Custom Vision

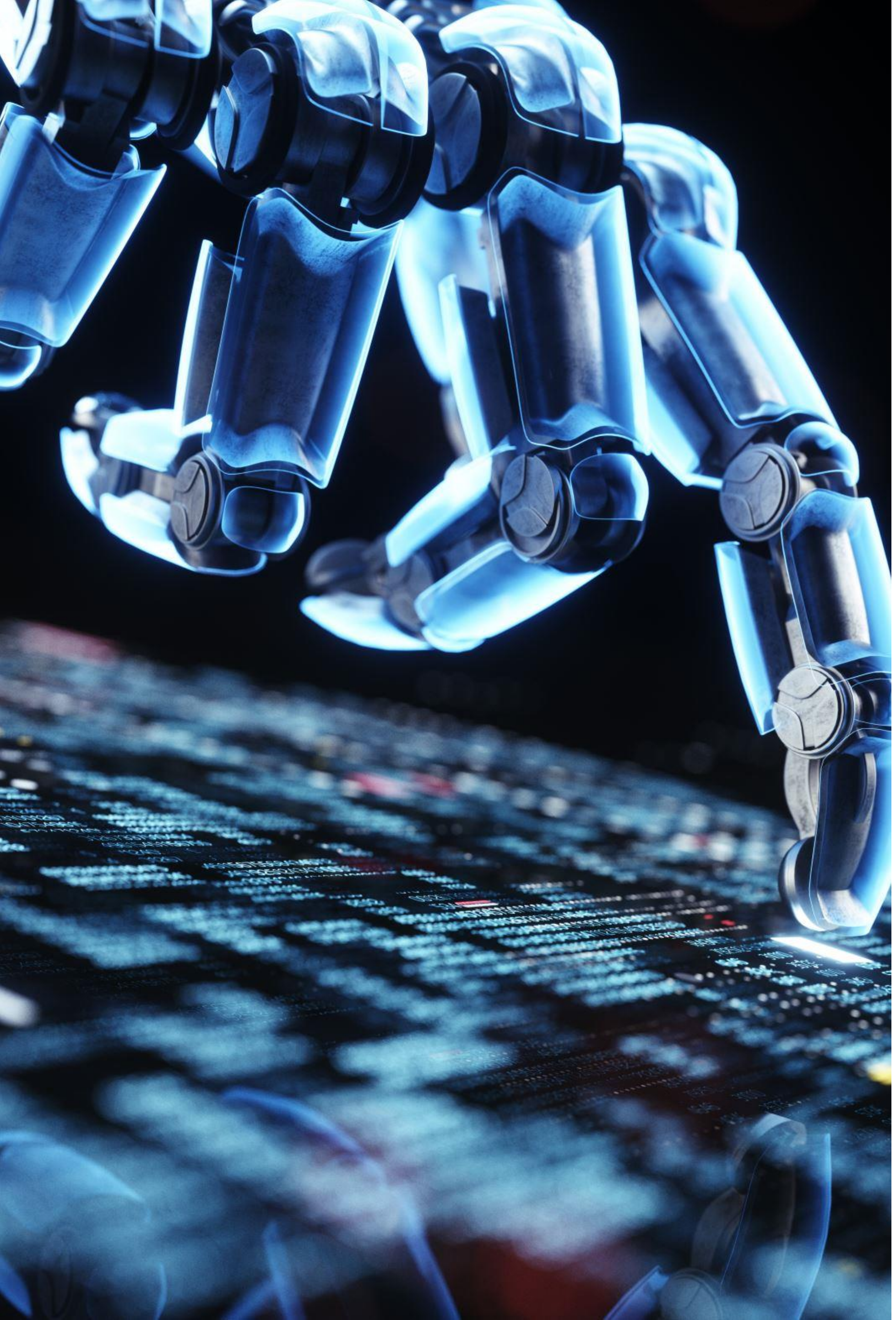
Conf.Dr. Cristian KEVORCHIAN
Mathematics and Computer Science Faculty

