

OTDK-dolgozat

2024.

BUDAPESTI METROPOLITAN EGYETEM
ÜZLETI, KOMMUNIKÁCIÓS ÉS TURISZTIKAI KAR



**A TŐZSDEI KERESKEDEÉS OPTIMALIZÁLÁSA GÉPI INTELLI-
GENCIA ALKALMAZÁSÁVAL**
**OPTIMIZING STOCK TRADING USING MACHINE INTELLI-
GENCE**

Bárdi Alex

Konzulens: Dr. Erdeiné Dr. habil Késmárki-Gally Szilvia PhD.

Kézirat lezárásának dátuma: 2023. 11. 23.

TARTALOMJEGYZÉK

1.	BEVEZETÉS	1
2.	TÖRTÉNETI ÁTTEKINTÉS AZ ALGORITMIKUS KERESKEDÉSI MODELLEKRŐL.....	2
3.	AZ ALGORITMIKUS KERESKEDÉS ALAPELVEI ÉS MECHANIZMUSAI	5
3.1.	AZ ALGORITMIKUS KERESKEDÉS DEFINÍCIÓJA ÉS FŐ JELLEMZŐI	5
3.2.	KERESKEDÉSI ALGORITMUSOK TÍPUSAI.....	5
3.3.	ALGORITMUSOK KIVÁLASZTÁSA.....	6
3.4.	AZ ALGORITMIKUS KERESKEDÉS KIHÍVÁSAI.....	7
4.	MAGAS FREKVENCIÁS KERESKEDÉS: ELŐNYÖK, KIHÍVÁSOK, PIACI HATÁS.....	9
4.1.	A HFT SZEREPE AZ ALGORITMIKUS KERESKEDÉSBEN	9
4.2.	A MAGAS FREKVENCIÁS KERESKEDÉS TECHNOLÓGIAI ALAPJAI	9
4.3.	ELŐNYÖK ÉS ELŐSEGÍTŐ TÉNYEZŐK	10
4.4.	KIHÍVÁSOK ÉS KOCKÁZATOK	11
4.5.	A PIACI HATÁSOK ÉS SZABÁLYOZÁSI TÉMAKÖRÖK	12
5.	A GÉPI TANULÁS ALKALMAZÁSA AZ ÁRFOLYAMOK ELŐREJELZÉSÉBEN	15
5.1.	HIPOTÉZISTÉR KORLÁTOZÁSA.....	16
5.2.	FELÜGYELT TANULÁS	19
5.3.	FELÜGYELET NÉLKÜLI TANULÁS.....	20
5.4.	A TÚLILLESZTÉS CSÖKKENTÉSE KÁLMÁN-SZŰRŐVEL	21
6.	AZ ALGORITMIKUS KERESKEDÉSI PLATFORMOK	24
6.1.	A METATRADER 5 (MT5) RENDSZER HASZNÁLATA AZ ALGORITMIKUS KERESKEDÉSBEN	25
7.	A TECHNIKAI MUTATÓK ÉS A NEURÁLIS HÁLÓZATOK HASZNÁLATA A RÉSZVÉNYPIACI TRENDÉK ELŐREJELZÉSÉBEN.....	40
7.1.	A NEURÁLIS HÁLÓ IMLEMENTÁLÁSA	44
7.2.	TESZTELÉS A VÉTELI JELZÉSEK ALAPJÁN JÖVŐBELI ADATOKRA VONATKOZTATVA.....	47
8.	A SZENTIMENTÁLIS ELEMZÉS, MÓDSZEREK ÉS ALKALMAZÁSOK	51
8.1.	A SZENTIMENTÁLIS ELEMZÉS ÉS MŰKÖDÉSE	51
8.2.	MÓDSZEREK ÉS ESZKÖZÖK A SZENTIMENTÁLIS ELEMZÉSBEN	52
8.3.	A NYELVI INFORMÁCIÓK ÁTALAKÍTÁSA	52
8.4.	A SZENTIMENTÁLIS ELEMZÉS INTEGRÁLÁSA AZ ALGORITMIKUS KERESKEDÉSI STRATÉGIÁKBA...	54
8.5.	A SENTIMENT ANALÍZIS TESZTELÉSE.....	54
8.6.	OPTIMALIZÁLÁSI LEHETŐSÉGEK	58
9.	KÖVETKEZTETÉSEK ÉS JAVASLATOK.....	63
10.	ÖSSZEFOGLALÁS	66
IRODALOMJEGYZÉK		68
MELLÉKLETEK.....		ERROR! BOOKMARK NOT DEFINED.
1. MELLÉKLET: HELYZETJELENTÉS A VÉGEKRŐL.....		ERROR! BOOKMARK NOT DEFINED.

TÁBLÁZAT- ÉS ÁBRAJEGYZÉK

1. ábra: Tőzsdei árfolyam előrejelzés (Mamdani) (Forrás: saját szerkesztés)	15
2. ábra: Alul-, illetve felülillesztés (Forrás: saját szerkesztés)	16
3. ábra: Vételi és eladási jelzések (2022-01-01 és 2023-11-09) között. (Forrás: saját szerkesztés)	17
4. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	17
5. ábra: Vételi és eladási jelzések (2023-08-01 és 2023-11-09) között. (Forrás: saját szerkesztés)	17
6. ábra: Vételi és eladási jelzések (2022-01-01 és 2023-11-09) között. (Forrás: saját szerkesztés)	17
7. ábra: A Kálmán-szűrő alkalmazása az APPLE részvény árfolyam-változásán (Forrás: saját szerkesztés)	23
8. ábra: A neurális hálózat működése a 4 technikai indikátorral (Forrás: saját szerkesztés)	45
9. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	46
10. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	46
11. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	48
12. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	49
13. ábra: Word2Vec modell (Nayak, 2019)	53
14. ábra: A program által generált szentiment értékek (Forrás: saját szerkesztés) ..	56
15. ábra: A szentiment elemzés eredménye (Forrás: saját szerkesztés)	57
16. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	57
17. ábra: A szentiment elemzés eredménye (Forrás: saját készítésű ábra)	58
18. ábra: A program által generált haszonelemzés (Forrás: saját szerkesztés)	59
19. ábra: A szentiment elemzés eredménye (Forrás: saját szerkesztés)	59
20. ábra: A program által generált haszonelemzés (Forrás: saját készítésű ábra) ...	60
21. ábra: A komplex rendszer vázlata (Forrás: saját szerkesztés)	62

1. BEVEZETÉS

A tőzsdei kereskedési környezet folyamatosan fejlődő és rendkívül dinamikus terület, ahol a másodpercek is értékesek, és az információ minősége közvetlenül befolyásolja a kereskedési eredményeket. Ebben a gyors tempójú világban az algoritmikus vagy automatizált kereskedés vált a modern kereskedés egyik alapkövévé, melynek révén a kereskedők képesek azonnal reagálni a piaci változásokra, és minimalizálni az emberi tényezőből fakadó kockázatokat. Ebben a kontextusban a gépi intelligencia (MI) és a mélytanulás technológiáinak bevezetése forradalmasította a pénzügyi döntéshozatalt, és új utakat nyitott a tőzsdei kereskedés optimalizálásához. Az utóbbi években a mesterséges intelligencia azon képessége, hogy komplex adatsorokból mintákat ismerjen fel és prognózisokat állítson elő, alapvető fontosságúvá vált a befektetési stratégiák kidolgozásában. Az MI alkalmazásával az algoritmikus kereskedés képes adaptálódni, tanulni és fejlődni a változó piaci körülmények között, ami növeli a stratégiák hatékonyságát és profitabilitását.

A szentimentális elemzés is új dimenziót nyitott, hiszen a piaci hangulat és a befektetői magatartás mélyebb megértését teszi lehetővé. Ebben a digitális korban a közösségi média platformok, hírportálok és szakmai blogok áradatnyi adatot generálnak, amelyek, ha helyesen elemezve és értelmezve vannak, értékes betekintést nyújthatnak a piaci trendekbe és a befektetői pszichológiába. Az Application Programming Interface (API) technológia segítségével történő adatgyűjtés és -feldolgozás lehetővé teszi a kereskedők számára, hogy valós időben hozzáférjenek ezekhez az információkhoz és integrálják őket kereskedési döntéseikbe.

Jelen dolgozat célja, hogy bemutassa az MI, a technikai- és a szentimentális elemzés integrációjának potenciálját a tőzsdei kereskedésben, különös tekintettel arra, hogy miként javíthatják a kereskedési algoritmusok a piaci adatok, közösségi média bejegyzések és hírek elemzésével előállított információk felhasználásával a kereskedési döntéseket. Megvizsgálom a technológia jelenlegi határait és a jövőbeli fejlesztési lehetőségeit a pénzügyi kereskedésben. A kutatás arra is törekszik, hogy feltárja azokat a legjobb gyakorlatokat és eljárásokat, amelyek maximalizálják az MI által generált előnyöket, miközben minimalizálják a vele járó kockázatokat, így hozzájárulva a tőzsdei kereskedési folyamatok és stratégiák optimalizálásához.

2. TÖRTÉNETI ÁTTEKINTÉS AZ ALGORITMIKUS KERESKEDÉSI MODELLEKRŐL

Az algoritmikus kereskedés, amelynek alapját a bonyolult matematikai modellek és informatikai megoldások képezik, a 20. század második felében vált széles körben elfogadottá. Ebben az időszakban, az 1970-es évek során, a piacelemzők és a kereskedők olyan alapvető eszközök segítségével kezdték megismerni és alkalmazni a matematikai modelleket, mint a mozgóátlagok és az oszcillátorok, ezzel próbálva megjósolni a különböző árfolyamok mozgását (Murphy, 1999).

Az 1980-as években a személyi számítógépek megjelenése és az adatfeldolgozás fejlődése radikálisan megváltoztatta a kereskedési technikákat. Ekkor jöttek létre azok a bonyolultabb stratégiák és algoritmusok, amelyek már részletesebb adatanalízisen alapultak. Ebben az évtizedben vált széles körben elterjedté a backtesting módszer is, amely lehetővé tette a kereskedők számára, hogy múltbeli adatok segítségével teszteljék stratégiáik hatékonyságát (Chaboud et al.,2014; Pardo, 2008).

Az 1990-es évekre a mesterséges intelligencia, valamint a neurális hálózatok előtérbe kerülése révén új, korszerű technológiák jelentek meg a pénzügyi piacokon. Ilyen újítások voltak például a neurális hálózatok, a genetikusan algoritmusok és a fuzzy logikai rendszerek, amelyeknek köszönhetően a kereskedők immár képesek voltak az addig rejtett, kaotikus piaci minták kiaknázására és ezekre épülő összetettebb, alkalmazkodóképes stratégiák kialakítására (Trippi & Turban, 1996; Deboeck, 1994).

Az ezredfordulóval egy időben a High-Frequency Trading (HFT) teljes mértékben új alapokra helyezte az algoritmikus kereskedést. Az internetes technológiák és az informatikai infrastruktúra fejlesztéseinek köszönhetően a döntéshozatali folyamatok gyorsasága milliszekundumokra csökkent. Ezek a gyorsasági és technológiai fejlesztések alapjaiban formálták át a pénzügyi piacok szerkezetét és működését (Aldridge, 2013).

Az elektronikus kereskedelem forradalmi fejlődése az 1960-as években vette kezdetét, amikor a kereskedési információkat számítógépes terminálokra irányították át. Az Amerikai Értékpapír- és Tőzsdei Bizottság (Securities and Exchange Commission, SEC) 1997-ben bevezetett új rendelete komoly lökést adott az elektronikus kereskedelmi rendszereknek,

különösen az Elektronikus Kommunikációs Hálózatok (Electronic Communication Networks, ECN) használatának engedélyezésével. A sötét medencék - amelyek zárt, magántulajdonú Alternatív Kereskedési Rendszereknek (Alternative Trading Systems, ATS) minősülnek - új kereskedési lehetőségeket biztosítottak a nagyobb intézményi befektetők számára. A Magas Frekvenciájú Kereskedelem (High-Frequency Trading, HFT) ma már meghatározó szereplője a globális pénzügyi piacoknak, kiemelten az amerikai és európai részvénypiacok területén (Mitchell, 1997; Schölkopf & Smola, 2002).

A 2000-es évek közepétől kezdődően az algoritmikus kereskedési modellek további fejlődésnek indultak a kvantum számítástechnika, a big data, és a gépi tanulás területén elért áttöréseknek köszönhetően. A hatalmas mennyiségű, strukturálatlan adat feldolgozásának képessége lehetővé tette a kereskedési algoritmusok számára, hogy finomítsák és bővítsék előrejelzési modelljeiket, ami elősegítette a döntéshozatali folyamatok automatizálását és a piaci ineffektivitások kihasználását (James, Witten, Hastie & Tibshirani, 2013).

Ebben az időszakban a „sötét medencék” és az ATS-ek széles körű alkalmazása lehetővé tette a kereskedők számára, hogy kevésbé átlátható módon, kevesebb piaci hatással cserélhessenek értékpapírokat. Ez a kereskedési forma jelentős hatást gyakorolt a piaci likviditásra és a kereskedési költségekre, különösen a nagy volumenű kereskedők esetében.

Az elmúlt évtizedben a mesterséges intelligencia (MI) és a mélytanulási modellek, mint az összetett rekurzív neurális hálózatok, új fejezetet nyitottak az algoritmikus kereskedésben. Ezek a modellek képesek a hagyományos statisztikai modelleknél sokkal bonyolultabb kapcsolatok és mintázatok azonosítására, beleértve a nemlineáris összefüggéseket is, amelyek eddig rejtve maradtak a kereskedők előtt. Az MI-n alapuló rendszerek képessé váltak az önkorrekcióra és az adaptív kereskedési stratégiák meghatározására real-time, változó piaci körülmények között.

Ezenfelül a blokklánc technológia és a kriptovaluták megjelenése új kereskedési eszközökké és piacokká váltak, amelyek újabb kihívások elé állították az algoritmikus kereskedési modelleket. A kriptopiacok volatilitása és 24/7 működése új stratégiák és adatelemzési módszerek kidolgozását igényelte.

Összefoglalva, az algoritmikus kereskedési modellek fejlődése folyamatosan összefonódott a technológiai innovációkkal. Mindez lehetővé tette a piaci szereplők számára, hogy maximalizálják profitjukat, minimalizálják a kockázatot, és hatékonyabban navigáljanak a pénzügyi piacokon (Narang, 2013). A jövőbeli fejlődés valószínűleg továbbra is ezen az úton halad előre, ahol a gépi tanulás, a mesterséges intelligencia és az új technológiák, mint például a kvantum számítástechnika, új lehetőségeket és stratégiákat hoznak a tőzsdei kereskedésben. Az új kor kihívásai, beleértve a kiberbiztonságot és az etikai mérlegeléseket is, új kérdéseket és szabályozási szempontokat is magukkal hoznak, amelyekkel a piaci szereplőknek és szabályozóknak egyaránt foglalkozniuk kell a fenntartható és stabil pénzügyi piacok érdekében.

3. AZ ALGORITMIKUS KERESKEDÉS ALAPELVEI ÉS MECHANIZMUSAI

3.1. AZ ALGORITMIKUS KERESKEDÉS DEFINÍCIÓJA ÉS FŐ JELLEMZŐI

Az algoritmikus kereskedés egy előre meghatározott szabályokon alapuló, számítógépes algoritmus által vezérelt kereskedési módszer (Jansen, Stefan 2020). A döntéshozatali folyamat során az algoritmusok a piaci adatokat elemezik, és kereskedési utasításokat adnak ki anélkül, hogy emberi beavatkozásra lenne szükség (Chlistalla, 2011; Johnson, B. 2010).

Fő jellemzői:

- Automatizált: Az algoritmikus kereskedés minimalizálja az emberi tényezőt.
- Gyorsaság: Az algoritmusok képesek gyorsan reagálni a piaci változásokra.
- Pontosság: Csökkenti az emberi hibák lehetőségét.

3.2. KERESKEDÉSI ALGORITMUSOK TÍPUSAI

Néhány alapvető típus:

Arbitrázs

Az arbitrázs a pénzügyi eszközök piacán fennálló árkülönbségek kihasználására irányuló kereskedési stratégia. A koncepció az, hogy bizonyos eszközök vagy eszközpárok ára időről időre eltérő lehet különböző piacokon, és ezeket az árkülönbségeket lehet használni nyereség eléréséhez azzal, hogy az alacsonyabb áron vásárolt és a magasabb áron eladott (Földes, 1996). Az arbitrázsstratégia a gyakorlatban gyakran azonnali nyereséget jelent kockázat nélkül, feltéve, hogy a tranzakciókat gyorsan és hatékonyan hajtják végre (Gatev, Goetzmann & Rouwenhorst, 2006).

Trendkövető

A trendkövető stratégia az aktuális piaci trendeket használja fel a kereskedési döntések meghozatalához. A kereskedők olyan technikai elemzéseket alkalmaznak, mint az átlagárak mozgása, az árrelációk, és a piaci momentumok azonosítása annak érdekében, hogy megállapítsák az árfolyamok várható irányát (Covel, 2009). A stratégia lényege, hogy a kereskedők a trenddel együtt vásárolnak vagy eladnak, abban a reményben, hogy a trend folytatódni fog (Chan, 2009; Chan, 2013).

Statisztikai arbitrázs

A statisztikai arbitrázs (vagy párok kereskedelme) egy olyan piaci semleges stratégia, amely két vagy több kapcsolódó pénzügyi eszköz árának statisztikai kapcsolatán alapszik. Az alapötlet az, hogy az árak közötti kapcsolat idővel állandó, és ha ez a kapcsolat eltér az átlagtól, várható, hogy visszatér az átlagos értékhez. Így a kereskedők az egyik eszközt eladják (rövidre veszik), míg a másikat megvásárolják (hosszúra veszik) az árkülönbség kihasználása érdekében (Gatev et al., 2006).

Piaci időzítés

A piaci időzítés stratégiája azt a koncepciót használja, hogy a kereskedők képesek előre jelezni a piaci ármozgásokat és a ciklusokat. Ezek az előrejelzések gyakran kombinálják a fundamentális és technikai elemzéseket annak érdekében, hogy meghatározzák, mikor kell belépni vagy kilépni a piacról. Bár ez egy nagyon vonzó stratégia, magas kockázatot hordoz, mivel a piacok gyakran viselkednek előre nem látható módon és a kereskedők döntései helytelenek lehetnek (Lo & MacKinlay, 1999), (Hendershott, T., Jones, C. M., & Menkveld, A. J. 2011).

3.3. ALGORITMUSOK KIVÁLASZTÁSA

A kereskedési algoritmus kiválasztásakor figyelembe kell venni a következőket:

Piaci környezet

Nem minden kereskedési algoritmus működik egyformán hatékonyan minden piaci környezetben. Az algoritmusok teljesítménye erősen függ a piaci feltételektől, mint például a likviditás, a volatilitás, és a résztvevők viselkedése. Egyes stratégiák, például a trendkövető algoritmusok, jól teljesíthetnek felfelé irányuló piacokon, de veszteségeket szenvedhetnek oldalirányú vagy lefelé irányuló trendek esetén (Chincarini & Kim, 2006). Ezért fontos az algoritmusok folyamatos felülvizsgálata és adaptációja a piaci körülményekhez (Sinclair, 2013).

Kockázattűrés

Az algoritmikus kereskedési stratégiák különböző kockázati profilokkal rendelkeznek és fontos, hogy ezek összhangban legyenek a kereskedő vagy a befektetési intézmény kockázattűrés szintjével. A kereskedési algoritmusoknak rugalmasnak kell lenniük a különböző kockázati tolerancia szintek kezelésére és képeseknek kell lenniük a lehetséges veszteségek minimalizálására a kereskedési paraméterek, például a stop-loss megbízások és a kockázatkezelési modellek beállításával (Lopez de Prado, 2018; Wilmott, 2007).

