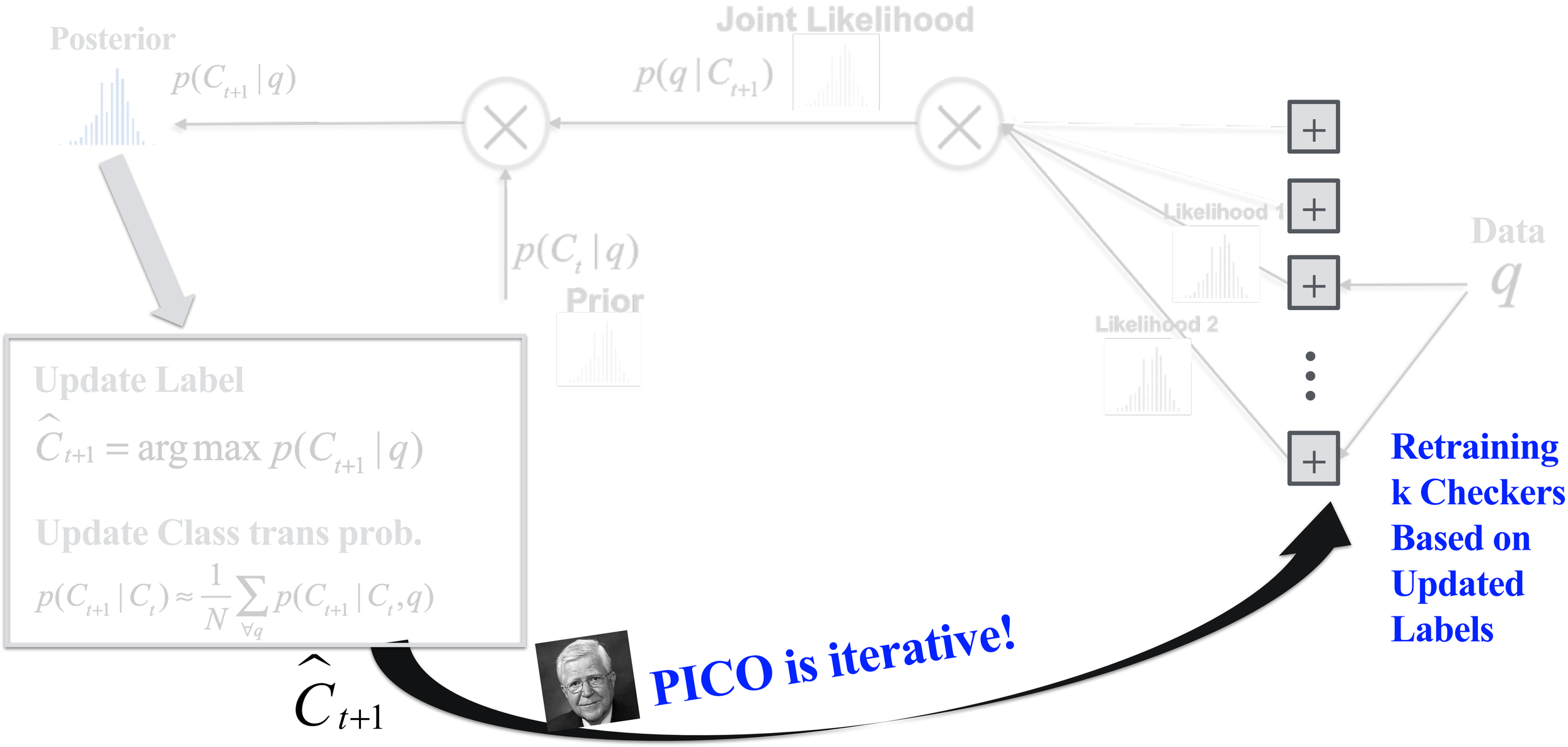
Update Label

$$\hat{C}_{t+1} = \arg\max p(C_{t+1} \mid q)$$

Update Class trans prob.

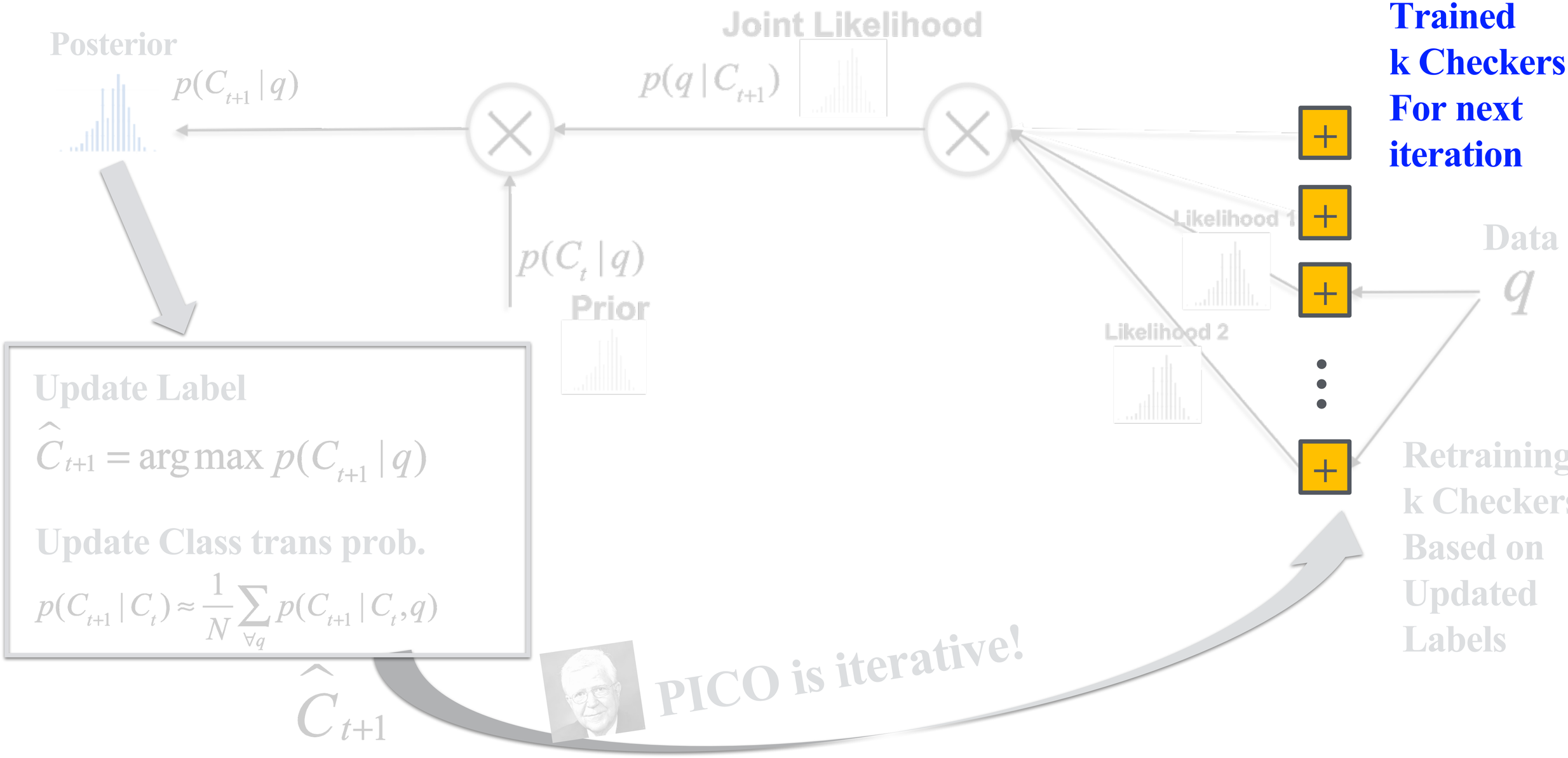
$$p(C_{t+1} \mid C_t) \approx \frac{1}{N} \sum_{\forall q} p(C_{t+1} \mid C_t, q)$$

5) 다음 Iteration을 위한 확률값 전달



0) 다음 Iteration 시작!

