**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ**

Факултет Приложна математика и информатика

**ДИПЛОМНА РАБОТА**

Тема: Прогнозиране на цени на акции с помощта на алгоритми от машинното обучение

|  |
| --- |
| *Дипломант:* Антон Златков Божидаров |

Фак. №: 181220007

*Специалност:* Приложна математика и информатика

*Образователно- квалификационна степен*: **бакалавър**

Дипломен ръководител: доц. д-р Златко Захариев

София 2024

# Съдържание

**Съдържание……………………………………………………………………………………1**

**Глава 1. Въведение……………………………………………………………………………2**

**Глава 2. Технологии и инструменти……………………………………………..………....5**

2.1 Преглед на използваните технологии и библиотеки……………………………..5

2.2 Инструменти за разработка………………………………………………………. 6

**Глава 3. Модели…………………………………………………………………………….... 7**

3.1 Компоненти на времеви редове………………………………………………….. 7

3.2 Моделът Prophet……………………………………………………………………9

3.3 Моделът TimeGPT…………………………………………………………………11

3.4 Метрики за грешки за оценка на производителността на модела…………….. 14

**Глава 4. Набор от данни……………………………………………………………………. 15**

4.1 Използвани данни………………………………………………………………… 15

4.2 Предварителна обработка на данни……………………………………............... 16

4.3 Обзор на данните…………………………………………………………………. 17

**Глава 5. Реализация………………………………………………………………………… 19**

5.1 Изпълнение на Prophet…………………………………………………………… 19

5.1.1 Внедряване на нужните библиотеки……………………………………….. 19

5.1.2 Настройка на данните…………………………………….…………………. 20

5.1.3 Трениране на модела………………………………………………………... 20

5.1.4 Прогноза и графики…………………………………………………………. 21

5.2 Изпълнение на TimeGPT………………………………………………………… 26

5.2.1 Внедряване на нужните библиотеки………………………………………. 26

5.2.2 Трениране на модела………………………………………………………... 26

5.2.3 Прогноза и графики………………………………………………………… 27

**Глава 6. Резултати…………………………………………………………………………. 31**

6.1 Исторически срещу прогнозирани данни.......................................................... 31

6.2 Сравнение на представянето............................................................................... 33

6.3 Използване на прогнозите в реална среда......................................................... 35

**Глава 7. Заключение………………………………………………………………………. 40**

**Глава 8. Литература……………………………………………………………………….. 42**

# Глава 1. Въведение

Търговията на акции е един от ключовите аспекти на финансовите пазари, който привлича вниманието на множество инвеститори и търговци. С развитието на технологиите и научните подходи, възниква въпросът за възможността за прогнозиране на цените на акциите, което би предоставило предимство в инвестиционните стратегии и вземането на решения.Прогноза, която може да показва бъдещето движение на цена на акция, води до по-добро управление на риска и по-високи финансови печалби. Целта на тази дипломна работа е да изследва възможностите за прогнозиране на цените на акции, използвайки алгоритми от машинното обучение, в частност модели за времеви редове. Основната цел е да се разгледат модели, които могат да предсказват бъдещи ценови движения на акции, на база исторически данни, да се провери ефективността им върху различни акции и дали могат прогнозите от тези методи да бъдат уред за вземане на правилни инвестиционни решения в реална среда.

Акцията е финансов инструмент, който представлява собственост или дял от капитала на дадена компания. Придобивайки акции на дадена компания, инвеститорите стават собственици на част от тази компания. Търговията с акции се осъществява на борсови площадки или чрез електронни търговски платформи, където инвеститорите купуват и продават акции. Цената на акциите се определя от търсенето и предлагането на пазара. Инвеститорите търгуват с акции с цел да реализират печалба от разликата между покупната и продажната цена.

Машинното обучение, е подгрупа от изкуствен интелект, който се фокусира върху разработването на компютърни алгоритми, които се подобряват автоматично чрез опит и чрез използване на данни, то позволява на компютрите да се учат от данни и да вземат решения или прогнози, без да са изрично програмирани да го правят. В основата си машинното обучение е свързано със създаването и внедряването на алгоритми, които улесняват тези решения и прогнози.

Тези алгоритми са предназначени да подобрят ефективността си с течение на времето, като стават по-точни и ефективни, когато обработват повече данни. В традиционното програмиране компютърът следва набор от предварително дефинирани инструкции за изпълнение на задача. При машинното обучение обаче на компютъра се дава набор от данни и задача, която трябва да изпълни, но зависи от компютъра да разбере как да изпълни задачата въз основа на дадените му примери.

Моделите за времеви редове са подмножество на общите модели в машинното обучение и представляват набор от точки от данни, подредени във времето, където времето е независимата променлива. Моделите за времеви редове се използват за работа с времеви данни и целите са анализ, прогнозиране или моделиране на тези данни въз основа на времето.

Времеви редове могат да бъдат приложени към всеки набор от променливи, които се променят във времето. За цените на акции, времевите редове са обичайни за проследяване във времето. Това може да се проследи в краткосрочен план, като например цената на акцията от момента на отваряне до затваряне на работния ден или цената на затваряне на ежедневен работен ден или последния ден на всеки месец през последните години. Сезонната тенденция и измененията са връхната точка на фондовия пазар.

Основният фокус е върху прогнозирането на цените на различни видове акции, използвайки дневната цена на затваряне и идентифицирани тенденции и модели в данните, използвайки два различни алгоритъма. Приложените алгоритми са Prophet на Facebook и TimeGPT на Nixtla. Двата модела представляват инструменти за прогнозиране на времеви редове, но използват различни подходи и технологии. Prophet се основава на традиционните статистически методи и е по-лесен за настройка и използване, докато TimeGPT използва модерни методи от областта на машинното обучение и предлага по-голяма гъвкавост и точност при работа със сложни данни. Тяхната ефективността се определя количествено с помощта на средна абсолютна грешка (MAE), Средна абсолютна процентна грешка (MAPE) и средна квадратна грешка (RMSE). Тези показатели ни показват колко точни са прогнозите и каква е степента на отклонение от действителните стойности.

Тази дипломна работа обхваща цените на различни видове акции, като се разглеждат акции от две категории на компании, базирани на тяхната пазарна капитализация, която представлява общата стойност на всички издадени акции на дадена компания. Тези категории са mega-cap и small-cap компаниите. Прогнозите се правят за акциите на Tesla (TSLA), Apple (AAPL) и Microsoft (MSFT), които са примери за mega-cap компании и CarParts.com(PRTS), ACM Research (ACMR) и Perion Network (PERI), които са спадат към small-cap компаниите.

Mega-cap компаниите представляват най-големите компании на пазара с пазарна капитализация над 200 милиарда долара, те са утвърдени лидери в своите индустрии и имат глобално присъствие. Mega-cap компаниите са стабилни и по-малко податливи на икономически колебания, което ги прави по-ниско рискови инвестиции, като инвеститорите търсят в тях сигурност и дивиденти, въпреки че темпът на растеж може да бъде по-нисък в сравнение с по-малките компании. Small-cap компаниите представляват малки компании с пазарна капитализация между 300 милиона и 2 милиарда долара, те са новосъздадени фирми или такива, които оперират в специфични ниши. Small-cap компаниите често имат потенциал за бърз растеж и значително увеличаване на стойността, но също така са по-рискови инвестиции.

# Глава 2. Технологии и инструменти

## 2.1 Преглед на използваните технологии и библиотеки

1. **Python:** е един от най-популярните езици за програмиране. Python е лесен за интерпретиране обектно-ориентиран език за програмиране от високо ниво с лесен за четене синтаксис. Поради повсеместното си разпространение и възможността да работи на почти всяка системна архитектура, Python е универсален език, намиращ се в множество различни приложения, включително уеб приложения, автоматизация, софтуер, data science, машинно обучение и AI.
2. **Pandas:** е библиотека на Python, която позволява работа с набори от данни за почистване, изследване и анализиране. Библиотеката предоставя синтаксис на високо ниво, който позволява работа с бързи и гъвкави структури от данни. Pandas е предназначен да работи с всяка индустрия, включително с финанси, статистика, социални науки и инженерство.
3. **Numpy:** е основният пакет за научни изчисления в Python. Numpy е библиотека на Python, която предоставя обект с многоизмерен масив, маскирани масиви и матрици и набор от рутинни процедури за бързи операции с масиви, като математически и логически.
4. **Matplotlib:** е популярна библиотека на Python за начертаване на графики. Използва се за създаване на статични, анимирани и интерактивни визуализации. Matplotlib позволява създаването на интерактивни фигури, които могат да бъдат мащабирани, актуализирани и да бъдат персонализирани визуалния стил и оформление.
5. **Io**: предоставя инструменти за работа с различни типове входно-изходни операции, включително работа с файлове в паметта.
6. **Os**: предоставя инструменти за взаимодействие с операционната система, включително работа с файловата система и изпълнение на команди на системно ниво.
7. **Sys**: предоставя функции и променливи, които се използват за манипулиране с интерпретатора на Python и неговата среда.
8. **Setuptools**: е библиотека за създаване, разпространение и инсталиране на Python пакети.
9. **Tokenize**: е модул в Python, който предоставя функции за раздробяване на текст на Python код на токени - най-малките единици, като ключови думи, идентификатори, символи.
10. **Sklearn metrics:** е модул в Scikit-Learn, който прилага функции, оценяващи грешка при прогнозиране. Позволява да се оцени качеството на прогнози за различни набори от данни, функции за оценка и показатели за ефективност.
11. **Yfinance:** е библиотека, която предоставя достъп до финансови данни, предоставени от Yahoo Finance. Пакетът позволява на потребителите да пишат команди на Python и да изтеглят широк набор от финансови данни на компаниите, като финансови коефициенти и стойности.