Likviditás és volatilitás

A likviditás és a volatilitás központi szerepet játszanak az algoritmikus kereskedési döntésekben. Az alacsony likviditás megnövelheti a költségeket és korlátozhatja a kereskedési lehetőségeket, míg a magas volatilitás növelheti a kockázatot és a potenciális nyereséget egyaránt. Az algoritmusoknak képesnek kell lenniük azonosítani és alkalmazkodni a piaci feltételek változásaihoz, kezelve a megbízások végrehajtását a piaci likviditás és a volatilitás ingadozásainak függvényében (Aldridge, 2013).

3.4. AZ ALGORITMIKUS KERESKEDEÉS KIHÍVÁSAI

A technológiai fejlődés ellenére az algoritmusok még mindig sérülékenyek a nem várt hibákra és rendellenességekre, amelyek komoly pénzügyi veszteségekhez vezethetnek. Az overfitting problémája továbbá kérdésessé teszi, hogy a stratégiák, amelyek a múltbeli adatokon jól teljesítenek, képesek-e ugyanolyan sikerrel működni a jövőben. Emellett a stratégiai degradáció is aggodalomra ad okot, mivel a piaci viselkedés változékonysága miatt a korábban sikeres kereskedési taktikák hatékonysága idővel csökkenhet. Az alábbiakban részletesen elemezem ezeket a központi kihívásokat és a lehetséges megoldásokat (Harris, 2002).

Az algoritmikus kereskedés jellemző hibái:

Technológiai hibák

Az algoritmikus kereskedésben használt technológiák, legyenek azok hardverek vagy szoftverek, nem mentesek a hibáktól. Ezek a hibák rendkívül költségesek lehetnek, mivel befo-

lyásolhatják a kereskedési műveleteket, torzíthatják a piaci árakat vagy akár komoly veszteségeket is okozhatnak. Példák lehetnek a rendszerösszeomlások, hálózati késések, rosszul megírt kód vagy technológiai kompatibilitási problémák. A kereskedési vállalkozásoknak hatékony hibaelhárítási folyamatokat kell fenntartaniuk, hogy minimalizálják ezeknek a hibáknak a hatásait.

Overfitting

Az overfitting az egyik leggyakoribb probléma az algoritmikus kereskedési modellek kifejlesztésében. Ez akkor fordul elő, amikor egy modell túlságosan "illeszkedik" a múltbeli adatokra, annyira specifikus lesz, hogy nem képes általánosítani és hatékonyan működni új, ismeretlen adatokon. A túlilleszkedés komoly veszélyt jelent, mivel jelentősen torzíthatja a modell teljesítményének értékelését és hamis biztonságérzetet adhat a stratégia hatékonyságával kapcsolatban (Bailey et al., 2014). Az overfitting elkerülése érdekében fontos a modell szigorú validációja olyan adatokon, amelyeket nem használtak a tanítás során.

Stratégiai degradáció

A stratégiai degradáció jelensége, amikor egy kereskedési stratégia hatékonysága az idő múlásával csökken. Ennek oka lehet az, hogy a piaci körülmények változnak, és a stratégia már nem alkalmazkodik jól, vagy az, hogy a stratégia „elhasználódik”, mivel egyre több szereplő kezdi el használni azt, csökkentve ezzel a stratégia egyedi előnyeit. A kereskedőknek és befektetőknek rendszeresen felül kell vizsgálniuk és módosítaniuk kell stratégiáikat, hogy fenntartsák azok relevanciáját és hatékonyságát a folyamatosan változó piaci környezetben (Huck, 2009).

4. MAGAS FREKVENCIÁS KERESKEDÉS: ELŐNYÖK, KIHÍVÁSOK, PIACI HATÁS

A magas frekvenciás kereskedés (HFT) egy olyan algoritmikus kereskedési módszer, melynek célja, hogy minimális idő alatt nagy mennyiségű tranzakciót hajtson végre, gyakran milliszekundumok vagy mikroszekundumok alatt. Ezáltal a HFT képes kihasználni a legkisebb árkülönbségeket is, amelyek rövid időszakokra jelennek meg a piacokon.

A HFT gyökerei az 1980-as évekbe nyúlnak vissza, amikor a tőzsdék elektronikussá váltak és a kereskedés automatizálásának első formái megjelentek. A technológia és az internet fejlődésével a 2000-es években a HFT rohamosan növekedett, és ma már jelentős részét képezi a globális pénzügyi tranzakcióknak (Zhang, Frank, 2010).

4.1. A HFT SZEREPE AZ ALGORITMIKUS KERESKEDÉSBEN

Az algoritmikus kereskedés egy széles kategória, mely magában foglalja azokat a módszereket, amelyek matematikai modelleken és automatizált algoritmusokon alapulnak a kereskedési döntések meghozatalához. A HFT ezen belül egy specifikus szegmens, amely az extrém gyorsaságra és reakcióidőre koncentráل.

A HFT két fő jellemzője a gyorsaság és a mennyiség. Míg egy hagyományos algoritmikus kereskedési stratégia órákig vagy akár napokig is eltarthat, egy HFT stratégia milliszekundumok alatt hoz döntéseket és hajt végre tranzakciókat. Ennek köszönhetően a HFT képes előnyt kovácsolni a piacon jelentkező pillanatnyi inhomogenitásokból és áringadozásokból.

4.2. A MAGAS FREKVENCIÁS KERESKEDÉS TECHNOLÓGIAI ALAPJAI

A magas frekvenciás kereskedés (HFT) gyorsasága és precizitása miatt kritikusan fontos, hogy a technológiai alapok megfelelőek legyenek. A hardvertől kezdve a szoftveren át a kereskedési platformokig minden elemnek tökéletesen kell működnie a hatékony és költséghatékony HFT működtetéséhez (Brogaard, Li & Yang, 2022).

4.2.1. *Hardver és infrastruktúra*

A HFT hardverkomponensei közé tartoznak az erős szerverek, melyek képesek kezelni a nagy mennyiségű adatfolyamot, valamint a gyors hálózati kapcsolatok, amelyek minimális késleltetéssel továbbítják az információkat. Sok HFT cég szerverközpontjait közvetlenül a

tőzsdék mellett helyezik el, hogy minimalizálják az adatátviteli időt (ezt „co-location”-nek nevezik) (Aldridge, 2013; Kirilenko, Kyle, Samadi & Tuzun, 2017).

4.2.2. Szoftver és algoritmusok specifikációja

A HFT szoftverei között megtalálhatók azok az algoritmusok, melyek a piaci adatok elemzésén és a kereskedési döntések meghozatalán alapulnak. Ezen algoritmusoknak képesnek kell lennie az adatok gyors feldolgozására, azaz milliszekundumok vagy akár mikroszekundumok alatt döntéseket kell hozniuk. A szoftvereknek magas rendelkezésre állásúaknak és skálázhatóknak is kell lenniük, hogy a változó piaci körülmények között is hatékonyan működjenek (Lopez de Prado, 2018).

4.2.3. Kereskedési platformok és rendszerek

Ezen platformok biztosítják az alapvető keretet a HFT számára, lehetővé téve a kereskedési stratégiák végrehajtását és az erőforrások kezelését. Ezeknek a rendszereknek a megbízhatósága kritikus fontosságú, hiszen bármilyen rendszerhiba óriási veszteségeket okozhat rövid idő alatt. Emellett a kereskedési platformoknak rendkívül gyorsnak és rendkívül alacsony késleltetési idővel kell rendelkezniük a tranzakciók során, hogy az algoritmusok teljes potenciáljukat kihasználhassák (Huck, 2009).

4.3. ELŐNYÖK ÉS ELŐSEGÍTŐ TÉNYEZŐK

4.3.1. Likviditás növelése és piaci hatékonyság

A magas frekvenciás kereskedés (HFT) növeli a piaci likviditást azzal, hogy nagy mennyiségű kereskedési rendeltetést ad le rövid idő alatt. Matematikailag a likviditást a következő képlettel lehet leírni:

$$L = Q \times F \quad (1)$$

ahol L a likviditás, Q a kereskedett mennyiség és F a kereskedés gyakorisága. Mivel a HFT nagy Q és F értékekkel rendelkezik, L is magas lesz.

4.3.2. Szűkített spreadek és költséghatékonyság

Azokban a piacokon, ahol a HFT dominál, gyakran szűkebbek a bid-ask spreadek, ami a kereskedők számára kevesebb költséget és jobb árakat jelent. A spread a következőképpen számítható:

$$\text{Spread} = \text{Ask} - \text{Bid} \quad (2)$$

A magasabb likviditás és a gyors reakcióidő csökkenti a spreadeket és növeli a kereskedelmi hatékonyságot.

4.3.3. Alacsonyabb tranzakciós költségek

A magas frekvenciás kereskedés képes arra, hogy gyorsan reagáljon a piaci változásokra és minimalizálja a tranzakciós költségeket. A tranzakciós költségek számíthatók a következő képlettel:

$$TC = P \times N \times C \quad (3)$$

Ahol TC a tranzakciós költségek, P az ár, N a kereskedett mennyiség és C a tranzakciós díj egységenként. A HFT alacsony C értékkel rendelkezik, így TC is alacsonyabb lesz.

4.4. KIHÍVÁSOK ÉS KOCKÁZATOK

4.4.1. Technológiai kockázatok és rendszerhibák

A magas frekvenciás kereskedés függ a komplex technológiai rendszerektől. Mivel a kereskedelmi döntéseket milliszekundumok alatt kell meghozni, a rendszerhibák vagy késések súlyos kockázatot jelenthetnek a kereskedők számára. Ezek a hibák nemcsak a kereskedési veszteségekhez vezethetnek, hanem károsíthatják a piaci integritást is, ahogyan egy technológiai hiba keresztülvihet egy nem kívánt tranzakciót vagy torzíthatja a piaci információkat (Aldridge, 2013).

4.4.2. Piaci manipuláció és „flash crash”-ek

Néhány kritikus úgy véli, hogy a magas frekvenciás kereskedés elősegítheti a piaci manipulációt, például a „quote stuffing” vagy az „order layering” módszereivel. Ezenkívül a HFT gyors természete miatt hozzájárulhat olyan rendkívüli piaci eseményekhez, mint a

„flash crash”, amikor a piaci értékek pillanatok alatt drasztikusan csökkennek és gyorsan visszaállnak, anélkül hogy konkrét gazdasági okuk lenne (Lopez de Prado, 2018).

4.4.3. *Piaci instabilitás és „toxic” likviditás*

Bár a HFT növeli a likviditást, ez nem mindig stabil. Néhány piaci esemény során a magas frekvenciás kereskedők visszavonulhatnak a piacról, hirtelen csökkentve a rendelkezésre álló likviditást, ami „toxic” vagy mérgező likviditásnak tekinthető. Ezen túlmenően, mivel a HFT stratégiák gyakran reagálnak egymásra, ez fokozhatja a piaci volatilitást és a nem kívánt események valószínűségét (Perez, Edgar 2011).

4.5. A PIACI HATÁSOK ÉS SZABÁLYOZÁSI TÉMAKÖRÖK

4.5.1. *A HFT hatása a hagyományos kereskedőkre és brókerekre*

A HFT-nek vannak előnyei, mint például a likviditás növelése és a spreadek szűkítése, egyes kritikusok szerint hátrányosan befolyásolja a hagyományos kereskedőket és brókereket. A HFT hatása a hagyományos piaci szereplőkre a következőképpen értékelhető:

- **Likviditás:** A HFT kereskedők nagy számú ügyletet hajtanak végre rövid idő alatt, ami növelheti a piaci likviditást. Ennek eredményeként a hagyományos kereskedőknek könnyebb lehet vételi vagy eladási megbízásokat teljesíteni.
- **Spread:** A HFT tevékenysége gyakran szűkíti a vételi és eladási árak közötti különbséget (spread). Ez csökkenti a kereskedési költségeket, ami előnyös lehet a hagyományos kereskedők számára.
- **Piaci volatilitás:** Egyes kritikusok szerint a HFT növelheti a piaci volatilitást, különösen stresszes piaci körülmények között. Ez a fokozott volatilitás kockázatosabbá teheti a kereskedést, különösen a hosszú távú befektetők és a hagyományos kereskedők számára.
- **„Front running” és információs előny:** A HFT kereskedők képesek lehetnek arra, hogy gyorsaságuk révén a hagyományos kereskedők megbízásait megelőzve ügyleteket bonyolítsanak, amit egyesek digitális „front running”-nak tekintenek. Emellett a HFT cégek gyakran rendelkeznek fejlett technológiai infrastruktúrával, ami információs előnyt biztosíthat számukra.

- **Infrastruktúra és technológiai fejlesztések:** A HFT kereskedők jelenléte arra kényszeríti a tőzsdéket és a brókercégeket, hogy fejlesszék technológiai infrastruktúrájukat, hogy versenyképesek maradhassanak. Ez növelheti a költségeket, ami áthárulhat a hagyományos kereskedőkre és brókerekre.
- **Stratégiai változások:** A HFT jelenléte miatt a hagyományos kereskedőknek és befektetési menedzsereknek meg kell változtatniuk kereskedési stratégiáikat. Például sokan elkezdtek használni az algoritmikus kereskedési stratégiákat, hogy elkerüljék a HFT kereskedők „szkennerét”.

4.5.2. Nemzetközi eltérések a HFT szabályozás terén

A HFT szabályozása jelentősen eltér a különböző országokban és régiókban. A következőkben néhány globális különbséget ismertetek:

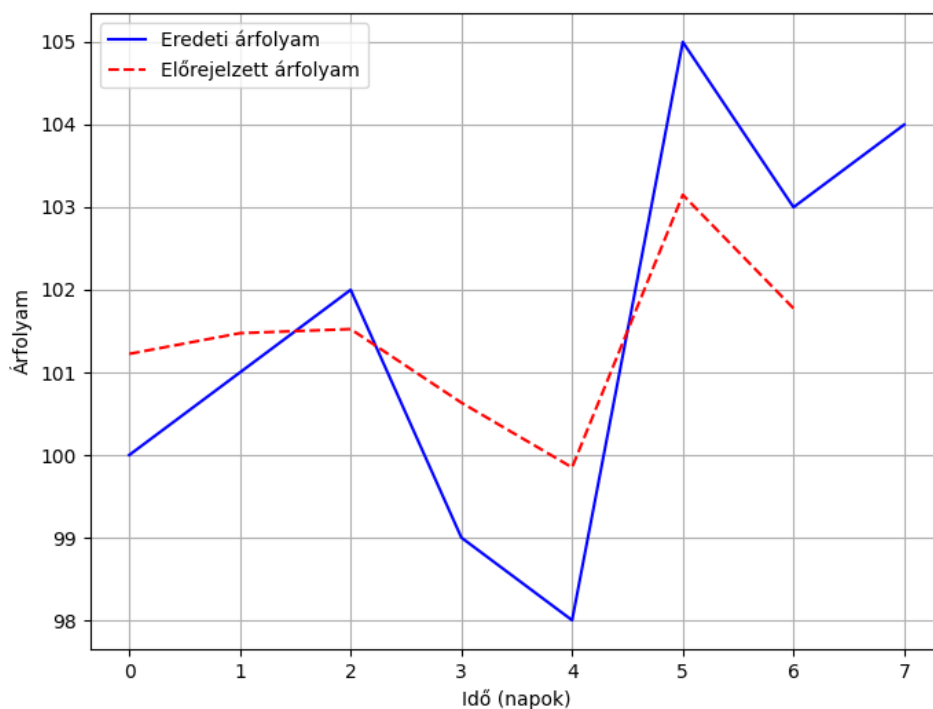
1. **Amerikai Egyesült Államok:** Az USA-ban a HFT tevékenységet a Securities and Exchange Commission (SEC) és a Commodity Futures Trading Commission (CFTC) szabályozza. Az USA-ban bevezetett szabályozások között szerepel az úgynevezett „Market Access Rule”, ami megköveteli a brókercégektől, hogy alkalmazzanak kockázatkezelési kontrollokat és felügyeleti gyakorlatokat, valamint a Dodd-Frank Act, ami befolyásolja a származékos ügyletek piacát.
2. **Európa:** Az Európai Unióban a HFT tevékenység szabályozását nagyrészt a MIFID II (Markets in Financial Instruments Directive II) rendelkezései határozzák meg. A MiFID II előírja, hogy a HFT kereskedéssel foglalkozó vállalatoknak engedélyt kell szerezniük, továbbá bevezetett bizonyos piaci integritást biztosító intézkedéseket, mint például a rendelési, levonási arányra vonatkozó korlátozások.
3. **Ázsia:** Ázsiában a HFT szabályozása országonként változó. Például Japánban a Tokyo Stock Exchange szigorú szabályokat vezetett be a HFT cégek számára, míg Kínában a kormány korábban korlátozásokat vezetett be a HFT tevékenységekre, de később ezeket a korlátozásokat enyhítették.
4. **Egyéb piacok:** Sok feltörekvő piacon a HFT még viszonylag új jelenség és a szabályozó hatóságok még alakítják ki a megfelelő szabályozási keretet. Azonban a HFT

növekedésével ezeken a piacokon is egyre inkább előtérbe kerül a szabályozási kérdés.

A HFT szabályozása világszerte változó és a különböző országok különböző megközelítéseket alkalmaznak. A HFT hatásának értékelése, valamint a technológiai fejlődés és a piaci változások követése érdekében a szabályozók világszerte folyamatosan felülvizsgálják és módosítják a szabályozási kereteket.

5. A GÉPI TANULÁS ALKALMAZÁSA AZ ÁRFOLYAMOK ELŐREJELZÉSÉBEN

A gépi tanulás területén az egyik kulcsfontosságú kérdés, hogy hogyan optimalizáljuk az algoritmusokat egy adott feladatra. Ez a kihívás számos bonyolult és mélyreható aspektust tartalmaz. Ebben a részben alaposabban elemzésre kerül ez a probléma és az, hogy ezek milyen hatással vannak az algoritmusok kiválasztására és használatára. A gépi tanulás célja, hogy azonosítsa azokat a mintákat a tanulási adatok között, amelyek fontosak az általánosítás szempontjából (Mitchell, 1997). Az általánosító képesség kulcsfontosságú, hiszen az algoritmusoknak nem csak a tanulási adatokra kell jól működniük, hanem az újonnan érkező, eddig ismeretlen adatokra is (Hastie, Friedman 2011). Az 1. ábrán egy tőzsdei árfolyam előrejelzés látható.