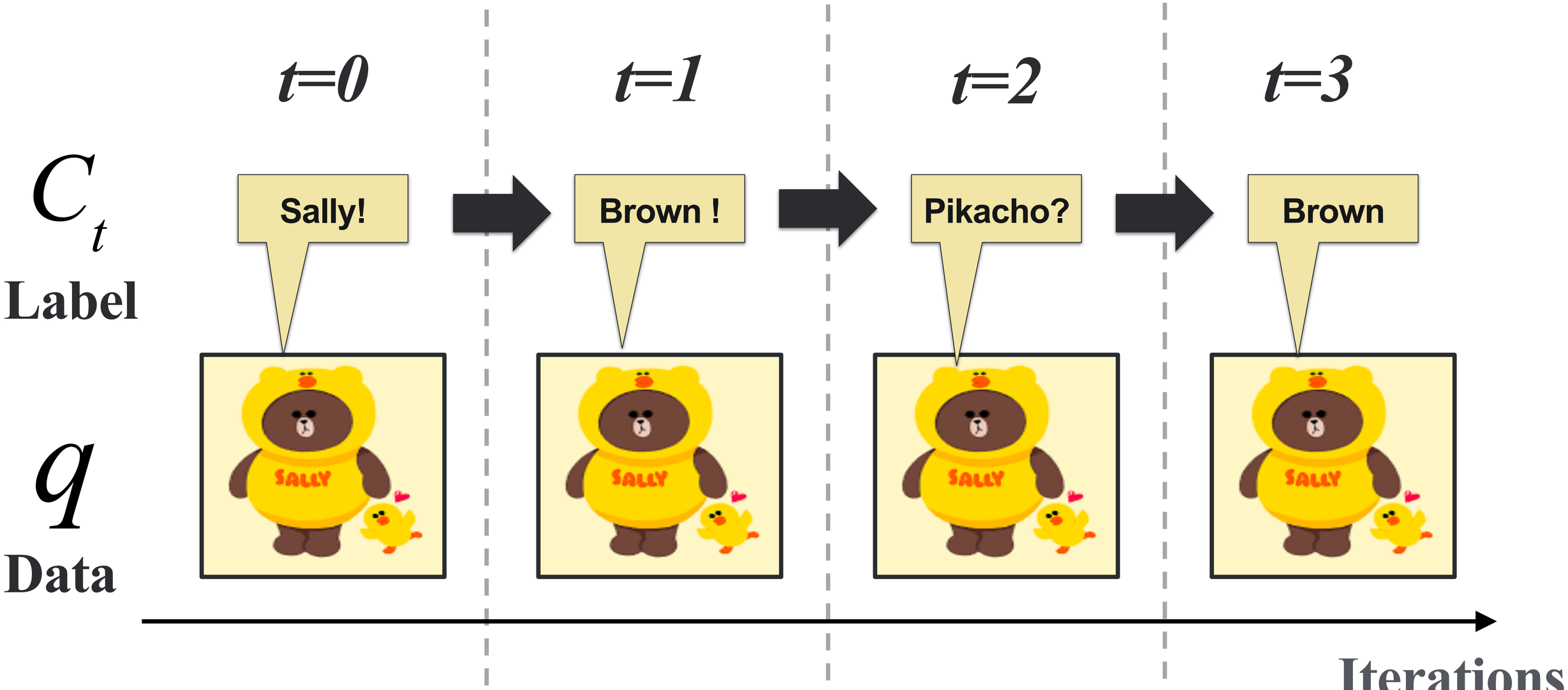
Trained
k Checkers
For next
iteration



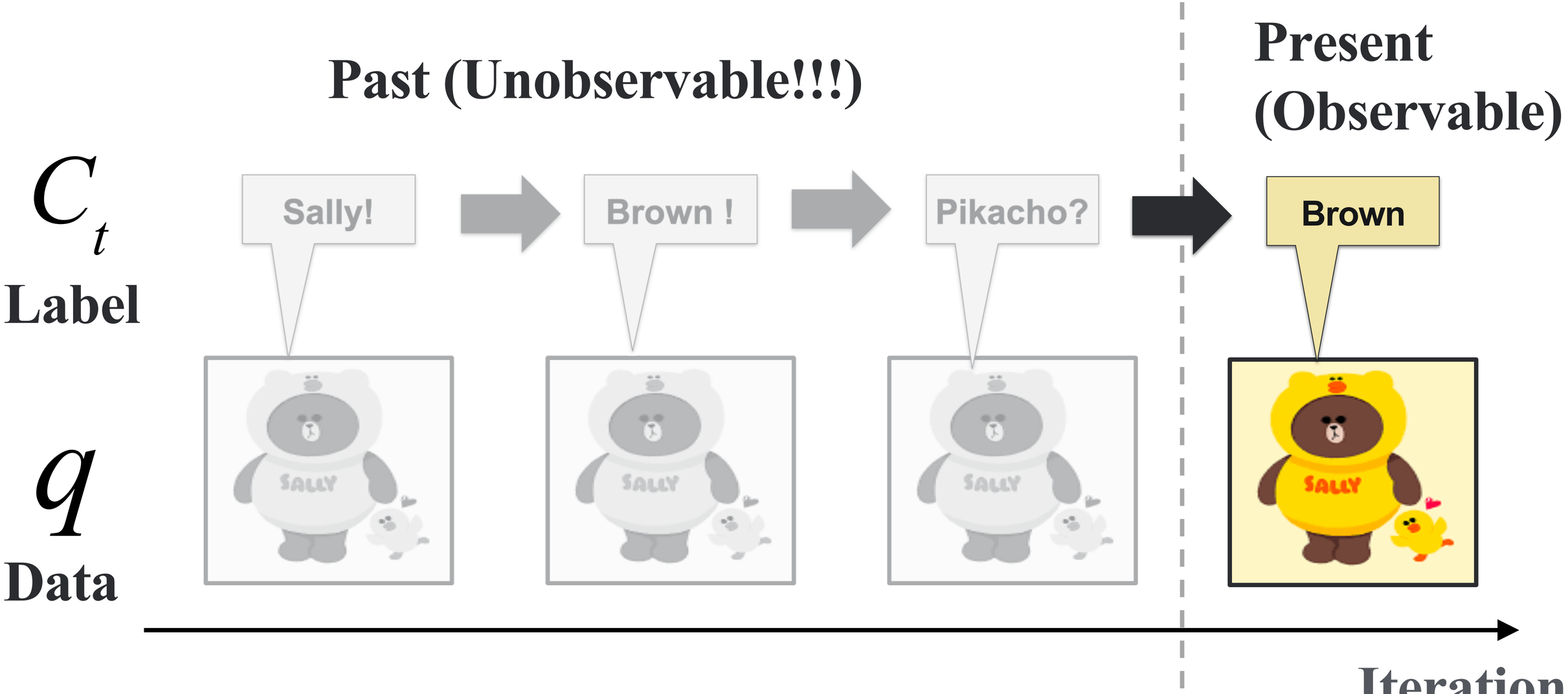
반복적 확률적 Vote를 통해서
점진적으로 레이블 노이즈를 제거한다

PICO는 Iteration에 따라서 어떻게 확률정보를 전달할까?

레이블링 히스토리 HMM 모델링



레이블링 히스토리 HMM 모델링



레이블링 히스토리 HMM 모델링

Past (Unobservable)

Present
(Observable)

C_t
Label

$$p(C_{t=0}, C_{t=1}, C_{t=2} \mid q)$$

q
Data

Iteration에 걸쳐서 확률값을
저장하여 전달해야함!

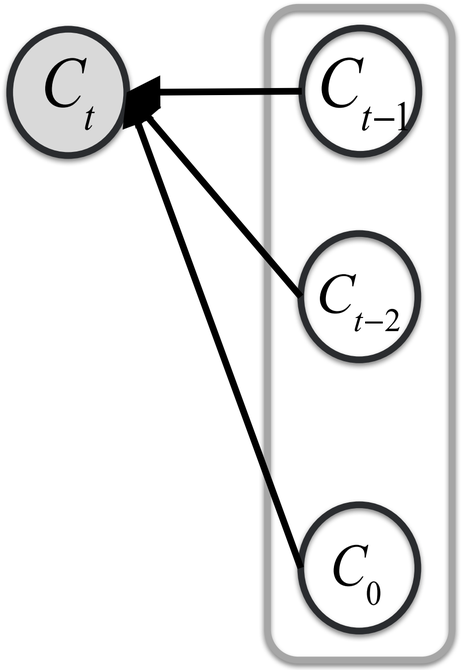
Iterations

레이블링 히스토리 HMM 모델링

DEVIEW 2019

❖ 레이블링 히스토리의 확률적 모델링

Naïve Model



Fully connected

- 과거 모든 히스토리를 고려

# of double precision values to save

$$N \times N^T \text{ per iteration}$$

T: # of prev iterations

N: # of classes for labeling

레이블링 히스토리 HMM 모델링

❖ 레이블링 히스토리의 확률적 모델링

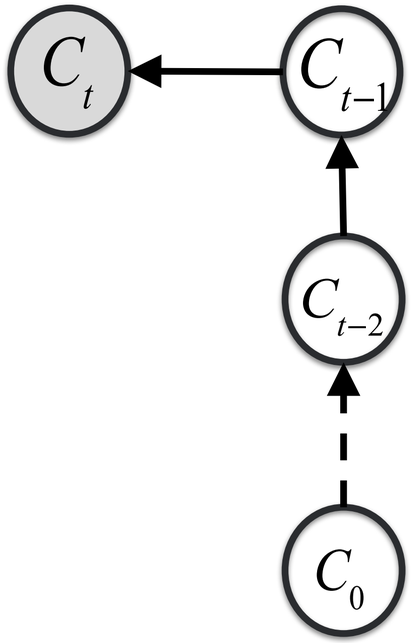
of double precision values to save

$$N \times N \text{ per iteration}$$

T: # of prev iterations

N: # of classes for labeling

Hidden Markov Model



- 직전 iter 히스토리만 고려

레이블링 히스토리 HMM 모델링

❖ **Prior density 계산 비교**

Naïve Model:

$$p(C_{t+1} \mid q) = \sum_{\{C_t, C_{t-1}, \ldots C_0\}} p(\underbrace{C_{t+1}}_{\text{Present is observable}} \mid \underbrace{C_t, C_{t-1}, \ldots, C_0}_{\text{Past is not observable}}) \underbrace{p(C_t, C_{t-1}, C_{t-2} \ldots, C_0 \mid q)}_{\text{We preserve the past info from the prev iteration !!}}$$

Hidden Markov Model:

