```
import tensorflow.keras as keras

from .DQN import DQN

class TargetDQN(DQN):

    def __init__(self, target_update_freq, **kwargs):
        super().__init__(**kwargs)

        self.type = "TargetDQN"

        self.setup_target_model()
        self.target_update_freq = target_update_freq

    def setup_target_model(self):
        self.target_model = keras.models.clone_model(self.model)
        self.update_target_model()

    def update_target_model(self):
        self.target_model.set_weights(self.model.get_weights())

    def get_next_Q_values(self, next_states):
        return self.target_model.predict(next_states)

    def execute_episode(self, **kwargs):
        super().execute_episode(**kwargs)

        if self.episode % self.target_update_freq == 0:
            self.update_target_model()
            if "verbose" in kwargs.keys():
                if kwargs["verbose"]:
                    print("\tUpdate target model")
```