Here’s your text translated into clear, professional English:

**Code Modification Notes**

**Transformer Part**

1. The START\_DATE and END\_DATE in config must match the start and end times of the dataset. This has already been fixed in the current dev-crypto branch.
2. The USE\_CSI\_300\_TICKET parameter in config lists all dataset filenames. If datasets are added or removed, this must be updated manually. You should also update the use\_ticker\_dict below. The CSI\_date parameter has already been set according to the current data time span, but if you plan to train on longer datasets later, this parameter will also need to be adjusted.
3. Note that the first starting time of the CSI\_date parameter must be at least 60 rows after the actual start of the dataset, since the two Transformer branches for time-series prediction require looking back 60 rows.
4. In the Transformer MAE training script, the three parameters enc\_in, dec\_in, and c\_out should be set to **the number of training dataset files + 8 (8 technical indicators)**, and must be modified manually if you change the dataset file number or the technical indicators in the config.
5. In the Transformer branch, line 91 of Stock\_data\_handle.py must be modified to read the time format as year-month-day hour:minute:

df['date\_str'] = df['date'].apply(lambda x: datetime.datetime.strftime(x,'%Y-%m-%d %H:%M'))

1. You may have some function call issues like .fillna, or \_seed(). This is due to the python libs version. Some suggested lib versions in the requirements.txt file are out of date. We need to install the latest version and not compatable with the original code. I have modified some, and feel free to correct the rest issues if you meet.

**RL Part**

1. Config modifications are similar to those in the Transformer part, except the time\_window\_start parameter in config has not been changed. In practice, it should be regenerated as a random sequence according to the data length.
2. Around line 200 of DRLAgent.py, you can save the raw outputs of actions. This has already been implemented in my code.
3. In train\_rl.py, the MAESAC\_PARAMS parameters (enc\_in, etc.) should match those in the Transformer MAE branch, i.e., **the number of dataset files + 8**.
4. In train\_rl.py, check the following line: model\_path = os.path.join('trained\_models/', version, model\_name, 'best\_model.zip')

Confirm whether the best\_model should be used here.