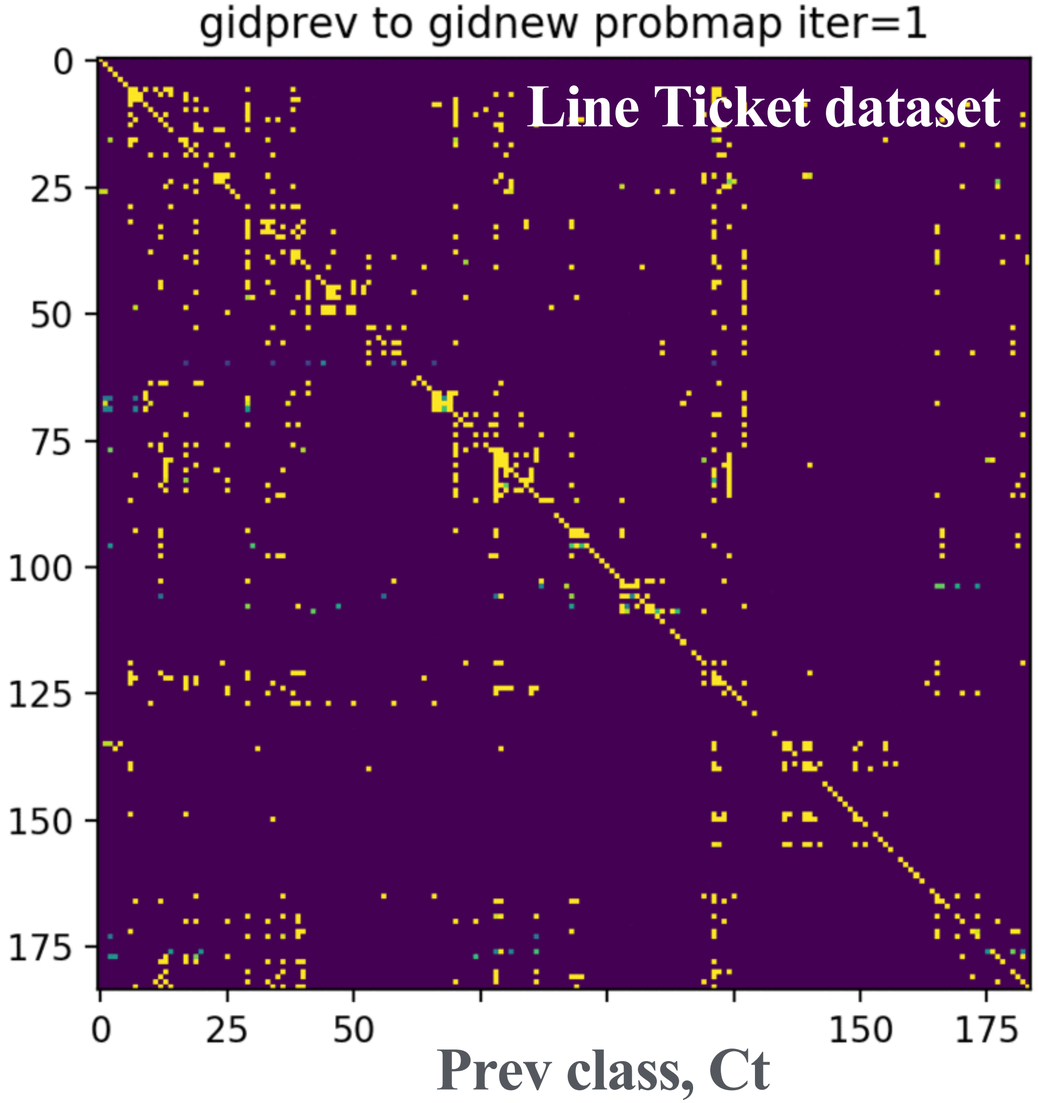
$$p(C_{t+1} \mid q) = \sum_{C_t} \underbrace{p(C_{t+1} \mid C_t)}_{\text{Assuming Markov process}} \quad p(C_t \mid q)$$

레이블링 히스토리 HMM 모델링

❖ Class Prior Density Evoluti

$$p(C_t \mid C_{t-1})$$

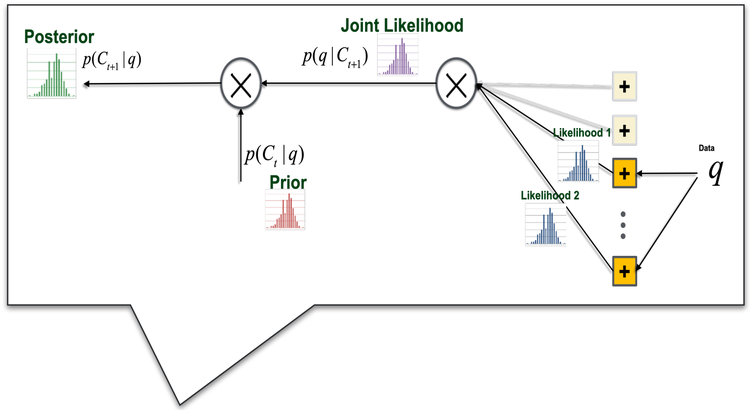
New class, Ct+1



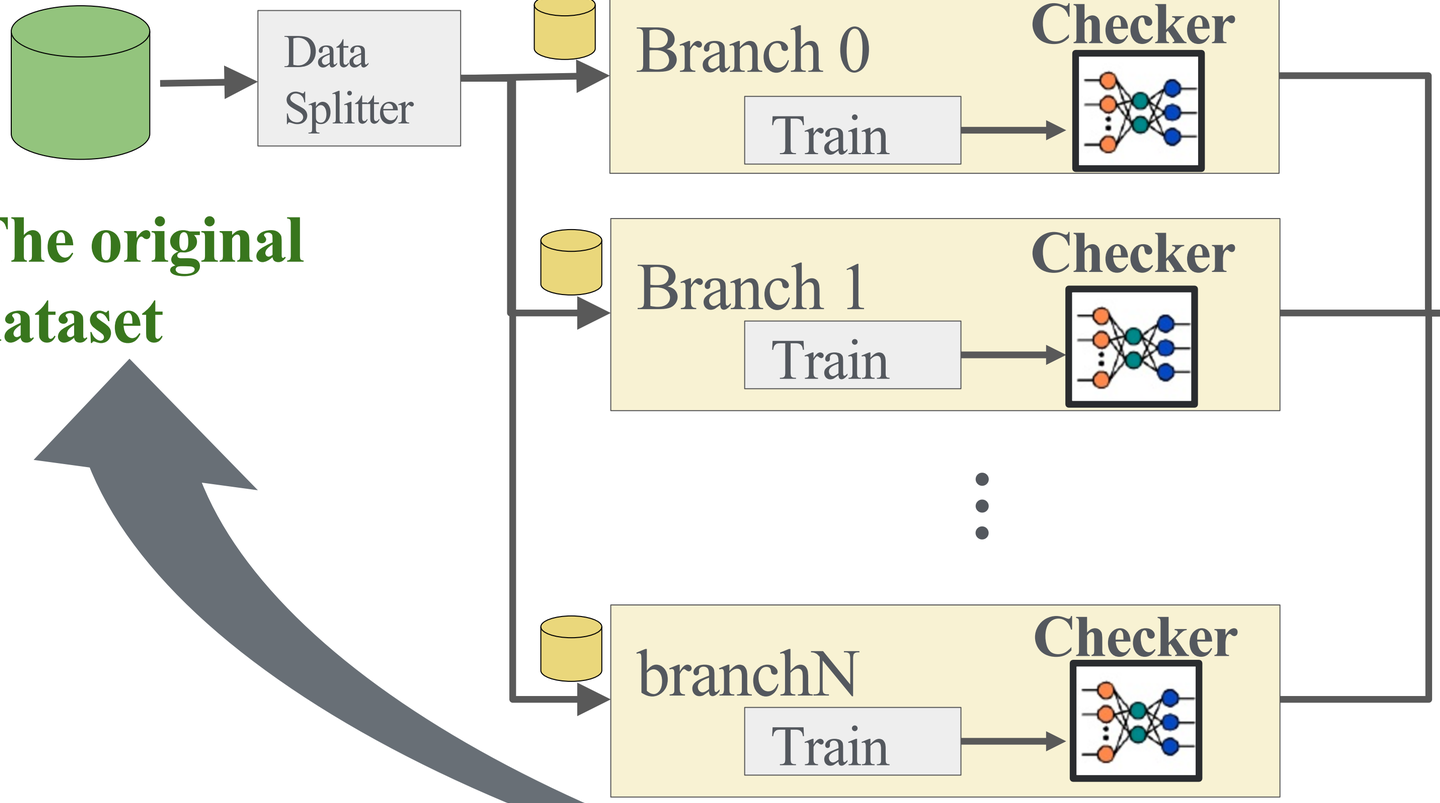
전체를 한그림으로 !!!

