with the same data split I won’t have any difference?

If you keep the **same data split** (i.e., no changes to the division between training, validation, and test sets) but still see a significant difference between validation and test performance, this discrepancy would likely persist in subsequent model training. This is because the root cause of the performance drop isn’t related to a particular run but rather to **how the model is fitting the data** and the nature of the data itself.

Here's why this happens and some further insights:

**Why the Same Data Split Still Shows a Difference**

1. **Overfitting to Training and Validation Data**:
   * If the model achieves high accuracy on the training and validation data but performs poorly on the test set, it likely indicates **overfitting** to patterns specific to the training and validation data that do not generalize to the test set. Overfitting is not inherently solved by keeping the same data split—it typically requires model tuning, simplification, or regularization to improve generalization.
2. **Validation and Test Set Represent Different Distributions**:
   * If your validation set is more similar to the training set than the test set, the model will tend to perform well on both training and validation data but struggle on the test set. This could be due to **sampling bias** during the data split, where the training and validation sets are inadvertently more similar to each other than to the test data.
   * For instance, in time-series or geographically structured data, certain patterns or clusters may be present in training and validation sets but not in the test set. This will lead to validation performance appearing high, while the model underperforms on truly unseen data in the test set.
3. **Insufficient Regularization**:
   * Without sufficient regularization (e.g., dropout, weight decay), the model may memorize specific details of the training and validation data, giving it high accuracy on these sets but poor generalization on the test set.
   * Keeping the same data split without introducing regularization won’t change the fact that the model is overly complex and prone to memorizing details rather than generalizing to new patterns.

**What You Can Do to Improve Generalization**

If you must keep the same data split, there are several strategies you can try to make your model generalize better:

1. **Increase Regularization**:
   * Add or increase regularization techniques like **L2 weight decay**, **dropout layers**, and **batch normalization** to help the model avoid overfitting specific features of the training/validation data.
2. **Simplify the Model Architecture**:
   * A simpler model (with fewer parameters or layers) is less likely to overfit than a highly complex one. Try reducing the number of layers or nodes in the model, which can force the model to focus on more general patterns rather than fitting noise or irrelevant patterns in the data.
3. **Cross-Validation**:
   * Even with the same test set, using **k-fold cross-validation** on the training and validation data can give you a better sense of the model’s ability to generalize. By rotating the training and validation sets, you get more comprehensive validation accuracy, which could help you understand if the high validation accuracy was an anomaly due to the specific split.
4. **Data Augmentation** (if applicable):
   * For image, text, or certain structured data types, augmenting the training data can help the model generalize better by learning from variations in the input. This is particularly useful in domains like computer vision or NLP.
5. **Early Stopping**:
   * Use **early stopping** during training, which stops training when the validation loss stops improving. This can prevent overfitting by ensuring that training doesn’t continue once the model’s performance on the validation data starts to decline.

**Summary**

Maintaining the same data split without changing the model or training approach will likely lead to the same outcome: high validation accuracy but low-test accuracy. To improve test accuracy, focus on regularization, simplifying the model, or using data augmentation and cross-validation techniques. These strategies will encourage the model to learn patterns that are likely to generalize better, even with the same dataset split.