## 2.2 Инструменти за разработка

1. **Среда за разработка (IDE):** Jupyter Notebook е инструмент за интерактивно разработване и представяне на проекти за data science. Jupyter Notebooks е част от проекта IPython. Името Jupyter идва от основните поддържани езици за програмиране, които поддържа - Julia, Python и R. Jupyter се доставя с ядрото IPython, което позволява написването на програмите на Python, като в момента има над 100 други ядра, които могат да се използват. Той комбинира код, визуализации и текст, за да бъде работният процес улеснен. Jupyter Notebook е документ, който безпроблемно съчетава код и неговото изпълнение. Той позволява да се изпълнява код, показва резултатите и да добавя обяснения, формули и диаграми на едно място. Това прави работата по-прозрачна, разбираема и възпроизводима. Jupyter Notebook е неизменна част от работния процес в компании и организации по целия свят, като позволява на data science специалистите да изследват данни, да тестват хипотези и да споделят ефикасно разсъжденията си.
2. **Платформа за проверка на резултати:** TradingView е социална медийна мрежа, платформа за анализ и мобилно приложение за търговци и инвеститори. TradingView е цялостна платформа за графики за анализ на финансови пазари, която предоставя набор от инструменти за технически и фундаментален анализ, включително графики, индикатори и предупреждения, които могат да се персонализират. Предлага се социална общност с функции за споделяне на идеи, обучение и работа в мрежа с търговци по целия свят. TradingView поддържа различни пазари и класове активи, обслужвайки различни видове търговци.

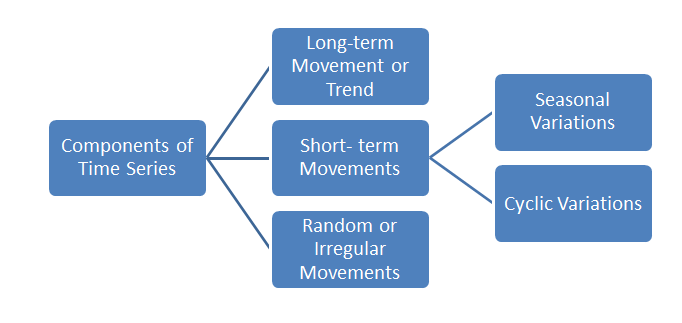
# Глава 3. Модели

## 3.1 Компоненти на времеви редове

Различните причини или сили, които влияят на стойностите на едно наблюдение във времеви редове, са компонентите на времеви редове. Четирите категории на компонентите на времеви редове са:

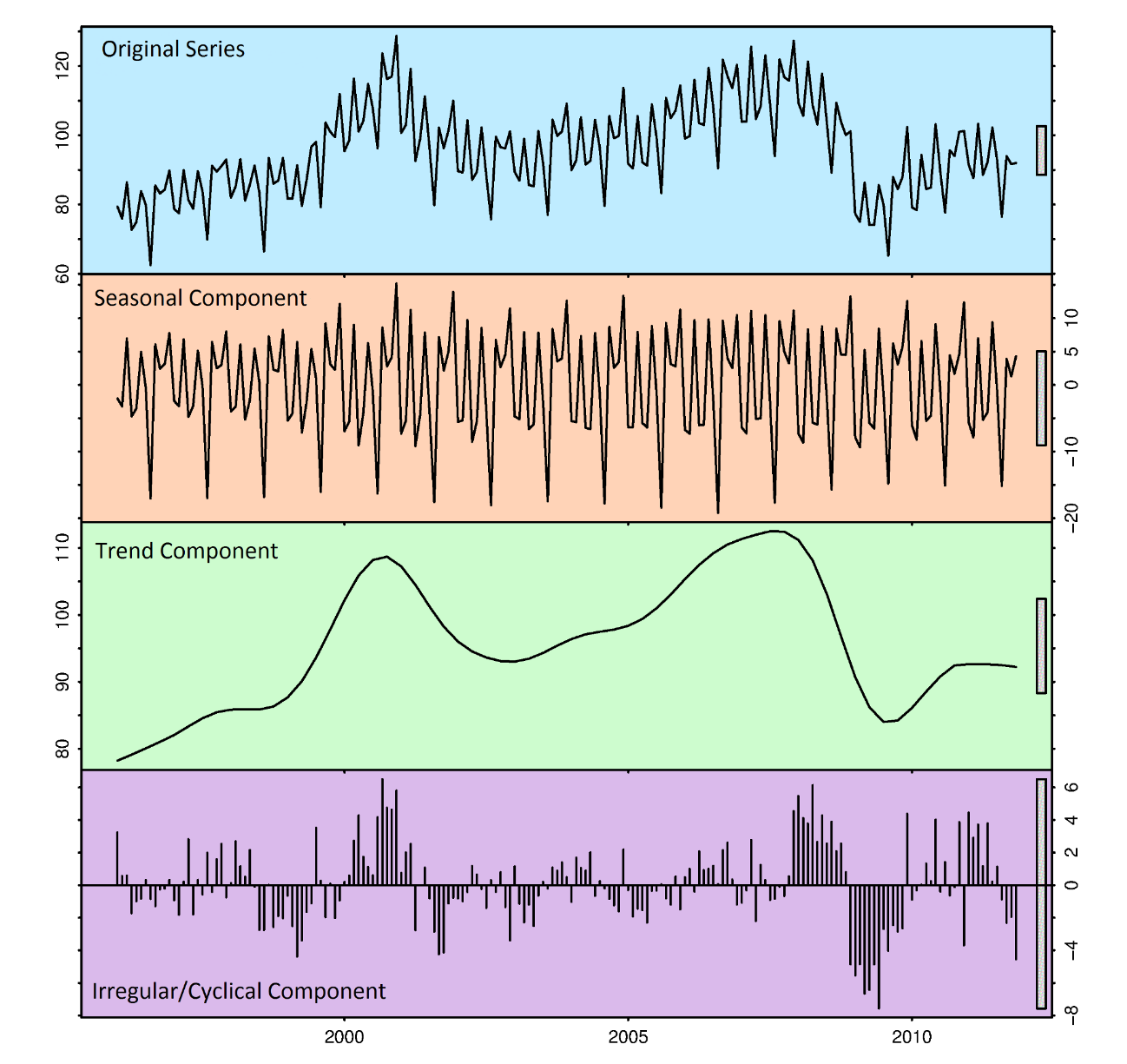
* Тренд
* Сезонност
* Цикличност
* Случайни или неправилни движения

[1] Сезонните и цикличните вариации са периодични промени или краткосрочни колебания.



*Фигура 1:* *Компонентите за анализ на времеви редове.*

1. **Тренд:** отнася се до дългосрочно движение нагоре или надолу в данните, което показва общо увеличение или намаляване с течение на времето. Трендът представлява основната структура на данните, като улавя посоката и големината на промяната за по-дълъг период.
2. **Сезонност:** отнася се до модели, които се повтарят през определен период от време, като ден, седмица, месец или година. Тези модели възникват поради редовни събития, като празници, уикенди или смяна на сезоните, и могат да присъстват в различни видове данни от времеви редове, като продажби, време или цени на акции. Има няколко вида сезонност в данните за времеви редове:
   * Седмична сезонност
   * Месечна сезонност
   * Годишна сезонност
   * Празнична сезонност
3. **Цикличност:** отнася се до повтарящи се модели или периодични колебания, които се появяват в данните за определен интервал от време. Може да се дължи на различни фактори като сезонност (ежедневна, седмична, месечна, годишна), тенденции и други основни модели.
4. **Случайни или неправилни движения:** представляват нередовни вариации и са чисто случайни или нередовни. Тези колебания са непредвидени, неконтролируеми, непредсказуеми и непостоянни. Примери за случайни или неправилни движения са пандемии, земетресения, войни, наводнения, глад и всякакви други бедствия.

*Фигура 2: Визуализация на компонентите за анализ на времеви редове.*

Компонентите на времевите редове са много важни за анализиране на променливата, която представлява интерес, за да се разбере поведението му, какви модели има и да се избере и напасне подходящ модел на времеви редове.

## 3.2 Моделът Prophet

Prophet е процедура за прогнозиране на данни от времеви редове, базирана на адитивен модел, при който нелинейните тенденции се съгласуват с годишна, седмична и дневна сезонност, плюс празнични ефекти. Работи най-добре с времеви редове, които имат силни сезонни ефекти и няколко сезона исторически данни. Prophet е устойчив на липсващи данни и промени в тенденцията и обикновено се справя добре с отклоненията.

e Prophet Model. e Prophet model is a model

developed by Facebook for large-scale time series analysis,

which has an open access software package to facilitate the

implementation of model algorithms [17, 18].

rough the ET phase error data and its analysis, it can

be seen that the ﬂuctuation of ET phase error in July is

obviously periodic. e Prophet model is designed to handle

complex features in time series, which has very powerful

predictive capabilities and is especially good at handling

daily periodic data with shifts in trend [19], and thus is

suitable for the prediction based on ET measurement error.

e Prophet model is robust to missing data and handles

large outliers well, which can have good performance when

the actual operation data of ET is not complete or has

outliers. In addition, this model is also designed to have

intuitive parameters that can be adjusted without knowing

the details of the underlying model [20

e Prophet Model. e Prophet model is a model

developed by Facebook for large-scale time series analysis,

which has an open access software package to facilitate the

implementation of model algorithms [17, 18].

rough the ET phase error data and its analysis, it can

be seen that the ﬂuctuation of ET phase error in July is

obviously periodic. e Prophet model is designed to handle

complex features in time series, which has very powerful

predictive capabilities and is especially good at handling

daily periodic data with shifts in trend [19], and thus is

suitable for the prediction based on ET measurement error.