PICO Architecture

DEVIEW 2019



Split-Train-Check branches



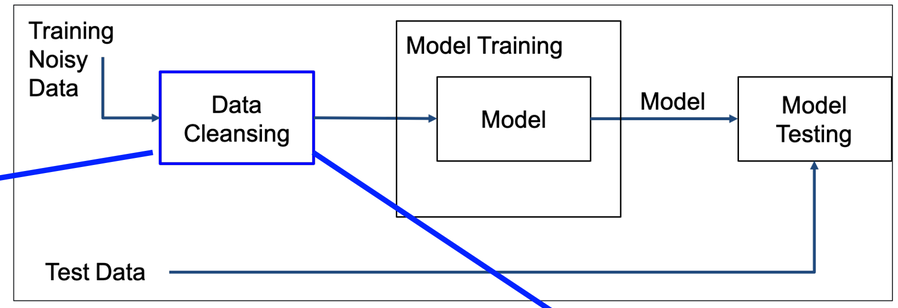
The original dataset

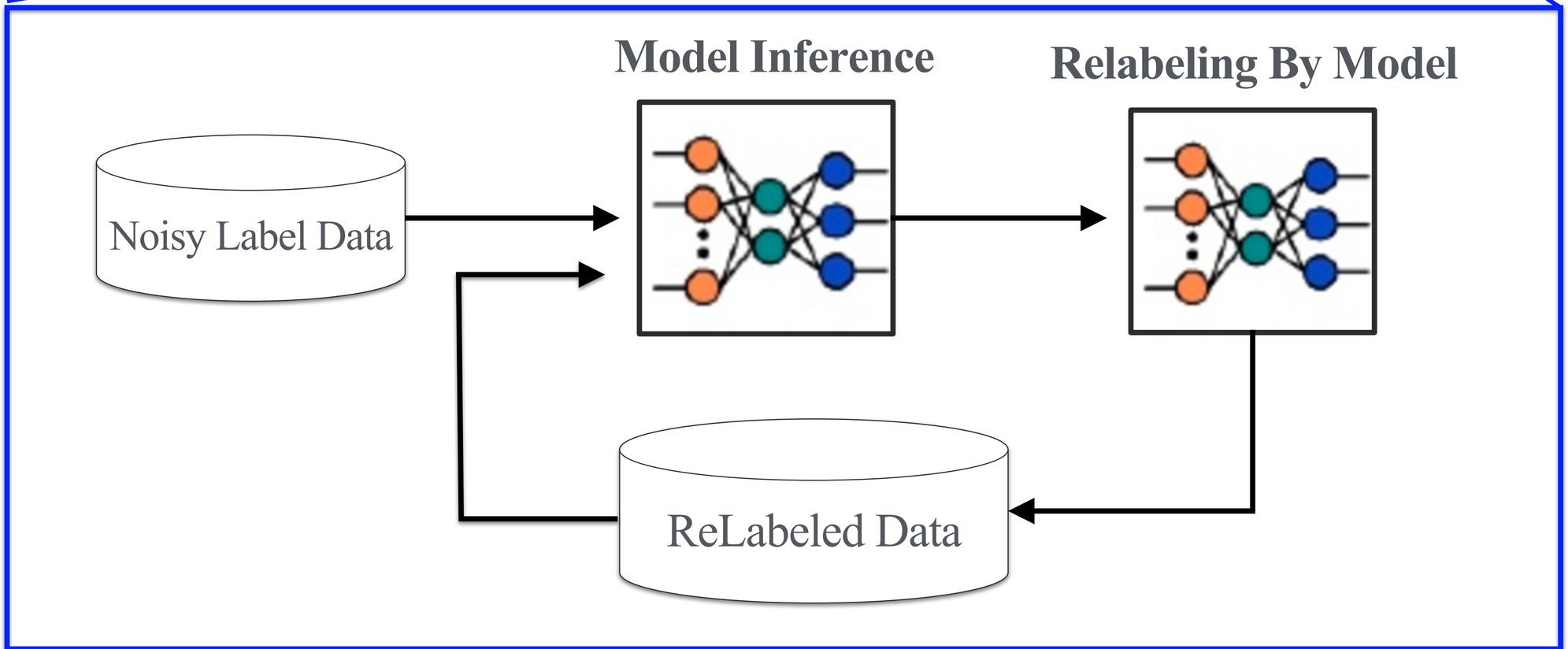
PICO

Relabeled dataset

PICO is iterative!

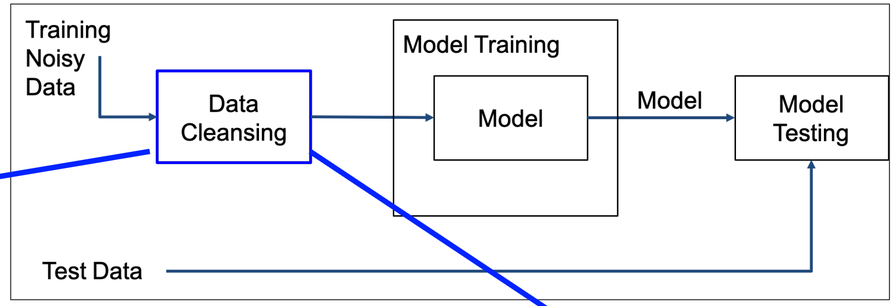
Our Plan

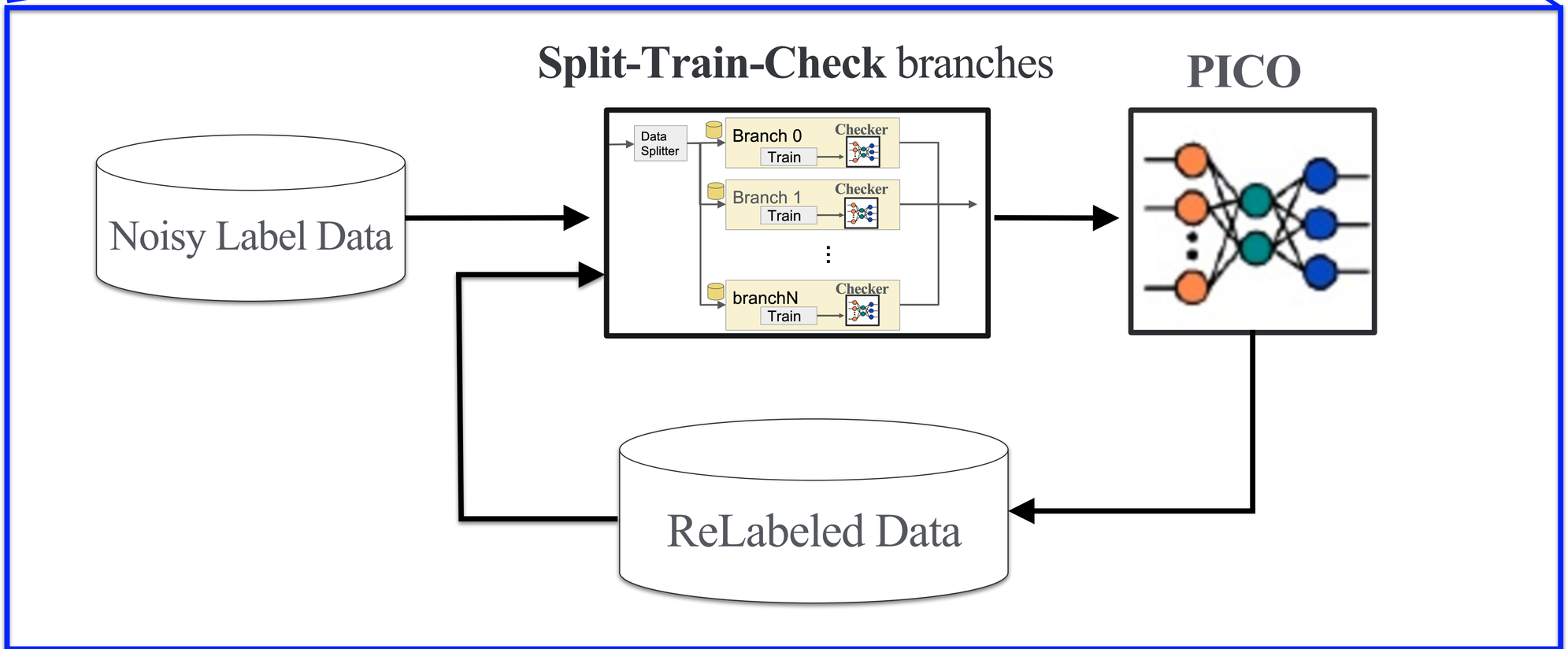

Training
Noisy
Data
Data
Cleansing
Model Training
Model
Model
Model
Testing
Test Data


Model Inference
Relabeling By Model
Noisy Label Data
ReLabeled Data

Our Plan became PICO!

DEVIEW 2019





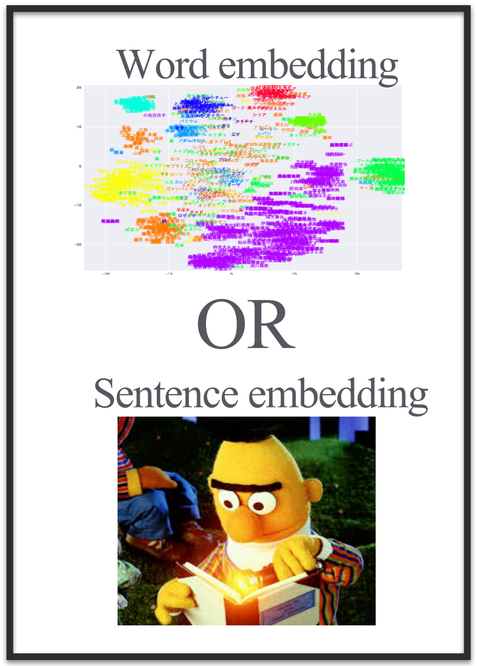
FAQ/Chat 데이터 셋에 적용해보기!

FAQ Chatbot

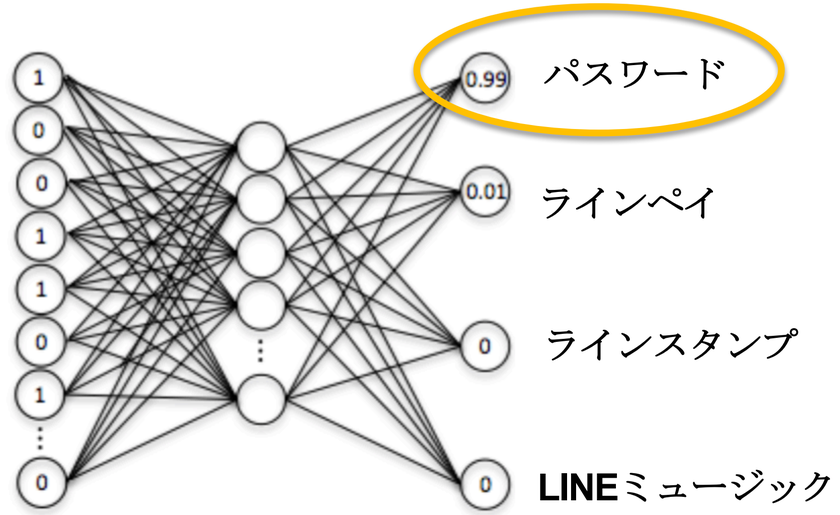
- 대화 도메인을 한정하여 높은 서비스 품질을 제공 가능
- Query-Intent Classification방식이 좋은 결과를 내고 있음

