MLOps

Training Data

“Designing Artificial Intelligence Systems

Introduction to ML system design

Data Engineering fundamentals

Training data & Feature Engineering

Model development and offline evaluation

Model deployment & prediction services

Introduction to MLOps

“Designing Artificial Intelligence Systems

Designing Artificial Intelligence Systems

Training Data

Handling Lack of Labels

Sampling

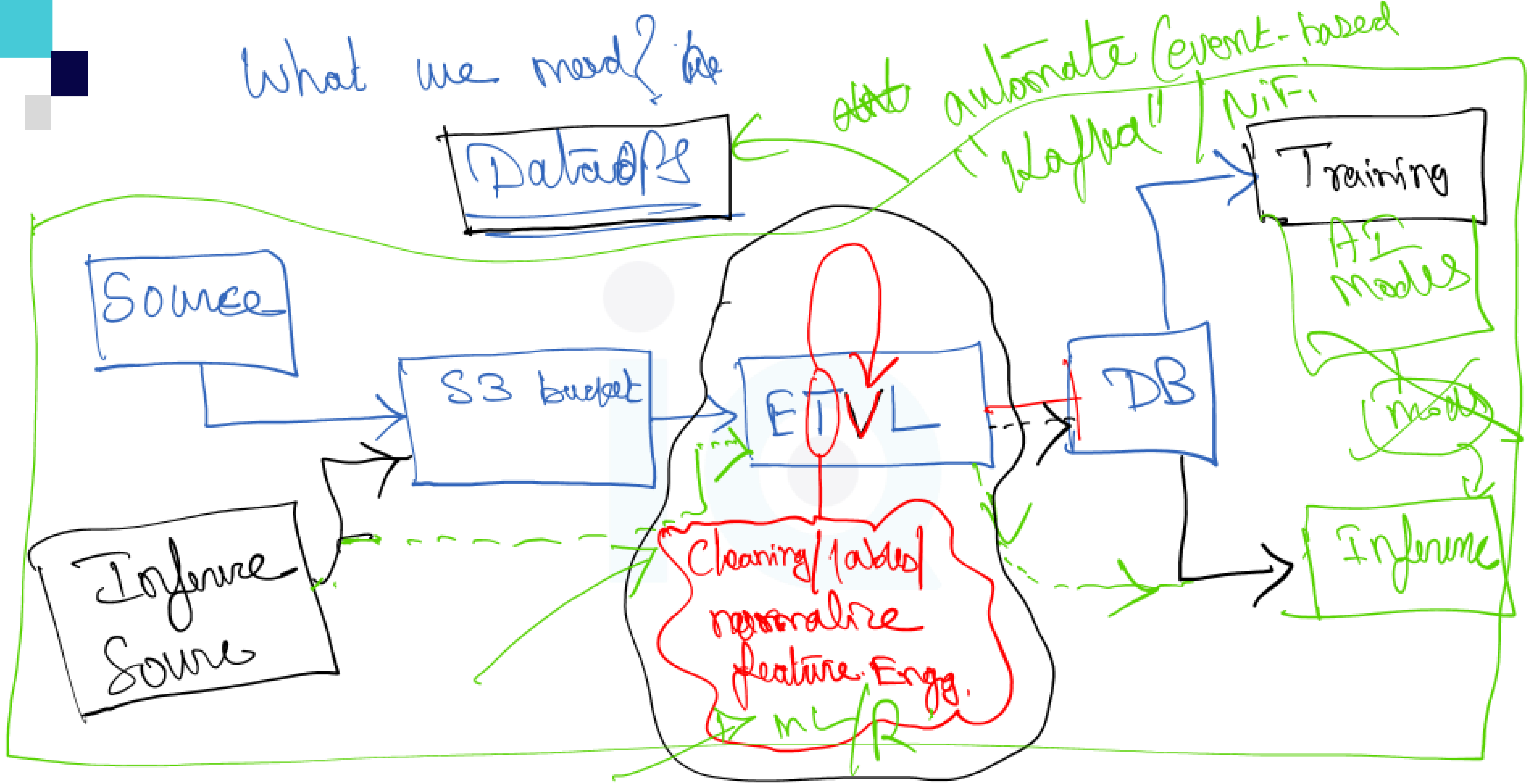
Class Imbalance

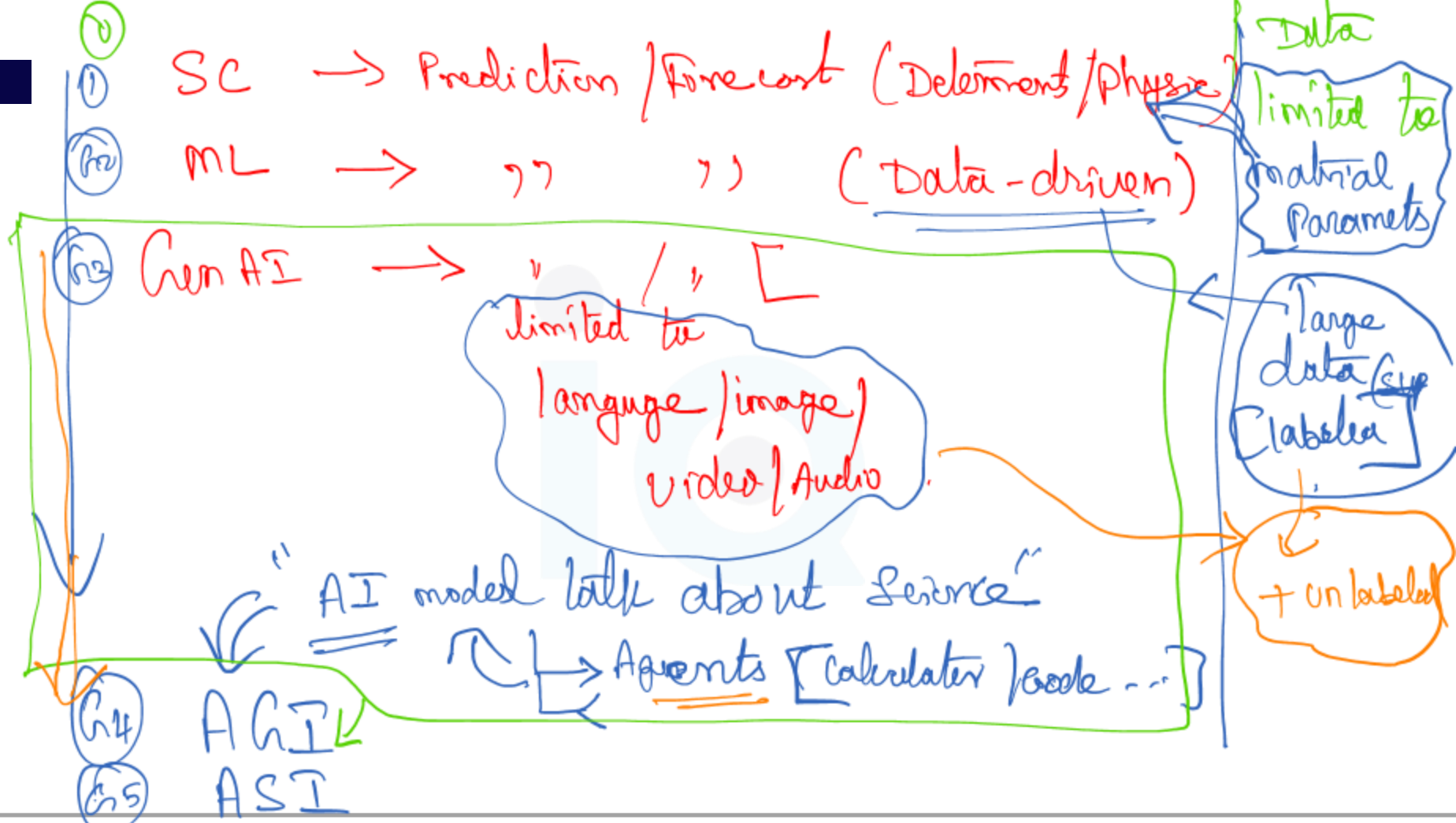
Labeling

Data Augmentation

“Training Data

What we need? be




0
1
2a
2b
SC → Prediction / Forecast (Deterministic / Physics)
ML → " " (Data-driven)
Gen AI → " / " [
limited to
language / image
video / Audio
Data
limited to
material
Parameters
large
data
labeled
+ unlabeled
"AI model talk about Science"
Agents [Calculator / code ...]
AGI
ASI