e Prophet model is robust to missing data and handles

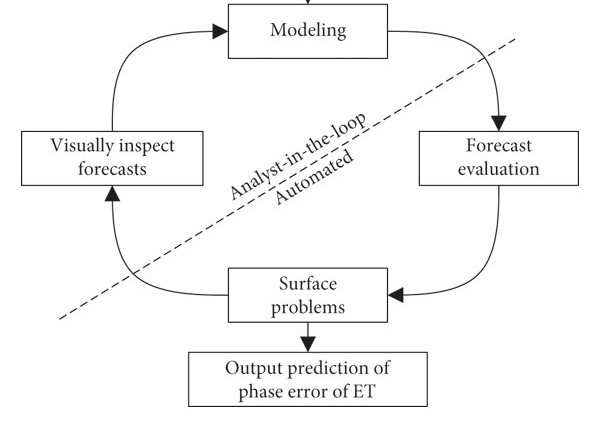
large outliers well, which can have good performance when

the actual operation data of ET is not complete or has

outliers. In addition, this model is also designed to have

intuitive parameters that can be adjusted without knowing

the details of the underlying model [20

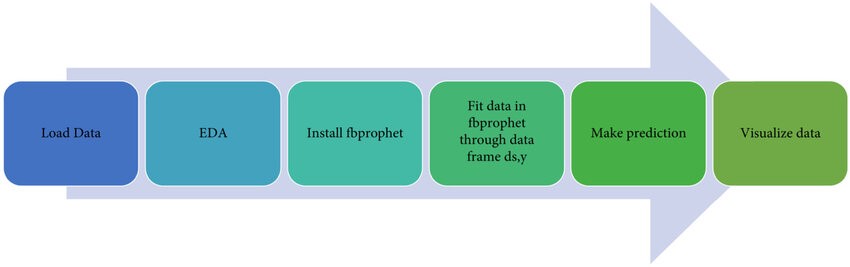
  
*Фигура 3:Поток на прогнозиране на Prophet.*

Моделът използва разложим модел на времеви редове с три основни компонента на модела: тенденция, сезонност и празници. Те са комбинирани в следното уравнение:

* y(t) = g(t) + s(t) + h(t) + εt

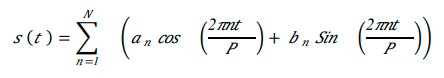
където:

* **g(t):** е функцията на тренда, която моделира непериодични промени в стойността на времевия ред.
* **s(t):** представлява периодични промени напр. седмична и годишна сезонност.
* **h(t):** представлява ефектите от празниците, които възникват при потенциално нередовни графици за един или повече дни.
* Терминът за грешка **εt** представлява всякакви идиосинкратични промени, които не са приети от модела.

  
*Фигура 4: Работен модел на Prophet*

[2]Параметърът на тренда има за задача да проследява още два параметъра, които са нарастване на наситеността и точки на промяна. Нарастването на наситеността следи колко последователни са данните, например броят на клиентите в магазин може да намалее, когато неговият конкурент отвори магазин наблизо. Параметрите на точките на промяна следят всяка внезапна промяна, която може да причини значително увеличение или понижение. Основна разлика между моделът Prophet и традиционните подходи е, че той се опитва да пасне на адитивни регресионни модели, той се опитва да пасне на кривата. Може да издържи данни, показващи годишна, месечна, седмична и дневна сезонност.

Сезонността е друг параметър, който се разглежда от Prophet. Той използва сериите на Фурие, за да осигури точен краен модел.



Тук s(t) означава сезонност, P се отнася до периода от време, който може да се вземе на месечна, седмична, дневна, тримесечна и годишна база. N се отнася до честотата на промените и параметрите aₙ, bₙ зависят от дадения N. Идентифицирането на сезонността е важно за извличане на значими наблюдения, което помага за изграждането на много по-точни модели.

В Prophet [3] една от полезните функции е, че има възможност за потребителите на модела, да приложат своите способности и информация без да имат познание по статистическите науки. Някои от основните спецификации са:

* Капацитет: Потребителите могат да използват всякаква външна информация от всеки източник за абсолютна оценка на пазара и след това находчивата информация като научено може да се приложи чрез определяне на лимити.
* Точки на промяна: Известни дати на точки на промяна, като например дати на промени в продукта, могат лесно да бъдат посочени в модела.
* Празници и сезонност: Има възможност за въвеждането на определени събития, като да се провери кои поводи оказват влияние върху моделът и да се използва като вход за съхраняване на съответните дати на поводи и съответните времеви размери на редовност или сезонност.
* Параметри на изглаждане: Променяйки по този начин стойността на тези параметри, могат да бъдат избрани по-глобални или по-локални гладки модели. Параметрите за изглаждане на сезонността и празниците (σ, ν) помагат за моделиране на количеството на историческото сезонно разнообразие, което се очаква в бъдеще.

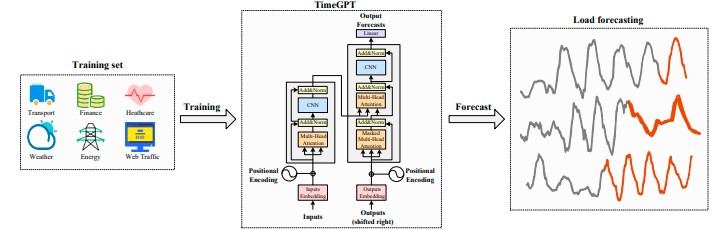
## 3.3 Моделът TimeGPT

TimeGPT е предварително обучен генеративен модел за прогнозиране на времеви серии от данни. Основната му функционалност е да прогнозира бъдещи стойности във времева серия въз основа на исторически данни. Основните характеристики и възможности на модела са:

1. Zero-shot прогноза: TimeGPT може да генерира прогнози и да открива аномалии директно, без да изисква данни за предварително обучение.
2. Фина настройка: Позволява фина настройка на модела към конкретни набори от данни, позволявайки на модела да се адаптира към нюансите на уникалните данни от времеви серии и подобрявайки производителността на персонализирани задачи.
3. Добавете на външни променливи: Включване на допълнителни променливи, като специални дати или събития , които могат да повлияят на прогнозите, за подобряване на точността на прогнозата.
4. Прогнозиране на множество серии: Едновременно прогнозиране на множество времеви серии данни, оптимизиране на работни потоци и ресурси.
5. Интервали на прогнозиране: Осигуряване на интервали в прогнозите, за да се определят ефективно количествената несигурност.
6. Нередовни времеви данни: Обработване на данни с нередовни времеви отпечатъци, приспособяване на неравномерни интервални серии без предварителна обработка.
7. Откриване на аномалии: Автоматично откриване на аномалии във времеви серии и използване на външни функции за подобрена производителност.

**Архитектура на модела:** [4], [5] TimeGPT чете данни от времеви серии подобно на начина, по който хората четат изречение – отляво надясно. Той разглежда времеви интервали с минали данни, които можем да смятаме за „жетони“, и прогнозира какво следва. Тази прогноза се основава на насоки, които моделът идентифицира в минали данни и екстраполира в бъдещето.

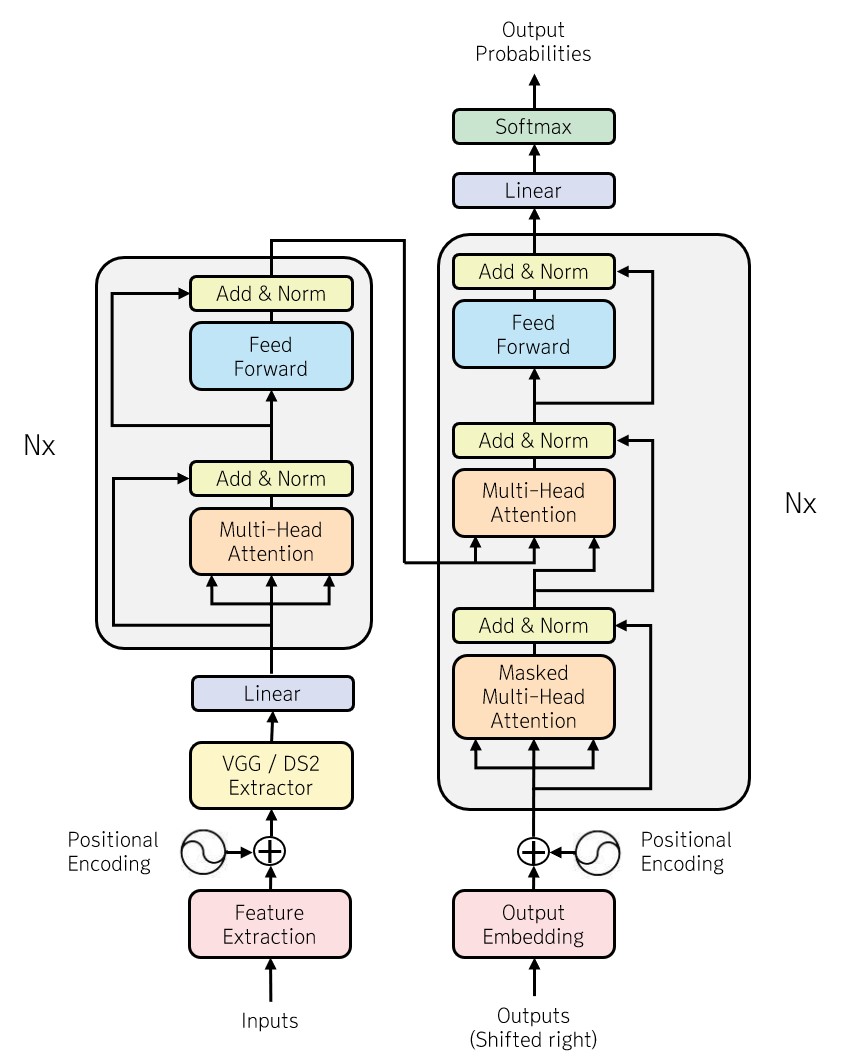
Моделът не е базиран на съществуващ голям езиков модел (LLM). Той е независимо обучен върху огромен набор от данни за времеви серии като модел на голям трансформатор и е проектиран така, че да сведе до минимум грешката при прогнозиране.



