



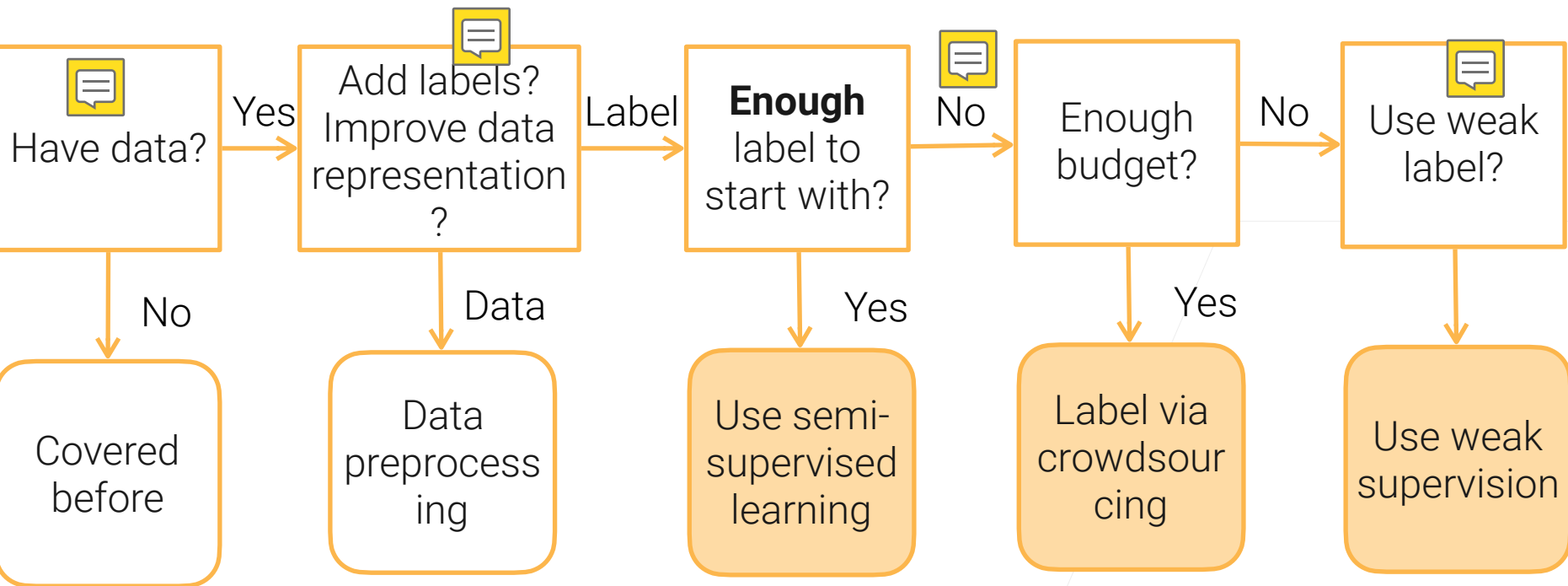
CS 329P : Practical Machine Learning (2021 Fall)

1.4 Data Labeling

Qingqing Huang, Mu Li, Alex Smola


<https://c.d2l.ai/stanford-cs329p>

Flow Chart for Data Labelling



Semi-Supervised Learning (SSL)