Training Data

Importance

- ML curriculum is heavily skewed towards modeling
- Data is messy, complex, unpredictable and potentially treacherous. If not handled properly, it can easily sink the entire ML system
- Data scientists and ML engineers should learn how to handle data well
  - How to obtain or create good training data?
  - How to split data?
  - Data is full of potential biases, and ML model can perpetuate it
- Dataset denotes a finite and stationary data, whereas training data need not be finite or stationary!

“Sampling

Training Data

Sampling

- An integral part of AI system
- Can be an experiment design to collect data or to create training data or splits for training, validation and testing
- Unfortunately, often overlooked in typical ML curriculum
- Happens in many stages
  - Sampling from all possible real-world data
  - Sampling from given dataset
  - Sampling from all events/processes that happen within ML system for monitoring
- Much needed for regulatory and certification

Training Data

Sampling

Training data sampling plays a crucial role in the performance and generalization ability of machine learning models.

The quality and representativeness of the training data directly impact the model's ability to learn patterns and make accurate predictions.

“Non-probability sampling

Sampling

Non-probability sampling

- Convenience Sampling
  - Selected based on their availability
  - Quick and easy to implement but may introduce bias since the sample may not be representative of the entire population
- Purposive Sampling
  - Purposive sampling involves selecting individuals who meet specific criteria or possess certain characteristics of interest
  - Researchers intentionally select participants based on their expertise, knowledge, or unique perspectives relevant to the research objective
  - While purposive sampling allows researchers to target specific populations, it may result in a non-representative sample

Sampling

Non-probability sampling

- Snowball Sampling
  - Starts with an initial set (snowball) of participants who meet the desired criteria
  - Future samples are selected based on existing samples
  - Often used when studying hard-to-reach or hidden populations but may introduce bias since participants are reliant on referrals
    - E.g.; Scrap legitimate Twitter accounts without having database. Start with small number of accounts and scrape all the accounts they follow
- Quota Sampling
  - Select sample that reflects specific characteristics or proportions of the population
  - Quotas based on certain criteria (e.g., age, gender, occupation) and then select individuals to meet those quotas
  - Does not involve random selection and may result in a non-representative

Sampling

Non-probability sampling

- Volunteer Sampling
  - Involves individuals self-selecting to participate in a study
  - Participants actively volunteer, often in response to invitations or advertisements
  - While this method is convenient, it can introduce bias as participants may have different characteristics or motivations compared to the general population
- Judgment Sampling
  - Sampling relies on the expert's judgment or expertise to select participants who are deemed representative or relevant to the research question
  - Subjective and can be prone to expert bias

Sampling

Non-probability sampling

- Challenges
  - Samples selected by non-probabilistic criteria are not representative of the real-world data, and might riddled with selection biases
  - Not recommended to use this family of sampling to train ML models. Unfortunately, it may not be the case, and still driven by convenience
    - Language models are often trained with data – Wikipedia, Common crawl, Reddit, etc.
    - Movie review/sentiment analysis are often done with natural sampling (ratings), IMDB reviews and Amazon reviews. This dataset need not be representative enough, only volunteers are available
    - Data for training self-driving cars not representative of every region.

