Gülap, Isirafil

1. What is the meaning of *generalization* in the paper?

It is way of learning *„how to perform important tasks [...] from examples“* and adapting these *„beyond the examples in the training set“*. Which means that your algorithm don‘t only work with the given data that you already know, but ideally on any new data sets that you have not seen before.

2. What are the many faces of overfitting?

A situation where *„your learner outputs a classifier that is […] [noticeably more] accurate on the training data [...] [than] on test data, when in fact it could have output one that is [...] [similar] on both, it has overfit.“*

One form of it is *„caused by noise, like training examples labeled with the wrong class“*. An example for overtitting to occur when there is no noise is to *„learn a Boolean classifier that is just the disjunction of the examples labeled “true” in the training set. [...] This classifier gets all the training examples right and every positive test example wrong, regardless of whether the training data is noisy or not.“*

Another form of overfitting is reladet to *„multiple testing“*. *„Standard statistical tests assume that only one hypothesis is being tested, but modern learners can easily test millions before they are done. As a result what looks significant may in fact not be.“*

3. What is feature engineering?

A combination of which *„features [are] used“* and how to *„construct features“* from *„the raw data [...] [into] a form that is amenable to learning“*. This part is also described as the *„one of the most interesting parts“* of work*“where most of the effort [...] goes“ and „where intuition, creativity and “black art” are [] important“*. This *„iterative process of running the learner, analyzing the results, modifying the data and/or the learner, and repeating“* is very time-consuming because during this trial and error work of feature design you have to *„gather data, integrate it, clean it and pre-process it“*.

4. What is ensemble learning?

In contrary to the early days of machine learning, *„effort [...] went into trying many variations of many learners [...].“* But even later *„instead of selecting the best variation found, [...] [researchers] combine many variations, the results are better.“* Examples for such techniques, which are standard now, are bagging: *„generate random variations of the training set by resampling, learn a classifier on each, and combine the results by voting.“* Two more examples are called *„boosting“* and *„stacking“*.

5. Why does more data beats clever algorithms?

The author gives the following very impressive example: *„Suppose you’ve constructed the best set of features you can, but the classifiers you’re getting are still not accurate enough. What can you do now? There are two main choices: design a better learning algorithm, or gather more data“*. While *„machine learning is all about letting data do the heavy lifting“* it is very obvious that *„the quickest path to success is often to just get more data.“*

The author gives some background information by explaing that in *„most of computer science, the two main limited resources are time and memory. In machine learning, there is a third one: training data. [...] The bottleneck has changed from decade to decade. [...] Today it is often time. Enormous mountains of data are available, but there is not enough time to process it [...]. This leads to a paradox: even though [...] more complex classifiers can be learned, in practice simpler classifiers wind up being used, because complex ones take too long to learn.“*

The other given reasons are *„using cleverer algorithms has a smaller payoff [...] [because] to a first approximation, they all do the same.“* and the *„more sophisticated learners [...] are usually harder to use, because they have more knobs you need to turn to get good results“*.