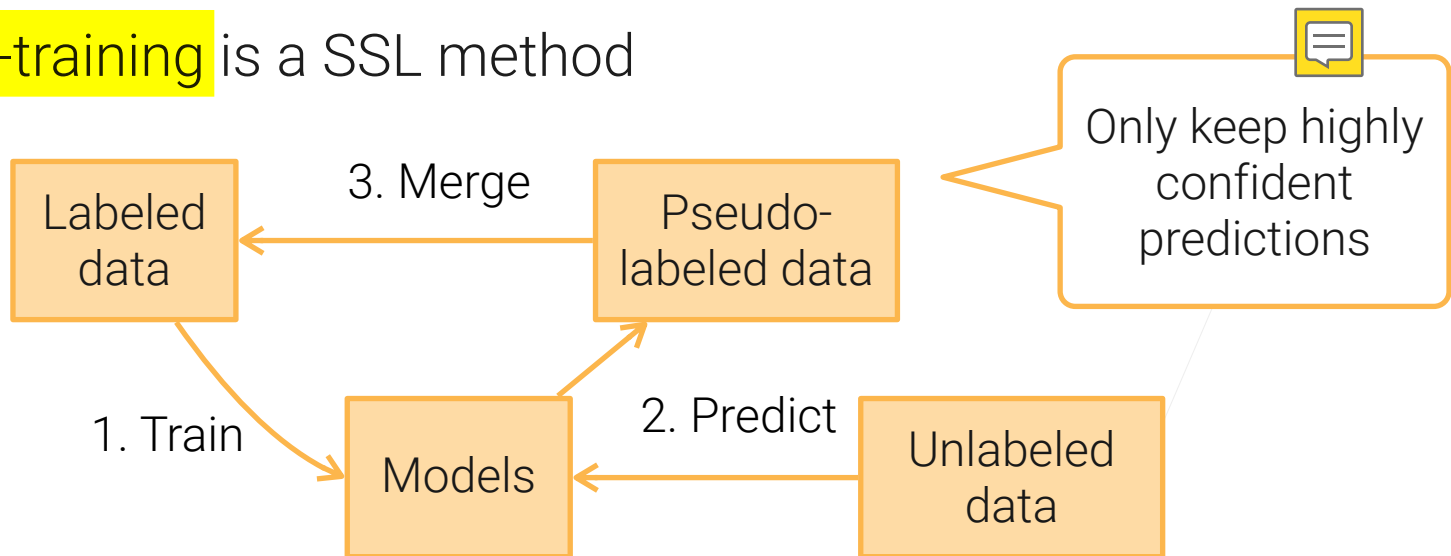



- Focus on the scenario where there is a small amount of labeled data, along with large amount of unlabeled data 
- Make assumptions on data distribution to use unlabeled data
 - **Continuity assumption:** examples with similar features are more likely to have the same label
 - **Cluster assumption:** data have inherent cluster structure, examples in the same cluster tend to have the same label
 - **Manifold assumption:** data lie on a manifold of much lower dimension than the input space

Self-training



- Self-training is a SSL method



- We can use expensive models 
- Deep neural networks, model ensemble/bagging

Label through Crowdsourcing



- ImageNet labeled millions of images through Amazon Mechanical Turk. It took several years and millions dollars to build
- According to Amazon SageMaker Ground Truth, the estimated price of using Amazon Mechanical Turk:



| | |
|---------------------------|-------------------|
| Image/text classification | \$0.012 per label |
| Bounding box | \$0.024 per box |
| Semantic segmentation | \$0.84 per image |

Challenges



- Simplify user interaction: design easy tasks, clear instructions and simple to use interface
 - Needs to find qualified workers for complex jobs (e.g. label medical images)
- Cost: reduce #tasks X #time per task sent to labelers
- Quality control: label qualities generated by different labelers vary



User interaction



- Example of user instruction and labeling task (MIT Place365)

Start **Is this a cliff scene?**

Definition: a high, steep or overhanging face of rock.

Task

For each of the **810** images, answer yes or no to the above question. Only answer **Yes** to **real photos**. Always answer **No** to **cartoon, drawing, CG rendering**, or real photos with a **large text overlay** on the photo. Here are some examples:

No Single Object No Text Overlay No Drawing No Screenshot No Graphics No Bad Photo

Drawing or Painting

Cartoon or Graphics Rendering

Not Only Logo No Magazine/Newspaper

No No Yes Yes

Yes Yes Yes Yes Yes Yes Yes Yes

Yes Yes Yes Yes Yes Yes Yes Yes

b)

Instruction **Is this a cliff scene?** **Submit (790 images left)**

Definition: a high, steep or overhanging face of rock.