Sampling

Intuitive sampling

- Simple Random Sampling
  - Each member of a population has an equal and independent chance of being selected
  - Easy to implement but rare category might not appear in the sample
- Stratified Sampling
  - Divide the population into distinct subgroups or strata based on certain characteristics or attributes
  - Each subgroup or category within the population is adequately represented in the sample
  - Each stratum should have a large enough sample size to provide reliable estimates within that subgroup
  - Random selection within each stratum is necessary
  - Multilabel samples are challenging

Sampling

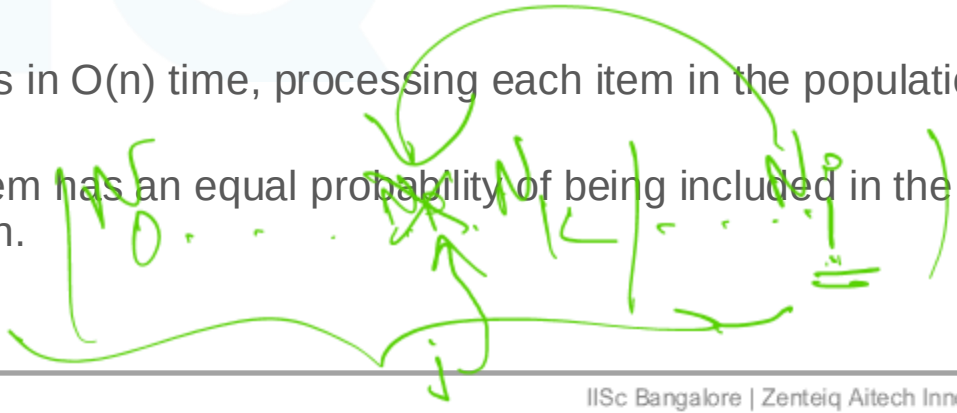
Intuitive sampling

- Weighted Sampling
  - Each sample is given a weight, which determines the probability of being selected
  - Particularly useful when working with imbalanced datasets
  - Weights should align with the business objective, expert knowledge, or the underlying characteristics of the dataset
- Reservoir sampling
  - Random sampling technique used to select a representative sample of a fixed size from a stream or large dataset with an unknown “N” or potentially infinite length.
  - Ensures that each item in the stream or dataset has an equal probability of being selected.

Sampling

Intuitive sampling

- Reservoir sampling
  - **Initialize the Reservoir:** Fill the reservoir with the first “k” items from the stream.
  - **Process the Stream:** For each subsequent item “i” (where i > k) in the stream:
    - Generate a random number j between 0 and i
    - If(j<k), then replace the item at index j in the reservoir with the current item.
- Key properties
  - Constant space complexity: Uses only O(k) space, regardless of the population size n.
  - Linear time complexity: Runs in O(n) time, processing each item in the population once.
  - Unbiased sampling: Each item has an equal probability of being included in the sample, even if n is unknown.



Sampling

Intuitive sampling

- Importance Sampling
  - Used in statistics and Monte Carlo simulations to estimate properties of a target distribution P(x) by sampling from a different, more easily sampled distribution Q(x)
  - Allows for efficient estimation in cases where direct sampling from the target distribution is challenging or impractical
  - The choice of the importance distribution Q(x) is critical for effective importance sampling
  - Sample x from Q(x) and weigh this sample by P(x)/Q(x)
    - Q(x) can be any distribution as long as Q(x)>0 non-zero P(x)

“Labeling

Labeling

Importance

- Labelling plays a crucial role in the development of ML systems
  - Most ML models in production today are supervised
- The process of labelling involves assigning meaningful and accurate annotations to data, which serves as the foundation for training machine learning models
- Proper labelling enables the models to learn patterns and make predictions or classifications based on the labelled data
- Inadequate or erroneous labelling can significantly impact the performance and reliability of machine learning systems
- We will explore the various aspects of labelling in ML, including the process, types of labels, techniques, challenges, and the importance of maintaining quality and fairness in labelling.

Labeling

Hand Labeling

- Involves several steps to ensure the availability of annotated data for model training
  - **Data Collection**: Collect relevant data that represents the ML problem.
    - This can involve scraping websites, accessing databases, or acquiring data through other means
  - **Annotation**: Once the data is collected, it needs to be annotated with labels.
    - Human annotators review the data and assign appropriate labels based on the desired outcome.
    - Annotation can be done manually or with the help of automated tools
  - **Iterative Feedback**: Iterative feedback loops are established between annotators and data scientists to address ambiguities and improve label quality
  - **Validation and Quality Control**: A validation set is often used to assess the quality of the labels.
    - Inter-annotator agreement (IAA) measures and other quality control techniques are employed to ensure consistency and accuracy in the labelling process
  - **Challenges:** Delay, data privacy, addition of new labels to the labeled data, etc.