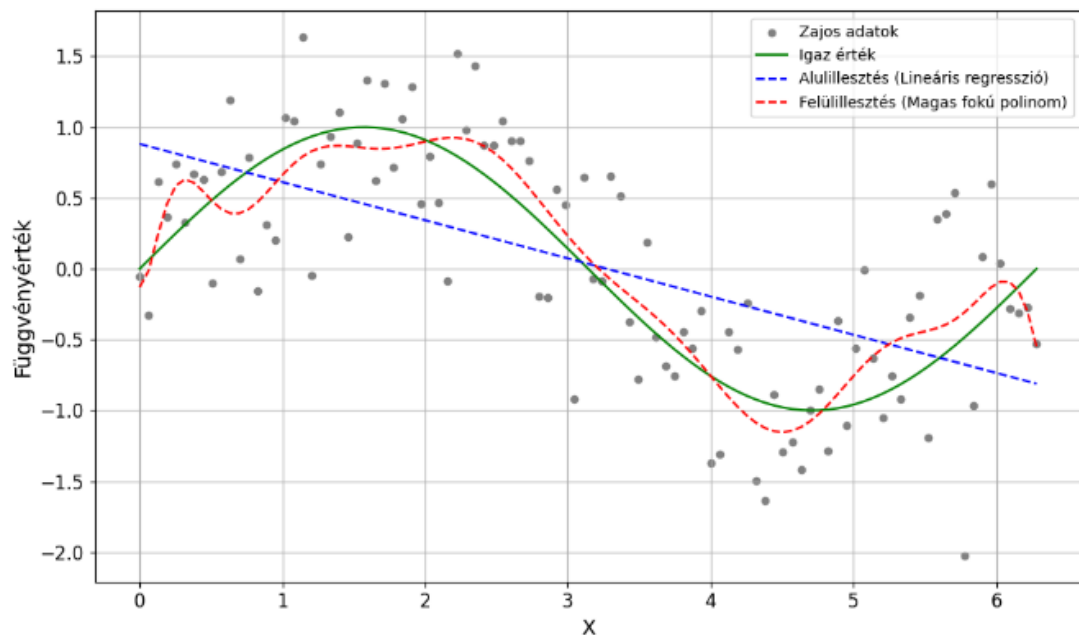


1. ábra Tőzsdei árfolyam előrejelzés (Mamdani) (Forrás: saját szerkesztés)

Számtalan mintázat létezik, és a rendelkezésünkre álló adatok csupán az összes potenciális minta töredékét tükrözik, ami megnehezíti az ideális minta kiválasztását. Az a végtelen mennyiségű függvény, amely képes bemeneti adatokból kimeneteket előállítani, jelentős kihívást és korlátozást jelent ebben a kontextusban.

5.1. HIPOTÉZISTÉR KORLÁTOZÁSA

Az algoritmusok tanulási képessége korlátozott és ezt a képességet az általuk képviselt hipotézistér definiálja, mely magába foglalja azokat a függvényeket, amelyeket az algoritmus képes modellezni (James et al., 2013). Ha ez a tér túl szűk, akkor az algoritmus nem lesz képes minden potenciális mintát felismerni. Ellenben, ha a tér túl tágas, fennáll a túlilleszkedés kockázata, ami azt jelenti, hogy az algoritmus túlzott mértékben igazodik a tanítóadatokhoz, és nem képes jól általánosítani az új információkra (2. ábra). A feladat tehát abban áll, hogy megtaláljuk azt az egyensúlyt az algoritmus és annak hipotézisterében, amely a lehető legjobban illeszkedik az adatokhoz, ugyanakkor elég flexibilis az általánosítás érdekében. Ez az egyensúly kritikus szempont az algoritmusok kiválasztásában és használatában (Bengio, 2013).

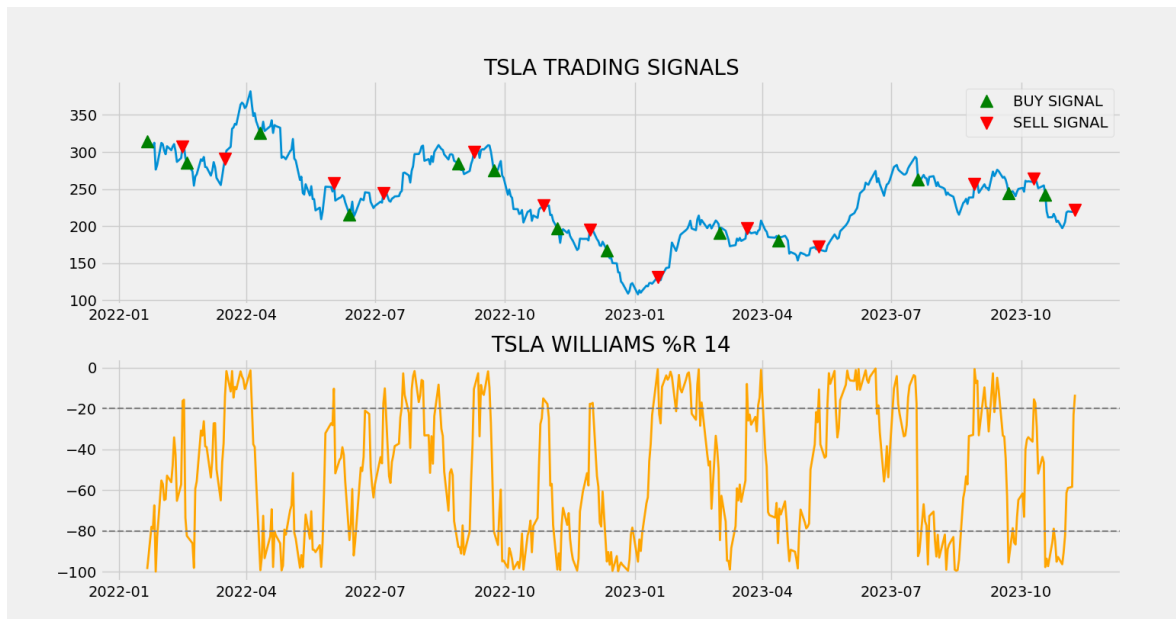


2. ábra Alul-, illetve felülillesztés (Forrás: saját szerkesztés)

Összességében elmondható, hogy az algoritmusoknak a feladathoz való igazítása bonyolult, de nélkülözhetetlen lépés a gépi tanulás folyamatában. Az általánosítás és túlilleszkedés közötti ideális egyensúly megtalálása elengedhetetlen a hatékony modellalkotáshoz.

A következő példában a TESLA részvény alapján mutatom be a hipotézistér megválasztásának jelentőségét:

A Williams%R indikátor alapján generáltam vételi és eladási jelzéseket, valamint profitelemzést végeztem. Az 3. ábrán a hipotézistér 2022-01-01 és 2023-11-09 közötti időszakra esik.

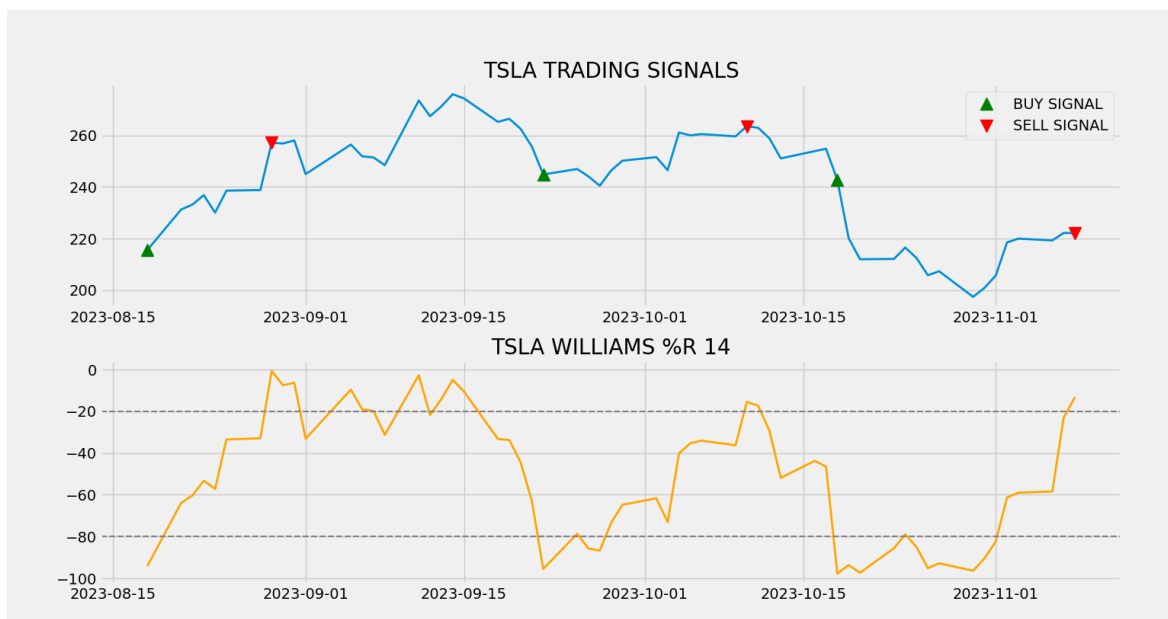


3. ábra Vételi és eladási jelzések (2022-01-01 és 2023-11-09) között. (Forrás: saját szerkesztés)

Profit gained from the W%R strategy by investing \$100k in TSLA : -54199.5
Profit percentage of the W%R strategy : -55%
Benchmark profit by investing \$100k : -9213.48
Benchmark Profit percentage : -10%
W%R Strategy profit is -45% higher than the Benchmark Profit

4. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

Az 5. ábrán a hipotézistér 2023-08-01 és 2023-11-09 közötti időszakra esik.



5. ábra Vételi és eladási jelzések (2023-08-01 és 2023-11-09) között. (Forrás: saját szerkesztés)

Profit gained from the W%R strategy by investing \$100k in TSLA : 18973.35

Profit percentage of the W%R strategy : 18%

Benchmark profit by investing \$100k : -4373.05

Benchmark Profit percentage : -5%

W%R Strategy profit is 23% higher than the Benchmark Profit

6. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

Az előző példák alapján jól látható, hogy mennyire lényeges a megfelelő hipotézistér kiválasztása, amikor egy kereskedési stratégia hatékonyságát értékeljük. Ebben az esetben a Williams%R indikátor alkalmazásával a TESLA részvényekre vonatkozó két különböző időszakban számított profitot hasonlítjuk össze egy benchmark-hoz, ami lehet például az S&P 500 index vagy valamilyen más, releváns piaci mutató.

Az első példában:

Az időszak 2022-01-01 és 2023-11-09 között zajlott, és a Williams%R stratégia jelentős, -55%-os veszteséget termelt, míg a benchmark csak -10%-os veszteséggel zárt. Ez azt mutatja, hogy ebben az időszakban a Williams%R stratégia nem bizonyult hatékonynak, és jelentősen alulteljesítette a piaci átlagot.

A második példában:

Az időszak rövidebb, 2023-08-01-től 2023-11-09-ig tart, itt viszont a Williams %R stratégia 18%-os profitot hozott, míg a benchmark -5%-os veszteséggel zárt. Ebben az esetben a Williams %R stratégia sokkal jobban teljesített a piaci átlagnál.

A fenti két példa alapján megállapítható, hogy a kereskedési stratégiák teljesítménye időszakválasztás függvényében jelentősen változhat. Egy adott stratégia, ami egyik időszakban jól működik, lehet, hogy egy másik időszakban rossz eredményeket hoz. Ezért fontos, hogy a kereskedők ne csak egy rövid időtávon vizsgálják a stratégia hatékonyságát, hanem hosszabb távon és különböző piaci körülmények között is.

5.2. FELÜGYELT TANULÁS

A felügyelt tanulás (supervised learning) azon gépi tanulási technikák közé tartozik, amelyek széles körben használatosak. Ebben a megközelítésben a tanuló algoritmus egy adott bemeneti adatsor alapján igyekszik előrejelzést tenni a kimeneti változó értékére (Bishop, 2006). Ha x_i jelöli az i -edik bemeneti mintát és y_i a hozzá tartozó kimeneti értéket, akkor az algoritmus célja egy f függvény megtanulása, melyre érvényes, hogy:

$$y_i = f(x_i) \quad (4)$$

Az ilyen módon tanított modellekben a kimeneti változót gyakran címkének (label), célnak (target) vagy eredménynek (output) is nevezik. A felügyelt tanulási probléma megoldásakor cél egy olyan modell létrehozása, amely a bemenet (X) és kimenet (y) közötti kapcsolatot jól közelíti. Matematikailag ezt a következőképpen fejezhetjük ki:

$$y = f(X) \quad (5)$$

Ahol X egy adathalmazt jelöl, melynek sorai az egyes bemeneti mintákat, oszlopai pedig a jellemzőket reprezentálják.

A felügyelt tanulás keretében nem elegendő csak az adathalmazon jól teljesítő modelleket alkotni, hanem az is elengedhetetlen, hogy ezek a modellek a későbbiekben, új adatokra is helyesen működjenek. A túlságosan bonyolult modellek gyakran esnek a túlilleszkedés csapdájába, mely során még az adatok apró zajaival is alkalmazkodnak, ami gyenge általánosítási képességet eredményez. Ugyanakkor az egyszerű modellek nem mindig képesek az

adatok mögött húzódó komplex összefüggéseket megragadni. Ezt a problémát a torzítás-variancia egyensúly (bias-variance trade-off) néven ismerjük (James et al.,2013):

$$\text{Error} = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error} \quad (6)$$

Ahol a „Hiba” a modell általánosítási hibáját, a „Torzítás” a modell előítéleteit, a „Variancia” pedig a modell variabilitását jelöli. Az "Irredukálható hiba" a tanító adathalmaz zajából származik, amelyet nem lehet elkerülni.

5.3. FELÜGYELET NÉLKÜLI TANULÁS

A felügyelet nélküli tanulás módszereiben nem az eredmények előrejelzése vagy változók közti összefüggések keresése a fő cél, hanem az adatok mögöttes szerkezetének felfedezése. Mivel csak a bemeneti adatokkal foglalkozunk és nincsenek hozzárendelt címkék vagy kimenetek, ezeknek az algoritmusoknak maguktól kell azonosítaniuk és újraértelmezniük az adatokban rejlő információkat. A sikeresség gyakran nem direkt módon mérhető, hanem abból a szempontból értékelhető, hogy az így kapott információk mennyire segítenek egy másik, későbbi feladat megoldásában.

A klaszterezés, ahol az adatokat azok hasonlósága alapján csoportosítjuk, és a dimenziócsökkentés, amely során az adatokat egy kevesebb dimenziós térben ábrázoljuk úgy, hogy közben a legfontosabb információt megtartsuk, a felügyelet nélküli tanulás két kiemelkedő alkalmazási területe. Az ilyen módszerekkel nyert eredmények gyakran hasznosak olyan feladatokban, mint a felügyelt tanulás vagy az adatok grafikus ábrázolása.

$$C=f(X) \quad (7)$$

ahol C a klaszter címkék halmaza, f a klaszterező algoritmus és X a bemeneti jellemzők mátrixa.

Példa: K-Means, Hierarchikus klaszterezés.

Dimenziócsökkentés: Egy technika, amely a magas dimenziójú adathalmazokat alacsonyabb dimenziójú térbe képes transzformálni úgy, hogy közben az eredeti adathalmaz lényeges információtartalmát megőrzi.

$$Y=g(X) \quad (8)$$

ahol Y az alacsony dimenziójú reprezentáció, g a dimenziócsökkentő függvény és X az eredeti bemeneti jellemzők.

Példa: PCA (Principal Component Analysis), t-SNE.

Ezen feladatok sajátossága, hogy olyan modelleket kell kialakítani, ahol nincs konkrét kimeneti változó (címke) a teljesítmény mérésére. Ezért a felügyelet nélküli tanulás eredményeit gyakran kombinálják más megközelítésekkel, mint például a felügyelt tanulással, vagy emberi értékeléseket alkalmaznak az értékelésükhöz.

5.4. A TÚLILLESZTÉS CSÖKKENTÉSE KÁLMÁN-SZŰRŐVEL

A jelfeldolgozásban az „adatokban előforduló zaj” kifejezés alatt a nem kívánt vagy irreleváns információkat értjük, amelyek eltorzíthatják vagy elnyomhatják a valódi jelzést. Ezen a területen a fő kihívás a zajos adatokból származó tiszta és pontos információ kinyerése. A tőzsdei kereskedés kontextusában a „zaj” a rövid távú, esetlegesen véletlenszerű áringadozásokra utal, amelyek eltérhetnek a vállalatok vagy a gazdaság alapvető értékeitől. A tőzsdei árfolyamok mögötti zajok megértése és értelmezése esszenciális a kereskedők számára, hogy jobban megértsék a piac dinamikáját és optimalizálják döntéseiket.

A tőzsdei árfolyamok zajai a következők közé tartozhatnak:

Likviditási zaj: Kisebb piacokon vagy alacsony likviditású értékpapíroknál az árak nagyobb volatilitást mutathatnak, mivel kevés ügylet képes nagyobb árváltozásokat generálni.

Hírek és események: Egy-egy hír vagy esemény, akár igaz, akár hamis, hirtelen és drámai árváltozásokat okozhat. Ezek az árfolyam-mozgások nem mindig tükrözik a vállalat valódi értékét vagy a piaci alapokat.

Algoritmikus és nagyfrekvenciás kereskedés: A gyors kereskedési algoritmusok gyakran okoznak rövid távú áringadozásokat, amelyek nem az alapvető piaci tényezőkön alapulnak.

Spekuláció: Sok kereskedő rövid távú profitért kereskedik, nem a hosszú távú értékelésen alapulva. Ez spekulatív „buborékok” és „összeomlások” kialakulásához vezethet.

Pszichológiai tényezők: Az emberi pszichológia, mint például a pánik vagy a kapzsiság, gyakran irracionális piaci mozgásokhoz vezet.

Technikai analízis: Sok kereskedő technikai indikátorokon és mintázatokon alapulva dönt, ami árfolyam-mozgásokat generálhat, függetlenül az alapvető piaci értékektől.

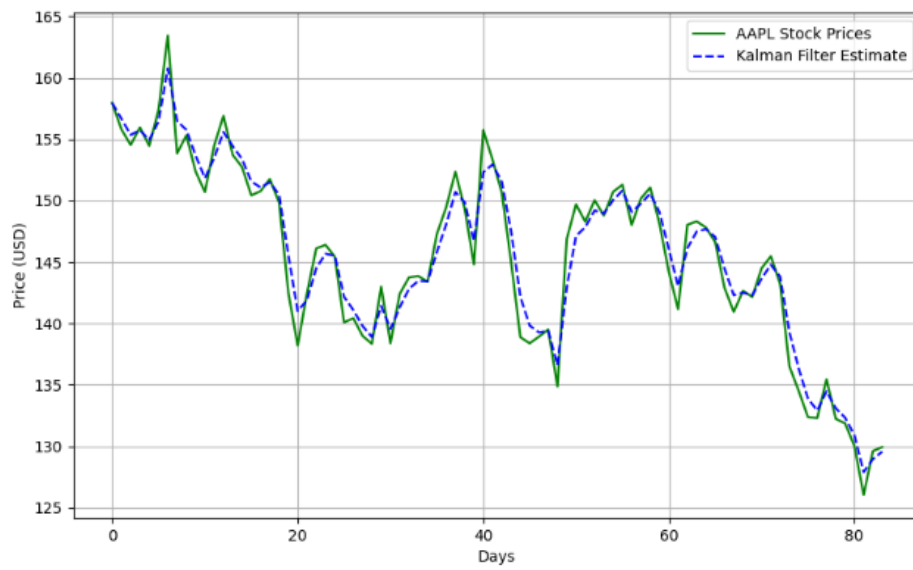
Külső sokkok: Egy váratlan makrogazdasági esemény, természeti katasztrófa vagy politikai válság hirtelen piaci reakciókat válthat ki.

Piaci manipuláció: Néhány nagyobb piaci szereplő képes lehet manipulálni az árakat rövid távú profit érdekében.

A Kálmán-szűrő az 1960-as évek óta rendkívül népszerű eszközzé vált a zajos adatok elemzésében és a valódi jel kiszűrésében. Ez az algoritmus hatékonyan képes felderíteni és kiszűrni a zajokat a mérésekben, és pontos becsléseket ad az állapotokról. Ennek köszönhetően széles körben alkalmazzák olyan különböző területeken, mint a számítógépes látás, repülési navigáció és robotika.

A Kálmán-szűrő lényege, hogy dinamikus lineáris modelleket használ az állapotok becslésére, és ezt szekvenciálisan, az újonnan érkező adatok alapján teszi. Bayes-féle statisztikai módszereket használva valószínűségi modelleként működik, amely a rejtett állapotok és a megfigyelések közötti összefüggéseket elemzi. Ugyanakkor, hogy a Kálmán-szűrő hatékony legyen, bizonyos feltételeknek teljesülniük kell. A rendszernek lineárisnak kell lennie, a rejtett állapotoknak Markov-lánc jellegűeknek kell lenniük, és a zajnak Gauss-eloszlású, nem korrelált zajnak kell lennie. A Kálmán-szűrő implementálásához Pythonban használható a *pykalman* könyvtár, amely hatékony eszköz az állapotok és modellparaméterek becslésére. A szűrő egyik fő előnye, hogy folyamatosan frissíti a becsléseit az új megfigyelésekkel, nem támaszkodva egy fix időintervallumra (Barber, 2012).