*Фигура 5:* *Обучението и използването на TimeGPT.*

TimeGPT се обучава на голям брой публично достъпни набори от данни със 100 милиарда точки от данни, като финанси, транспорт, банкиране, уеб трафик, време, енергия, здравеопазване. По отношение на времевите характеристики, данните за обучение включват различни набори с различна сезонност, цикли с различна продължителност и различни типове тенденции. В допълнение към времевите характеристики, наборът от данни показва променливост в нивата на шума и наличието на отклонения, осигурявайки стабилно условие за обучение. Някои набори от данни показват подредени, предсказуеми модели, докато други показват значителни пикове на шум или неочаквани събития, предоставяйки широк набор от сценарии за асимилация на модела

Моделът използва архитектура на енкодер-декодер с множество слоеве, включващи остатъчни връзки и нормализиране на слоевете. Изходът на декодера се съпоставя с измерението на прозореца за прогнозиране чрез линеен слой. Основната идея е, че механизмите, базирани на вниманието, ефективно улавят разнообразието от минали събития, позволявайки точна екстраполация на потенциални бъдещи разпределения.



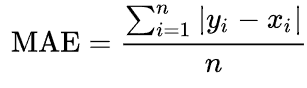
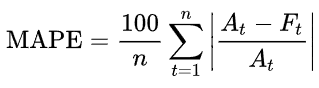
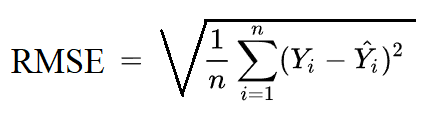
*Фигура 6:* *Основната структура на трансформаторния блок.*

Архитектурата включва главно позиционно кодиране (PE), multi-head attention и конволюционна невронна мрежа (CNN), в които остатъчната връзка и нормализирането на слоя са интегрирани, за да се предотврати влошаване на градиента и да се ускори сходимостта на алгоритъма.

Чрез използване на сегмент от исторически стойности, TimeGPT генерира прогнози, като същевременно включва местни PE за подобрено представяне на входните данни. След това изходните данни на декодера се проектират за прогнозиране на стойности чрез линеен слой. За да се запази това авторегресивно свойство, входната последователност на декодера са неговите преди това генерирани токени, но изместени с една позиция надясно.

## 3.4 Метрики за грешки за оценка на производителността на модела

[6] Измерването на производителността на всеки модел на машинно обучение е важно от техническа и от бизнес гледна точка. Особено когато бизнес решенията зависят от наблюденията, генерирани от моделите за прогнозиране. Има няколко различни показатели за ефективност, които могат да се използват за измерване на точността и ефикасността на модел за прогнозиране на времеви серии:

* Средната абсолютна грешка MAE: се определя като средната стойност на абсолютната разлика между прогнозираните и истинските стойности. Където yi е очакваната стойност, а xi е действителната стойност. n представлява общия брой стойности в тестовия набор.   
    
  MAE показва колко неточност се очаква средно от прогнозата. MAE = 0 означава, че очакваните стойности са правилни и статистиката за грешките е в оригиналните единици на прогнозираните стойности.Колкото по-ниска е стойността на MAE, толкова по-добър е моделът.
* Средна абсолютна процентна грешка (MAPE): е пропорцията на средната абсолютна разлика между прогнозираните и истинските стойности, разделена на истинската стойност. Очакваната стойност е Ft, а истинската стойност е At. Числото n се отнася до общия брой стойности в тестовия набор.  
    
  Работи по-добре с данни, които не съдържат нули и екстремни стойности поради вътрешния знаменател. Стойността на MAPE също приема екстремна стойност, ако тази стойност е изключително малка или огромна. Моделът е по-добър, ако MAPE е ниска.
* Средноквадратична грешка (RMSE): дефинира се като корен квадратен от средната квадратична грешка. Където y’ означава прогнозираната стойност, а y означава действителната стойност. Числото n се отнася до общия брой стойности в тестовия набор.   
    
  Тази статистика винаги е положителна, като по-ниските стойности показват по-висока производителност.

# Глава 4. Набор от данни

Моделите на времевите серии - Prophet и TimeGPT бяха използвани за прогнозиране на цените на акциите. Примерни данни, събрани от Yahoo Finance във формат CSV (текстови формат за таблични данни). Двата модела бяха приложени за 6 акции (TSLA, AAPL, MSFT, PRTS, ACMR, PERI), за да се сравнят алгоритмите за времеви редове.

## 4.1 Използвани данни

Наборът от данни се състои от акции, листвани на фондовата борса Nasdaq за период от 5 години, от 1 януари 2018 година до 31 декември 2023 година, като от 2018 до 2022 година данните са използвани за трениране на моделите, а данните от цялата 2023 година са използвани за тестване и оценяване на прогнозите. За по-добро разбиране на развитието на цената на акциите, разнообразни прогнози и резултати са разгледани набори от данни от различни сектора от технологичната индустрия, автомобилната и производителната индустрия, също и компании, които са с различна пазарна капитализация: mega-cap и small-cap компании. За дипломната работа примерите избрани за проучване са Tesla, Apple, Microsoft, CarParts, ACM Research и Perion Network. Данните за акциите бяха извлечени от наличните данни на Yahoo Finance чрез библиотеката yfinance в средата за разработка Jupyter Notebook.

За използване на Yahoo Finance е нужно да бъде инсталиран yfinance библиотеката в терминала на Jupyter Notebook:

* !pip install yfinance

За изтеглянето на историческите данни на 6-те акции в терминала на Jupyter Notebook се използва следния код:

import yfinance as yf

tesla = "TSLA"  
apple = "AAPL"   
microsoft = "MSFT"  
carparts = "PRTS"  
acmresearch = "ACMR"  
perionnetwork = "PERI"

start = "2018-1-1"  
end = "2023-12-31"

tesla\_data = yf.download(tesla, start = start, end = end)  
apple\_data = yf.download(apple, start = start, end = end)  
microsoft\_data = yf.download(microsoft, start = start, end = end)  
carparts\_data = yf.download(carparts, start = start, end = end)  
acmresearch\_data = yf.download(acmresearch, start = start, end = end)  
perionnetwork\_data = yf.download(perionnetwork, start = start, end = end)

## 4.2 Предварителна обработка на данни

Данните бяха почистени чрез изтриване на ненужните полета и добавяне на всички липсващи/нулеви данни:

* data.dropna(inplace=True)
* data.reset\_index(drop=True, inplace=True)

Библиотеката yfinance използва Date като индекс по подразбиране при изтегляне на данни. За да се обособи като колона Date и да бъде използвана е направено нулиране на индекса:

* data.reset\_index(inplace=True)

Също се изисква стойностите от колоната Date да бъдат преобразувани в тип дата-час:

* import pandas as pd  
  data['Date'] = pd.to\_datetime(data['Date'])

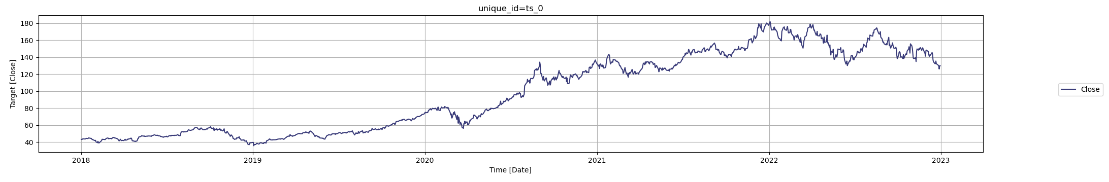
След изпълнението на кода, данните са готови за наблюдение и прогнозиране.

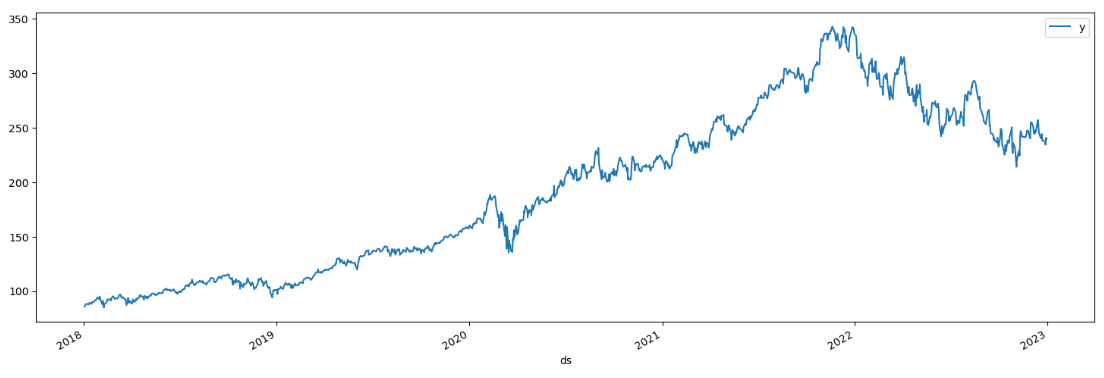
Извлечените исторически данни на акциите са дата, цена на отваряне и затваряне на акциите, най-високата и най-ниската търгувана цена, брой търгувани акции и коригирана цена. За прогнозата за бъдещите цени са разгледани полетата за дата и цена на затваряне, върху колоните Date и Close е извършено тренирането на моделите.