Computer Vision

- **Image classification** - This is where the API will give you a number of tags that classify the image. It should also give you a confidence score of how strongly the model predicts the image to be of that tag.
- **Content Moderation** - The API can give you a ”is Adult” and ”is Racy” flags to determine if the image meets those criteria. A confidence score is provided.
- **OCR** - The API can extract the text within the images and will provide you as a processable entity. This API can also work with handwritten text instead of just text on signs.
- **Facial Recognition** - This API will recognize the faces of celebrities or other well-known people within images.
- **Landmark Recognition** - This will recognize landmarks within images.

Custom Vision

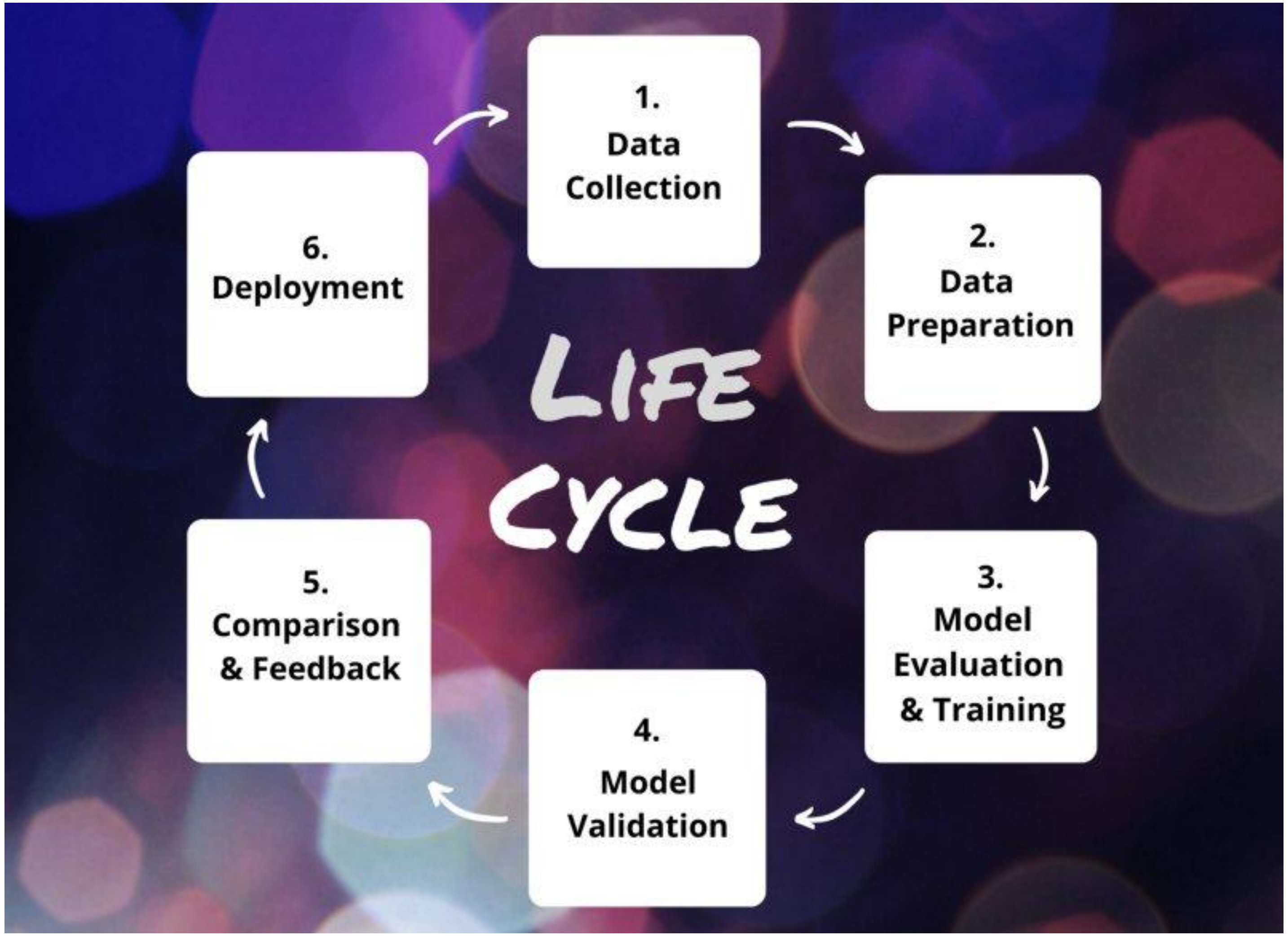
- The Custom Vision service is different in that it allows users to train a model with own data(photos) using a Microsoft prebuilt model. For starters, this can just classify images and detect objects. The object detection section will not only tell you what tag an image is, but will also show you where it is in the image. This aspect of the service is currently in beta, but I've had decent results with it thus far.
- Custom Vision service allow users to upload own images. For image classification, this means that users can upload own images and, for each image, associate it one or multiple tags. When user run an image through the model it will return the tag or tags it thinks it is along with the tag's confidence score. For object detection, you do the same process, but you pick in the images where the object is you want to detect and give that a tag.
- Each time the user will upload and tag new images and the model needs to be trained. From there you can evaluate how well is the model runs, give it test images, or even use the REST URLs or SDKs to interact with it.