A Kálmán-szűrő egy gyakori alkalmazása a tőzsdei árfolyamok simítása, ahol a rövid távú zajok és ingadozások kiszűrésével kaphatunk egy stabilabb, az alapvető trendeket követő árfolyamgörbét. Az APPLE részvényárfolyam simításának példája jól mutatja ezt az alkalmazást (7. ábra).



7. ábra A Kálmán-szűrő alkalmazása az APPLE részvény árfolyam-változásán (Forrás: saját szerkesztés)

6. AZ ALGORITMIKUS KERESKEDÉSI PLATFORMOK

A kereskedési platformok kulcsfontosságú szerepet töltenek be a modern pénzügyi piacokon, hiszen azokon keresztül lépnek kapcsolatba a befektetők és a piacok. Az elmúlt évtizedekben a technológia rohamos fejlődésének köszönhetően a kereskedési platformok is hatalmas átalakuláson mentek keresztül. Mára már számos olyan digitális platform áll rendelkezésre, amelyek lehetővé teszik a kereskedők számára, hogy gyorsan, hatékonyan és gyakran alacsony költségek mellett végezdhessék el a kereskedési műveleteiket. Ezek a platformok többnyire számos eszközt és funkciót kínálnak a felhasználóknak, beleértve a technikai és alapvető elemzési eszközöket, automatizált kereskedési stratégiákat, valamint a közösségi kereskedést és a hírekhez való hozzáférést. Az alábbiakban bemutatok néhány népszerű kereskedési platformot, amelyek széles körben használatosak a kereskedők és befektetők között.

MetaTrader 4 (MT4): Az MT5 elődje, amely a Forex kereskedésben vált rendkívül népszerűvé. Egyszerűsége és a kereskedési robotok (Expert Advisors) támogatása miatt sok bróker és kereskedő továbbra is előnyben részesíti.

MetaTrader 5 (MT5): Az MT4 utódjaként megalkotott platform, amely nem csak a Forex, hanem a részvények, áruk, és indexek kereskedésére is kiterjed. Az MT5 számos továbbfejlesztést és új funkciót tartalmaz az elődjéhez képest, beleértve egy fejlettebb grafikus felületet, több idősíkot, további technikai indikátorokat, és a gazdasági naptár integrációját. Az MT5 a kereskedési robotok (Expert Advisors) és az automatizált kereskedési stratégiák támogatása mellett több piaci eszköz és elemzési lehetőség kínálatával igyekszik megfelelni a mai kereskedők igényeinek.

cTrader: Egy másik népszerű kereskedési platform, amelyet kifejezetten Forex és CFD kereskedésre terveztek. A cTrader ismert a felhasználóbarát felületéről és algoritmikus kereskedési lehetőségeiről.

NinjaTrader: Ez a platform különösen népszerű a napkereskedők és a futures kereskedők körében. Támogatja az automatizált kereskedést, és széleskörű technikai elemzési eszközöket kínál.

Interactive Brokers (IBKR) Trader Workstation (TWS): Az Interactive Brokers saját kereskedési platformja, amely hozzáférést biztosít számos piachoz világszerte. Komplex és sokoldalú eszköztárát gyakran haladó és professzionális kereskedők használják.

Thinkorswim: A TD Ameritrade által kifejlesztett platform, amely a részvény-, opció-, és futures kereskedők számára kínál számos eszközt és kutatási lehetőséget.

TradingView: Egy webalapú platform, amely egyszerűen használható és közösségi funkciókat is kínál. A felhasználók egymással megoszthatják a kereskedési ötleteiket és a technikai elemzéseiket.

eToro: Egy szociális kereskedési platform, amely lehetővé teszi a felhasználók számára, hogy más kereskedők befektetéseit "másolják", ezzel is megkönnyítve a kereskedést kezdők számára.

IQ Option: Egy online kereskedési platform, amely egyszerű és intuitív felülettel rendelkezik, és lehetővé teszi a felhasználók számára, hogy különböző eszközökkel, például részvényekkel, kriptovalutákkal és bináris opciókkal kereskedjenek.

Mindegyik platformnak megvannak a maga előnyei és hátrányai, valamint eltérő funkciókat és eszközöket kínálnak különböző típusú kereskedők számára.

6.1. A METATRADER 5 (MT5) RENDSZER HASZNÁLATA AZ ALGORITMIKUS KERESKEDÉSBEN

Az MetaTrader 5 (röviden MT5) egy népszerű kereskedési platform, amelyet kifejezetten a pénzügyi piacokon való online kereskedésre terveztek. Az MT5-öt a MetaQuotes Software Corp. fejlesztette ki, és a következő főbb jellemzőkkel rendelkezik:

Támogatott piacok: Az MT5 nem csak a Forex piacokon, hanem részvények, árutőzsdei termékek, indexek és kriptovaluták kereskedésére is alkalmas.

Technikai elemzés: A platform széleskörű eszközöket kínál a piaci trendek elemzéséhez, beleértve az indikátorok, oszcillátorok és grafikonok használatát. A felhasználók testreszabhatják a kereskedési chartokat, és hozzáadhatnak különféle elemzési eszközöket.

Algoritmikus kereskedés: Az MT5 lehetővé teszi a felhasználók számára, hogy automatizált kereskedési rendszereket, úgynevezett Expert Advisors (EA) alkalmazzanak. Az EA-k MQL5 (MetaQuotes Language 5) programozási nyelven íródhatnak, ami a platform saját scriptnyelve.

Gazdasági naptár: Az MT5 integrált gazdasági naptárral rendelkezik, amelyben a felhasználók tájékozódhatnak a globális gazdasági eseményekről és hírekről, amelyek befolyásolhatják a piacokat.

Kereskedési rendszerek: Az MT5 támogatja a nettó és a bruttó pozíciórendszer kereskedési módszereit.

Mobil kereskedés: Az MT5 mobil verziói is elérhetők iOS és Android eszközökön, lehetővé téve a kereskedők számára, hogy útközben is nyomon követhessék a piacokat és kezelhessék a pozícióikat.

Demo számlák: A felhasználók a valódi pénz kockáztatása nélkül kipróbálhatják a kereskedési stratégiáikat demo számlákon.

Többnyelvű támogatás: Az MT5 támogatja a különböző nyelveken történő használatot, így a kereskedők szerte a világon használhatják a platformot a saját anyanyelvükön.

Ezek a jellemzők teszik az MT5-öt egy sokoldalú és erőteljes kereskedési platformmá, amely minden szintű kereskedő számára hasznos lehet.

6.1.1. Az MQL5 programozási nyelv

Az MQL5, vagyis a Meta Quotes Language 5, a MT5 kereskedési platformhoz kifejlesztett speciális programozási nyelv. Az MQL5 objektumorientált jellege és a C++ programozási

nyelvvél való szoros kapcsolata lehetővé teszi a kereskedők számára, hogy saját automatizált kereskedési rendszereket, más néven Expert Advisors (EA) alkotásokat, technikai indikátorokat és szkripteket hozzanak létre. Az MQL5 segítségével létrehozott kereskedési robotok képesek önállóan, emberi beavatkozás nélkül nyitni és bezárni pozíciókat a piacon, így jelentős mértékben növelve a kereskedési folyamatok hatékonyságát.

Az MQL5-nek köszönhetően a kereskedők nemcsak személyre szabott stratégiákat alkalmazhatnak, hanem kiterjedt tesztelési és optimalizálási lehetőségekkel is rendelkeznek. Az MQL5-ben írt stratégiák történelmi adatokon való tesztelése gyors és pontos, így a kereskedők a lehető legjobban felkészülhetnek a valós piaci körülmények közötti kereskedésre.

Továbbá, az MQL5.com közösségi portál révén a fejlesztők és kereskedők egyaránt megoszthatják saját kódjaikat, indikátorjaikat és kereskedési robotjaikat, vagy akár megvásárolhatnak mások által kifejlesztett eszközöket és stratégiákat. Emellett a nyelv magas szintű integrációja a MetaTrader 5 platformmal lehetővé teszi a valós idejű adatáramlás és a pénzügyi eszközökkel kapcsolatos információk feldolgozását is.

Az MQL5 egy rendkívül erőteljes és rugalmas eszköz a kereskedők számára, lehetővé téve számukra, hogy maximalizálják az algoritmikus kereskedés előnyeit, miközben személyre szabott és automatizált módon közelíthetik meg a piaci lehetőségeket.

6.1.2. Az MQL5 beépített fuzzy könyvtára

A „Fuzzy” egy olyan könyvtár, amely lehetővé teszi a fuzzy logika, vagyis a homályos logika alkalmazását. A fuzzy logika egyfajta kibővítése a klasszikus, arisztotelészi logikának, amelyben az igazság értékeit nyelvi változókként kezeli. A fuzzy logika saját műveletekkel rendelkezik, amelyek homályos halmazokon definiáltak, és hasonlóan működnek, mint a klasszikus halmazműveletek, de számításuk nehezebb. A homályos halmazok összetétele is homályos halmazt eredményez.

A fuzzy logika alapelvei a valóság tükrözésének maximális pontossága és a szubjektivitás nagy mértéke, ami jelentős számítási hibákhoz vezethet. Egy fuzzy modell olyan matematikai modell, amely fuzzy logikán alapuló számításokat végez. Ezek a modellek akkor alkalmazhatóak, amikor a vizsgált terület nehezen formalizálható és pontos matematikai leírása bonyolult vagy nem ismert. A modellek teljesítménye (hibáik száma) az Expert Advisor beállításaitól függ. A legjobb eredmények elérése érdekében a modellt részletesen kell elkészíteni és nagy méretű adathalmazon gépi tanulással finomítani.

A fuzzy modell felépítése három lépésből áll:

1. A modell bemeneti és kimeneti jellemzőinek meghatározása.
2. Tudásbázis, azaz szabálybázis létrehozása.
3. Egy következtetési módszer (pl.: Mamdani vagy Sugeno) kiválasztása.

Az első lépés alapvetően határozza meg a modell további működését. A tudásbázis olyan „ha, akkor” típusú homályos szabályok gyűjteménye, amelyek definiálják a bemenetek és a kimenetek közötti kapcsolatokat. A szabályok száma nem korlátozott és az Expert Advisor dönti el. A szabályok általános formája: „Ha szabályfeltétel, akkor szabálykövetkeztetés”. Minden szabálynak súlya van, ami a szabály fontosságát jelzi a modellben, a súlyok értéke $[0, 1]$ intervallumban van meghatározva.

A szabályok lehetnek:

- egyszerűek – egyetlen homályos változót tartalmaznak;
- összetettek – több homályos változót tartalmaznak.

A tudásbázis alapján választják ki a modellhez a homályos következtetési rendszert. A homályos következtetés a bemeneti értékek alapján egy homályos halmaz formájában történő következtetés levonása a tudásbázis és a homályos műveletek segítségével. A két legfontosabb homályos következtetési típus a Mamdani és a Sugeno.

Mamdani módszer:

A Mamdani módszer egy népszerű és széles körben használt fuzzy következtetési technika, amelyet 1975-ben Lotfi Zadeh és Ebrahim H. Mamdani fejlesztett ki. Az eljárás lényege, hogy homályos szabályok segítségével modellezi az emberi döntéshozatali folyamatokat. A Mamdani következtetés lépései a következők:

Feltételértékelés: Először minden egyes szabály feltételeit értékeljük ki az adott bemeneti értékek alapján. Ebben a lépésben a homályos halmazokra számított tagsági függvények segítségével meghatározzuk, mennyire teljesülnek a szabályok.

Következtetés: Miután minden szabály feltételét kiértékeljük, a következő lépés a szabályok következményeinek számítása. Itt az egyes szabályok súlya és a feltételértékelés eredménye alapján módosítjuk a következmények homályos halmazait.

Összegzés: A következő lépésben az összes szabály következményeit aggregáljuk egyetlen homályos halmazzá. Ez általában az egyes következmények maximumának vétele lesz az adott bemeneti értékekre.

Defuzzifikáció: Az utolsó lépésben a kapott homályos halmazt egy konkrét, „éles” értéké alakítjuk, amely legjobban képviseli a homályos halmaz információtartalmát. Az élesítési eljárások közül az egyik leggyakrabban használt a „centroid” vagy középponti módszer, amely a homályos halmaz súlypontját számítja ki.

Például, ha egy autóparkoló rendszert modellezünk, a Mamdani módszer segítségével meghatározhatjuk, hogy egy adott helyzetben milyen gyorsan és milyen szögben kell kormányozni az autónak a parkolóhelyre való beálláshoz. A szabályok ilyenek lehetnek: „Ha az autó távolsága a parkolóhelytől kicsi, akkor a kormányzási szög nagy legyen”, és így tovább. A Mamdani módszer segít a bonyolult és homályos helyzetek kezelésében, és a közelítőleges értékek alapján is képes döntéseket hozni.

Példa:

Az alábbiakban egy egyszerű példát ismertetek egy Mamdani típusú fuzzy inferenciarendszerre az MQL5 kereskedési platformon. A következmények itt fuzzy halmazok.

Tekintsünk két szabályt:

- Ha a piac alacsony volatilitású, akkor kis mennyiségű pozíciót nyitunk.
- Ha a piac magas volatilitású, akkor nagyobb mennyiségű pozíciót nyitunk.

A Mamdani rendszerben a következmények is fuzzy halmazok és az output az összes szabály következményeinek uniója, majd defuzzifikációval kapjuk a tényleges értéket.

```
input double lowVolatilityThreshold = 0.1;
input double highVolatilityThreshold = 0.2;
input double lowVolatilityPositionSize = 0.1;
input double highVolatilityPositionSize = 1.0;
```

```

// A példa egy egyszerűsített Mamdani rendszert mutat be
double CalculatePositionSize(double volatility)
{
    // Szabály 1: Alacsony volatilitás esetén kis pozíciót nyitunk
    double membershipLowVolatility = 1.0 - (volatility / lowVolatilityThreshold);
    double consequentLowVolatility = lowVolatilityPositionSize;

    // Szabály 2: Magas volatilitás esetén nagyobb pozíciót nyitunk
    double membershipHighVolatility = (volatility - lowVolatilityThreshold) /
(highVolatilityThreshold - lowVolatilityThreshold);
    double consequentHighVolatility = highVolatilityPositionSize;

    // Mamdani következtetés: a következmények fuzzy uniójának defuzzifikációja
    // Ebben a példában a centroid módszert használjuk a defuzzifikációra
    double positionSize = (membershipLowVolatility * consequentLowVolatility +
membershipHighVolatility * consequentHighVolatility) / (membershipLowVolatility
+ membershipHighVolatility);

    return positionSize;
}
void OnTick()
{
    double volatility = iATR(_Symbol, _Period, 14, 0);
    double positionSize = CalculatePositionSize(volatility);
}

```

Ebben a példában a Mamdani rendszer összegyűjti az egyes szabályok következményeit, majd egy defuzzifikációs lépés segítségével konvertálja ezeket egy konkrét értéké, amely meghatározza a nyitandó pozíció méretét. Az implementáció természetesen bővíthető további szabályokkal és komplexebb következményekkel.

Sugeno módszer:

A Sugeno-típusú fuzzy következtetés, amelyet Takagi, Sugeno és Kang fejlesztett ki, egy másik népszerű fuzzy rendszer, amelyet gyakran használnak komplex rendszerek modellezésére és irányítására. A Sugeno módszer néhány aspektusban eltér a Mamdani módszertől, és néha hatékonyabb lehet, különösen, amikor a probléma jól formalizált matematikai modellezést igényel.

A Sugeno típusú fuzzy következtetési folyamat a következőképpen néz ki:

Feltételértékelés: Hasonlóan a Mamdani módszerhez, itt is értékeljük a bemeneti változókat a szabályok feltételei alapján, meghatározva, hogy mennyire teljesülnek a szabályok.

Következtetés: A Sugeno módszerben a szabályok következményei általában lineáris függvények. Azaz, ahelyett, hogy homályos halmazok lennének, a következmények konkrét matematikai kifejezések, például $y = ax + b$, ahol y a kimenet, x a bemenet, a és b pedig konstansok.

Összegzés: Ebben a lépésben az egyes szabályok kimeneteit összegyűjtjük és súlyozzuk a szabályokhoz rendelt tagsági értékekkel. Az összegzett kimenet tehát egy súlyozott átlag lesz.

Defuzzifikáció: Mivel a Sugeno rendszerben a következmények már konkrét számértékek vagy lineáris függvények, ezért ebben az esetben nincs szükség külön erre a lépésre. Az összegzett érték közvetlenül a rendszer kimenetét képviseli.

A Sugeno típusú fuzzy rendszerek előnye, hogy gyakran kevesebb számítási erőforrást igényelnek, és könnyebben integrálhatók más matematikai modellekkel. Gyakran alkalmazzák olyan rendszerekben, ahol a bemenet és a kimenet közötti kapcsolatot könnyebb lineáris vagy egyéb matematikai függvényekkel leírni.

A Sugeno rendszerben, eltérően a Mamdani szabálytól, az input változó értéke nem egy fuzzy tagfüggvény segítségével van meghatározva, hanem egy lineáris függvényen keresztül. Egy fuzzy logikai szabály a Sugeno algoritmus számára a következőképpen írható le:

$$\text{if}(X_1 \text{ is } a_1) \wedge (X_2 \text{ is } a_2) \wedge \dots \wedge (X_n \text{ is } a_n) \text{ then } (Y = b_0 + b_1 \cdot X_1 + b_2 \cdot X_2 + \dots + b_n \cdot X_n)(W) \quad (9)$$

- $X = (X_1, X_2, X_3 \dots X_n)$ – a bemeneti változók;
- Y - kimeneti változó;
- $a = (a_1, a_2, a_3 \dots a_n)$ – a bemeneti változók vektora;
- $b = (b_1, b_2, b_3 \dots b_n)$ – a szabad tag a kimeneti érték lineáris függvényében;

- W – a szabály súlya.

Osztály Metódusok:

Calculate: Kiszámítja a fuzzy következtetést a rendszer számára.

CreateSugenoFunction: Létrehoz egy lineáris Sugeno függvényt a rendszer számára.

EmptyRule: Létrehoz egy üres fuzzy Sugeno szabályt az aktuális rendszer alapján.

Output: Megkapja a fuzzy Sugeno kimeneti változóinak listáját.

OutputByName: Egy meghatározott név alapján megkapja a fuzzy Sugeno kimeneti változót.

ParseRule: Létrehoz egy fuzzy Sugeno szabályt egy megadott sor alapján.

Rules: Visszaadja a fuzzy szabályok listáját.

A Sugeno rendszer lényege, hogy a kimeneti változók egyenes függvényeként vannak kifejezve, ami gyakran egyszerűsíti a számításokat és a rendszer implementációját.