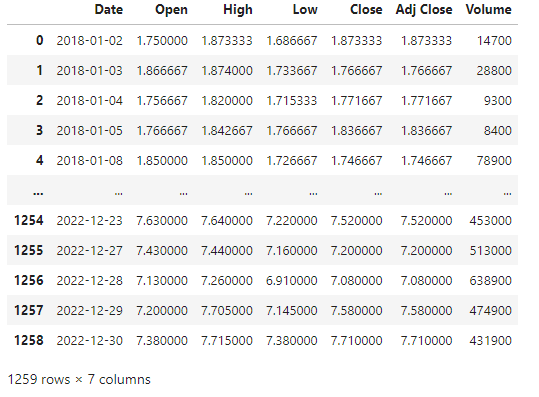
Цената на затваряне е най-важният фактор, независимо от колебанията на акциите през този ден, поради нестабилността на пазара. Тя служи като отправна точка за регулатори, търговци, инвеститори и финансови институции при определяне на ефективността за по-дълги периоди от време като години, месеци или седмици, както и за по-кратки периоди от време като минути или по-малко. Инвеститорите наблюдават цените на акциите при затваряне, за да направят преценка относно своите инвестиционни решения. В резултат на това цената на затваряне е целта за прогнозиране.

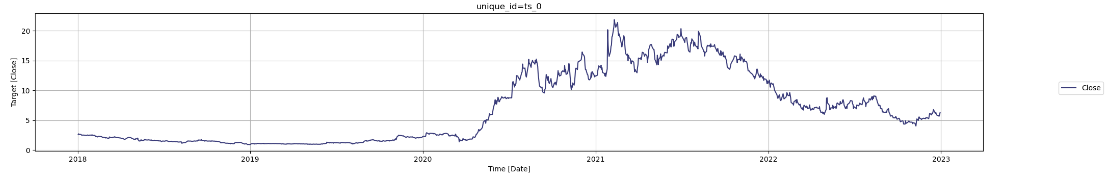
## 4.3 Обзор на данните

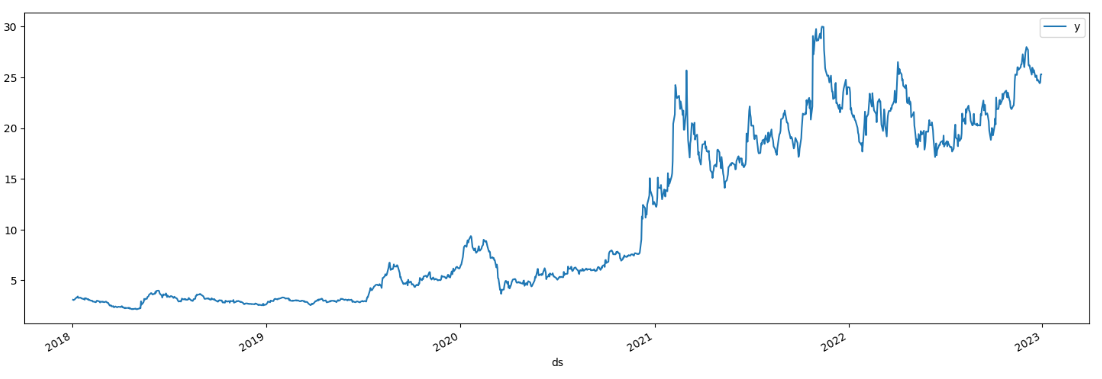
   
*Фигура 7: Рамка на историческите данни на Tesla.*

 *Фигура 8: Графика на цената на затваряне в периода от 1 януари 2018 година до 31 декември 2022 година от на apple от библиотеката на TimeGPT.*

  
*Фигура 9: Графика на цената на затваряне в периода от 1 януари 2018 година до 31 декември 2022 година на microsoft от библиотеката на Prophet.*

  
*Фигура 10: Рамка на историческите данни на ACM Research.*

*Фигура 11: Графика на цената на затваряне в периода от 1 януари 2018 година до 31 декември 2022 година на Carparts от библиотеката на TimeGPT*.

  
*Фигура 12: Графика на цената на затваряне в периода от 1 януари 2018 година до 31 декември 2022 година на Perion Network от библиотеката на Prophet.*

# Глава 5. Реализация

Двата модела се прилагат към извлечените данни на избраните шест акции, за да се получат прогнозите. След това данните и прогнозите се визуализират с помощта на графики, за да се покаже наборът от данни, което направи данните лесни за разбиране и помогна за разбирането на движенията на фондовия пазар.

Оригиналният набор от данни се разделя на тренировъчни и тестови данни. Тренировъчните данни съдържат цената на акциите от 1 януари 2018 година до 31 декември 2022 година, тестовите данни са данните, които биват прогнозирани и оценявани за една година напред – цялата 2023 година.

* train\_data = data[(data['Data'] >= '2018-01-01') & (data['Data'] <= '2022-12-31')]  
  test\_data = data[(data['Date'] >= '2023-01-01') & (data['Date'] <= '2023-12-31')]

Поради особеността с търгуването на акции, има нередовни времеви стъпки, може да се търгува само в работни дни и в определено време през деня, Nasdaq работи от 9:30 до 16:00 часа източно време (ET). Двата модела се справят по различен начин с това предизвикателство:

1. Prophet автоматично се справя с липсващи стойности, така че не е необходимо интерполиране на данните преди подаване към модела. Ако липсват стойности за определени дни, Prophet ги обработва правилно.
2. TimeGPT е оборудван да обработва този вид нередовни данни, но е нужно посочване на честотата на серията. В този случай честотата бива зададена като „B“. Без това методът може да не успее да открие автоматично честотата, особено когато клеймото за време е неправилно.

## 5.1 Изпълнение на Prophet

### 5.1.1 Внедряване на нужните библиотеки

* import pandas as pd
* import numpy as np
* import matplotlib as plt
* import yfinance as yf
* import io, os, sys, setuptools, tokenize
* from prophet import Prophet
* from sklearn.metrics import mean\_squared\_error, mean\_absolute\_error

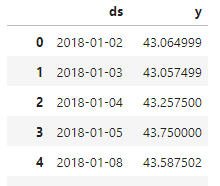
За изобразяване на използваните графики са използвани допълнителни функции от модул на prophet - prophet.plot:

* from prophet.plot import plot\_plotly
* from prophet.plot import add\_changepoints\_to\_plot

### 5.1.2 Настройка на данните

Преди трениране на модела е нужна промяна на имената на колоните Date и Close в набора от данни. Рамката от данни за прогнозиране трябва да има определен формат. Първата колона трябва да има името ds и да съдържа датите и часовете. Втората колона трябва да има име y и да съдържа наблюдаваните стойности.

* data = data[["Date","Close"]]  
  data.columns = ["ds","y"]

  
*Фигура 13: Таблица на данните от apple след промяна на имената на колоните от ‘Date’ и’ Close’ на ‘ds’ и ‘y’*

### 5.1.3 Трениране на модела

Командата Prophet() инициализира екземпляр на модела prophet за обучението на наборът от данни.

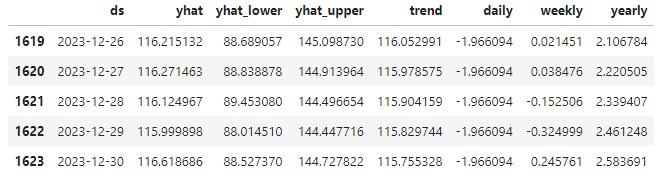
* p = Prophet(daily\_seasonality=True)  
  p.fit(train\_data)

Моделът прогнозира следващата една година. Честотата на данните е 1 ден, периодът на прогнозата е 365 дни. Бъдещата рамка от данни има само стойности на ds и трябва да предвидим стойностите на y.

* future = p.make\_future\_dataframe(periods=365)  
  data\_forecast = p.predict(future)

### 5.1.4 Прогноза и графики

След трениране на модела се разглеждат прогнозата чрез различни параметри.

  
*Фигура 14: Прогнозните стойности за акциите на apple.*

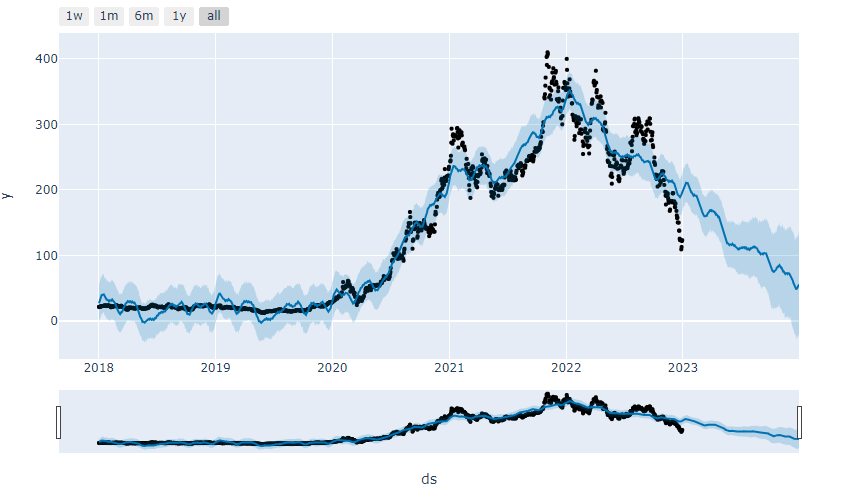
* ds: Датата, използвана за идентифициране на времевите точки.
* yhat: Основната прогноза на модела.
* yhat\_lower: Долна граница на прогнозния интервал.
* yhat\_upper: Горната граница на прогнозния интервал.
* trend: Показва дългосрочния растеж, свиване или застой на данните.
* daily: Дневна сезонност.
* weekly: Седмична сезонност.
* yearly: Годишна сезонност.