Labeling

Label Multiplicity

- An important concept, particularly for tasks involving structured data
- Number of labels associated with a single instance or row in a table
- In some cases, a single row in a table may have multiple labels, indicating multiple attributes or categories assigned to that instance
  - Can vary depending on the specific problem and the nature of the data being labeled
  - Handling label multiplicity requires careful consideration in the labelling process to ensure accurate representation and appropriate training of machine learning models
  - Techniques such as multi-label classification and hierarchical labelling are commonly used to address table multiplicity
  - Sometimes due to multiple annotators or data lineage (different source) sample gets multiclass wrongly

Labeling

Natural Labels

- We might be lucky to work on tasks with natural ground truth labels
  - Model prediction can immediately be validated
    - OTT recommendation
    - Google Map
- Even if the natural label is not inherently available, it might be possible to design a system to get labels through feedback.
  - Allow user to submit a correct translation and use it to retrain a model
    - Caution: reliability
- Not necessarily best but easier and cheaper

Handling Lack of Labels

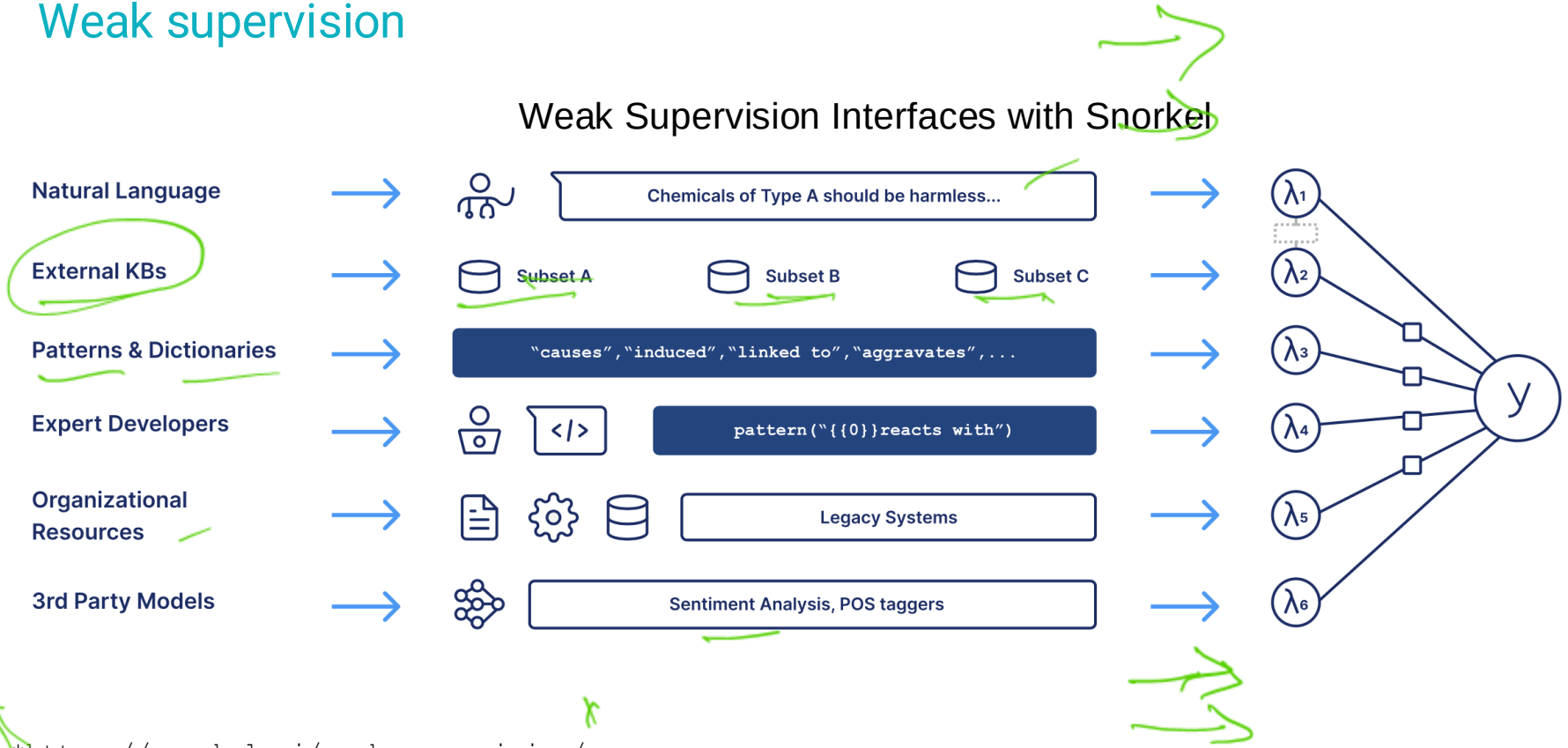
Handling Lack of Labels

Weak supervision

- An approach to ML in which high-level and often noisier sources of supervision are used to create much larger training sets much more quickly than could otherwise be produced by manual supervision
- A technique used in ML systems when obtaining fully labeled training data is challenging or expensive
- Instead of relying solely on carefully annotated data, heuristics, rules, or other forms of noisy labels are used for training
- Leverages the availability of large amounts of weakly labeled or partially labeled data
- Allows for faster data labeling and can overcome the limitations of manual annotation

Handling Lack of Labels

Weak supervision