Az alábbiakban egy egyszerű példaprogram található a Sugeno típusú fuzzy rendszer alkalmazására az MQL5 nyelven:

```
#include <Fuzzy.mqh> // Szükséges a fuzzy logikai könyvtár

// Definiáljuk az input változókat
input double X1 = 0.5;
input double X2 = 0.7;

void OnStart()
{
    // Hozzunk létre egy Sugeno típusú fuzzy rendszert
    CFuzzySystemSugeno fuzzySystem;

    // Hozzunk létre egy bemeneti változót X1-nek
    CFuzzyVariable &varX1 = fuzzySystem.Input("X1");
    varX1.AddTriangleTerm("LOW", 0, 0.5, 1);
    varX1.AddTriangleTerm("HIGH", 0.5, 1, 1.5);

    // Hozzunk létre egy bemeneti változót X2-nek
    CFuzzyVariable &varX2 = fuzzySystem.Input("X2");
    varX2.AddTriangleTerm("LOW", 0, 0.5, 1);
    varX2.AddTriangleTerm("HIGH", 0.5, 1, 1.5);
}
```

```

// Hozzunk létre egy Sugeno típusú szabályt
fuzzySystem.CreateSugenoFunction("OutputFunction", "X1 + 2 * X2");

// Hozzunk létre egy szabályt
fuzzySystem.ParseRule("IF X1 IS LOW AND X2 IS HIGH THEN OutputFunction WITH
1.0");

// Fuzzifikáljuk a bemeneti változókat
fuzzySystem.Fuzzify("X1", X1);
fuzzySystem.Fuzzify("X2", X2);

// Végrehajtjuk az inferenciát
double result = fuzzySystem.Calculate();

// Kiírjuk az eredményt
Print("Az eredmény: ", result);
}

```

Fontos megjegyezni, hogy ez csak egy egyszerű példa, és a valós alkalmazásokban sokkal összetettebb logikára és szabályokra lehet szükség. Ezen túlmenően a fenti kód csak a Sugeno fuzzy rendszer alapvető koncepcióját mutatja be az MQL5 használatával.

6.1.3. Az MT5 használata külső fejlesztő-környezetből

Az MT5 és Python integrációjával lehetővé válik, hogy a kereskedők és fejlesztők kiterjesszék az MT5 alapfunkcióit és automatizálják a kereskedési folyamatokat. A Python egy széles körben elterjedt programozási nyelv, amely sokrétű könyvtárakkal rendelkezik a pénzügyi adatok elemzéséhez és feldolgozásához, valamint gépi tanulási algoritmusok készítéséhez (Bird, Klein & Loper 2009).

A PyCharm, mint fejlesztői környezet, további lehetőségeket nyújt a fejlesztőknek a kód írásának, hibakeresésének és optimalizálásának folyamatában. A hibakereső eszközök, a kódelemzés és a grafikus felhasználói interfész segítik a fejlesztőt abban, hogy gyorsan és hatékonyan dolgozhasson.

A backtesting lehetősége a PyCharm-on belül lehetővé teszi a kereskedési stratégiák hatékonyságának értékelését történelmi adatok segítségével, így a kereskedők megítélhetik a potenciális teljesítményt anélkül, hogy kockáztatnák a tőkéjüket valós körülmények között.

Az MT5 platform és Python közötti integráció elősegíti a valós idejű adatfolyamok kezelését és az azokra épülő automatizált kereskedési rendszerek kialakítását. Ennek eredményeképpen a kereskedők képesek lehetnek gyorsabban reagálni a piaci változásokra, kihasználva a Python nyelv által biztosított adatkezelési és elemzési képességeket.

A közösségi támogatás szintén jelentős előny, mivel a fejlesztők és a kereskedők egyaránt hozzáférhetnek egy széleskörű tudásbázishoz, megoszthatják a problémákra vonatkozó megoldásokat és együttműködhetnek a kódok fejlesztésében.

Ez az integráció ötvözi a MT5 kereskedési platform előnyeit a Python mint programozási nyelv rugalmasságával és bővíthetőségével, ami átfogóbb és hatékonyabb kereskedési megoldások kialakítását teszi lehetővé.

Az MT5 platform és a PyCharm (Python fejlesztőkörnyezet) összekapcsolása általában az MT5 API-ján keresztül, vagy külső könyvtárak használatával történik, hogy a két rendszer között adatokat cserélhessenek és irányíthassák a kereskedést.

Az alábbiakban néhány módszert mutatok be, amit használhatunk az összekapcsoláshoz:

MetaTrader5 Python csomag

A MetaTrader5 Python package egy hivatalos könyvtár, amely közvetlenül kommunikál az MT5 terminállal az MT5 API-ján keresztül. Így lehet Python scripteket futtatni PyCharmban, amelyek képesek kommunikálni az MT5 platformmal.

A könyvtár telepítése:

```
pip install MetaTrader5
```

Ezután, a PyCharm projektben így lehet importálni és használni:

```
import MetaTrader5 as mt5
```

ZeroMQ

ZeroMQ egy másik módszer az MT5 és Python közötti aszinkron kommunikációra. Egy külön ZeroMQ bridge-t kell implementálni az MQL5 és a Python között. Ehhez az MQL5-nek kell egy DLL-t használnia, ami kezeli a ZeroMQ hívásokat, és a Python oldalon pedig használhatjuk a pyzmq könyvtárat.

Fájlok használata

Bár nem a leggyorsabb megoldás, a Python scriptek írhatnak és olvashatnak szöveges fájlokat vagy CSV fájlokat, amelyeket az MT5 kódol vagy az MQL5 szkriptek érzékelhetnek és feldolgozhatnak.

WebSocket vagy más hálózati protokollok

WebSockets és más hálózati protokollok használatával valós időben kommunikálhatunk az MT5 és a Python között. Ez több szintű hálózati és biztonsági tudást igényel, de erős és gyors megoldást kínálhat.

A konkrét implementációs lépések attól függenek, hogy milyen típusú interakciót szeretnénk az MT5 és a Python között. Mindig meg kell győződni arról, hogy megfelelő biztonsági és hibakezelési protokollok vannak használatban, különösen ha valós kereskedési környezetben dolgozunk.

Az alábbiakban egy példakódot mutatok be, amely demonstrálja, hogyan lehet a MetaTrader 5 kereskedési platformot integrálni Python nyelven írt szkriptekkel a ***MetaTrader5*** Python könyvtár és a ***pandas*** könyvtár segítségével. A példa bemutatja, hogy hogyan hozhatunk létre kereskedési műveleteket, mint például limit megbízásokat, hogyan számíthatunk ki nyereséget és margin-t, valamint hogyan kezelhetjük a pozíciókat és várakozó megbízásokat. Az itt látható metódusok segíthetik a felhasználókat az automatizált kereskedési stratégiák kialakításában és azok MT5 platformon való implementálásában. A kód tartalmazza az alábbi fő funkciókat és műveleteket:

Inicializálás: Az ***mt5.initialize()*** függvény segítségével csatlakozik az MT5 kereskedési számlájához, megadott felhasználónévvel (login), szerverrel és jelszóval.

```
import MetaTrader5 as mt5
import pandas as pd
```

```
mt5.initialize(login = 12345678, server = "ICMarketsSC-Demo", password = "kkkkkkkk")
```

Buy Limit és Sell Limit megadása: A *buy_limit()* és *sell_limit()* függvényekkel limitált vételi és eladási megbízásokat hoz létre a megadott árfolyamon és mennyiségben.

```
def buy_limit(symbol, volume, price):
    request = mt5.order_send({
        "action": mt5.TRADE_ACTION_PENDING,
        "symbol": symbol,
        "volume": volume,
        "type": mt5.ORDER_TYPE_BUY_LIMIT,
        "price": price,
        "deviation": 20,
        "magic": 100,
        "comment": "python market order",
        "type_time": mt5.ORDER_TIME_GTC,
        "type_filling": mt5.ORDER_FILLING_IOC,
    })

    print(request)
```

```
def sell_limit(symbol, volume, price):
    request = mt5.order_send({
        "action": mt5.TRADE_ACTION_PENDING,
        "symbol": symbol,
        "volume": volume,
        "type": mt5.ORDER_TYPE_SELL_LIMIT,
        "price": price,
        "deviation": 20,
        "magic": 100,
        "comment": "python market order",
        "type_time": mt5.ORDER_TIME_GTC,
        "type_filling": mt5.ORDER_FILLING_IOC,
    })

    print(request)
```

Nyereség Számítása: A *cal_profit()*, *cal_volume()*, *cal_buy_profit()*, *cal_sell_profit()*, *cal_buy_margin()* és *cal_sell_margin()* függvényekkel kiszámolja a nyereséget, a pozíciók összmenyiségét, a vételi és eladási nyereséget, valamint a szükséges margin mértékét.

```
def cal_profit(symbol):
    usd_positions = mt5.positions_get(symbol=symbol)
    df = pd.DataFrame(list(usd_positions), columns=usd_positions[0]._asdict().keys())
    df['time'] = pd.to_datetime(df['time'], unit='s')
    df.drop(['time_update', 'time_msc', 'time_update_msc', 'external_id'],
axis=1, inplace=True)
    profit = float(df["profit"].sum())
    return profit
```

Nyereség százalék számítása: A *cal_pct_buy_profit()* és *cal_pct_sell_profit()* függvények a vételi és eladási pozíciók nyereségének százalékos arányát számolják ki a felhasznált margin-hoz képest.

```
def cal_pct_buy_profit(symbol):
    profit = cal_buy_profit(symbol)
    margin = cal_buy_margin(symbol)
    pct = (profit/margin)*100
    return pct
```

Pozíciók bezárása: A *close_position()* függvény bezár egy adott pozíciót, míg a *close_all()* függvény az összes pozíciót bezárja egy adott szimbólumon.

```
def close_position(position):
    tick = mt5.symbol_info_tick(position.symbol)
    request = {
        "action": mt5.TRADE_ACTION_DEAL,
        "position": position.ticket,
        "symbol": position.symbol,
        "volume": position.volume,
        "type": mt5.ORDER_TYPE_BUY if position.type == 1 else mt5.ORDER_TYPE_SELL,
        "price": tick.ask if position.type == 1 else tick.bid,
```

```

        "deviation": 20,
        "magic": 100,
        "comment": "python script close",
        "type_time": mt5.ORDER_TIME_GTC,
        "type_filling": mt5.ORDER_FILLING_IOC,
    }
    result = mt5.order_send(request)
    return result
def close_all(symbol):
    position = mt5.positions_get(symbol=symbol)
    for position in position:
        close_position(position)

```

Várakozó megbízások törlése: A *delete_pending()* függvény eltávolít egy várakozó megbízást (limit vagy stop), a *close_limit()* függvény pedig az összes várakozó megbízást törli egy adott szimbólumra vonatkozóan.

```

def delete_pending(ticket):
    close_request = {
        "action": mt5.TRADE_ACTION_REMOVE,
        "order": ticket,
        "type_time": mt5.ORDER_TIME_GTC,
        "type_filling": mt5.ORDER_FILLING_IOC,
    }
    result = mt5.order_send(close_request)
    if result.retcode != mt5.TRADE_RETCODE_DONE:
        result_dict = result._asdict()
        print(result_dict)
    else:
        print('Delete complete...')
def close_limit(symbol):
    orders = mt5.orders_get(symbol=symbol)
    df = pd.DataFrame(list(orders), columns=orders[0]._asdict().keys())
    df.drop(['time_done', 'time_done_msc', 'position_id', 'position_by_id',
'reason', 'volume_initial',
        'price_stoplimit'], axis=1, inplace=True)
    df['time_setup'] = pd.to_datetime(df['time_setup'], unit='s')
    for ticket in df.ticket:
        delete_pending(ticket)

```

A kódokat a felhasználói igényeknek megfelelően módosíthatók. A pandas DataFrame-ek használata megkönnyíti az adatok kezelését és elemzését. A kód így lehetővé teszi az automatizált kereskedési stratégiák és a pozíciómenedzsment hatékony végrehajtását.

7. A TECHNIKAI MUTATÓK ÉS A NEURÁLIS HÁLÓZATOK HASZNÁLATA A RÉSZVÉNYPIACI TRENDEK ELŐREJELZÉSÉBEN

A részvénykereskedésben a technikai elemzés már hosszú ideje alkalmazott módszer a piaci irányzatok kiderítésére. A technikai indikátorok olyan hagyományos eszközök, melyeket kereskedők és piaci szakértők használnak a piaci trendek előrejelzésére, múltbeli ár- és piaci adatok alapján. A neurális hálózatok megjelenése új dimenziót hozott a technikai elemzésben. Az emberi megfigyelés helyett a neurális hálózatok a bonyolult és emberi szemmel rejtett mintákat és kapcsolatokat is képesek kiszűrni az adatokból. A két módszer kombinálásakor, amikor a technikai indikátorokat a neurális hálózat bemeneti adataiként használjuk, az eredmény egy olyan rendszer lehet, amely egyesíti mindkét módszer erősségeit. Míg a technikai indikátorok az árváltozások és piaci trendek rövidebb távú információit szolgáltatják, addig a neurális hálózatok a mélyebb összefüggések és összetett kapcsolatok feltárására képesek. Számptalan piaci indikátor létezik, és az újabbak folyamatosan fejlesztés alatt állnak. Azonban néhányukat kiemelten gyakran alkalmazzák és ismertek a szakmában (Cervelló-Royo, Guijarro & Michniuk, 2015; Fernandez-Rodriguez, Gonzalez-Martel & Sosvilla-Rivero, 2000; Arnold, G. 2011):.

Mozgóátlagok:

- Egyszerű Mozgóátlag (SMA)
- Exponenciális Mozgóátlag (EMA)
- Súlyozott Mozgóátlag (WMA)

1. Momentum Indikátorok:

- Relatív Erő Index (RSI)
- Stochasztikus Oscillátor
- MACD (Moving Average Convergence Divergence)
- CCI (Commodity Channel Index)
- ROC (Rate of Change)
- Awesome Oscillator
- Larry William's R%

2. Volatilitás Indikátorok:

- Bollinger Bands
- Average True Range (ATR)

- Keltner Channels

3. Trendkövető Indikátorok:

- Mozgóátlagok (mint fentebb)
- ADX (Average Directional Index)
- Parabolic SAR

4. Volume (forgalom) Indikátorok:

- Volume
- On-Balance Volume (OBV)
- Chaikin Money Flow (CMF)
- Money Flow Index (MFI)

5. Overlap Studies:

- Pivot Points
- Fibonacci Retracements és Extensions

6. Ciklikus Indikátorok:

- Elliott Wave Theory

7. Formáció Alapú Indikátorok:

- Head and Shoulders
- Double Top/Bottom
- Triangles (ascending, descending, symmetrical)

8. Egyéb Indikátorok:

- Ichimoku Cloud
- Gator Oscillator
- Alligator Indicator
- Market Depth
- TRIX

E projekt keretében számos technikai indikátorral végeztem kísérletet, amelyeket a neurális hálózat bemeneti adataként használtam fel. A kísérletek alapján sikerült kiválasztani négy olyan indikátort, amelyek működése a profit realizálása szerint az optimumot adták. Ezek az indikátorok a piaci viselkedés különböző szempontjait fedik le, mint az ártrendek, a dinamika, a volatilitás és a tranzakciós volumen. A neurális hálózat segítségével kereskedelmi

jelzéseket hoztam létre ezen indikátorok felhasználásával. Az implementáció során több részvényen is lefolytattam a vizsgálatot, az alábbi módon:

A vizsgálatot Python programozási nyelven végeztem, amely rendkívül népszerű a pénzügyi adatelemzés és a gépi tanulás területén. A Python nyelv lehetővé teszi számunkra, hogy különböző könyvtárakat és eszközöket használjunk, amelyek egyszerűsítik az adatok letöltését, manipulációját és vizualizációját. Ezen kívül, a Python széles körű támogatást nyújt a neurális hálózatok létrehozásához és képzéséhez (McKinney, 2012).

Az elemzés során a *yfinance* könyvtárat alkalmaztam a részvényár adatok letöltésére, míg a neurális hálózatok építéséhez és tanításához a *keras* és *tensorflow* csomagokat használtam. A vizualizációk elkészítéséhez a *matplotlib* és *seaborn* könyvtárakat vettem igénybe.

Technikai indikátorok kiszámítása: Három különböző technikai indikátort számolunk ki, a negyediket közvetlenül megkapjuk:

- SMA (Simple Moving Average): Az egyszerű mozgóátlagot a záróár 15 napos ablakában számoljuk ki.
- RSI (Relative Strength Index): A relatív erőindexet a záróár napi változásainak alapján számoljuk, és 15 napos ablakot használunk hozzá.
- Bollinger Bands: A Bollinger sávokat az SMA és a záróárak szabványos eltéréseinek alapján számoljuk. Az alsó és felső sávokat az SMA +/- 2 szabványos eltéréssel adjuk meg.

1. Ártrendet meghatározó indikátor:

Simple Moving Average (SMA) Egyszerű mozgóátlag adott t időtartamra:

$$MA_t = \frac{1}{t} \sum_{i=1}^t c p_i \quad (10)$$

- $c p_i$ az i . napi záróár.
- t az időtartam (a példában $t=15$).

2. Lendületet meghatározó indikátor:

Relative Strength Index (RSI) Relatív erő index

$$RSI = 100 - \left(\frac{100}{1+RS} \right) \quad (11)$$

Ahol:

$$RS = \frac{\text{Átlag } t \text{ napi emelkedő zárasi árak}}{\text{Átlag } t \text{ napi csökkenő zárasi árak}} \quad (12)$$

3. Volatilitást meghatározó indikátor:

Bollinger Bands (Bollinger Sávok): A Bollinger sávok egy középső sávból (egyszerű mozgóátlag) és két szabványeltérésen alapuló külső sávból állnak.

- **Középső sáv (Middle Band):**

$$MB = SMA(n) \quad (13)$$

SMA(n) az árak n napos egyszerű mozgóátlaga.

- **Felső sáv (Upper Band):**

$$UB = SMA(n) + (k \times SD(n)) \quad (14)$$

SD(n) az árak n napos szabványos eltérése és k egy konstans, mely általában 2.

- **Alsó sáv (Lower Band)**

$$LB = SMA(n) - (k \times SD(n)) \quad (15)$$

A SD(n) képlete, azaz az árak n napos szabványos eltérése:

$$SD(n) = \sqrt{\frac{\sum_{i=1}^n (P_i - SMA(n))^2}{n}} \quad (16)$$

Ahol P_i i-edik napi záróár.

Ez az indikátor segíthet azoknak a pillanatoknak a meghatározásában, amikor az ár „túl magas” (a felső sáv felett) vagy „túl alacsony” (az alsó sáv alatt), ami esetleges fordulatokat

jelezhet. Ugyanakkor fontos megjegyezni, hogy az indikátor önmagában nem ad kereskedési jeleket, hanem a többi technikai elemzési eszközzel kombinálva használják a leggyakrabban.

4. Forgalmat meghatározó indikátor

Volume (forgalom) indikátor:

A „Volume” (vagy forgalom) ténylegesen az adott időszakban kereskedett értékpapírok vagy egyéb eszközök számát jelzi. Önmagában nincs bonyolult képlete, de egyes kereskedési stratégiák vagy technikai indikátorok számolnak a volume-al vagy annak változásaival.

Napi forgalom:

V_t az aktuális nap (t) forgalma

Forgalom változás:

$$\Delta V = V_t - V_{t-1} \quad (17)$$

Ahol V_t az aktuális nap (t) forgalma és V_{t-1} az előző nap (t-1) forgalma.

Átlagos forgalom (pl. 20 napra):

$$AV20 = \frac{\sum_{i=1}^{20} V_{t-i}}{20} \quad (18)$$

V_{t-i} az i-edik nappal ezelőtti forgalom

A forgalom az egyik legkritikusabb adatforrás a technikai analízis során, mert jelzi a tranzakciók számát és a befektetők iránti érdeklődés erősségét egy adott értékpapírra vonatkozóan. Bár a kereskedett mennyiség magában véve nem tartalmaz komplex képleteket, annak változékonysága, irányzatai és együttese más technikai mutatókkal mélyebb rálátást kínálnak a piaci tendenciákra.