LINEのパスワードがわからないので、どうしたらいいですか？

Input query



Embedding



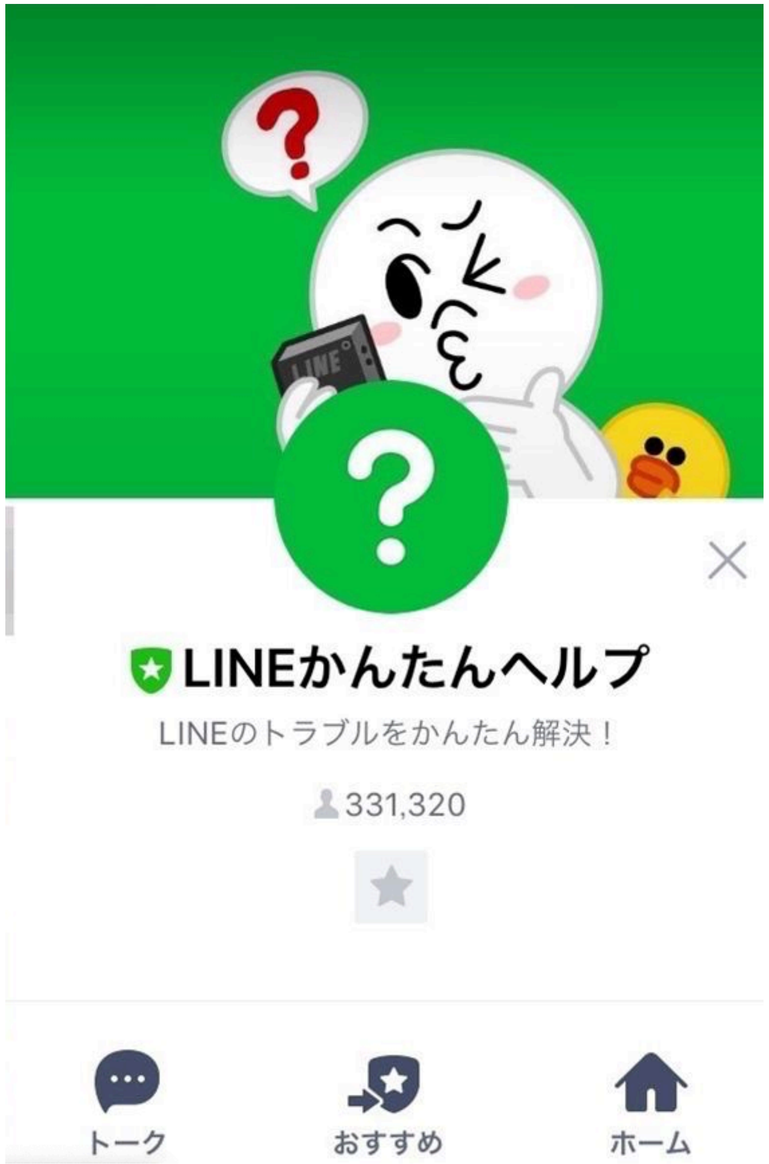
Neural Intent Classifier

LINE KanTanHelp FAQ

LINE사용에 관한 FAQ 톡 서비스

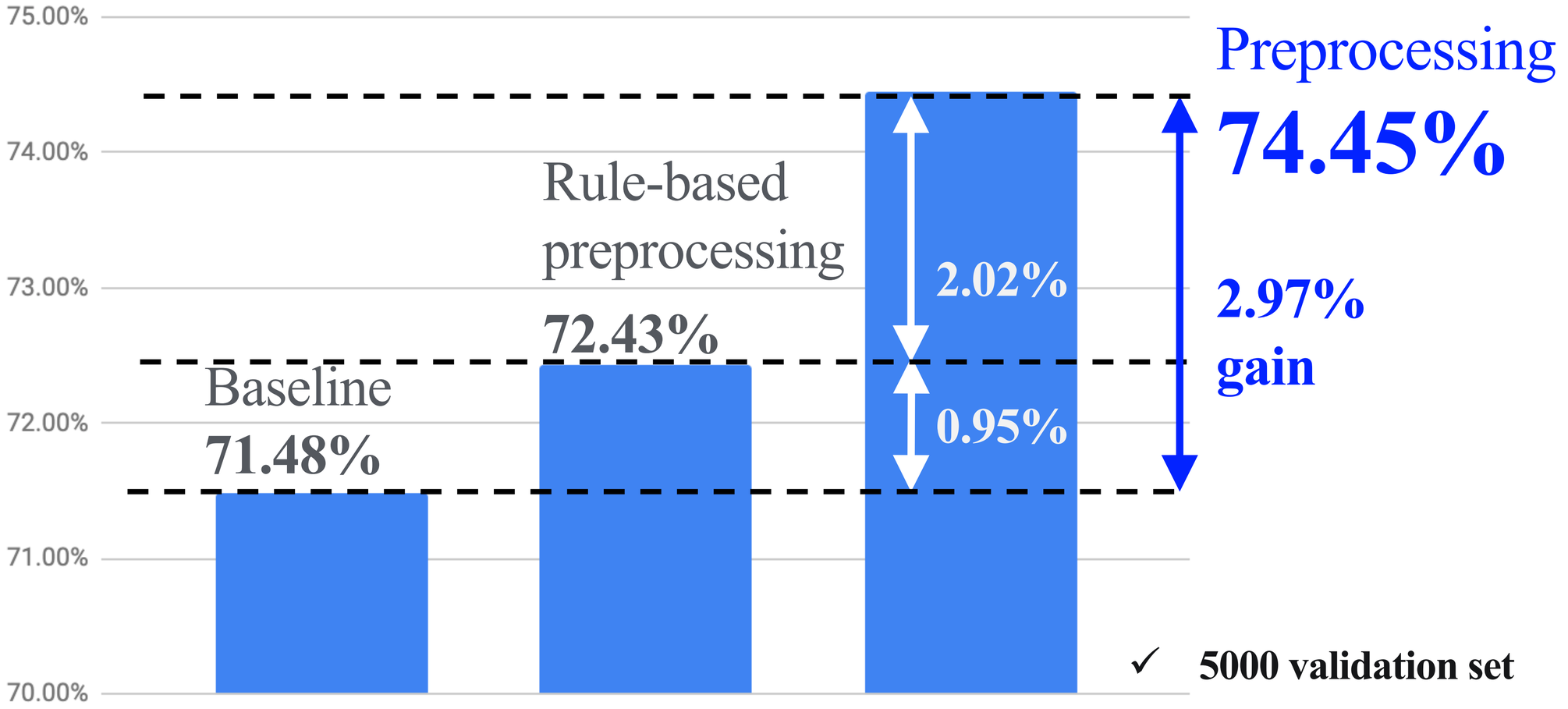
- Appx 90000 data / 1000 intent classes
- Checker model: classifier-BERT
  - ✓ **Train “classifier” only for the checkers in PICO**

- 특징
  - ✓ Imbalance class problem
  - ✓ Query-intent mapping ambiguity problem
  - ✓ Some noisy queries



LINE KanTanHelp FAQ

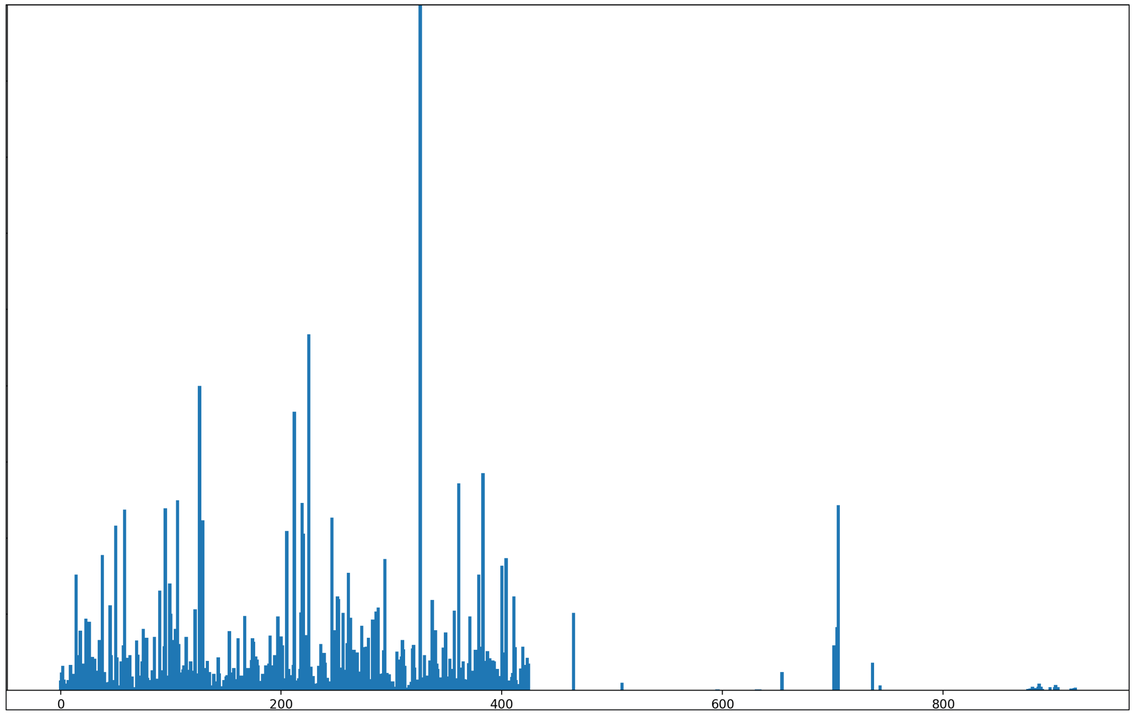
Top-1 accuracy: Classifier-BERT, single model



