

Sieci Neuronowe w Pythonie: wprowadzenie, struktura i zastosowanie w strategii tradingowej - Raport podsumowujący

Jakub Kępka
Damian Kąkol
Karol Gutkowski

Maj 2024

Spis treści

1	Wstęp	2
2	Dane wejściowe	2
3	Architektura modelu	2
3.1	LSTM - Long Short Term Memory	2
3.2	Architektura	3
4	Eksperymenty i wyniki	3
4.1	Study Case 1 (akcje Game Stop)	3
4.2	Study Case 2 (akcje APPLE)	5
4.3	Study Case 3 (TESLA)	7
4.4	Study Case 4 (ADOBE)	8
4.5	Study Case 5 (gld)- złoto	8
5	Wnioski	10

1 Wstęp

W tym podsumowaniu przedstawiamy efekty pracy nad przygotowaniem modelu uczenia maszynowego w celu predykcji zmian cen akcji na rynku giełdowym. Przedstawimy wybrany model danych wejściowych, architekturę modelu oraz przedstawimy wyniki działania systemu.

2 Dane wejściowe

Źródłem naszych danych były notowania publicznych spółek z głównie amerykańskich giełd NASDAQ oraz NYSE. Z notowań pobieraliśmy ceny otwarcia oraz zamknięcia dla danego oraz maksymalną i minimalną cenę akcji dnia. W naszej analizie przyjęliśmy model w którym, gdy algorytm mówi, że danego dnia cena powinna wzrosnąć to podejmujemy pozycję długą (ang. long), w przeciwnym przypadku pozycja krótka (ang. short).

Pobrane dane wejściowe zmodyfikowaliśmy do wektora wejściowego dla modelu poprzez stworzenie następujących pól

- H-L - różnica między najwyższym, a najniższym notowaniem dnia
- O-C - różnica między ceną otwarcia, a zamknięcia danego dnia
- 3 day MA - średnia krocząca z trzech ostatnich dni
- 10 day MA - średnia krocząca z trzech ostatnich dni
- 30 day MA - średnia krocząca z trzech ostatnich dni
- std_dev - odchylenie standardowe z 5 ostatnich dni

W naszym drugim modelu wykorzystaliśmy powyższe dane historyczne z pewnymi dodatkowymi elementami. Dodaliśmy danego pogodowe z kilku stacji z różnych lokalizacji ze stanów zjednoczonych. Konkretnie z Jacksonville, Los Angeles i Nowy Jork. Dane pobralismy z api ze strony www.ncdc.noaa.gov (National Centers for Environmental Information). Kolejnym dodatkowym elementem były indeksy giełdowe, w tym sp500. Próbowaliśmy dodać jeszcze wyniki realnej gospodarki mierzone w pkb. Niestety dane historyczne były podawane co kwartał, co nas nie zadowalało, dane potencjalnie mogłyby zakłamać realny stan gospodarki (początek kwartału może różnić się bardzo od końca kwartału).

3 Architektura modelu

3.1 LSTM - Long Short Term Memory

W naszym modelu wykorzystaliśmy sieci LSTM. LSTM, są rodzajem sieci rekurencyjnych tzn. takich, które posiadają pamięć swoich poprzednich outputów. W klasycznych RNN'ach pojawia się często problem zależności od starszych wyników wyjścia, to często prowadzi do zmniejszonych wpływów nowych danych na zmianę parametrów modeli. Tutaj sprawdzają się właśnie LSTM, które pozwalają modelowi regulować poziom istotności nowych faktów względem starych poprzez mechanizm zapamiętywania i zapominanie.

3.2 Architektura

```
1 def generate_model_architecture(shape):
2     model = Sequential()
3     model.add(LSTM(units=50, return_sequences=True, input_shape=shape))
4     model.add(Dropout(0.2))
5     model.add(LSTM(units=50, return_sequences=True))
6     model.add(Dropout(0.2))
7     model.add(LSTM(units=50, return_sequences=True))
8     model.add(Dropout(0.2))
9     model.add(LSTM(units=50))
10    model.add(Dropout(0.2))
11    model.add(Dense(units=1))
12    return model
```

Listing 1: Kod do stworzenia architektury modelu

4 Eksperymenty i wyniki

Na wykresach zwroty są rozumiane jako wartość bezwymiarowa ponieważ korzystamy do obliczenia ich wartości za pomocą logarytmu (często używanych w analizie finansowej) z ilorazu ceny zamknięcia i otwarcia (cena zamknięcia z dnia poprzedniego), pomimo że wartości nie oddają w pełni zwrotów to użycie logarytmu umożliwia łatwe obliczanie skumulowanych zysków, umożliwia wyznaczenie zwrotów z operacji short oraz zachowuje prawidłową relację względną między użyciem modelu a zwrotami z rynku.

Market returns na poniższych wykresach reprezentuje zyski z rynku tj. sytuację gdy na giełdzie jest zajmowana pozycja long i utrzymywana przez cały czas, co w wyniku długoterminowego wzrostu rynku daje wymierne korzyści (Market returns nie uwzględnia możliwości zajęcia pozycji short).