*https://snorkel.ai/weak-supervision/

Handling Lack of Labels

Weak supervision

- Various techniques can be applied to handle weak supervision, including:
  - Snorkel*: A framework for creating weakly labeled training data using label models and data programming
  - Co-training: Training models on different views of the same data to enhance learning
  - Self-training: Using the model's own predictions as pseudo-labels to further train the model iteratively
- Weak supervision is a valuable tool in scenarios where fully labeled data is scarce or expensive, but careful consideration is required to ensure the quality and reliability of the training process

Handling Lack of Labels

Semi-Supervised Labelling

- Combines both labeled and unlabeled data for training machine learning models
- A small set of labeled data is supplemented with a larger set of unlabeled data
- The labeled data provides explicit supervision, while the unlabeled data contributes to the learning process by capturing underlying patterns and improving model performance
- Useful when acquiring labeled data is time-consuming or expensive, but unlabeled data is readily available

Handling Lack of Labels

Semi-Supervised Labelling

- Commonly used techniques include:
  - **Self-training:** Initially, the model is trained on the labeled data. Then, the model predicts labels for the unlabeled data and treats these predictions as additional labeled examples for training
  - **Co-training:** Two or more models are trained independently on different subsets of features or views of the data. Each model then provides labels for the unlabeled data, which are used to improve the training of the other models
  - **Cluster-based methods**: Unlabeled data is clustered, and representative points from each cluster are labeled. These labels are then used to train the model
- Semi-supervised labelling can significantly improve model performance by leveraging the abundance of unlabeled data while minimizing the reliance on expensive labeled data