7.1. A NEURÁLIS HÁLÓ IMLEMENTÁLÁSA

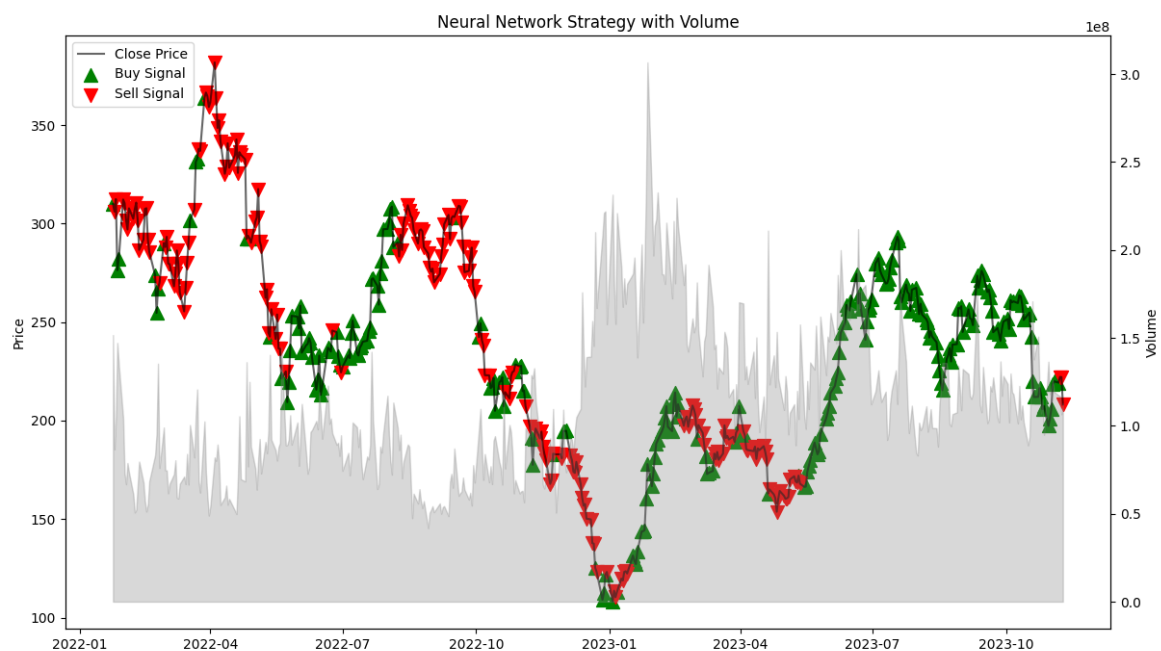
Az adatokat előkészítjük a neurális hálózat tanításához. A modell bemeneteként az SMA-t, RSI-t, Bollinger sávokat és a napi forgalmat (Volume) használjuk. A célváltozó (y) azt mutatja, hogy az adott napon a záróár emelkedni vagy csökkenni fog-e a következő napon.

A tanítási és tesztelési adatokat 80-20%-os arányban osztjuk fel. Az adatokat normalizáljuk a *StandardScaler* segítségével.

Ezután létrehozunk egy egyszerű neurális hálózatot, amely három rétegből áll, a középső rétegben pedig néhány kiesési réteget használunk az overfitting elkerülése érdekében. A hálózatot az „Adam” optimalizálóval és a bináris keresztentropia veszteségfüggvénnyel tanítjuk. Az „Adam” optimalizáló egy népszerű módszer a gépi tanulásban, különösen a mély tanulás területén. A neve az „adaptive moment estimation” kifejezésből származik. Az Adam optimalizáló különlegessége, hogy adaptívan módosítja a tanulási rátákat a különböző paraméterek számára, ami gyakran gyorsabb konvergenciához vezet, mint más optimalizálási módszerek, például a gradient descent vagy a momentum alapú módszerek.

Vételi és eladási jelek generálása: A tanított modell alapján előrejelzéseket végzünk az összes adatpontra. Ezt követően vételi és eladási jeleket generálunk a predikciók alapján.

Grafikus ábrázolás: Az árfolyamot, a vételi és eladási jeleket egy grafikonon ábrázoljuk. Ezen kívül egy második Y tengelyen az adott időszakban történő napi forgalmat is megjelenítjük (8. ábra).



8. ábra A neurális hálózat működése a 4 technikai indikátorral (Forrás: saját szerkesztés)

Visszatesztelés: A generált jelek alapján egy egyszerű visszatesztelést végzünk, ahol a kezdeti tőkénk 100.000 US-dollár. Ezt követően összehasonlítjuk a neurális hálózat alapján elért nyereséget a benchmark nyereséggel (az egész időszak alatt tartott pozíció nyeresége).

Profit gained from the Neural Network Strategy by investing \$100k in TSLA : \$271284.04
Profit percentage of the Neural Network Strategy : 271.28%
Benchmark profit by investing \$100k : \$-32851.61
Benchmark Profit percentage : -32.85%
Neural Network Strategy profit is 304.13% higher than the Benchmark Profit

9. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

Ez nagyon jó eredmény. A neurális háló alapján kialakított stratégia 271,28%-os nyereséget hozott a vizsgált időszakban (2022.01.01 – 2023.11.09), míg, ha az egész időszak alatt tartottuk volna a részvényeket (benchmark), akkor 32,85%-os veszteséget szenvedtünk volna el. A neurális hálózat alapú stratégia tehát 304,13%-kal teljesített jobban a benchmarkhoz képest. Ez azt mutatja, hogy a modell képes volt jól előrejelezni a piaci mozgásokat és ennek megfelelően vételi/eladási jeleket adni.

Visszatesztelés (S&P500 Benchmark):

A tesztelés folytatásaként az S&P 500 indexet használhatjuk benchmarkként a kereskedési stratégia teljesítményének értékelésére, amelyet ugyanarra a részvényre és időszakra alkalmaztam (10. ábra).

Profit gained from the Neural Network Strategy by investing \$100k in TSLA : \$186593.44
Profit percentage of the Neural Network Strategy : 186.59%
Benchmark profit by investing \$100k in S&P 500 : \$-9327.93
Benchmark Profit percentage : -9.32%
Neural Network Strategy profit is 195.92% higher than the Benchmark Profit

10. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

A megadott eredmények alapján a neurális hálózatot használó kereskedési stratégia rendkívül magas hozamot ért el a TSLA részvényekre vonatkozóan, amennyiben 100.000 US-dolláros kezdőtőkével számolunk:

- A Neural Network Strategy által elért profit: \$186 593,44
- A Neural Network Strategy által elért profit százalékban: 186,59%

Ugyanakkor az S&P 500 index, mint benchmark, negatív hozamot produkált ugyanebben az időszakban:

- A Benchmark által elért profit: -\$9 327,93
- A Benchmark által elért profit százalékban: -9,33%

A Neural Network Strategy és a Benchmark közötti különbség: A Neural Network Strategy profitja 195.92%-kal magasabb, mint a Benchmark profitja.

Azonban fontos megjegyezni, hogy a visszatesztelés csak a múltbeli adatokon történt, és a jövőbeli teljesítményt nem garantálja. Mindig ajánlott további vizsgálatokat és teszteket végezni más időszakokon és más részvényeken is, mielőtt valós kereskedési döntéseket hoznánk egy modell alapján.

7.2. TESZTELÉS A VÉTELI JELZÉSEK ALAPJÁN JÖVŐBELI ADATOKRA VONATKOZTATVA

Az alábbiakban listázott adatok a Tesla, Inc. (TSLA) részvényeire vonatkozó vételi és eladási jelzéseket tartalmazzák, amelyeket a neurális hálózat alapú kereskedési modell generált 2023. október 23. és november 9. között. A jelzések potenciálisan előrejelzik a piaci fordulópontokat vagy a kereskedési lehetőségeket, amelyeket a modell az adatok alapján azonosított.

```
{ "Date": "2023-10-23", "Price": 212.080002, "Signal": "Buy"},  
{ "Date": "2023-10-24", "Price": 216.520004, "Signal": "Buy"},  
{ "Date": "2023-10-25", "Price": 212.419998, "Signal": "Buy"},  
{ "Date": "2023-10-26", "Price": 205.759995, "Signal": "Buy"},  
{ "Date": "2023-10-27", "Price": 207.300003, "Signal": "Buy"},  
{ "Date": "2023-10-30", "Price": 197.360001, "Signal": "Buy"},  
{ "Date": "2023-10-31", "Price": 200.839996, "Signal": "Buy"},  
{ "Date": "2023-11-01", "Price": 205.660004, "Signal": "Buy"},  
{ "Date": "2023-11-02", "Price": 218.509995, "Signal": "Buy"},  
{ "Date": "2023-11-03", "Price": 219.960007, "Signal": "Buy"},  
{ "Date": "2023-11-06", "Price": 219.270004, "Signal": "Sell"},  
{ "Date": "2023-11-07", "Price": 222.179993, "Signal": "Sell"},  
{ "Date": "2023-11-08", "Price": 222.110001, "Signal": "Sell"},  
{ "Date": "2023-11-09", "Price": 208.020004, "Signal": "Sell"},
```

A tesztelésnél a kereskedési stratégia az, hogy az első „Buy” jelzésnél történik a vásárlás, és a pozíciót az első „Sell” jelzésig tartjuk. A következő „Buy” jelzéseket figyelmen kívül hagyjuk, egészen addig, amíg egy „Sell” jelzés nem érkezik, és csak akkor adjuk el a pozíciót. A kereskedés eredménye a 11. ábrán látható.

Final Portfolio Value: 103386.49 Profit from Trading: 3386.49 Buy Operations: [('2023-10-23', 471.0, 212.080002, 99889.68)] Sell Operations: [('2023-11-06', 471.0, 219.270004, 103276.17)]
--

11. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

A megadott időszakban a következő történt:

A program által generált adatok alapján a Tesla részvényeire vonatkozó kereskedési modell a következő eredményeket produkálta 2023. október 23. és november 9. között:

Kezdeti portfólió érték: \$100.000

Vételi műveletek:

2023. október 23-án 471 részvényt vásárolt 212.08 USD/részvény áron, összesen \$99.889,68 értékben.

Eladási műveletek:

2023. november 6-án az összes, azaz 471 részvényt eladta 219.27 USD/részvény áron, összesen \$103 276.17 értékben.

Végső portfólió érték: \$103 386.49

Kereskedésből származó profit: \$ 3 386.49

Ez azt jelenti, hogy az adott időszakban a modell által generált kereskedési stratégia pozitív hozamot eredményezett. A kezdeti tőke 100.000 dollárról 103.386,49 US-dollárra nőtt, ami körülbelül 3.39%-os profitot jelent. A kereskedési modell csak egyetlen vételi és egyetlen eladási műveletet hajtott végre, amelyek sikeresnek bizonyultak ebben az időszakban.

Egy másik kereskedési stratégiát alkalmazva:

Minden 'Buy' jelzés esetén 1000 dollár értékű részvényt vásárolunk és minden 'Sell' jelzésnél az összes addigi pozíciót eladjuk. A tranzakciók listája tartalmazza az összes vételi és eladási műveletet. Az összesített profit a kezdő tőke és a kereskedési műveletek eredményeként fennmaradó készpénz különbségeként jelenik meg.

A kereskedési jelzések alapján a stratégia enyhe nyereséget termelt a megadott időszak alatt, az alábbiak szerint (12. ábra):

Final Portfolio Value: 100471.55057799588
Profit from Trading: 471.55057799587667
Tranzactions: [('Buy', '2023-10-23', 4.715201766171239, 212.080002), ('Buy', '2023-10-24', 4.618510906733587, 216.520004), ('Buy', '2023-10-25', 4.707654690779161, 212.419998), ('Buy', '2023-10-26', 4.8600312222985815, 205.759995), ('Buy', '2023-10-27', 4.823926606503715, 207.300003), ('Buy', '2023-10-30', 5.066882827995121, 197.360001), ('Buy', '2023-10-31', 4.979087930274605, 200.839996), ('Buy', '2023-11-01', 4.862394148353707, 205.660004), ('Buy', '2023-11-02', 4.576449695127218, 218.509995), ('Buy', '2023-11-03', 4.546280997345122, 219.960007), ('Sell', '2023-11-06', 47.75642079158205, 219.270004)]

12. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

Végső portfólió értéke: \$100.471,55

Profit a kereskedésből: \$ 471,55

Tranzakciók:

Vételi műveletek sorozata 2023. október 23-tól 2023. november 3-ig, ahol minden egyes vételnél \$1000 értékben vásárolt részvényeket.

Egy eladási művelet 2023. november 6-án, ahol a stratégia az összes korábban vásárolt részvényt eladta 219.27 USD/részvény áron.

A kereskedési műveletek eredményeképpen a kezdeti 100.000 dolláros tőke enyhe növekedést mutatott. Az előbbi stratégiához képest a kevesebb profitot a kockázatkerülőbb stratégia is okozta, hiszen csak kevés részvénnnyel kereskedtünk. Amennyiben 10.000 dolláros tranzakciókat hajtottunk volna végre a profit az előző stratégiához képest nagyobb hasznot tudott volna hozni (\$4 715).

Ezek az eredmények azt mutatják, hogy a modell jól teljesített a tesztelt időszakban. Fontos megjegyezni azonban, hogy a kereskedési modellek teljesítménye változó lehet más időszakokban és piaci körülmények között. A valós piaci kereskedés során további költségek, például tranzakciós díjak és adók is felmerülhetnek, amelyek befolyásolhatják a tényleges hozamot.

8. A SZENTIMENTÁLIS ELEMZÉS, MÓDSZEREK ÉS ALKALMAZÁSOK

A digitális korban hihetetlen mennyiségű szöveges adat halmozódik fel nap mint nap. A közösségi médiás bejegyzésektől kezdve az üzleti jelentéseken át a hírportálok cikkeiig, a szöveges adatok gyakorlatilag mindenhol jelen vannak. Ezen információforrások kiaknázása kritikus szerepet játszik a gépi tanulás területén.

Az első és legfontosabb ok, amiért a szöveges adatok a gépi tanulás szempontjából fontosak, hogy ezek az adatok hatalmas mennyiségű információt tartalmaznak az emberek viselkedéséről, érzéseiről és véleményéről. Egy egyszerű tweet vagy termékértékelés képes értékes információt szolgáltatni egy cég számára a termékei, vagy szolgáltatásai iránti közönségérzésről (Russell, 2011; Russell, 2013)).

Ezen túlmenően a szöveges adatok segítségével mélyebb betekintést nyerhetünk az emberi nyelv működésébe. A nyelvi modellezés, a szemantikai elemzés vagy a beszédfelismerés mind olyan területek, melyek a gépi tanulás által hatalmas fejlődést mutatnak, és mindezek középpontjában a szöveges adatok állnak.

Azonban a szöveges adatoknak a gépi tanulásban való felhasználása nem megy problémák nélkül. A nyelv változó és sokszínű természete miatt a modelleknek képesnek kell lenniük az idiómákra, szlengekre és a nyelv dinamikus változásaira való reagálásra. Emellett az adatok gyakran struktúrálatlanok, ami kihívást jelent a hagyományos adatelemzési módszerek számára.

8.1. A SZENTIMENTÁLIS ELEMZÉS ÉS MŰKÖDÉSE

A szentimentális elemzés, más néven véleménybányászat, a természetes nyelvfeldolgozás (NLP) és a szövegbányászat területére esik. Célja annak meghatározása, hogy egy adott szöveg (pl. termékértékelés, hírcikk vagy közösségi médiás poszt) milyen érzelmi hangulatot tükröz: pozitív, negatív vagy semleges (Bollen, Mao & Zeng 2010).

Végrehajtása:

- Adatgyűjtés: Szöveges adatok gyűjtése a kívánt forrásból (pl. Twitter, ügyfélértékelések).
- Adat előfeldolgozás: A szöveg tisztítása és előkészítése (pl. sztopszavak eltávolítása, lemmatizálás).
- Modell építés: A szentiment osztályozó modell építése korábban címkézett adatokon (pl. pozitív és negatív értékelésekkel).

- Modell alkalmazása: Az épített modell alkalmazása az új szövegekre a szentiment meghatározásához.

8.2. MÓDSZEREK ÉS ESZKÖZÖK A SZENTIMENTÁLIS ELEMZÉSBEN

Szabályalapú módszerek: Szabályok és listák (pl. pozitív és negatív kulcsszavak listája) segítségével történik az érzelmek osztályozása.

Gépi tanulás alapú módszerek: Felügyelt tanulási modellek, mint például a logisztikus regresszió, döntési fa vagy mély tanulási modellek használata az érzelmek osztályozásához.

Hibrid módszerek: A szabályalapú és gépi tanulási módszerek kombinálása az elemzés javítása érdekében.

Eszközök:

- Python könyvtárak: NLTK, TextBlob, SpaCy, Keras, TensorFlow
- SaaS megoldások: IBM Watson Tone Analyzer, Google Cloud Natural Language, Amazon Comprehend stb.

8.3. A NYELVI INFORMÁCIÓK ÁTALAKÍTÁSA

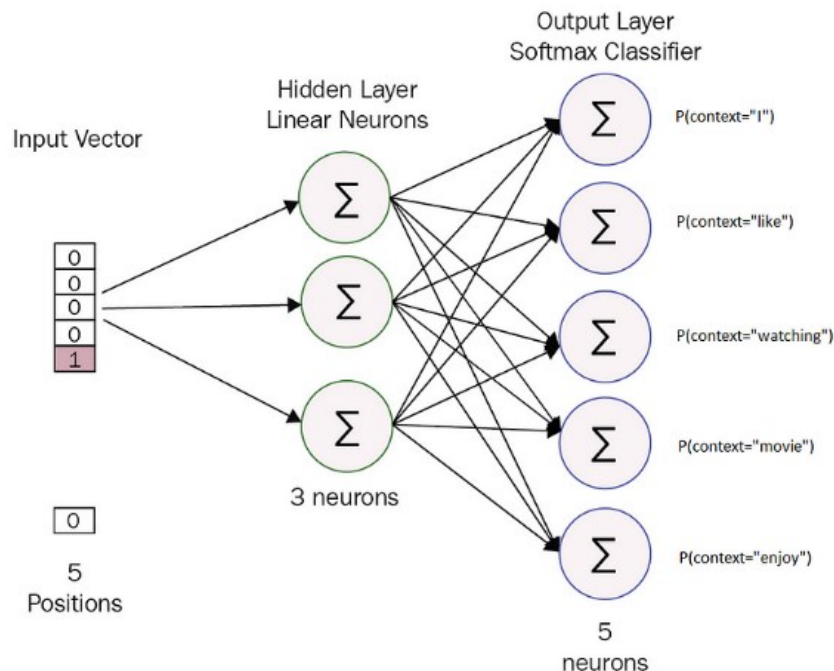
Tokenizálás: A tokenizálás a szövegfeldolgozás első lépése. Egy hosszabb szövegrészt szavakra, mondatokra vagy akár karakterekre bonthatunk. Ezen lépés célja, hogy a szövegben található információt kisebb, kezelhető részekre osztjuk, amelyekkel aztán tovább dolgozhatunk.

Bag-of-Words (BoW) Modell: Az BoW modell az egyik leginkább alapvető szövegvektor-képző módszer. Az elkészült mátrixban egy dokumentum egyedi szavakból álló „zsákját” képviseli. Hátránya, hogy elveszti a szavak közötti sorrendi információt, viszont egyszerű és gyors.

TF-IDF (Term Frequency-Inverse Document Frequency): A TF-IDF egy statisztikai módszer, amely kiemeli a szavak fontosságát egy dokumentumgyűjteményben. Míg a TF a szó gyakoriságát méri egy dokumentumban, addig az IDF azt méri, hogy mennyire egyedi egy szó az összes dokumentumhoz képest.

Word Embeddings: A szóbeágyazások mély tanulási modelleket használnak a szavak vektoros reprezentációjának létrehozásához. A kapott vektorok nem csak az adott szó gyakoriságát tartalmazzák, hanem a szemantikai jelentését is képesek hordozni, így képesek a szavak közötti kapcsolatok megállapítására is.