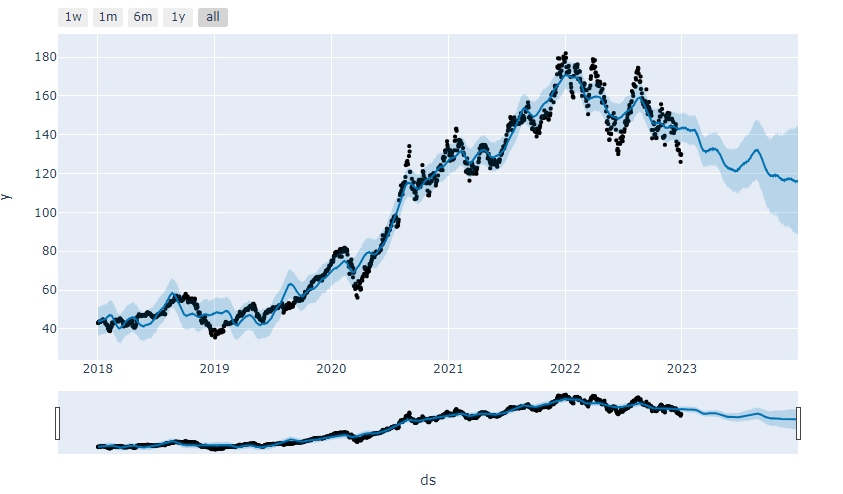
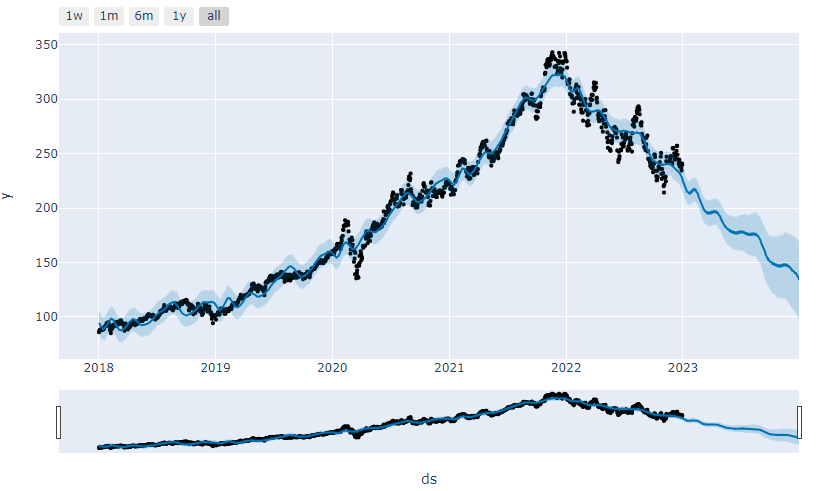
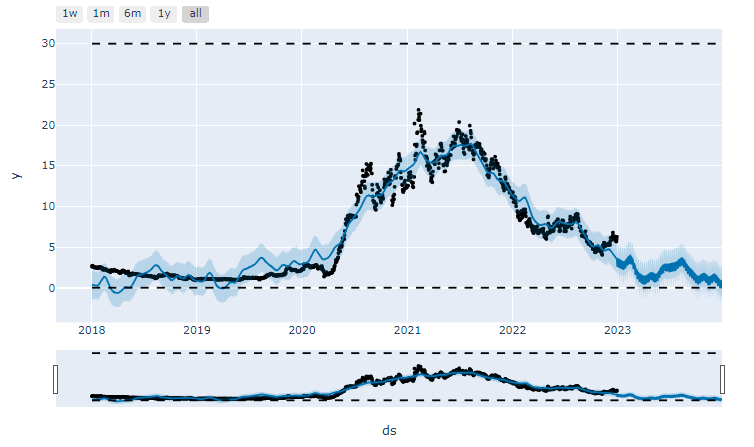
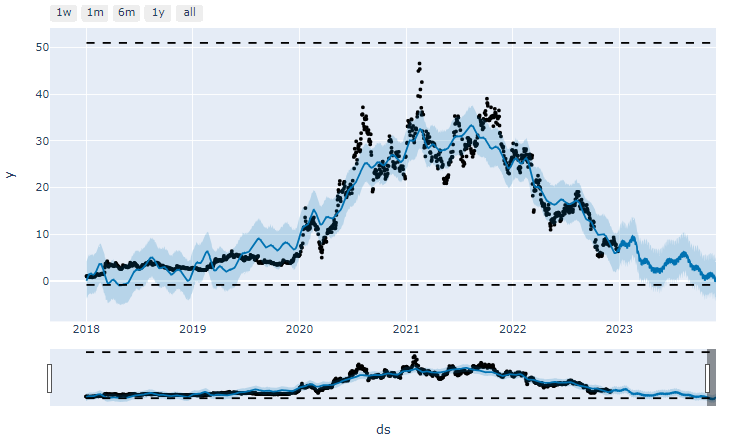
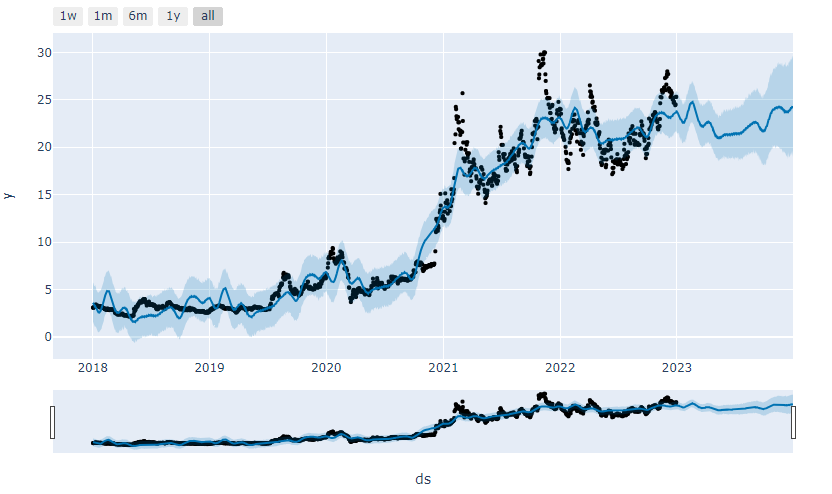
Визуализацията на историческите и прогнозираните стойности на цените на акциите е изпълнена чрез:

* plot\_plotly(p, data\_forecast)

Графиките визуализират прогнозите, трендът, и доверителните интервали. Те представляват сравнение на историческите данни с прогнозите на модела. Графиката се състои от:

1. Черни точки: представляват историческите данни.
2. Синя линия: представлява прогнозираните стойности (yhat).
3. Сива зона: представлява доверителните интервали (yhat\_lower и yhat\_upper). Тази зона показва диапазона, в който моделът прогнозира, че реалните стойности ще попаднат с определена вероятност (80% или 95%). Тя дава представа за несигурността на прогнозите.

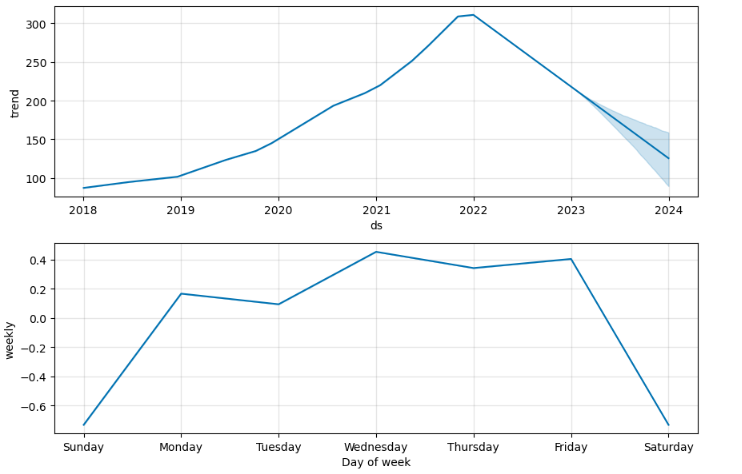
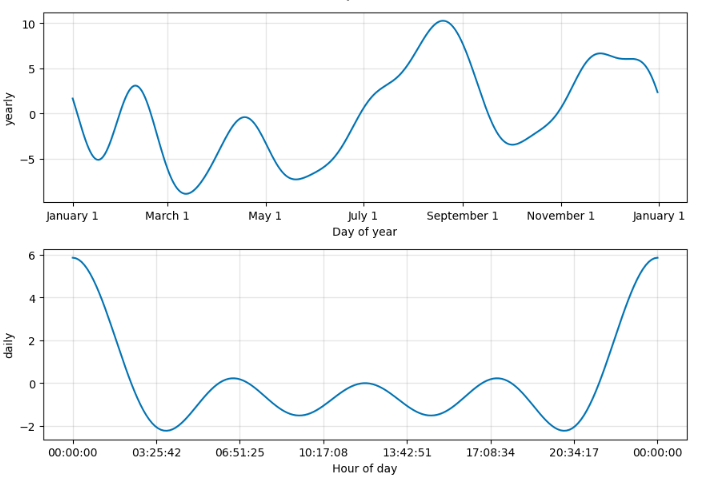
****  
*Фигура 15: Графика на прогнозата на Prophet за цената на Tesla.*

  
*Фигура 16: Графика на прогнозата на Prophet за цената на Apple.*  
 *Фигура 17: Графика на прогнозата на Prophet за цената на Microsoft.*  
  
*Фигура 18: Графика на прогнозата на Prophet за цената на Carparts.*  
*Фигура 19: Графика на прогнозата на Prophet за цената на ACM Research.*  
 *Фигура 20: Графика на прогнозата на Prophet за цената на Perion Network.*

* data\_plot = p.plot\_components(data\_forecast)

Функцията plot\_components представя различни компоненти на прогнозния модел. Тези компоненти включват:

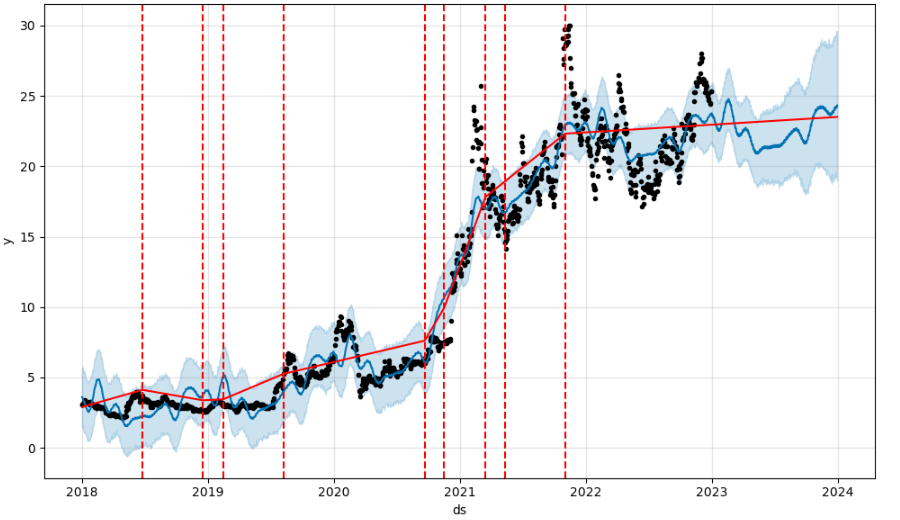
1. Тренд: показва дългосрочния тренд в данните. Трендът представлява основната посока, в която се движат данните през времето, независимо от сезонните или други повтарящи се модели.
2. Сезонност: показва сезонните модели, които се повтарят периодично:
   1. Седмична сезонност.
   2. Годишна сезонност.
   3. Дневна сезонност.

