

1 Streszczenie

Optymalizacja pracy pieca poprzez dostosowanie przepływu powietrza dystrybucyjnego wymaga od nas dokładnego zrozumienia jego zasady działania. W tym celu stworzyliśmy rekurencyjną sieć neuronową (**LSTM**) i wytrenowaliśmy ją na danych pomiarowych. Dało nam to możliwość predykcji skutków potencjalnych zachowań, którą dzięki algorytmom przeszukującym zastosowaliśmy do znalezienia najlepszego ciągu akcji sterownik.

2 Informacje techniczne

W tej sekcji rozwiniemy zagadnienia dotyczące modelu, który stworzyliśmy, przedstawimy plusy oraz minusy zastosowanych algorytmów, wymienimy powody zastosowania określonych rozwiązań.

2.1 Funckja celu

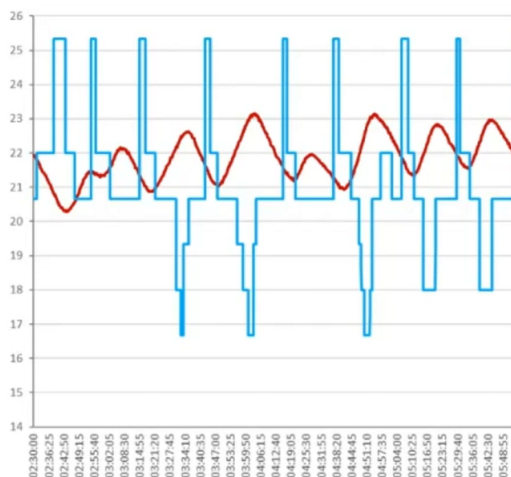
Aby dobrze poradzić sobie z zadaniem musielimy najpierw szczegółowo określić jaki efekt chcemy uzyskać. Konkretnie zależy nam na określeniu funkcji, która dla danych opisujących straty KSR zwróci nam wartość liczbową określającą, jak efektywnie działał piec.

Zdecydowaliśmy się na funkcję:

$$F(D) = \frac{1}{|D|} \sum_{d \in D} (d - \delta)^2$$

Gdzie D to zbiór pomiarów strat KSR, a δ to średnia wartość straty. Własności funkcji F :

- Jest tym mniejsza im bardziej wykres strat się wypłaszcza
- Znacząco kara duże odstępstwa od normy (znaczny wzrost/spadek strat)
- Jest niezależna od rozmiaru próbki. Można porównywać w prosty sposób efektywność pieca zarówno na podstawie dziesięciu minut jego pracy, jak i dziesięciu godzin.



Objective function loss: 1.714
Mean: 24.94

Tak określony wzór daje nam dobrze zdefiniowany problem optymalizacyjny. Mianowicie im niższa wartość funkcji F tym piec wydajniej pracuje.

2.2 Model sieci neuronowej

Maszynista stojący przed podjęciem decyzji o zmianie przepływu powietrza musi zadać sobie pytanie - jaką zmianę spowoduje? Kiedy on odpowiada sobie na nie na podstawie wiedzy i intuicji, my zaprzęgniliśmy algorytm sieci neuronowych badający tendencje i ukryte zależności danych. Nasz model potrafi podać jak zachowa się piec dla dowolnej akcji jaką można na nim wykonać. Rezultat ten otrzymaliśmy dzięki zastosowaniu rekurencyjnych sieci neuronowych (**LSTM**)...

2.3 Algorytm przeszukujący

Gdy umiemy już przewidzieć jaką reakcję spowoduje dana akcja, możemy starać się odpowiedzieć na pytanie - jaki jest optymalny ciąg akcji na najbliższe kilka minut? A w szczególności na pytanie - jaką akcję powinno się podjąć w danym momencie? Zauważmy, że przeszukiwanie wszystkich możliwych akcji dla każdej sekundy z kolejnych paru minut, daje astronomicznej wielkości problem. Aby zmniejszyć przestrzeń poszukiwań i zarazem skupić się na rozwiązaniach sensowniejszych ograniczyliśmy możliwe ustawienia przepływu powietrza do dziewięciu opcji, natomiast dane zbierane co sekundę pogrupowaliśmy w sekcje 30-sekundowe. Chociaż liczba kombinacji możliwych akcji wciąż jest przerażająco wielka, to daje już pewne nadzieje na znalezienie ciągu sensownych decyzji. Aby tego dokonać zastosowaliśmy algorytm **Monte Carlo Tree Search** będący jednym z najbardziej zaawansowanych algorytmów tej dziedziny informatyki. Jego własności:

- W przeszukiwaniu rozwiązań skupia się znacznie bardziej na tych dających lepsze rezultaty. Nie zapomina jednak całkiem o gorszych akcjach, ponieważ czasem ich drobna modyfikacja prowadzi do lepszych rezultatów, niż znalezione obecnie.
- Jest algorytmem ANY-TIME, to znaczy, że skończy się kiedy my powiemy mu, aby się skończył. Innymi słowy im dłużej będzie działał, tym lepsze rezultaty osiągnie.
- Daje możliwość zrównoleglenia, dzięki czemu stosując jego wersję zaimplementowaną na karcie graficznej uzyskuje się niesamowite przyspieszenie, a co za tym idzie poprawę efektywności.

Aby podkreślić skuteczność zastosowanego algorytmu można podać jego osiągi. Algorytm MCTS był podstawą do stworzenia systemu AlphaGo, który to jako pierwszy na świecie pokonał mistrza świata w chińskiej grze w Go.

3 Podsumowanie

Nasze podejście wykorzystuje jedno z najbardziej zaawansowanych i zoptymalizowanych algorytmów w danych dziedzinach, jednak nierozsądnym byłoby rezygnować z wiedzy, doświadczenia i intuicji osób pracujących z piecem na co dzień. Te dwa podejścia do problemu można połączyć. Intuicje opisane odpowiednim językiem można przekazać komputerowi, a algorytmy które stosujemy są wprost przygotowane pod dołożenia kluczowego elementu jakim jest wiedza ekspertów. Dlatego najlepsze osiągi moglibyśmy uzyskać mierząc się z problemem wspólnie.