Word2Vec: A Word2Vec egy népszerű módszer a szavak vektoros ábrázolására. Két fő architektúrája van: CBOW (Continuous Bag of Words) és skip-gram. Míg a CBOW a kontextus szavakat használja a központi szó becslésére, a skip-gram az ellenkezőjét teszi: a központi szót használja a kontextus szavak becslésére. A Word2Vec tanítása során a modell a szavak közötti kapcsolatokat tanulja meg a nagy mennyiségű szöveges adat alapján.



13. ábra Word2Vec modell (Nayak, 2019)

GloVe (Global Vectors for Word Representation): A GloVe egy másik módszer a szavak vektoros ábrázolására. A GloVe kifejezetten a szó-szó társulások globális statisztikáin alapszik. A módszer egy ko-occurrence mátrixot épít, amely megmutatja, hogy egy adott szó hányszor fordul elő egy másik szó közelében, és ezt a mátrixot használja a vektoros ábrázolások tanításához.

FastText: A Word2Vec továbbfejlesztett változata, amely képes a szavak belső struktúráját is figyelembe venni.

N-gramok: Az N-gramok lehetővé teszik a kontextus megőrzését a szavak kombinációinak létrehozásával. A bigramok két, a trigramok három egymást követő szót kombinálnak, így a szavak közötti sorrendi információt is képesek hordozni.

Morphológiai Elemzés: Ez a lépés a nyelv szerkezeti elemeire bontja a szavakat. A morfológiai elemzéssel képesek vagyunk azonosítani a szavak gyökereit, töit és a hozzájuk kapcsolódó toldalékokat. Ez különösen hasznos olyan nyelveknél, mint a magyar, ahol a szavak formája változatos lehet a mondatban betöltött szerepüknek megfelelően.

8.4. A SZENTIMENTÁLIS ELEMZÉS INTEGRÁLÁSA AZ ALGORITMIKUS KERESKEDÉSI STRATÉGIÁKBA

A szentimentális elemzést használva a kereskedők képesek lehetnek azonosítani a piaci hangulatot, amely befolyásolhatja a részvényárakat és egyéb pénzügyi eszközöket. Az algoritmikus kereskedésben a szentimentális elemzés folyamata az alábbiak szerint valósítható meg:

1. Adatgyűjtés: Hírcikkek, közösségi médiás posztok, elemzői jelentéseket stb. gyűjtése
2. Szentimentális elemzés: Meg kell határozni a szöveg szentimentjét (pozitív, negatív, semleges).
3. Jelzések generálása: Kereskedési jelek létrehozása a szentiment alapján. Például, ha egy céghez kapcsolódó hírek pozitívak, az algoritmus vásárlási jelzést ad.
4. Automatizált kereskedés: A jelek alapján az algoritmus eldönti, hogy vásárol, elad vagy tartja a pozíciót.

8.5. A SENTIMENT ANALÍZIS TESZTELÉSE

A szentiment analízis tesztelésére a python eszköztárában van megfelelő metódus.

Az alábbi kódrészletben a **transformers** könyvtárt használtam, amely a Hugging Face által fejlesztett és tartalmazza a BERT modellt. A Yahoo Finance-ből származó hírek érzelmi elemzéséhez a **nlptown/bert-base-multilingual-uncased-sentiment** előre betanított modellt alkalmaztam, amely képes több nyelven is érzelmi hordalékot meghatározni. A GDELT API segítségével hírcímeket gyűjtöttem be a kívánt témában (jelen esetben „BRENT OIL”).

```
# GDELT API lekérdezés
```

```
BASE_URL = "https://api.gdeltproject.org/api/v2/doc/doc"
```

```

QUERY_PARAMS = {
    "query": "BRENT OIL",
    "mode": "ArtList",
    "maxrecords": 250,
    "sort": "datetime",
    "format": "json"
}

```

Részvényárfolyam lekérdezése: A yfinance könyvtár segítségével letölti az adott részvény (jelen esetben a Brent olaj, 'BZ=F') tőzsdei adatait a hírek időpontjai alapján.

```

# Részvényárfolyam lekérdezése
start_date = min(sorted_timestamps)
end_date = max(sorted_timestamps)
stock_data = yf.download('CL=F', start=start_date, end=end_date)

stock_prices = stock_data['Close'].reindex(sorted_timestamps, method='ffill')

```

Szövegelemzési függvény: A sentiment_analysis függvény a BERT modellt használja a szövegek szentimentjének meghatározására. A kimenet pozitív (1), semleges (0) vagy negatív (-1) értékek lehetnek.

```

def sentiment_analysis(text):
    inputs = tokenizer(text, return_tensors="pt", truncation=True, padding=True,
max_length=512)
    outputs = model(**inputs)
    probs = torch.nn.functional.softmax(outputs.logits, dim=-1)
    sentiment = torch.argmax(probs)
    if sentiment == 0 or sentiment == 1:
        return -1
    elif sentiment == 2:
        return 0
    else:
        return 1

```

A BERT (Bidirectional Encoder Representations from Transformers) egy előre betanított nyelvi reprezentációs modell, amelyet számos NLP (Natural Language Processing) feladatra

lehet alkalmazni anélkül, hogy a specifikus feladatra újra kéne tanítani. A BERT modell képes érteni a szöveg kontextusát, mivel mindkét irányban (előre és hátra) figyelembe veszi a szavakat a mondatban.

A kódban:

- A *BertTokenizer.from_pretrained* függvénnyel betöltjük az előre betanított tokenizálót, amely a szövegeket tokenekre bontja (általában szavakra vagy szó részekre) a BERT modell számára.
- A *BertForSequenceClassification.from_pretrained* függvénnyel betöltjük az előre betanított BERT modellt, amely képes osztályozni a szöveges adatokat.

Jelen esetben azt szeretnénk meghatározni, hogy egy hír pozitív, negatív vagy semleges hatással van-e a tesztelt vállalatra.

A kódban létrehoztunk egy *sentiment_analysis* nevű függvényt, amely:

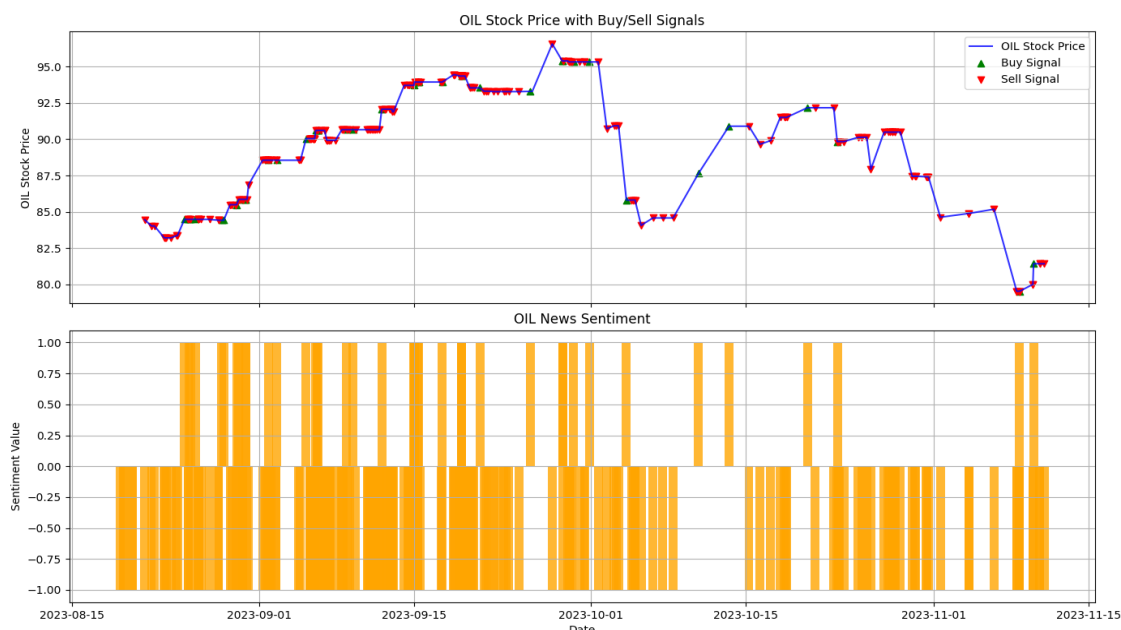
- A tokenizer segítségével tokenekre bontja a bemeneti szöveget.
- A tokenizált szöveget átadja a BERT modellnek, amely előállítja az osztályozási eredményt.
- A kapott eredmény alapján döntést hozunk a sentiment értékéről (-1: negatív, 0: semleges, 1: pozitív).

Ezen függvény segítségével automatikusan meghatározhatjuk a hírek érzelmi hordalékát anélkül, hogy manuálisan olvasnánk és értékelnénk őket (14. ábra).

Date: 2023-11-11 00:00:00, Title: Цената на руския петрол Уралс се върна малко над наложения от Г - 7 таван, Sentiment: -1
Date: 2023-11-10 14:15:00, Title: Barclays lowers 2024 Brent oil price forecast to \$93 / b on demand concerns Hellenic Shipping News Worldwide, Sentiment: -1
Date: 2023-11-10 14:15:00, Title: أسعار النفط تعاود الارتفاع عالمياً, Sentiment: -1
Date: 2023-11-09 23:30:00, Title: Petróleo Brent ultrapassa US\$80 após recuar nesta semana, Sentiment: -1
Date: 2023-11-09 23:30:00, Title: 저가 매수세에 3일만에 반등 ... 브렌트유 80달러 되찾아, Sentiment: -1
Date: 2023-11-08 18:15:00, Title: Brent crude oil price forecast : technical analysis, Sentiment: 1
Date: 2023-11-08 17:30:00, Title: Petrolio (Brent) giù a \$80 : al minimo da luglio . A quando lo stop della discesa ? , Sentiment: -1

14. ábra A program által generált szentiment értékek (Forrás: saját szerkesztés)

A hírek program szerinti elemzés a 15. ábrában található. A lekérdezést a szolgáltató csak rövid időtávon biztosítja, ezáltal a visszatesztelés csak rövid időtávon lehetséges.



15. ábra A szentiment elemzés eredménye (Forrás: saját szerkesztés)

A profit-elemzési metódust, az előző részben használt struktúra szerint végeztem, amely a szentiment-értékek alapján próbálja meghatározni a potenciális profitot. Ehhez néhány feltételezésem:

Vételi jelzés esetén 100.000 USD értékben veszünk Brent olaj részvényeket.

Eladási jelzés esetén az összes részvényt eladjuk.

Ezután összehasonlítjuk a szentiment alapú stratégia profitját azzal, ha egyszerűen csak tartottuk volna a részvényeket az elemzett időszak alatt (ez lesz a benchmark). Továbbá, az egyszerűség érdekében nem vesszük figyelembe a tranzakciós költségeket és egyéb díjakat. A program kimenté a 16. ábrán látható.

Profit gained from the Sentiment Analysis Strategy by investing \$100k in OIL : \$9725.55
 Profit percentage of the Sentiment Analysis Strategy : 9.72%
 Benchmark profit by investing \$100k : \$-3587.49
 Benchmark Profit percentage : -3.58%
 Sentiment Analysis Strategy profit is 13.80% higher than the Benchmark Profit

16. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

Az eredmények alapján látható, hogy a szentiment alapú stratégia jobban teljesített, mint a benchmark stratégia (amely egyszerűen csak tartja a részvényeket az elemzett időszak alatt).

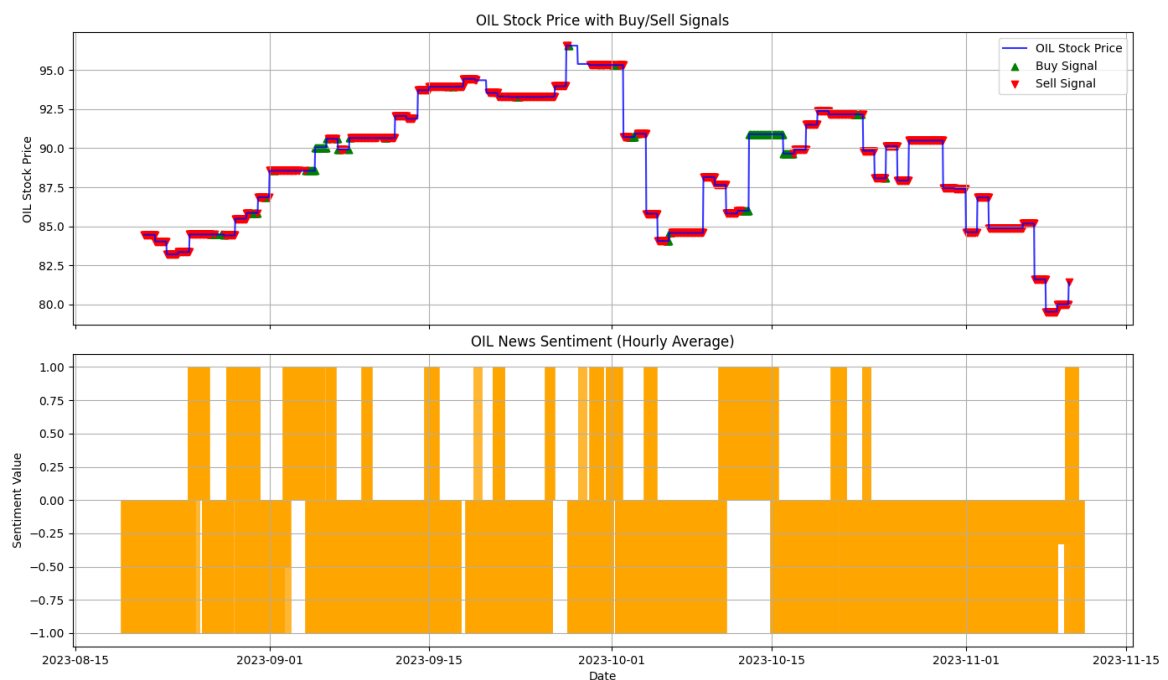
Ez azt mutatja, hogy a hírek szentimentjének figyelembevétele értékes információkat nyújthat a befektetési döntésekhez. Azonban fontos megjegyezni, hogy ez a stratégia egy adott időszakra és egy adott részvényre vonatkozik.

8.6. OPTIMALIZÁLÁSI LEHETŐSÉGEK

8.6.1. Szentimentek átlagolása

Az ötlet, hogy az összegyűjtött szentiment értékekből óránkénti átlagot készítünk, mely segíthet csökkenteni a zajt és stabilabb jeleket generálhatunk. Jelenleg a stratégia vételt és eladást jelez minden pozitív vagy negatív szentiment érték esetén (17. ábra).

```
sentiment_df = pd.DataFrame({'timestamp': sorted_timestamps, 'sentiment': sorted_sentiments})
sentiment_df.set_index('timestamp', inplace=True)
hourly_sentiments = sentiment_df.resample('1H').mean().ffill()
```



17. ábra A szentiment elemzés eredménye (Forrás: saját készítésű ábra)

Profit gained from the Sentiment Analysis Strategy by investing \$100k in OIL : \$8635.80

Profit percentage of the Sentiment Analysis Strategy : 8.63%

Benchmark profit by investing \$100k : \$-3587.49

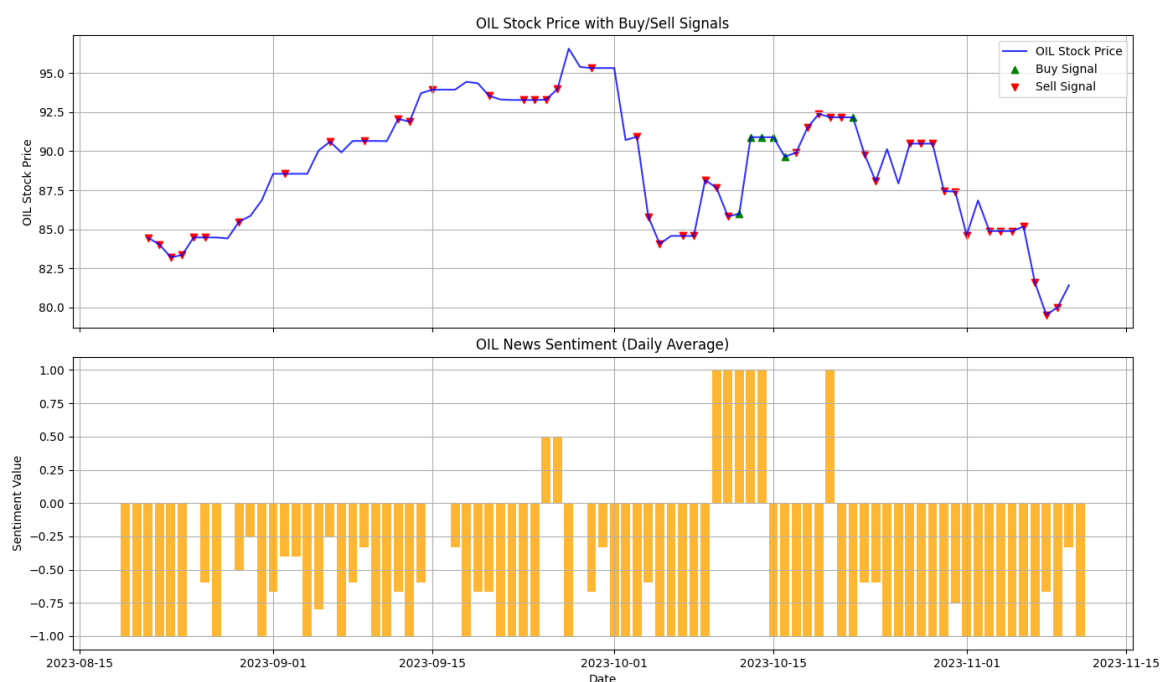
Benchmark Profit percentage : -3.58%

Sentiment Analysis Strategy profit is 12.67% higher than the Benchmark Profit

18. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

A módosított stratégia nem teljesített jobban, mint az előző. Az eredmények alapján (18. ábra) az új szentiment alapú stratégia 8,63%-os profitot ért el, míg a benchmark stratégia 3,58%-os veszteséget termelt. A kevesebb profit kevesebb kockázattal is jár, mert az átlagolással az esetleges zajokat ki tudjuk szűrni.

Naponkénti átlag: A 19-20. ábrákon a napi átlagolás hatását vizsgálhatjuk, melyben a zaj nagyon csekély mértékben van jelen, a kockázat is alacsony szinten van, de a teljesítmény is ennek megfelelően csökken.



19. ábra A szentiment elemzés eredménye (Forrás: saját szerkesztés)

Profit gained from the Sentiment Analysis Strategy by investing \$100k in OIL : \$1889.57

Profit percentage of the Sentiment Analysis Strategy : 1.88%

Benchmark profit by investing \$100k : \$-3587.49

Benchmark Profit percentage : -3.58%

Sentiment Analysis Strategy profit is 5.68% higher than the Benchmark Profit

20. ábra A program által generált haszonelemzés (Forrás: saját szerkesztés)

A különböző szentiment átlagolási módszerek eredményei közötti eltérések a következő okokra és kockázatokra utalhatnak:

Átlagolás nélküli szemlélet:

- **Nagyobb Reakció:** Amikor minden egyes hírcímet külön-külön értékelünk, a kereskedési stratégia gyorsabban reagál a piaci hangulat változásaira. Ez lehetővé teszi a gyors mozgások kihasználását.
- **Több kereskedési jelzés:** Több vételi és eladási jelzés keletkezik, ami több tranzakciót és potenciálisan nagyobb nyereséget eredményezhet.
- **Nagyobb kockázat:** Azonban ez a módszer növelheti a „fals pozitív” vagy „fals negatív” jelek számát, ami felesleges vagy rossz időzítésű tranzakciókhoz vezethet.