  
  
*Фигура 21: Трендът и сезонността на данните на Microsoft.*

* fig = p.plot(data\_forecast)  
  a = add\_changepoints\_to\_plot(fig.gca(), p, data\_forecast)

Функциите p.plot(data\_forecast) и add\_changepoints\_to\_plot(fig.gca(), p, data\_forecast) показват точките на промяна.

Точките на промяна са моменти във времето, където моделът идентифицира значителни промени в тренда на времевата редица. Тези точки се изобразяват като вертикални линии върху графиката. Те показват къде моделът е открил промени в наклона на тренда, което може да се дължи на събития или фактори, които са повлияли на времевият ред. Червената линия представлява тренда и неговия наклон.

  
*Фигура 22: Потенциалните точки на промени в стойностите на Perion Network.*

## 5.2 Изпълнение на TimeGPT

### 5.2.1 Внедряване на нужните библиотеки

* import yfinance as yf
* import pandas as pd
* import numpy as np
* import matplotlib as plt
* from nixtla import NixtlaClient
* from datasetsforecast.long\_horizon import LongHorizon
* from sklearn.metrics import mean\_squared\_error, mean\_absolute\_error

Нужно е предоставяне на API ключ за удостоверяване, за да се използва класа NixtlaClient.

* nixtla\_client = NixtlaClient(  
  api\_key = 'my\_api\_key\_provided\_by\_nixtla')

### 5.2.2 Трениране на модела

Тренирането на TimeGPT е изпълнено по следният начин:

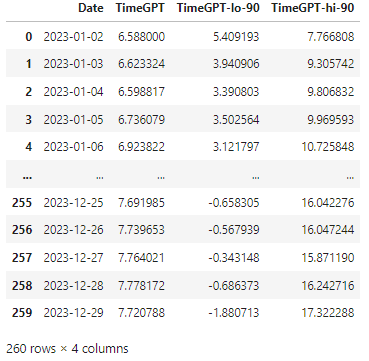
* data\_forecast = nixtla\_client.forecast(  
   df = train\_data,  
   h = len(test\_data),  
   freq = 'B',  
   level = [90],  
   model = 'timegpt-1-long-horizon',  
   time\_col = 'Date',  
   target\_col = 'Close'  
   )

Методът на прогнозата се състои от следните аргументи:

1. df = train\_data: подаване на тренировъчния набор от данни към функцията.
2. h = len(test\_data): подаване на броя стъпки за прогнозиране.
3. Freq = 'B': задава честотата на данните. 'B' означава бизнес дни, което е нужно, за да обработи правилно цените на акции, които се търгуват само в работни дни.
4. level = [90]: задава нивото на доверителния интервал за прогнозата. Нивото на доверителния интервал е 90%, което означава, че моделът предоставя прогнозата с 90% вероятност реалните стойности да попаднат в този интервал.
5. model = 'timegpt-1-long-horizon': създава по-дълги прогнози, предвижда повече от един сезонен период от данните.
6. time\_col = 'Date': идентифицира колоната за дата.
7. target\_col = 'Close': променливата, която се прогнозира.

### 5.2.3 Прогноза и графики

След трениране на модела се разглеждат прогнозата чрез различни параметри.

  
*Фигура 23: Прогнозните стойности за акциите на ACM Research.*

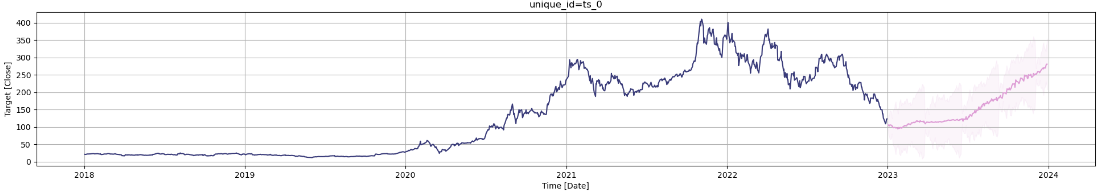
* Date: съдържа датите, за които е направена прогнозата.
* TimeGPT:съдържа основната прогнозна стойност за всяка дата.
* TimeGPT-lo-90: съдържа долната граница на 90% доверителен интервал за прогнозите. Това означава, че моделът предвижда с 90% вероятност, че истинската стойност ще бъде над тази граница.
* TimeGPT-hi-90: съдържа горната граница на 90% доверителен интервал за прогнозите. Това означава, че моделът предвижда с 90% вероятност, че истинската стойност ще бъде под тази граница.

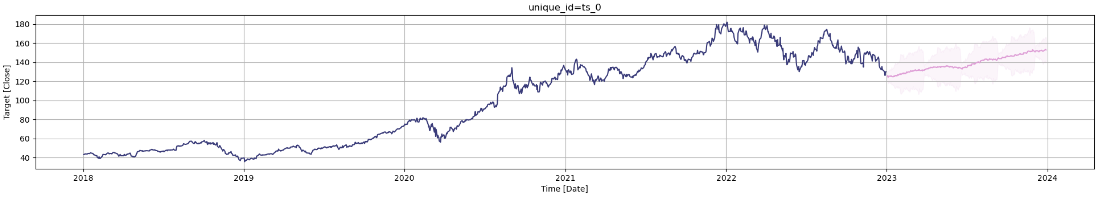
Визуализацията на историческите и прогнозираните стойности на цените на акциите е изпълнена чрез:

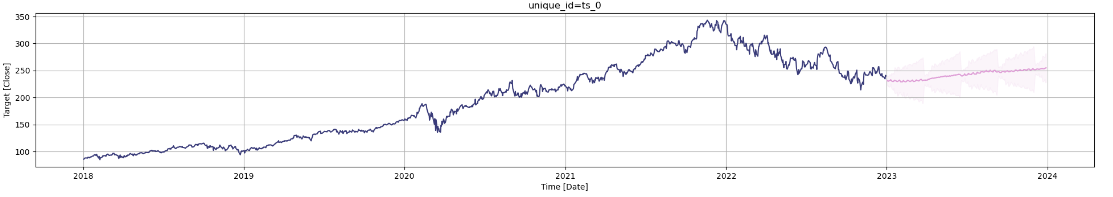
* nixtla\_client.plot(train\_data, data\_forecast, models=['TimeGPT'], level=[90], time\_col='Date', target\_col='Close')

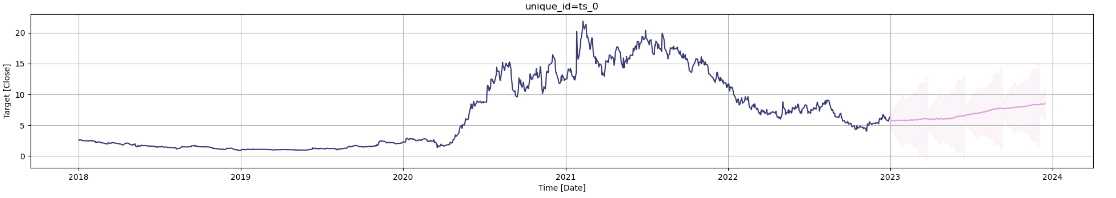
Графиките визуализират прогнозите и доверителните интервали. Те представляват сравнение на историческите данни с прогнозите на модела. Графиката се състои от:

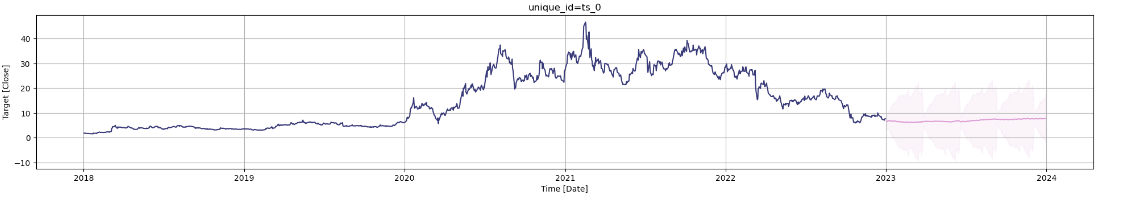
1. Синя линия: представя историческите данни.
2. Лилава линия: представлява прогнозираните стойности.
3. Светло лилава зона: представлява 90% доверителни интервали около прогнозните стойности. Долната граница (TimeGPT-lo-90) и горната граница (TimeGPT-hi-90) на доверителния интервал дават представа за диапазона, в който се очаква да попаднат реалните стойности с 90% вероятност.