Computer Vision
vs.
Custom Vision

The Custom Vision service is a little different in that it allows you to train a model of your own photos using a Microsoft prebuilt model. To begin, this can just classify images and detect objects. The object detection section will not only show what tag is associated with the image is, but will also show you where it is in the image. This aspect of the service is currently in beta, but I've had decent results with it thus far.

Computer Vision Life Cycle Project



Gold Rule for ML Projects : ”Start Small, Fail Fast

In terms of workload and result quality, machine learning projects are always full of risk.

”The start small, fail fast” strategy lead us to minimize risk and investment for our customers. This means we design a feature-complete system with the least amount of labor necessary in order to gain quick feedback on how effectively the model and accessible data perform together.

In the iterations (one iteration means a complete life cycle run), we refine the data and model to achieve the desired output quality.

Data Collection

Machine learning models should solve a given problem on the basis of data. Therefore everything starts with collecting enough samples with proper metadata.

Quality, quantity, and the balance of the data are the decisive points in data collection. The more data we have and the better the quality and balancing is, the better the model will learn and predict accurately.

The quality of the samples is important because wrong or misleading samples or metadata (called noisy data) will confuse the model and dramatically lower the prediction quality. We can improve the quality with data cleaning (phase 2 of the life cycle).

Having balanced data means to have roughly the same amount of training data for each class. Unbalanced training data can lead to biased models as classes are not represented equally.

In computer vision projects we often face a lack of training data (correctly labeled images). To increase quantity and improve balance of the data we might be able to use data synthesis to create training data programmatically ourselves. This process can be very complex and there are various methods to do this.

Another common method to create more data is called data augmentation. We create additional data by modifying existing samples, e.g. through random cropping, adding noise, changing colors or brightness.

Data Preparation

- If there are enough collected data, then we need to create a procedure to load the model with this data.
- It needs to clean the data by identifying noise, false or misleading data and correct or removing it from the training set. Additionally, we preprocess the data to normalize it. In our cases, this mostly means scaling or cropping images, converting them into a relevant format, and creating a folder structure we can use for training.
- Collecting, cleaning, and preprocessing data are the most time-consuming challenges. It is not unusual to spend a major part of the project time on these tasks(about 60% of the total time of the project).

