This article describes doppelganger, which is a situation where the training and validation sets are very similar due to coincidence or other reasons. Such data may lead to the problem that even though the model is underfitted to the data, the model can still perform well on the validation set due to the similarity of the validation and training data.

This article summarizes the previous methods for detecting doppelganger in data. Firstly, logical approach can be used to reduce the dimensionality of data to the point where it can be visualized through dimensionality reduction algorithms such as PCA, or embedding methods, to see the distribution of samples in the reduced space. However, this approach is not feasible because doppelganger is not necessarily distinguishable in the reduced dimensional space. Secondly some studies use dupChecker method to detect doppelganger, this method is to determine the duplicate samples by comparing the MD5 fingerprint of these data CEL files. But this detection method only detects duplicate samples by fingerprint, not really detecting doppelganger in the data, and this method cannot detect the kind of doppelganger samples that are generated by chance. Finally, another method is the PPCC (pairwise Pearson's correlation coefficient) method, which is generally used to capture the relationship between sample pairs from different data sets. An unusual high value of PPCC indicates the possible presence of doppelganger in a pair of samples.

I think the doppelganger effects are very common in data science including biology, computer science, computer vision, natural language processing and other fields. For example, in the work of sentiment analysis, I have done a small project that used CNN for sentiment analysis, although the data set is small, the model can correctly distinguish the sentiment of words or phrases such as "good", "very good", "bad", "not good", etc., but the training samples and validation samples are more similar, so the model can achieve 80% The model can achieve more than 80% correct rate. However, due to the doppelganger effect, when the model is used to predict phrases such as "not bad", it classifies them as negative comments, so the model is not well trained, but the correct rate of the doppelganger effect model is only very high in the validation set. And similarly, in computer vision, for example, in target recognition work, if we use a CNN

model to recognize cats and dogs, and we use the same cat (e.g., Domestic short-haired cat) and the same dog (e.g., Chihuahua) in both the training and validation sets, then the model may classify very well on the validation set. But if I use a picture of a Sphynx cat, then the model may recognize the Sphynx cat as a dog, and this problem is also due to doppelganger effects.

According to the suggestions in the article doppelganger effects can be avoided in several ways, an approach of data layering, an approach of implementing metadata cross-checking and an approach of including as much data as possible for data robustness checking.

By reading the literature, I refer to other machine learning directions for mining doppelganger in data. First, I think that for small sample datasets, it is possible to maintain, for all samples, a list of similarities between that sample and other samples before starting training, and then obtain smaller batches with lower similarity by sampling similar samples, and the validation set is generated in the same way. The training and validation sets generated in this way can reduce the similarity between samples and thus avoid the doppelganger effect, but the disadvantage of this method is that it increases the computational overhead, so I think this method can only be used in small sample datasets (Smirnov et al., 2017).

Also, I have an idea that if we can organize the doppelganger data, we might be able to get a dataset on doppelganger. If the data is sufficient, we can identify the doppelganger in the data by designing a machine learning based classifier, and in this way identify the doppelganger in the data so as to avoid the doppelganger effect

References

[1] Smirnov, E., Melnikov, A., Novoselov, S., Luckyanets, E., & Lavrentyeva, G. (2017). Doppelganger mining for face representation learning. In *Proceedings of the IEEE international conference on computer vision workshops* (pp. 1916-1923).

[2] Pennekamp, J., Henze, M., Hohlfeld, O., & Panchenko, A. (2019, May). Hi doppelgänger: Towards detecting manipulation in news comments. In *Companion*

Proceedings of The 2019 World Wide Web Conference (pp. 197-205).

[3] Rathgeb, C., Fischer, D., Drozdowski, P., & Busch, C. (2022). Reliable detection of doppelgängers based on deep face representations. *IET Biometrics*, *11*(3), 215-224.