Handling Data Sources for Effective Machine Learning

Machine learning models are highly reliant on training data. In this talk, I would like to share my personal experience working with various data sources and the lessons I've learned along the way. There are two crucial aspects of data: quality and quantity. When it comes to data obtained through human raters, there is an additional practical factor to consider: economic feasibility. In the talk, I will discuss how to strike a balance among these three factors

Types of Data Sources

There are two major types of data depending on their source that require different processing approaches before they can be used for machine learning. First, there is user-generated data. This can be obtained by carefully logging user activity and behaviour. For instance, in online advertising, user actions like clicks, conversions can be collected, logged and finalised as labelled data.
Second, there is data where no labelling originally exists, however there is still a need to train an ML model. One way to tackle this issue is employing human annotators to label data manually. For example MNIST dataset, one of the earliest computer vision datasets compiled and labelled by humans.

Another strategy of handling non-labelled data is useful when there are no resources for staffing enough annotators to label the training data. In such a case, we can opt for the proxy-label method. For example, users can report content that they find inappropriate and then it can be labelled as such, although sometimes such data can be noisy and might require some attention of the annotators' team.

Let us focus on the ways to handle the most challenging type, the non-labelled data.

Gathering Datasets with Human Annotators

Building datasets employing human annotators is a commonly used technique in machine learning. Its two main limitations are costs and time consumption.

Costs can turn into a major challenge if a dataset is large and processing it means hiring many people. Also, the nature of data may require a high level of expertise and, consequently, employing expensive specialists. For instance, annotating medical images or legal documents can only be done by highly trained staff.

Time can be an issue for several reasons. First, it is obvious that large datasets take a long while (or an impractical number of people) to process. Sometimes it can also be difficult to recruit annotators willing to engage in long-time projects. Second, annotators should be adequately trained to deliver the desired level of quality. Such training can be time consuming, especially when high expertise levels are required.

Here are the lessons I have learned while working on various ML projects throughout my career. Often, the first step in building an ML solution is to gather a training sample. Here are the key factors that made the process more effective

- **Annotate data in batches**
  Often, ML teams have a limited budget for human raters, so splitting the annotation process into batches can be extremely useful. Sometimes, initially, human raters may not produce high-quality results, and it is important to identify this early on to save the majority of the budget. To address this issue, I have found it extremely important to collaborate closely with the team of annotators to refine the guidelines and process, aiming to minimise errors (more on this in the next section)..

  Another advantage of sending data in batches is that it allows to implement active learning. If the entire dataset were sent to the annotators all at once, we would not have been able to utilise this strategy.

- **Sample batches with active learning**
  Random Sampling is generally a good initial approach, but it has two caveats
  - The complete dataset can be heavily biassed. For example, imagine we would like to train an image classifier to detect unprofessional images in an online shop. If large shops are onboarded on the platform, they will likely dominate the complete dataset with high-quality images.
  - Random sampling does not easily allow the addition of "difficult" samples to the training data. In the case of an image classifier, such difficult samples would include images with predictions that are borderline, around 0.5.

  One of the solutions to this problem could be to sample with Active Learning. To implement it, one can follow three steps:
  - Randomly sample some data
  - Get predictions for every sample using the current model
  - Send borderline samples to human annotators. Alternatively, we can send samples where false positives or false negatives are more likely to be added to the final dataset by sorting the samples based on their predictions and sending either the top N first samples or the bottom N last samples

  Switching from random sampling to an active learning strategy often allows for an increase in the percentage of the underrepresented class in the final labelled dataset

- **Track annotators' quality**
  It's quite common when working with human annotators to collect several responses per item and assign the final label using the supermajority rule.
  For example, continuing the image classifier use case. Let's assume we collect three responses per image and if two answers indicate that the image is "unprofessional" we assign it as the final label. Intuitively it seems like this procedure should significantly increase accuracy of the data.
  However, if we make simple calculations we will see that the increase in probability of the final label to be correct is small and it's more important to work on the accuracy of the individuals.
  Let's assume each annotator has a minimum probability to give a correct answer for a given question; let us put this probability as *p*. Now let us calculate what is the minimum probability for at least two raters to give a correct answer; we shall put this probability as *q*. Then, *q* is a sum of probabilities of two events: all three raters giving a correct answer (p^3) and any two raters giving a correct answer 3*(1-p)*p^2. In the table below we can see how *q* changes depending on *p*.

Annotator's minimum probability of providing a correct answer (*p*)	Final label's minimum probability of being correct (*q*)
0.6	0.648
0.7	0.784
0.8	0.896

  As we can see, the final probability (*q*) does not change much if we apply the supermajority rule with a minimum of three responses. That is why it is very important to track the quality of each annotator.
  I found the following process to be very effective in improving the quality of human raters
  - Introducing a "golden set" created by well-trained annotators. This set is then used to calculate the accuracy for each annotator assigned to the project
  - Having a biweekly AMA where annotators could ask the questions about controversial cases
  - Introducing final exam and minimum performance threshold. An annotator can only start rating if they pass the final exam and get the score higher than threshold

Reducing human involvement

Sometimes, a human-based approach may fail or become too impractical to employ. In many cases, there might be an elegant solution to bail you out. Let us have a look at some of them:

- **Proxy Values**

Sometimes we could be creative and use a proxy value for a label instead of building a dataset employing human annotators. For instance, we could use images of the product of the well known brands as a source of high quality images

- **Data augmentation**
  In this case, data from an already processed dataset is reproduced in an altered form and then fed back into the model. In the example of image classifier we can take the images rated as “professional” and use basic graphic tools to make the high quality images unprofessional. In particular, we could blur the good images and add them as "bad" images to the training set.

As you see, human input in machine learning is essential. However managing it may be a matter of survival for many tech projects. The rule of thumb is opting for less but more qualified staff, setting reference datasets and extracting as much as possible from actions of users