Preparation-Technical Aspects

- get lots of pictures anyway
- see that you have different images
- you need pictures with different perspectives(Washington monument) and the surroundings
- it should also be possible to identify different weather conditions
- the quality of the images should perhaps also vary
- you can get a lot out of it with a little experimentation
- As a base setup, we need at least two tags (a tag denotes a category of objects and is often called a label).
  Yes – Monument is present
  No – Monument is not present

Model Evaluation and Training

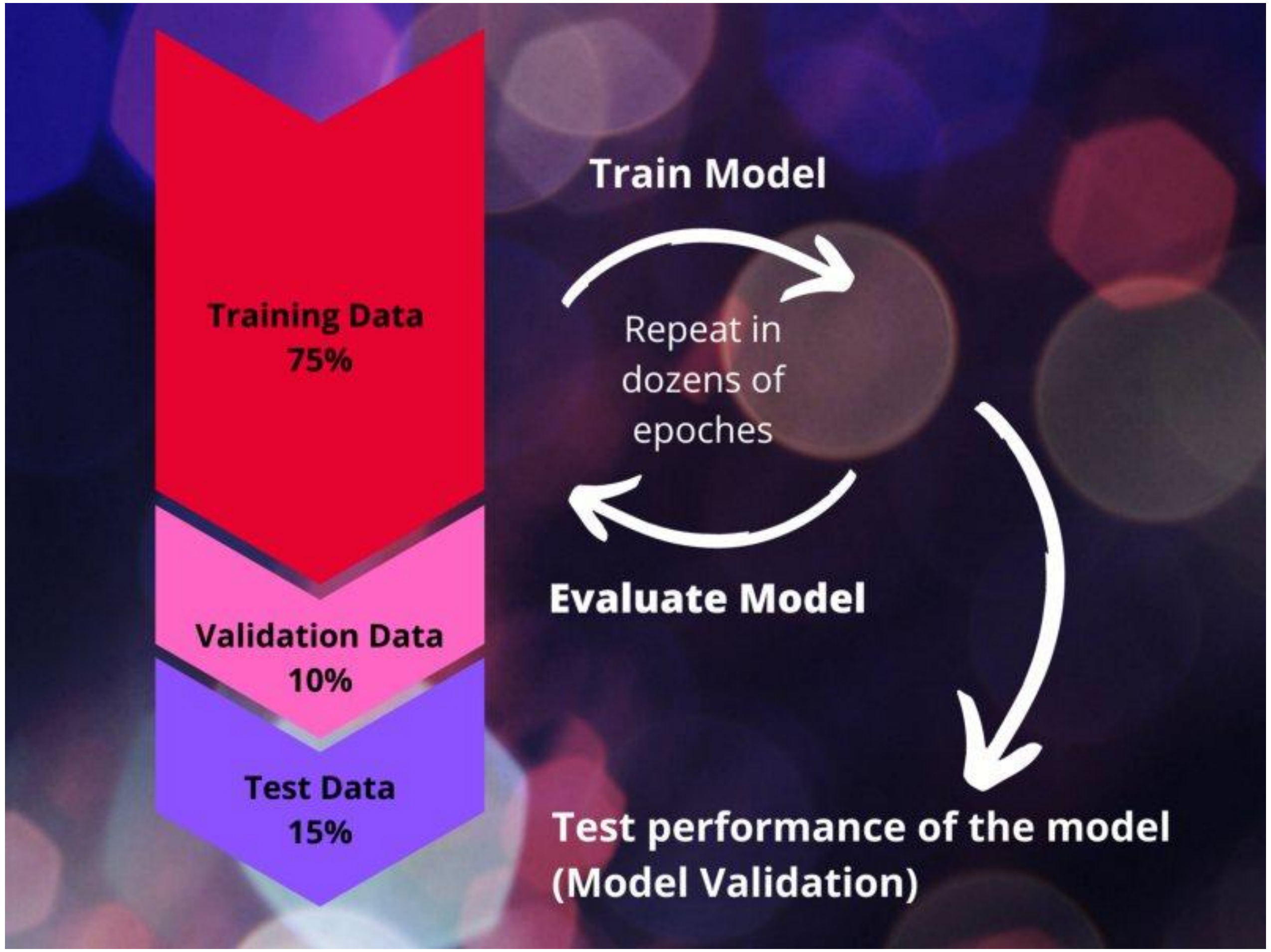
- During the model evaluation, we examine many models and model architectures in order to determine which designs work best with certain data and challenges.
- Some models, such as translation and word categorization, function well with text. Other models, such as categorization, detection, and localization models, function well with images. Our expertise, commitment to best practices, and scientific research led us to the ideal approach for our present project..
- Before we begin training the model, we divide the training data set into actual training data (about 75% of the data), validation data (10%), and test data (15%). Depending on the amount of data available, the actual distribution may differ. For model training, both training and validation data are used. After the training, the test data is used to confirm the model's performance with previously unseen data.