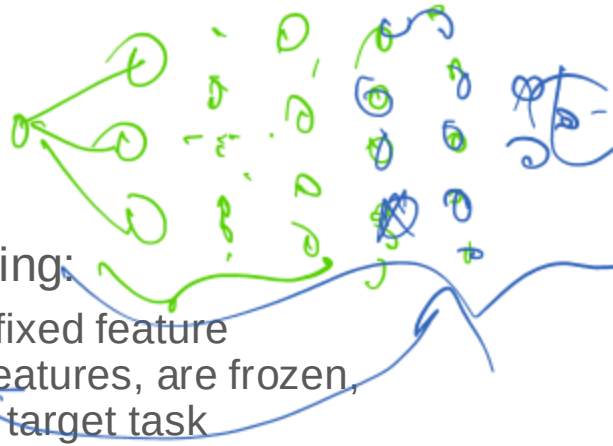
Handling Lack of Labels

Transfer Learning

- A powerful technique in machine learning that allows the reuse of knowledge gained from one task to improve performance on another related task
- Instead of starting the learning process from scratch, transfer learning leverages pre-existing knowledge and models trained on a different but related task or domain
- Especially useful when the labeled data for the target task is limited or when training a model from scratch is computationally expensive
- Popular pre-trained models for transfer learning include ImageNet models (e.g., VGG, ResNet) for image-related tasks and BERT, GPT, and RoBERTa for natural language processing tasks.
    - E.g. https://github.com/google-research/bert
    - https://devopedia.org/bert-language-model
    - https://huggingface.co/docs/transformers/model_doc/roberta

Handling Lack of Labels

Transfer Learning



- There are two common approaches to transfer learning:
  - **Feature Extraction**: Pre-trained models are used as fixed feature extractors. The earlier layers, which capture generic features, are frozen, and the later layers are replaced and retrained for the target task
  - **Fine-tuning**: Pre-trained models are used as initial weights, and the entire model is further trained on the target task. The earlier layers may be frozen, but the later layers are fine-tuned to adapt to the new task
- By leveraging transfer learning, models can benefit from the knowledge and representations learned from vast amounts of labeled data in related tasks or domains, resulting in improved performance and faster convergence on the target task.

Handling Lack of Labels

Active Learning

- A strategy that allows machine learning models to actively query human annotators (only when needed) for additional labels on uncertain or informative instances
- Unlike traditional supervised learning, where all labels are pre-defined and provided upfront, active learning iteratively selects the most valuable instances for labelling
- The goal of active learning is to reduce the annotation effort and improve model performance by focusing on the most informative data points

Handling Lack of Labels

Active Learning

- Typically involves the following steps
  - **Initial Model Training**: Train a model on a small labeled dataset
  - **Uncertainty Sampling**: Apply the trained model to unlabeled instances and select the most uncertain or ambiguous examples for annotation
  - **Annotator Labeling**: Request human annotators (SMEs) to label the selected instances
  - **Model Update**: Incorporate the newly labeled instances into the training set and retrain the model
  - **Repeat**: Iterate the process by selecting the next set of uncertain instances for annotation

Handling Lack of Labels

Active Learning

- Common uncertainty sampling methods used in active learning include
  - **Least Confidence**: Select instances where the model's predicted probability for the most probable class is the lowest
  - **Margin Sampling**: Select instances where the difference between the top two predicted probabilities is the smallest
  - **Entropy Sampling**: Select instances with the highest entropy, representing the highest level of uncertainty.
- Enables efficient utilization of annotation resources by focusing on challenging instances, resulting in improved model performance with fewer labeled examples.

“Class Imbalance in ML Systems

Class Imbalance

Introduction

- Refers to a situation where the distribution of classes in the training data is significantly skewed
- Can pose challenges and impact model performance in machine learning systems
- Understanding class imbalance is crucial for developing effective and fair machine learning models

Class Imbalance

Introduction

- Occurs when the distribution of classes in the training data is heavily skewed, with one class being significantly more prevalent than others
- The majority class refers to the class with a larger number of instances, while the minority class has fewer instances
  - E.g.: Cancer deduction dataset
- Class imbalance can lead to biased model predictions and difficulties in accurately classifying the minority class
- Visual representation of class imbalance can help illustrate the imbalance in the dataset

Class Imbalance

Handling class imbalance

- Resampling methods, such as oversampling the minority class or undersampling the majority class, can help balance the class distribution in the training data
- The Synthetic Minority Over-sampling Technique (SMOTE) generates synthetic samples of the minority class to increase its representation and address class imbalance
- Cost-sensitive learning assigns different misclassification costs to different classes, considering the imbalance and adjusting the model's behavior accordingly
- Ensemble methods, such as combining multiple models trained on balanced subsets of the data, can improve the prediction of the minority class and overall model performance
- Employing these techniques can help alleviate the impact of class imbalance and improve the fairness and accuracy of machine learning

Class Imbalance

Handling class imbalance

- Use right evaluation metric
  - **Precision, recall, and F1-score**
    - Metrics that consider true positives, false positives, and false negatives, providing a more accurate evaluation of performance.
    - Precision = TP/(TP+FP)
    - Recall = TP/(TP+FN)
    - F1 = 2 * Precision * Recall/(Precision + Recall)
  - **Receiver Operating Characteristic (ROC) curve**
    - Evaluating the trade-off between true positive rate and false positive rate
  - **Area Under the Curve (AUC)**
    - Assessing the overall performance of the model.