LINE KanTanHelp FAQ

❖ 데이터 셋 인텐트 분포의 변화 특징

- 쿼리의 이동: 일반적인 답변 인텐트 → 구체적인 답변 인텐트
  - ✓ 일반적인 답변으로 묶여있는 인텐트 class 사이즈 감소

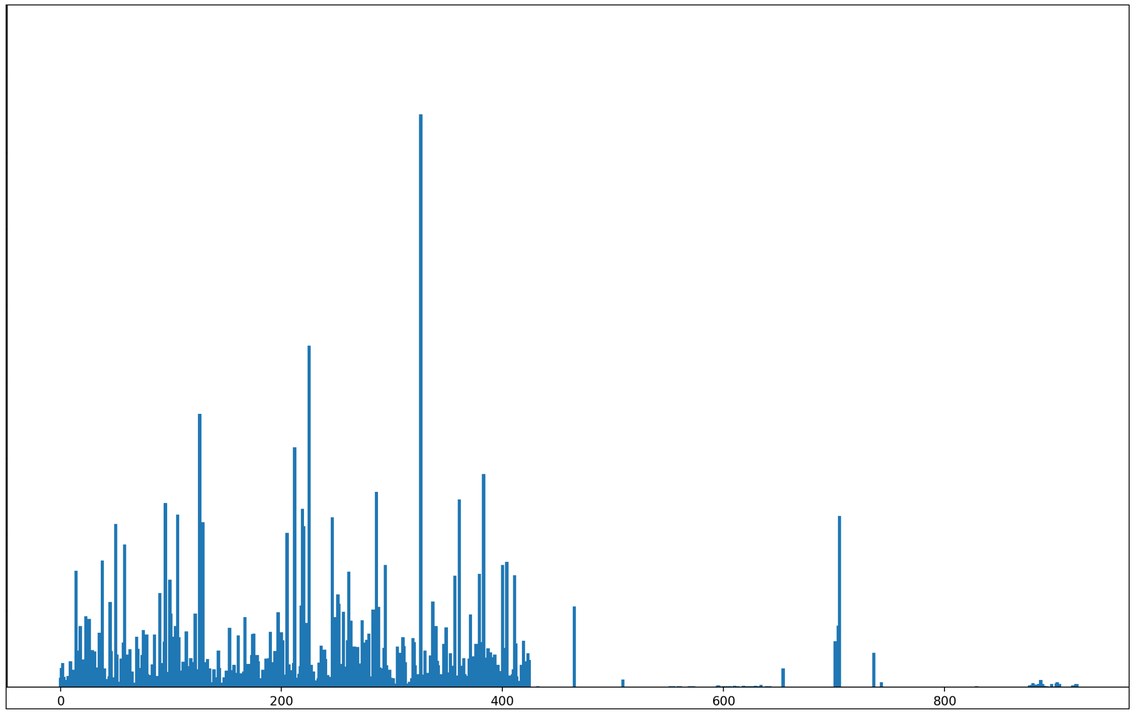


Iter-0

LINE KanTanHelp FAQ

❖ 데이터 셋 인텐트 분포의 변화 특징

- 쿼리의 이동: 일반적인 답변 인텐트 → 구체적인 답변 인텐트
  - ✓ 일반적인 답변으로 묶여있는 인텐트 class 사이즈 감소

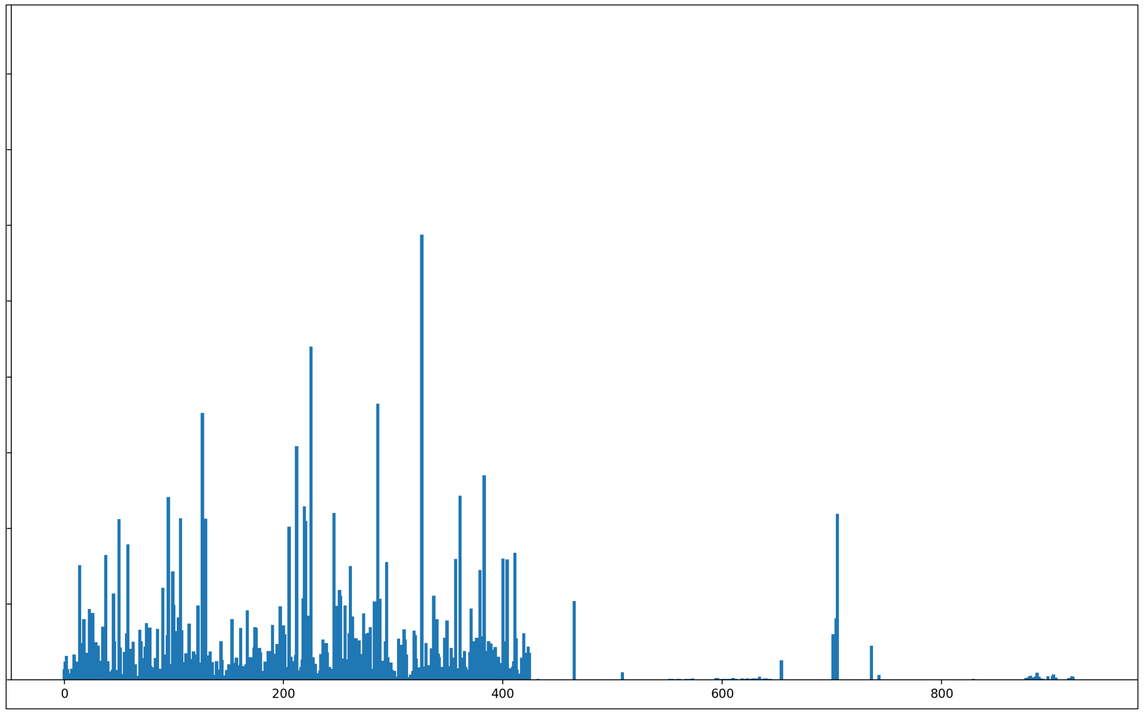


Iter-1

LINE KanTanHelp FAQ

❖ 데이터 셋 인텐트 분포의 변화 특징

- 쿼리의 이동: 일반적인 답변 인텐트 → 구체적인 답변 인텐트
  - ✓ 일반적인 답변으로 묶여있는 인텐트 class 사이즈 감소



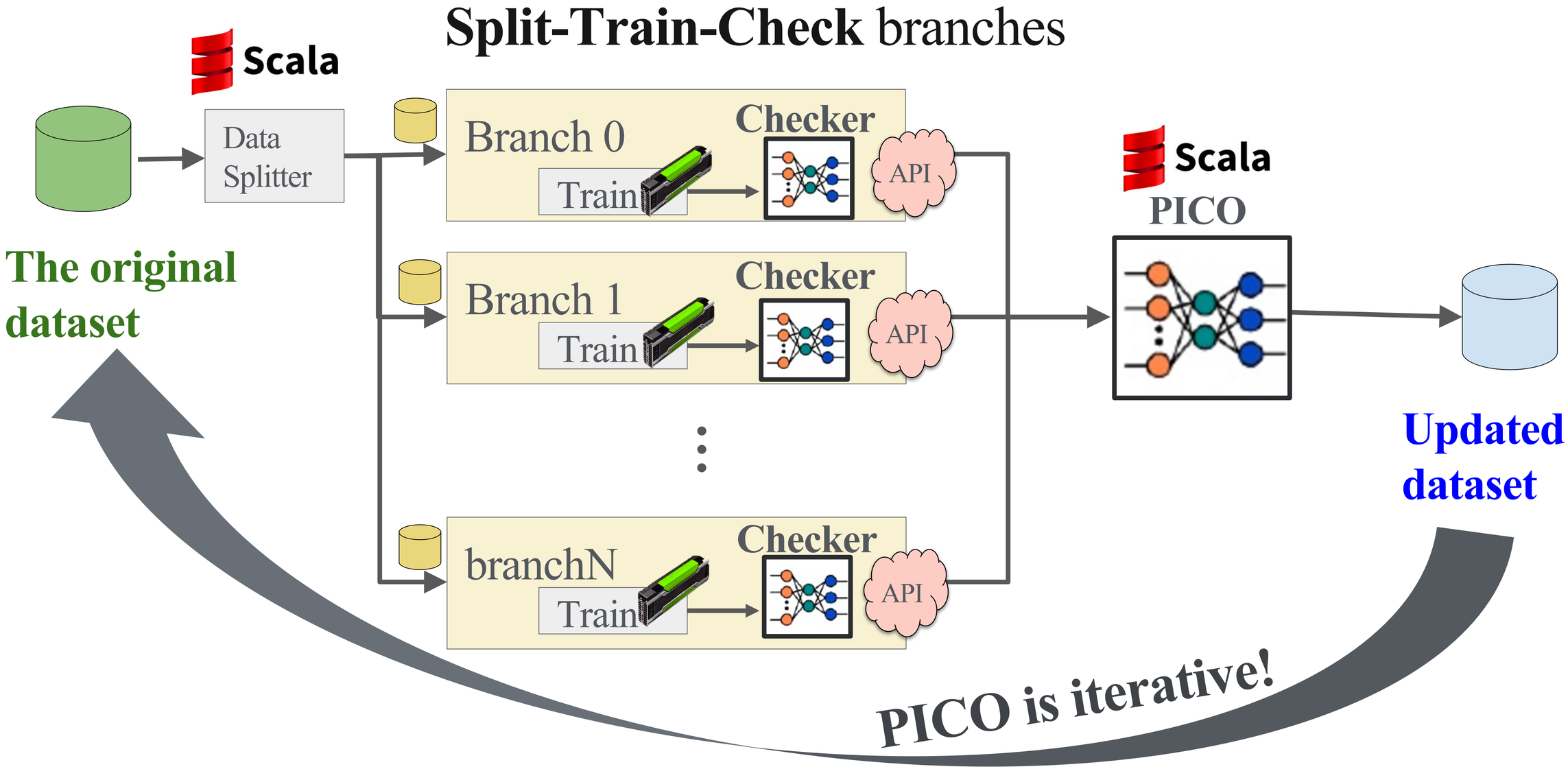
Iter-2

DEVIEW
2019

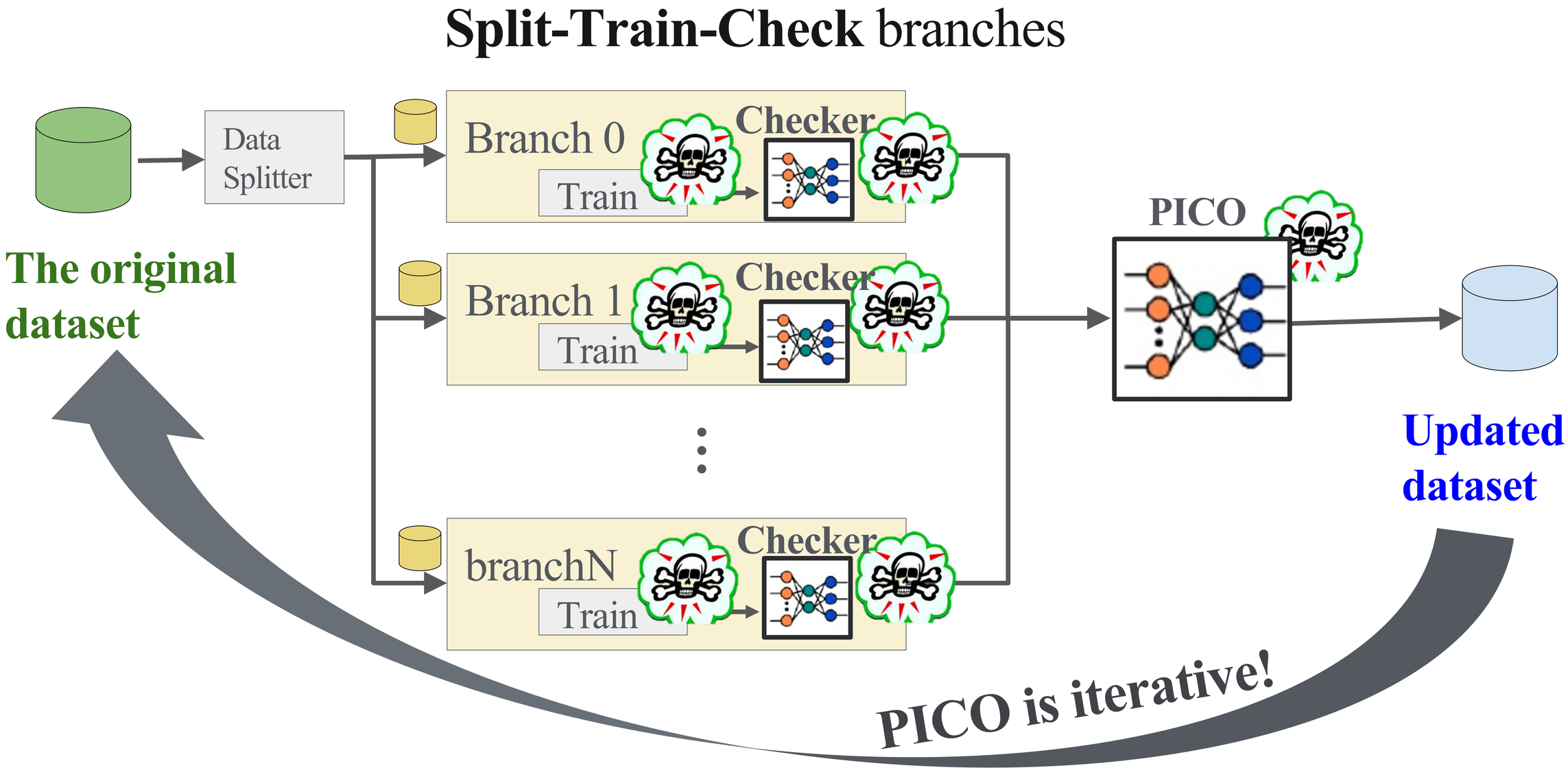
구현 삽질기
및
향후 해결과제

