

Progettazione e sviluppo di test suite automatica e modulo raccolta dati per intelligenze artificiali che si occupano di creare strategie di investimento

Garion Musetta - 920623

6 aprile 2020

1 Ente presso cui è stato svolto il lavoro di tesi

La tesi è stata svolta presso Nexid s.r.l (<https://nexid.it/>), nella divisione Blockchain e Artificial Intelligence, Nexid Edge. L'azienda si occupa dello sviluppo di strumenti intelligenti per l'analisi del mercato delle criptovalute e la creazione di strategie di investimento. Opera inoltre sulla Blockchain Ethereum per la realizzazione di Smart Contract e per la certificazione di opere in diversi tipi di settori.

2 Contesto iniziale

Nell'ambito di AI per trading finanziario, Nexid ha sviluppato un agente in grado di creare strategie di investimento intelligenti (Sentymment AI). Il software è da perfezionare e testare e necessita della riprogettazione dell'infrastruttura per l'acquisizione dati, oltre ad altri componenti secondari.

Sentymment è composta da diverse AI che si comportano in modo simile, ma soltanto una di queste deve essere scelta periodicamente come miglior rappresentante, da utilizzare in produzione. Ogni valuta trattata ha associato un insieme specifico di AI. La scelta della migliore si basa su una misura di prestazioni calcolata per le varie AI.

3 Obiettivi del lavoro

Gli obiettivi della tesi sono diversi. È stata richiesta innanzitutto la progettazione di un modulo di raccolta dati allo scopo di rendere le informazioni disponibili alle AI, insieme ad un coordinatore che espone delle API per interrogare lo stato del sistema e attivare le diverse funzionalità dello stesso.

Concentrandosi sulle AI per trading, l'obiettivo principale è lo sviluppo di uno strumento intelligente in grado di scegliere la migliore fra le AI che compongono Sentyment. Il software sviluppato allo scopo, qui chiamato *meta-learner*, verrà integrato in Sentyment e agirà in automatico applicato ad ogni criptovaluta trattata. *Meta-learner* impara dai risultati prodotti dalle AI nel tempo e, dopo un sufficiente numero di dati, è in grado di scegliere una AI migliore fra tutte le altre, in base ad alcune metriche definite.

Lo scopo di questa tesi è anche testare Sentyment, trattandola come un'unica AI composta dal suo rappresentante migliore, scelto in precedenza. È eseguito un confronto delle prestazioni rispetto ad altri strumenti simili, principalmente altre strategie di investimento utilizzate nell'ambito di applicazione.

4 Descrizione lavoro svolto

In primo luogo si sono definite alcune metriche attraverso cui paragonare due strategie di investimento, producendo un risultato numerico. Poi è stato approssiato il problema di scelta della migliore AI.

Sfruttando un algoritmo di *reinforcement learning*, detto *multi-armed bandit*, si è creato un software che utilizza le metriche ed è capace di adattarsi al problema e produrre risultati soddisfacenti. Tutti i parametri e le scelte affrontate sono analizzate e discusse con attenzione. Lo strumento verrà infine applicato a molti più insiemi di AI, ovvero quelli relativi alle altre criptovalute trattate da Sentyment.

Per quanto riguarda i test con altri strumenti, sono prese in esame alcune strategie di investimento note ed ampiamente utilizzate e sono usate come confronto rispetto alle strategie prodotte da Sentyment. Sono analizzati diversi istanti temporali e lunghezze di periodo, applicando le metriche definite al fine di dare una misura qualitativa del software.

Durante l'analisi dei risultati si scopre un modello ottimale di scelte di investimento a breve termine (*maximum profit*), è sviluppato più di un algoritmo per riprodurlo e il modello è impiegato per confrontare ulteriormente i risultati di Sentyment e delle altre strategie. Fornisce anche spunti per eventuali lavori futuri.

5 Tecnologie coinvolte

Durante tutto il progetto si è fatto largo uso del linguaggio Python (Python3) e delle principali librerie di data science come *pandas*, *numpy* e *matplotlib*, usate per l'analisi dati.

Lo sviluppo degli agenti intelligenti è stato facilitato da alcune librerie Python open source per *multi-armed bandit*, poi riadattate per lo scopo. Da citare anche i numerosi strumenti per l'analisi finanziaria come *pyfolio* e *zipline*.