Ponadto zakładamy, że rynek zawsze jest płynny (transakcje są wykonywane po cenie rynkowej natychmiast) oraz pomijamy opłaty brokerskie.

4.1 Study Case 1 (akcje Game Stop)

Game Stop stał się niezwykle popularny za sprawą swojego nagłego wzrostu akcji w 2021 za sprawą grupy redditowców z r/wallstreetbets. Na początku 2021 roku akcje firmy wzrosły z ok 1 USD do 80 USD w szczycie. Akcje GameStop zaliczyły rekordowy skok ceny jednego dnia o całe 135%.

Z poniższego wykresu możnaby wywnioskować, że model radzi sobie dobrze przy reagowaniu na wahania rynku nawet tak szybkie jak game stop.



Rysunek 1: Zwrot z tradingu na akcjach GAME STOP

4.2 Study Case 2 (akcje APPLE)



Rysunek 2: Zwrot z tradingu na akcjach APPLE - krótki horyzont czasowy



Rysunek 3: Zwrot z tradingu na akcjach APPLE - dłuższy horyzont czasowy

Akcje apple pokazują, że model radzi sobie korzystnie (a przynajmniej akceptowalnie) kiedy operuje na danych na nieodległym horyzoncie czasowym w stosunku do okresu uczenia, tak jakby "wyczuwał" aktualny nastrój rynku w procesie uczenia i po upływie odpowiedniej ilości co raz bardziej odbiegał od konkurencji w postaci rynku.

4.3 Study Case 3 (TESLA)



Rysunek 4: Zwrot z tradingu na akcjach TESLA

4.4 Study Case 4 (ADOBE)



Rysunek 5: Zwrot z tradingu na akcjach ADOBE

Na akcjach firmy ADOBE model radzi sobie dużo gorzej, analogicznie jak na akcjach APPLE im więcej czasu mija tym rozjazd jest większy.

4.5 Study Case 5 (gld)- złoto

O ile złoto nie występuje na rynku w postaci zwykłej akcji to łatwo można odtworzyć kurs złota za pomocą kontraktu ETF, który replikuje kurs złota. Należy tutaj zaznaczyć, że w rzeczywistości na kontraktach ETF nie wolno dokonywać analogicznego tradingu jak na akcjach, i kurs ETF GLD nie oddaje idealnie kursu złota, ale występuje tutaj w ramach atrapy złota. W okolicach 2013 roku złoto zanotowało istotny spadek ceny w swoim kursie, ale od tego czasu konsekwentnie rok w rok notuje wzrosty. Widać, że model nie odstaje znacząco od rynku to jednak w przypadku złota traktowanych w tradingu jako "safe heaven" (dobro które stanowi bezpieczny wybór dla inwestora, bez niecodziennych wahań kursu) dzienny trading może być gorszy od zwykłej pozycji long.



Rysunek 6: Zwrot z tradingu na kontrakcie etf GLD

5 Wnioski

W naszej analizie korzystaliśmy z podejścia, które umożliwiło wykonanie zadania tj. uczenie modelu na jednej wybranej akcji do pewnego czasu, i testowanie go na przedziale czasowym występującym bezpośrednio po okresie trenowania. Pomimo, że korzystaliśmy z platform chmurowych to brakuje mocy obliczeniowej do stworzenia dużego modelu wytrenowanego np. na 100 największych spółkach nasdaq, które mogły reprezentować rynek. Założenie o wykonywaniu transakcji na początku jednego dnia tworzy lukę do "popsucia" modelu, wystarczy znaleźć akcję, w której w ramach jednego dnia następuje drastyczny spadek lub wzrost, a poprzednie dni nie wskazują na taki ruch. Takie akcje to np. spółki biotechnologiczne, które w wyniku jakiegoś odkrycia notują spektakularne wzrosty w ramach jednego dnia, niestety wiele z tych spółek jest dosyć młoda i wzrosty następowały niedługo po wejściu na rynek i brakowało danych do trenowania modelu, dla tego nie wystąpiły one w analizie. Pominięcie w analizie opłat brokerskich znacząco poprawia wyniki modelu, w końcu dokonuje on codziennie jednej transakcji i nie ponosi konsekwencji jeśli dokonuje błędnych decyzji w ramach małych fluktuacji ceny akcji (opłaty mogły by być znacząco większe od zysków lub strat). W kwestii efektywności modelu należy również zauważyć, że niekoniecznie nadają się do grania na takich aktywach jak np. złoto, które krótkoterminowo zalicza małe fluktuacje ceny, a w długim terminie notuje wzrosty, w tej sytuacji model może niepotrzebnie wykonywać tradnig na nieznających ruchach cenowych.

Wyniki drugiego modelu nie różniły się od wyników modelu pierwszego. Wynika to najprawdopodobniej z małej ilości danych. Użyteczność podejścia przy drugim modelu wynika z efektu skali i większej ilości danych.

Literatura

- [1] Wpis na temat LSTM - <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [2] tworzenie modelu - <https://blog.quantinsti.com/neural-network-python/?fbclid=IwAR3HZkBYZkuuIfVKGddgZemmEKRLJZIKrBCCr1HrRbDuCNMtZl19UuhSPaA>