Óránkénti átlagolás:

- **Kiegyensúlyozottabb stratégia:** Az óránkénti átlagolás kisimítja a szentiment ingadozásait, ami kevesebb, de átgondoltabb kereskedési jelet eredményez.
- **Csökkentett zaj:** Az óránkénti átlagolás csökkenti a hírek "zaját", ami kiküszöböli a rövid távú, irreleváns ingadozásokat.
- **Lassabb reakcióidő:** Ez a megközelítés lassabb reakciót jelent a piaci változásokra, ami lemaradást okozhat a gyorsan változó piaci helyzetekben.

Napi átlagolás:

- **Nagyon simított adatok:** A napi átlagolás még tovább simítja a szentiment adatokat, ami nagyon stabil, de talán túlságosan óvatos stratégiát eredményez.
- **Csökkentett reakcióképesség:** A napi átlagolás jelentősen csökkenti a rendszer reakcióképességét, így a fontos piaci eseményekre való reagálás lassú lehet.
- **Alacsonyabb tranzakciószám:** Kevesebb tranzakció jelentkezik, ami csökkenti a tranzakciós költségeket, de lehetőségeket is veszít.

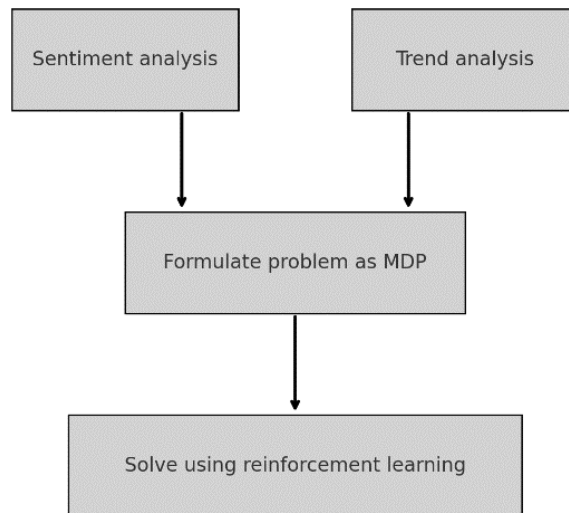
Következtetések és kockázatok

- **Túlzottan reagáló stratégia:** Az átlagolás nélküli megközelítés túlzottan reagálhat a piaci zajra, ami felesleges kereskedési döntésekhez vezethet.
- **Elmaradás a piaci változásoktól:** A napi átlagolás jelentősen lassítja a reakcióidőt, ami lemaradást okozhat a fontos piaci eseményekre való reagálásban.
- **Kereskedési költségek:** A több tranzakció magasabb tranzakciós költségeket jelenthet.
- **Piaci zaj és valós információk:** A piaci zaj és a valós, fontos információk megkülönböztetése kulcsfontosságú minden stratégiában.

Ezek az eredmények azt mutatják, hogy a különböző átlagolási módszerek jelentősen befolyásolhatják a kereskedési stratégia teljesítményét, és fontos megtalálni az egyensúlyt a gyors reagálás és a stabil, jól megalapozott döntések között.

8.6.2. Egyéb indikátorok hozzáadása

Az elemzési módszerek továbbfejlesztésére azáltal is van lehetőség, hogy kiegészítjük a jelenlegi modellt további indikátorokkal. Az alapvető és technikai jelzők kombinálásával még pontosabbá és megbízhatóbbá tehetjük az előrejelzéseinket. Például, ha az előző fejezetben kifejlesztett kombinált stratégiát integráljuk a jelenlegi elemzési modellünkbe, komplexebb és sokoldalúbb rendszert alkothatunk. Az MDP (Markov Döntési Folyamat) alkalmazása pedig egy újabb dimenziót adhat a modellünknek, lehetővé téve a pénzügyi trendek és mozgások még pontosabb előrejelzését. Ezen túlmenően, a modellezés során fontos a megfelelő aktivációs funkció kiválasztása: bináris döntések alkalmazása esetén a szigmoid függvény, míg többsztályos döntések során a softmax függvény ideális választásnak bizonyulhat. Az összetett modell blokkvázlatát a 21. ábrán látható.



21. ábra A komplex rendszer vázlata (Forrás: saját szerkesztés)

9. KÖVETKEZTETÉSEK ÉS JAVASLATOK

A kutatásom alapján megállapítható, hogy az MI, a technikai- és a szentimentális elemzés együttes alkalmazása jelentős potenciált hordoz a tőzsdei kereskedésben. Az MI képessége a bonyolult kapcsolatok, mint a piaci adatok, közösségi média bejegyzések és hírek elemzése révén, lehetővé teszi a kereskedési döntések pontosabbá és megalapozottabbá tételét. A technikai mutatók és neurális hálózatok kombinálása modern, hatékony módszert jelent a piaci trendek azonosítására és a kereskedési stratégiák fejlesztésére. Saját készítésű Python programjaim segítségével rugalmasan és pontosan tudtam azonosítani a piaci dimenziókat, melyek hozzájárultak a kereskedési jelek hatékony generálásához.

Technológiai határok és fejlődési lehetőségek: A technológia jelenlegi korlátai ellenére számos lehetőség kínálkozik a továbbfejlődésre. A kutatás rámutatott, hogy az MI, különösen a neurális hálózatok, képesek felismerni és kiaknázni a piaci adatokban rejlő bonyolult mintázatokat és összefüggéseket.

Kockázatok és eljárások: A kutatás során azonosítottam és feltártam azokat a legjobb gyakorlatokat és eljárásokat, amelyek segíthetnek az MI által kínált előnyök maximalizálásában, miközben minimalizálják a kockázatokat. Ez hozzájárulhat a tőzsdei kereskedési folyamatok és stratégiák további optimalizálásához.

Technikai elemzés és neurális hálózatok együttes alkalmazása: A tanulmány kiemeli a technikai elemzés és a neurális hálózatok együttes alkalmazásának előnyeit, különösen az ártrendek, lendület, volatilitás és forgalom analízisében.

Egyedi python programok hatékonysága: A saját készítésű Python programok használata lehetővé teszi a kutatási célokhoz való rugalmas igazodást és az algoritmusok optimalizálását.

Hírek és részvényárak közötti kapcsolat: A hírek és részvényárfolyamok közötti kapcsolatok feltárása új lehetőségeket nyit meg a piaci trendek előrejelzésében.

A dolgozatomban arra mutat rá, hogy a jövőbeli kereskedési stratégiák fejlesztésében és optimalizálásában nagy szerepet játszhat az MI és a korszerű adatelemzési módszerek. Ezek a technológiák jelentős előnyöket kínálhatnak a tőzsdei kereskedésben, de fontos, hogy a kockázatokat is figyelembe vegyük és kezeljük. Az MI képessége arra, hogy összetett mintázatokat azonosítson és előrejelzéseket tegyen a nagy adathalmazokból, potenciálisan forradalmasíthatja a piaci elemzést és döntéshozatalt. Ugyanakkor az ilyen rendszerek alkalmazása során szükség van a pénzügyi piacok mélyebb megértésére és a modellek folyamatos ellenőrzésére és finomhangolására.

A kutatásom alapján az alábbi javaslatokat fogalmazom meg:

Kockázatkezelési stratégiák fejlesztése: Fontos, hogy a fejlett MI-alapú kereskedési rendszerek mellett kifinomult kockázatkezelési stratégiákat is kifejlesszünk. Ezeknek a stratégiáknak figyelembe kell venniük a piaci volatilitást, likviditási kockázatot és a rendszer belső hibáinak lehetőségét.

Adatvédelmi és biztonsági megfontolások: A személyes adatok védelme és a kiberbiztonsági fenyegetések kezelése elengedhetetlen, különösen amikor a kereskedési döntéseket közösségi média és más online források adatai alapján hozzuk meg.

Egyensúly a modellezés és intuíció között: Miközben az MI-alapú modellek hatékonyak lehetnek, fontos, hogy ne hagyatkozzunk kizárólag ezekre. Az emberi intuíció és a piaci tapasztalat továbbra is kulcsfontosságú szerepet játszik a kereskedési döntésekben.

További adatforrások integrálása: Az MI alapú rendszerek további fejlesztéséhez érdemes lenne bevonni további adatforrásokat, mint például gazdasági jelentések és piaci indikátorok, a még átfogóbb elemzés érdekében.

Real-time adatelemzés: Az élő adatfolyamok elemzésének beépítése lehetővé teheti a kereskedési rendszerek további finomhangolását és a piaci változásokra való gyorsabb reagálást.

További integráció és innováció: A technológia fejlődésével újabb lehetőségek nyílhatnak meg a piaci adatok még részletesebb elemzésére, ezeket folyamatosan szem előtt kell tartani.

Fejlesztés és tesztelés: A kereskedőknek folyamatosan fejleszteni és tesztelni kell a saját kereskedési algoritmusokat és stratégiáikat, hogy megfeleljenek a piaci változásoknak és a technológiai fejlődésnek.

Oktatás és kutatás: Az oktatási és kutatási intézményeknek javaslom, hogy bővítsék a tőzsdei kereskedés és az MI területén folytatott kutatásaikat és oktatási programjaikat, hogy jobban felkészítsék a jövő kereskedőit a modern piacok kihívásaira.

10. ÖSSZEFOGLALÁS

A dolgozatomban elemeztem, hogy a mesterséges intelligencia (MI) és a technikai elemzés, valamint a szentiment elemzés integrációjának mennyire nagy a jelentősége a tőzsdei kereskedés terén. Alaposan megvizsgáltam, hogyan segíthetnek ezek a technológiák a kereskedési algoritmusoknak abban, hogy a piaci adatokat, a közösségi média tartalmakat és a híreket elemezve értékes információkat szűrjenek ki, ezáltal növelve a kereskedési döntések hatékonyságát. Kutatásom során különös figyelmet fordítottam a technológiai innovációk és korlátozások feltárására, valamint arra, hogy milyen irányokban várható további fejlődés. Ezzel párhuzamosan törekedtem arra is, hogy feltárjam azokat a módszereket és legjobb praktikákat, amelyek kiaknázzhatják az MI nyújtotta előnyöket, miközben csökkentik a hozzá kapcsolódó kockázatokat, ezzel is elősegítve a tőzsdei kereskedési eljárások és stratégiák finomítását. A részvények jövőbeli irányainak előrejelzése a pénzügyi világban kiemelkedő jelentőségű, figyelembe véve, hogy ezek a trendek számos tényezőtől függnének. E trendek meghatározására a technikai elemzés és a neurális hálózatok összekapcsolása mutatkozott a legígéretesebbnek. A módszer, ahol a technikai mutatók közvetlenül a neurális hálózatok bemeneteként szolgálnak, lehetőséget teremt a két eszköz előnyeinek ötvöztetésére. Míg a technikai mutatók rövid- és középtávú piaci trendeket és ármozgásokat tükröznek, a neurális hálózatok a komplex összefüggések feltárásában nyújtanak segítséget.

Projektemben nem csak meglévő eszközöket vettem igénybe, hanem egyedi Python programokat is fejlesztettem az adatelemzés és a kereskedési stratégiák teszteléséhez. Ezek a személyre szabott megoldások lehetővé tették számomra, hogy pontosan és rugalmasan igazodjak a kutatási céljaimhoz, optimalizálva ezzel az algoritmusokat a legjobb eredmények elérése érdekében. A projektben azonosítottam számos technikai mutatót, amelyeket a neurális hálózat bemeneteként használtam. Ezek a mutatók különböző piaci dimenziókat, mint például ártrendet, lendületet, volatilitást és forgalmat tükröznek, és segítségükkel kereskedési jeleket generáltam.

A dolgozatban kifejtettem a hírek és a részvényárfolyamok közötti lehetséges kapcsolatokat is. Feltételezésem szerint a hírek pozitív vagy negatív tartalma számottevően befolyásolhatja a részvények mozgását. A piaci trendek előrejelzésének lehetőségeit vizsgálva arra jutottam, hogy a hírek és a múltbeli árfolyam adatok kombinált elemzése ígéretes eredményeket hozhat. Automatizált módszereket alkalmaztam a hírek érzelmének azonosítására, abból a megfontolásból kiindulva, hogy a pozitív hírek kedvezően, míg a negatív hírek kedvezőtlenül

hathatnak a piacra. A hírek érzelmének pontos elemzéséhez a BERT modellt használtam, amely alkalmas a szövegek kontextusának mélyebb megértésére a természetes nyelvfeldolgozás területén. A szentimentelemzés alapján generáltam vételi és eladási jeleket a részvény idősorán, és profítanalízist végeztem, melyből arra a következtetésre jutottam, hogy a modell jól teljesített a kísérleti mintán.

A technikai elemzések és a neurális hálózatok kombinálása modern és hatékony módszert kínál a kereskedési jelek előrejelzésére és az algoritmikus stratégiák fejlesztésére. Ezek az eszközök, kombinálva a szentiment elemzéssel, mely a közösségi média és egyéb digitális források adatait veszi alapul, új szintre emelhetik az algoritmikus kereskedést. Az integrált megközelítés, amely magába foglalja mind a technikai, mind a szentimentális elemeket, forradalmi lépést jelenthet a pénzügyi piacokon. Emellett a neurális hálózatok kimeneti rétegének aktivációs függvénye kulcsfontosságú a rendszer hatékonyságában, ahol a bináris döntések esetén a szigmoid függvény, míg a többosztályos döntések alkalmazása során a softmax függvény ideális lehet. Ezek az összetevők együttesen ígéretes jövőt kínálhatnak a kereskedőknek az eredményes döntések meghozatalában, bár ezeknek a tesztelése további kutatást igényel.

IRODALOMJEGYZÉK

1. Aldridge, I. (2013). *High-Frequency Trading: A Practical Guide to Algorithmic Strategies and Trading Systems*. John Wiley & Sons.
2. Bengio, Y. (2013). *Deep Learning of Representations: Looking Forward*. arXiv preprint.
3. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
4. Brogaard, Jonathan; Li, Dan; Yang, Jeffrey (2022). *Does High Frequency Market Manipulation Harm Market Quality?* URL: <https://ssrn.com/abstract=4280120>
5. Cervelló-Royo, R., Guijarro, F., & Michniuk, K. (2015). *Stock Market Trading Rule Based on Pattern Recognition and Technical Analysis: Forecasting the DJIA Index with Intraday Data*. Expert Systems with Applications, ScienceDirect. URL: <https://doi.org/10.1016/j.eswa.2015.03.017>
6. Chaboud, A. P., Chiquoine, B., Hjalmarsson, E., & Vega, C. (2014). *Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market*. The Journal of Finance, 69(5), 2045-2084.
7. Chlistalla, M. (2011). *High-Frequency Trading: Better Than its Reputation?* Deutsche Bank Research, Focus 09/2011, pp. 7-11; World Federation of Exchanges, Paris
8. Perez, Edgar (2011). *The Speed Traders: An Insider's Look at the New High-Frequency Phenomenon That is Transforming the Investing World*. McGraw-Hill, c2011. ISBN 978-0071768283
9. Deboeck, G. J. (1994). *Trading on the Edge: Neural, Genetic, and Fuzzy Systems for Chaotic Financial Markets*. John Wiley & Sons.
10. Hendershott, T., Jones, C. M., & Menkveld, A. J. (2011). *Does Algorithmic Trading Improve Liquidity?* The Journal of Finance, Wiley Online Library. URL: <https://onlinelibrary.wiley.com/doi/10.1111/j.1540-6261.2010.01624.x>
11. Mitchell, T. (1997). *Machine Learning: A Guide to Current Research*. [Technikai jelentés]
12. Murphy, J. (1999). *Technical Analysis of the Financial Markets*. New York Institute of Finance.
13. Nayak, M. (2019). *An Intuitive Introduction of Word2Vec by Building a Word2Vec From Scratch*. Towards AI.

14. Schölkopf, B., & Smola, A. J. (2002). *Learning with Kernels*. MIT Press.
15. Trippi, R. R., & Turban, E. (1996). *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real-World Performance*. McGraw-Hill.
16. Jansen, Stefan (2020). *Machine Learning for Algorithmic Trading: Predictive Models to Extract Signals from Market and Alternative Data*. 2nd Edition. Birmingham, UK: Packt Publishing. ISBN 978-1839217715
17. Zhang, Frank, (2010). *High-Frequency Trading, Stock Volatility, and Price Discovery*. URL: <https://ssrn.com/abstract=1691679>
18. Gatev, Evan; Goetzmann, William N.; Rouwenhorst, K. Geert (2006). *Pairs Trading: Performance of a Relative-Value Arbitrage Rule*. Review of Financial Studies, Vol. 19, Issue 3, pp. 797-827.
19. Fernandez-Rodriguez, Fernando; Gonzalez-Martel, Christian; Sosvilla-Rivero, Simon (2000). *On the Profitability of Technical Trading Rules Based on Artificial Neural Networks: Evidence from the Madrid Stock Market*. Economics Letters, Elsevier, Vol. 69, Issue 1, pp. 89-94, URL: <https://www.sciencedirect.com/science/article/abs/pii/S0165176500002706>
20. Kirilenko, Andrei A.; Kyle, Albert S.; Samadi, Mehrdad; Tuzun, Tugkan (2017). *The Flash Crash: High-Frequency Trading in an Electronic Market*. Journal of Finance, American Finance Association, Vol. 72, Issue 3, pp. 967-998, URL: <https://ssrn.com/abstract=1686004>
21. Bailey, David H.; Borwein, Jonathan; Lopez de Prado, Marcos; Zhu, Qiji Jim (2014). *Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-of-Sample Performance*. Notices of the American Mathematical Society, Vol. 61, No. 5, pp. 458-471 URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1686004
22. Huck, Nicolas (2009). *Pairs Selection and Outranking: An Application to the S&P 100*. European Journal of Operational Research, Vol. 196, Issue 2, pp. 819-825. URL: <https://doi.org/10.1016/j.ejor.2008.03.025>
23. Lopez de Prado, M. (2018). *Advances in Financial Machine Learning*. Wiley.
24. Russell, M. A. (2014). *Mining the Social Web*, 2nd Ed. O'Reilly Media.
25. Sinclair, E. (2013). *Volatility Trading*, 2nd Ed. John Wiley & Sons.
26. Wilmott, P. (2007). *Introduces Quantitative Finance*, 2nd Ed. John Wiley & Sons.

27. Arnold, G. (2011). *Financial Times Guide to the Financial Markets*. Financial Times/Prentice Hall.
28. Barber, D. (2012). *Bayesian Reasoning and Machine Learning*. Cambridge University Press.
29. James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). *An Introduction to Statistical Learning: with applications in R*. Springer.
30. Johnson, B. (2010). *Algorithmic Trading & DMA: An introduction to direct access trading strategies*. 4Myeloma Press.
31. McKinney, W. (2012). *Python for Data Analysis*. O'Reilly Media.
32. Narang, R. K. (2013). *Inside The Black Box: The Simple Truth About Quantitative*. Wiley, ISBN 978-0470432068
33. Bird, S., Klein, E., Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.
34. Bollen, J., Mao, H., Zeng, X. (2010). *Twitter mood predicts the stock market*. CoRR, abs/1010.3003.
35. Chan, E. P. (2009). *Quantitative Trading: How to Build Your Own Algorithmic Trading Business*. John Wiley & Sons.
36. Chan, E. P. (2013). *Algorithmic Trading: Winning Strategies And Their Rationale*. John Wiley & Sons.
37. Harris, L. (2002). *Trading and Exchanges: Market Microstructure for Practitioners*. Oxford University Press.
38. Hastie, T., Tibshirani, R., Friedman, J. (2011). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd Ed. Springer.
39. Pardo, R. (2008). *The Evaluation and Optimization of Trading Strategies*, 2nd Ed. John Wiley & Sons.
40. Russell, M. A. (2011). *21 Recipes for Mining Twitter*. O'Reilly Media.