Per quanto riguarda lo sviluppo del modulo raccolta dati, si sono utilizzate le API Kraken (piattaforma online di trading) per lo scaricamento dati, un database relazionale *MySQL* per la memorizzazione e librerie Python per il mapping ORM quali *sqlalchemy*.

Infine sono esposte alcune API per l'interrogazione del sistema Sentyment, implementate attraverso *flask*.

Altri strumenti necessari e non già esistenti sono stati sviluppati, come *maximum profit* e parte del software di analisi.

Alcuni metodi che implementano le metriche erano a disposizione all'interno della codebase di Nexid.

6 Competenze e risultati raggiunti

I risultati raggiunti rispettano gli obiettivi iniziali. Il software di scelta di una migliore AI ha prodotto risultati accettabili, anche se non è stato possibile testarlo applicato a numerose altre valute. Sarebbero possibili alcuni miglioramenti, infatti inizialmente si voleva scegliere periodicamente una AI migliore a distanza di qualche settimana, cosa molto difficile da realizzare con lo strumento pensato. Si è appreso dunque il valore della ricerca continua e di come a volte una soluzione che sembra adatta al problema non può dare esattamente i risultati sperati a causa di altri fattori, che si scoprono soltanto quando lo strumento è ormai stato sviluppato. Queste caratteristiche dei dati però non sarebbero emerse se non si fosse intrapresa quella strada. Infatti, *meta-learner* non è in grado di cambiare scelta di AI ogni settimana, perchè il training di *multi-armed bandit* necessita di una quantità troppo grande di dati, non disponibili nel contesto. Per sviluppare qualsiasi strumento di AI si devono analizzare attentamente i *dati* e il loro contesto.

Il confronto con altre strategie ha favorito l'applicazione di nozioni già conosciute in ambito statistico ad un contesto particolare, dove si parla sempre di dati ma si fa riferimento ad una specifica applicazione; sarà più facile in futuro declinare i concetti generali alle diverse forme dei dati.

Va infine citato l'aspetto collaborativo dello sviluppo: gli strumenti creati sono utilizzati da altre persone, non necessariamente informatici. Parti del software sono state scritte da altri. Interfacciarsi con esperti di settori affini aiuta a sviluppare un ulteriore senso di astrazione necessario per il dialogo.

Riferimenti bibliografici

- [1] Sutton, R. S., & Barto, A. G. (1998). Introduction to reinforcement learning (Vol. 135). Cambridge: MIT press.

- [2] Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5), 988-999.
- [3] Yoon, Y., Swales Jr, G., & Margavio, T. M. (1993). A comparison of discriminant analysis versus artificial neural networks. *Journal of the Operational Research Society*, 44(1), 51-60.
- [4] Subramanian, V., Hung, M. S., & Hu, M. Y. (1993). An experimental evaluation of neural networks for classification. *Computers & operations research*, 20(7), 769-782.
- [5] Tsibouris, G., & Zeidenberg, M. (1995). Testing the efficient markets hypothesis with gradient descent algorithms. In *Neural networks in the capital markets* (Vol. 8, pp. 127-136). Chichester, UK: Wiley.
- [6] Maddala, G. S., & Lahiri, K. (1992). *Introduction to econometrics* (Vol. 2). New York: Macmillan.
- [7] Kim, K. J., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert systems with Applications*, 19(2), 125-132.
- [8] Kim, J. W., Weistroffer, H. R., & Redmond, R. T. (1993). Expert systems for bond rating: a comparative analysis of statistical, rule-based and neural network systems. *Expert systems*, 10(3), 167-172.
- [9] Kuo, R. J., Chen, C. H., & Hwang, Y. C. (2001). An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy sets and systems*, 118(1), 21-45.
- [10] Yang, H., Chan, L., & King, I. (2002, August). Support vector machine regression for volatile stock market prediction. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 391-396). Springer, Berlin, Heidelberg.
- [11] Hong, T., & Han, I. (2004). Integrated approach of cognitive maps and neural networks using qualitative information on the World Wide Web: the KBNMiner. *Expert Systems*, 21(5), 243-252.
- [12] Fung, G. P. C., Yu, J. X., & Lam, W. (2003, March). Stock prediction: Integrating text mining approach using real-time news. In *2003 IEEE International Conference on Computational Intelligence for Financial Engineering*, 2003. Proceedings. (pp. 395-402). IEEE.
- [13] Tsaih, R., Hsu, Y., & Lai, C. C. (1998). Forecasting S&P 500 stock index futures with a hybrid AI system. *Decision Support Systems*, 23(2), 161-174.