Data Augmentation

Data Augmentation

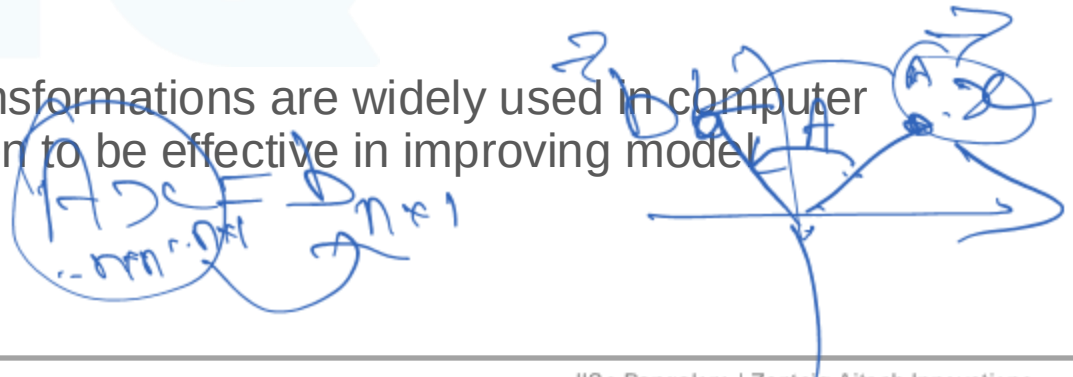
Introduction

- A technique used to increase the size and diversity of a training dataset by applying various transformations to the existing data
- Plays a crucial role in improving model generalization, mitigating overfitting, and enhancing the robustness of machine learning models

Data Augmentation

Simple Label-Preserving Transformations

- Involve applying basic modifications to the input data while preserving the original labels
- Common transformations in CV include rotation, scaling, flipping, and translation of images or adjusting brightness, contrast, or saturation
- Effectively increase the training data's variability without altering the ground truth labels, providing additional samples to improve model performance
- Simple label-preserving transformations are widely used in computer vision tasks and have proven to be effective in improving model generalization



Data Augmentation

Perturbation-Based Methods

- Introduce controlled perturbations to the training data, simulating real-world variations or uncertainties
- Perturbations can include random noise, occlusion, deformations, or alterations to specific features in the data
- By exposing the model to diverse variations of the input, perturbation-based methods enhance the model's ability to handle noisy or altered inputs, improving its robustness
- These methods are particularly useful in applications such as image classification, object detection, and speech recognition

Data Augmentation

Data Synthesis

- Involves generating synthetic data to supplement the training dataset, either by generating entirely new samples or by interpolating existing samples
- Techniques such as generative adversarial networks (GANs) and variational autoencoders (VAEs) are commonly used for data synthesis
- Data synthesis is beneficial when the available training data is limited or lacks diversity, enabling the model to learn from a broader range of examples
    - E.g: Self-driving cars, healthcare, etc
- It can also be used to balance class distributions in imbalanced datasets or generate samples for rare or specific classes
    - Audio Data Augmentation, Text Data Augmentation, Image Augmentation

Training Data

Summary

- Training data forms the foundation of Modern ML models
- No matter how good the ML model might be, if the dataset is bad, ML model won't be able to perform well
- Apply different sampling approaches
- Use appropriate technique to generate right label, if needed
- Handle class imbalance, if any, appropriately
- Augment training data, if needed

Training Data

Summary

Training Data

Sampling

Labeling

Handling Lack of Labels

Class Imbalance

Data Augmentation