Yes

No

No

Reduce #tasks: Active Learning

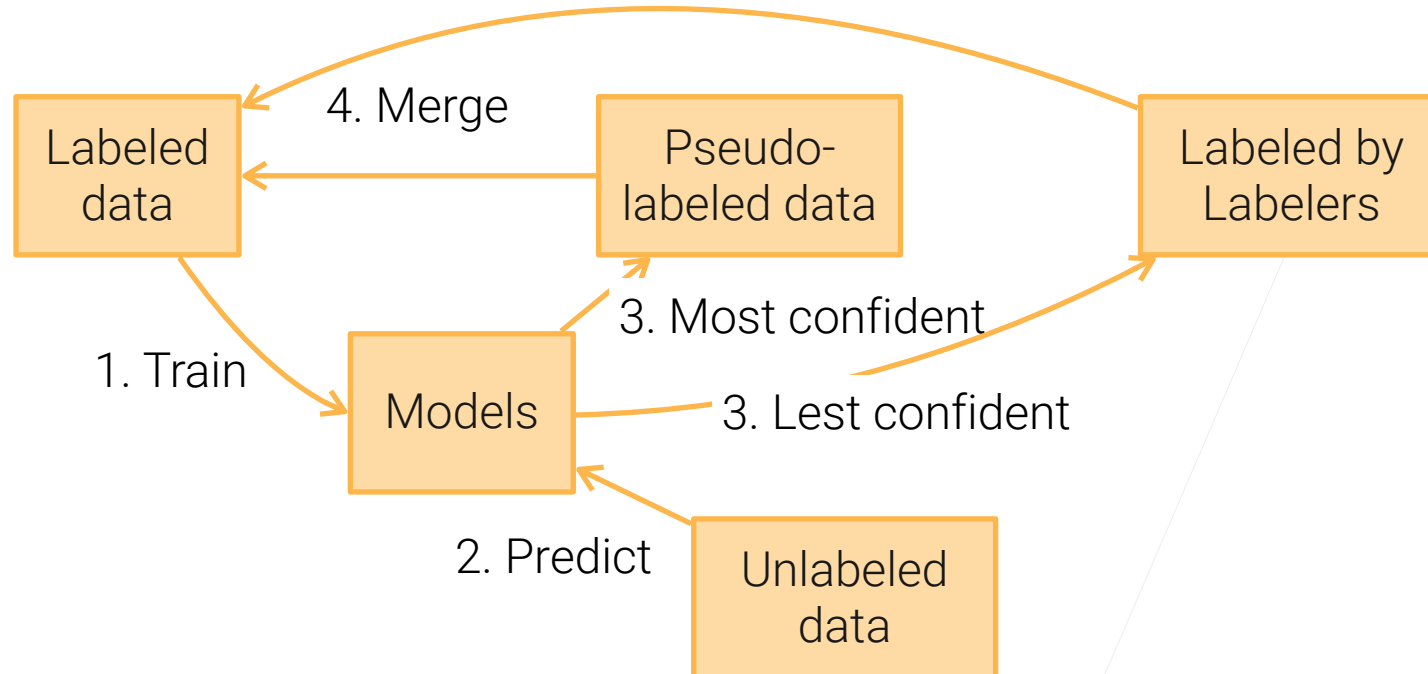


- Focus on same scenario as SSL but with human in the loop
 - Self training: Model helps propagate labels to unlabeled data
 - Active learning: Model select the most “interesting” data for labelers 
- **Uncertainty sampling** 
 - Select examples whose predictions are most uncertain
 - The highest class prediction score is close to random ($1/n$)
- **Query-by-committee**
 - Trains multiple models and select samples that models disagree with

Active Learning + Self-training



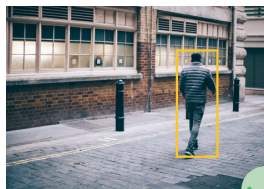
- These two methods are often used together



Quality Control



- Labelers make mistakes (honest or not) and may fail to understand the instructions



Too big



Too small



Wrong object



- Simplest but most expensive: sending the same task to multiple labelers, then determine the label by majority voting
- Improve: repeat more for controversial examples, prune low-quality labelers



Weak Supervision



- Semi-automatically generate labels
 - Less accurate than manual ones, but good enough for training
- Data programming:
 - Domain specific heuristics to assign labels
 - Keyword search, pattern matching, third-party models
 - E.g. rules to check if YouTube comments are spam or ham



```
1 def check_out(x):  
2     return SPAM if "check out" in x.lower() else ABSTAIN  
3 def sentiment(x):  
4     return HAM if sentiment_polarity(x) > 0.9 else ABSTAIN  
5 def short_comment(x):  
6     return HAM if len(x.split()) < 5 else ABSTAIN
```

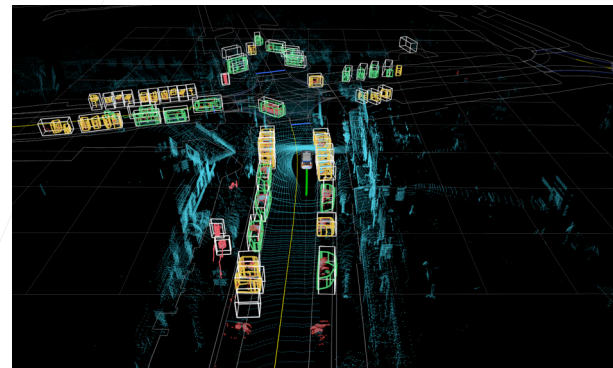


Adapted from snorkel.org

Data Labeling for Self-driving Car




- Tesla and Waymo both have large in-house data labeling teams
- Labels needed: 2D/3D bounding box, image semantic segmentation, 3D laser point cloud annotation, video annotation,...
- Use active learning to identify scenarios which need more data / label
- Use ML algorithms for automatic labeling
- Use simulation to generate perfectly labeled, unlimited data for rare situations



Summary



- Ways to get labels 
 - **Self-training**: iteratively train models to label unlabeled data
 - **Crowdsourcing**: leverage global labelers to manually label data
 - **Data programming**: heuristic programs to assign noisy labels
- Alternatively, You could also consider **unsupervised/self-supervised learnings**

Flow chart for data preprocessing