Probability threshold and Overlap Treshholg

- In the performance tab on the left side of your screen is the **Probability Threshold**. By default, it is set to 50%. This means that when the model is 50% sure that a tag applies to an image it will classify the image with that tag.
- **Overlap Threshold** – determines how high the deviation between the defined and the recognized bounding box is – it's all about object recognition in a scene.

Training in Computer Vision

- Computer vision tasks are more difficult and time-consuming to train than text-based machine learning tasks. This is due to the fact that we use deep and complicated models, and the data required for these models is typically vast, ranging from terabytes to petabytes. As a result, calculation takes a lot of time.



Model Validation

- We evaluate the model's quality after completing the training mentioned above. We work with the model to figure out how it behaves: which aspects are well-solved and which aren’t.
- We interpret essential adjustments to the training set by inspecting the visual data in order to optimize result quality.
- For example, an adjustment could be to acquire or synthesize additional data from a particular category.We may need to update the model architecture in some cases, especially if the model is incapable of grasping the goal or simply memorizes the training set (under- and overfitting).

- True Positive (TP) — Correct detection made by the model.
- False Positive (FP) — Incorrect detection made by the detector.
- False Negative (FN) — A Ground-truth missed (not detected) by the object detector.
- True Negative (TN) —This is backgound region correctly not detected by the model. This metric is not used in object detection because such regions are not explicitly annotated when preparing the annotations.

Precision and Recall

- Precision is the degree of exactness of the model in identifying only relevant objects. It is the ration of TPs over all detections made by the model.

$$P = \frac{TP}{TP + TN}$$

- Recall measures the ability of the model to detect all ground truths—proposition of TPs among all ground truths.

$$R = \frac{TP}{TP + FN}$$

Precision, Recall and mAP

Precision - shows you accuracy our model has been correctly detected when it detects objects in one of the test data images (regardless of whether the object was really a Monument or something else). E.g.: has a picture 10 Monuments and were of which 5 real, as well as 3 buildings, were wrongly recognized as Washington Monument, then I have a Precision of 5 / (5+3) = 62.5%

Recall - show us how likely your model will find a real Monument from all sorts of WM posts in picture collection. (simple: how good is our model) E.g. like Precision (5 real postcars + [false MW] obelisks, din afara celor 15 imagini) 5/10 = 20%

mAP (mean Average Precision) - this value reflects the performance of the object detector. Basically, it is a question of how exactly the detector can perceive the detected bounding box to the specified box. This is evaluated and averaged several times under different parameters.

Comparison And Feedback

- It's time to communicate the progress we've made so far with our customer in this stage. We provide our findings regarding the model's quality and condition, as well as what worked and what did not.
- Here, good teamwork with our customer is crucial. We talk about how we can improve the model together, such as acquiring more data and where we can get it. We are planning the next version of model training in close collaboration.

Deployment

- Our current model version's deployment serves as a quality baseline for further training iterations. If the model already brings value to the customer's experience, it might be included in his prototype or even production. Meanwhile, we begin the next training iteration, and the life cycle begins all over again.Keep an eye out for the next installment of this series. We'll go through all there is to know about data, including how to gather it, clean it, and preprocess it. Please do not hesitate to contact us if you require assistance with your AI project.

Thank you,
for your attention!