PICO를 설계 그대로
용감하게 구현했으나!

PICO Architecture



PICO Architecture에서 사고침



PICO 리소스 효율 개선

DEVIEW
2019

GPU 리소스 이슈 :

메모리 부족 이슈:

Inference API 서버 부하 이슈:

PICO 리소스 효율 개선

GPU 리소스 이슈 :

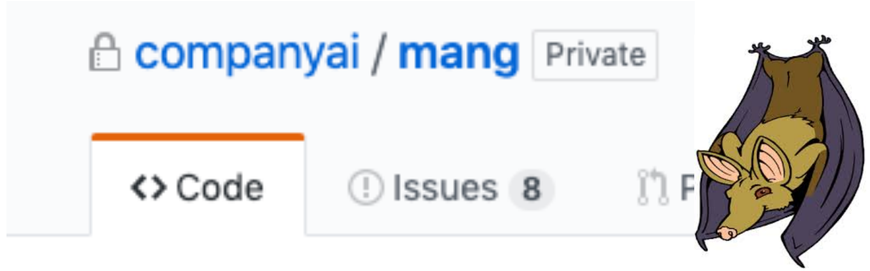
메모리 부족 이슈:

- SPARK 과 Sparse Matrix 도입하여 해결

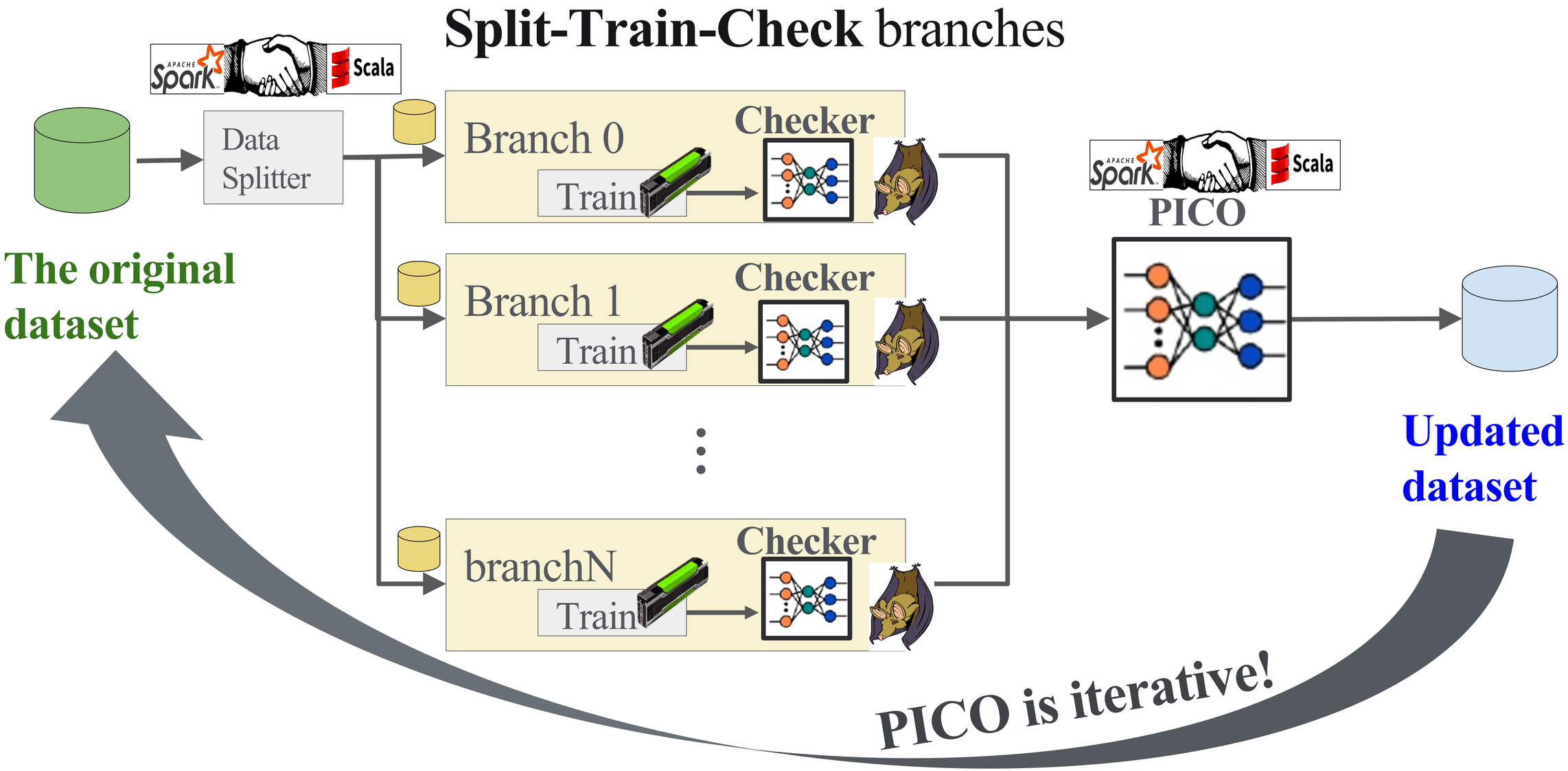
Inference API 서버 부하 이슈:

- 팀의 Local serving project의 비호로 해결

Project Mang



PICO 리소스 효율 개선



PICO 리소스 효율 개선

DEVIEW 2019

GPU 리소스 이슈 :

- **branch 수 최소화 + early stopping criterion 도입 예정**

❖ 메모리 부족 이슈:

- SPARK 과 Sparse Matrix lib 도입하여 해결

❖ Inference API 서버 부하 이슈:

- Local serving project 의 비호로 해결

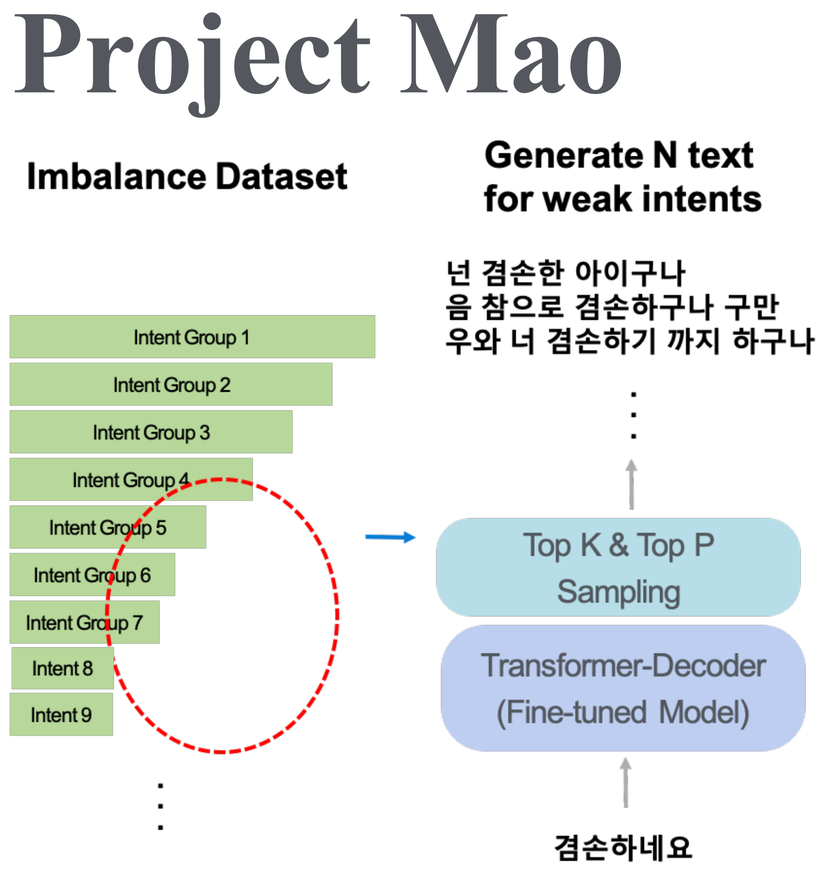
Further Works

❖ **효율개선**: PICO-trigger: 데이터 셋 품질 사전 검증 모듈

Further Works

❖ **효율개선:** PICO-trigger: 데이터 셋 품질 사전 검증 모듈

❖ **품질개선:** 생성모델를 통한 Imbalance Dataset 문제 해결



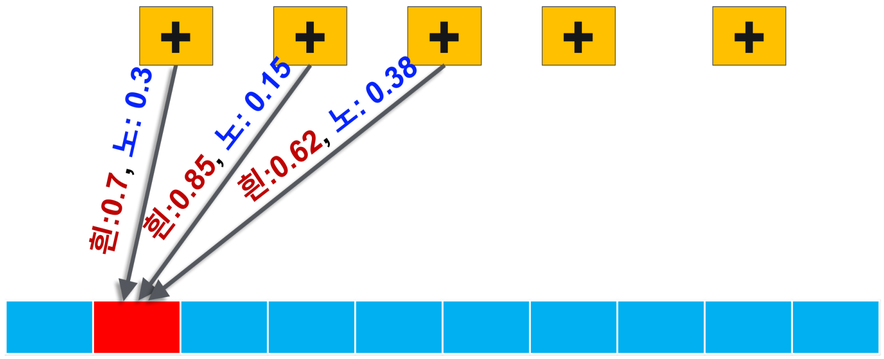
Further Works

DEVIEW 2019

❖ **효율개선:** PICO-trigger: 데이터 셋 품질 사전 검증 모듈

❖ **품질개선:** 생성모델를 통한 Imbalance Dataset 문제 해결

❖ **품질개선**: 다양한 Metric voting 방식 적용



Take Home Message!

데이터 레이블링 조금 잘못돼도 괜찮아요!

DEVIEW 2019

- 데이터 품질 전략이 없는 AI 프로젝트는 성공 하기 어려움
- AI 데이터 자동정제 파이프라인은 매우 큰 경쟁력
- **Naver Clova Chatbot Builder**는 **PICO**를 통해서 데이터 자동 정제하여 서비스 품질 개선
- **PICO** 아키텍쳐는 다른 종류 데이터 셋에도 적용 가능

DEVIEW
2019

Human-free data cleaning can improve your AI performance

Chatbot팀과 함께할 분을 모십니다!

DEVIEW 2019

Clova Chatbot 모델러:

Clova Chatbot ML 플랫폼 개발:

Clova Chatbot 빌더&웹 개발:

Contact:

dl_recruit_clova_ai_biz@navercorp.com

Clova 부스에 놀러오세욧!

부스에서 꿀잼 프로젝트 소개중!

DEVIEW 2019

Chatbot Model

Chatbot / NLP Model Research for AI Business

Khan

AutoML Project for Chatbot AI Builder

Overview

- End2End CB building pipeline
- Clean dataset
- Select appropriate features
- Select an appropriate models fam
- Optimize hyper-parameters
- Optimize service-parameters
- Automated model validation

Automated Feature Eng.

- Intent and slot clustering
- Noisy query filtering
- Auto label error correction

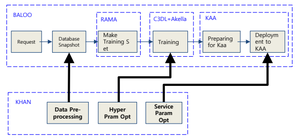
Hyper-Param Opt.

- Coordinate search
- Genetic algorithm
- Bayesian optimization

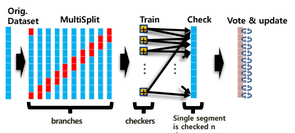
Service-Param Opt.

- Model ensemble weight
- Model selection

Chatbot Building Process

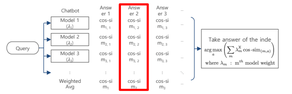


PICO: Auto Lable Correction



Coordinate Search for HiddenSizes

Model Ensemble Weight Calculation



Mao

Text Generation in Dialogue Model

Overview

Text generation research project for improving dialogue model qualities.
From normal generations to context based generations such as *GPT2 & *CTRL.

Text Augmentation for Imbalance dataset

Dealing with imbalance data by generating queries.

Imbalance Dataset

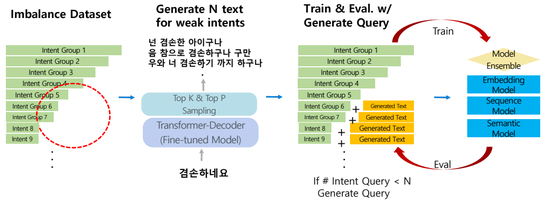
Generate N text for weak intents

넌 겸손한 아이구나
용 참으로 겸손하구나 구만
우와 너 겸손하기 까지 하구나

Top K & Top P Sampling
Transformer-Decoder (Fine-tuned Model)

겸손하네요

Train & Eval. w/ Generate Query

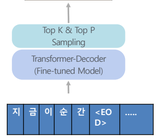


If # Intent Query < N
Generate Query

Answer Generation

Convers the retrieval model for weak domains.

Train & Generate



Retrieval vs. Generation

니 이름이 뭐야?	꽤 유명한데 아직 모르시나 봐요
식사는 언제 하실 꺼에요?	전 언제 어디서든 도움을 주는 친구 클로바입니다. 당신이 먹는 것만 봐도 배가 불러요
여행가고 싶어	와이파이 한 조각을 먹고 왔답니다. 바보는 방향을 하고 현명한 사람은 여행을 떠난다고 해요 여행은 인생의 좋은 경험이 될 거에요
오늘 달리기 시합 있어	커피 한잔의 여유는 필요하죠
어이 상실	건강을 위해 운동을 해 보세요 네, 부르셨나요
지금 이 순간	누구나 실수는 할 수 있어요 보고 싶은 사람이 있다는 건 좋은 일이에요 당신과 함께라면 항상 행복해요

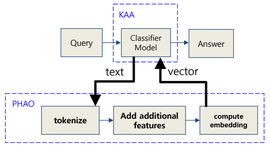
*GPT2: Language Models are Unsupervised Multitask Learners
*CTRL: A Conditional Transformer Language Model for Controllable Generation

Phao

API providing sentence embeddings

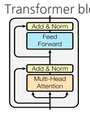
Overview

Sentence level embeddings
serving with the API



Model Details

Transformer blocks



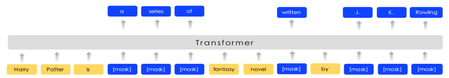
$$\text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_K}}\right)\mathbf{V}$$

Relative Position Representations

$$a^{V}_{2,1}=w^{V}_{-1} \quad a^{V}_{2,4}=w^{V}_{2} \quad a^{V}_{4,n}=w^{V}_{k}$$
$$a^{K}_{2,1}=w^{K}_{-1} \quad a^{K}_{2,4}=w^{K}_{2} \quad a^{K}_{4,n}=w^{K}_{k}$$

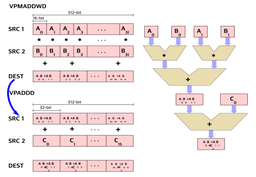
x_1 x_2 x_3 x_4 ... x_n

N-Gram Masked Language Model



Inference Optimization

Custom Tensorflow
Intel Math Kernel Library
Intel Advanced Vector Extensions 512



Bagheera

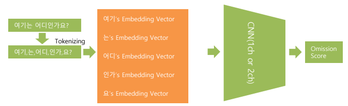
Evaluation Model of Omission and Continuity Dialogue

Overview

- Multi-turn dialogue understanding is a major challenge for building chatbot model. We have the following multi-turn tasks:

Omission Evaluation Model

- Detect sentences omitting words in dialogue.

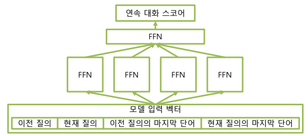


- 비생략: 한강에 갈래?
- 생략 : 가자

→ 생략 : "한강에"가 생략

2-Turn Continuity Dialogue Model

- Recognize the next sentence in conversation has the same intent



Answer Concatenation Model

- Regeneration of user query to more include the context of conversation based on the previous answer intent.

시스템 답변 : 오늘 하루 즐거우셨어요?
사용자 질의 : 그래
실제 사용자 질의 : 오늘 하루 즐거웠다.

References

- **[Bo Han, et al.: 2019]** Han, Bo et al.,“Co-teaching: Robust training of deep neural networks with extremely noisy labels,” in *NIPS*, pp. 8536-8546, 2018.
- **[Konyushkova’2017]** K. Konyushkova, et al., “Learning active learning from real and synthetic data,” in NIPS, 2017.
- **[Lu Jiang, et al.:2018]** Jiang, Lu et al., “Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels.” in ICML, pp. 2309-2318, 2018.
- **[Y. Bengio: 2009]** Bengio, Yoshua et al, “Curriculum learning,” in ICML, pp. 41–48, 2009
- **[Yoo’2019]** D. Yoo et al., “Learning loss for Active learning,” in CVPR, 2019