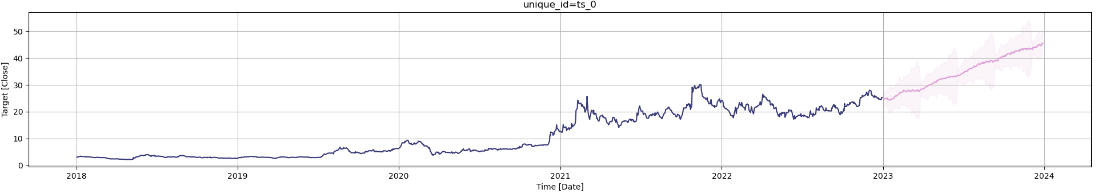
  
*Фигура 24: Графика на прогнозата на TimeGPT за цената на Tesla.*

  
*Фигура 25: Графика на прогнозата на TimeGPT за цената на Apple.*

  
*Фигура 26: Графика на прогнозата на TimeGPT за цената на Microsoft.*

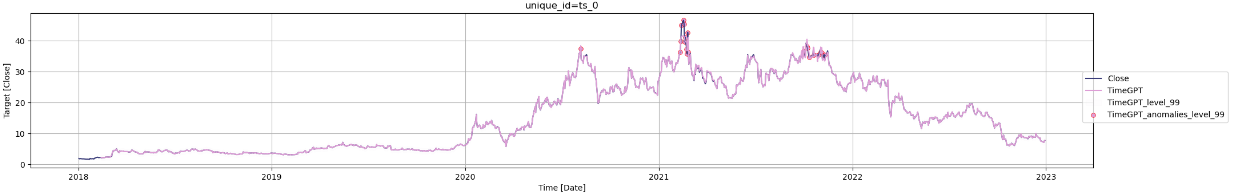
  
*Фигура 27: Графика на прогнозата на TimeGPT за цената на Carparts.*

  
*Фигура 28: Графика на прогнозата на TimeGPT за цената на ACM Research.*

  
*Фигура 29: Графика на прогнозата на TimeGPT за цената Perion Network.*

Като се има предвид променливостта и волатилността на акциите, е полезно да се опита да се идентифицират аномалии в данните. TimeGPT може да се използва за това чрез извикване на метода detect\_anomalies от класа NixtlaClient. Този метод оценява всяко наблюдение спрямо неговия контекст в серията, като използва статистически мерки, за да определи вероятността да бъде аномалия. По подразбиране той идентифицира аномалии въз основа на 99% доверителен интервал.

* anomalies = nixtla\_client.detect\_anomalies(tesla\_data, time\_col='Date', target\_col='Close', freq='B')  
  nixtla\_client.plot(tesla\_data, anomalies, time\_col='Date', target\_col='Close', plot\_anomalies=True)

  
*Фигура 30: Графика на откритите аномалии за 99% доверителен интервал за ACM Research.*

След изпълнението на кода, се визуализира графика, която включва:

* Линия за историческите данни: показва реалните цени на затваряне на акциите за историческия период.
* Аномалии: показва специално маркирани точки, които показват откритите аномалии. Тези точки показват дати, където цените на акциите се отклоняват значително от нормалното поведение.

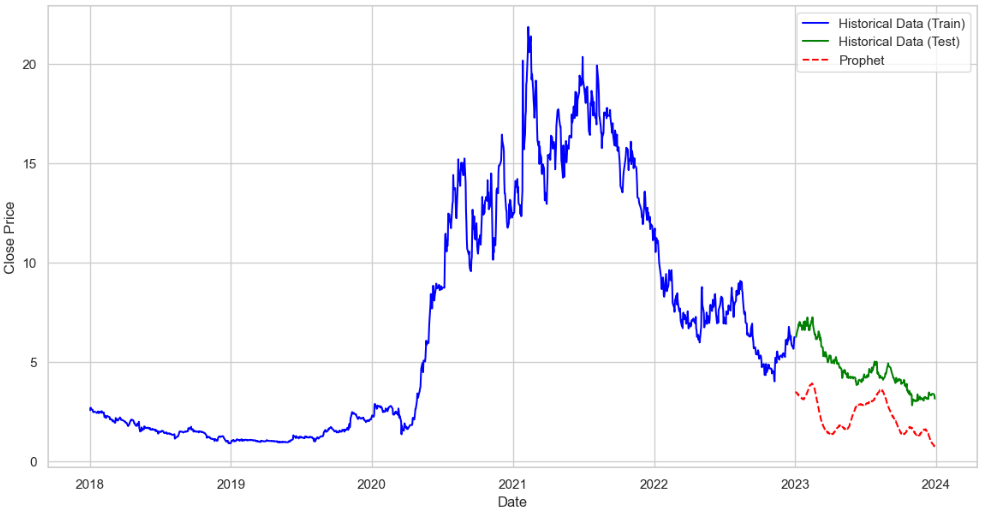
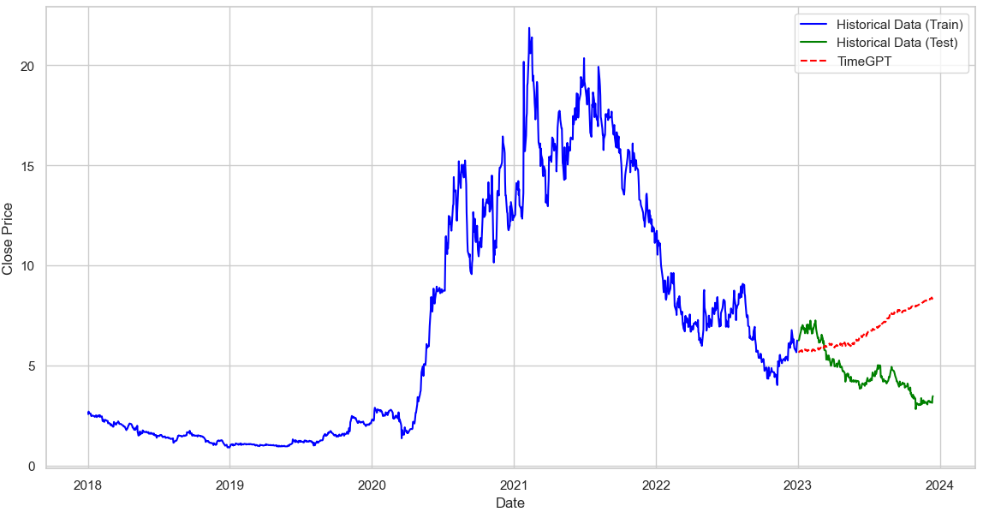
# 6. Резултати

## 6.1 Исторически срещу прогнозирани данни

****  
*Фигура 31: Прогнозираните и действителните цени на Tesla от Prophet (ляво) и TimeGPT (дясно).*

  
*Фигура 32: Прогнозираните и действителните цени на Apple от Prophet (ляво) и TimeGPT (дясно).*

****  
*Фигура 33: Прогнозираните и действителните цени на Microsoft от Prophet (ляво) и TimeGPT (дясно).*

** ****  
*Фигура 34: Прогнозираните и действителните цени на Carparts от Prophet (ляво) и TimeGPT (дясно).*

** ****  
*Фигура 35: Прогнозираните и действителните цени на ACM Research от Prophet (ляво) и TimeGPT (дясно).*

****  
*Фигура 36: Прогнозираните и действителните цени на Perion Network от Prophet (ляво) и TimeGPT (дясно).*

## 6.2 Сравнение на представянето

Резултатът от представянето демонстрира ефикасността на Prophet и TimeGPT, които бяха обучени за прогнозиране на цената на шест различни акции. Ефективността на моделите беше измерена с помощта на показателите за оценка RMSE, MAE и MAPE.

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 123.389 | 80.566 |
| MAE | 105.364 | 69.382 |
| MAPE | 36.403 | 31.744 |

*Таблица 1: Представяне на двата модела за прогнозите на цените на Tesla.*

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 51.615 | 36.341 |
| MAE | 46.630 | 34.062 |
| MAPE | 23.900 | 18.852 |

*Таблица 2: Представяне на двата модела за прогнозите на цените на Apple.*

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 150.402 | 77.872 |
| MAE | 136.423 | 70.610 |
| MAPE | 37.464 | 21.050 |

*Таблица 3: Представяне на двата модела за прогнозите на цените на Microsoft.*

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 2.526 | 2.852 |
| MAE | 2.354 | 2.434 |
| MAPE | 41.814 | 59.643 |

*Таблица 4: Представяне на двата модела за прогнозите на цените на Carparts.*

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 10.149 | 7.031 |
| MAE | 8.791 | 6.513 |
| MAPE | 52.090 | 46.050 |

*Таблица 5: Представяне на двата модела за прогнозите на цените на ACM Research.*

|  |  |  |
| --- | --- | --- |
| Метрики | Prophet | TimeGPT |
| RMSE | 10.617 | 8.492 |
| MAE | 9.809 | 6.887 |
| MAPE | 30.443 | 21.814 |

*Таблица 6: Представяне на двата модела за прогнозите на цените на Perion Network.*

## 6.3 Използване прогнозите на моделите в реална среда

За да се разбере дали алгоритмите на машинното обучение могат да помогнат за взимане на инвестиционни решения при търговия с акции, прогнозите на двата модела са тествани в платформата за графики за анализ на финансови пазари TradingView.

Съществуват много разнообразни стратегии в търговията на акции, като стратегия за търговия с новини, високочестотна търговия, скалпиране, търговия в рамките на деня, суинг търговия, позиционна търговия.

За тази дипломна работа стратегията за търгуване и проверката дали прогнозите са доходоносни са изпълнени по следния начин:

1. Вземат се датите при най-ниска и най-висока цена от периода на прогнозата за 2023 година от двата модела за разгледаните акции, като датата при най-ниска цена трябва да е преди датата на най-високата.
2. Купува се 1 брой акции от определените компании в деня на най-ниската цена и се продават в денят на най-високата.
3. Сравнява се печалбата (или загубата) в средата за тестване TradingView спрямо очакваната печалба от прогнозите на моделите.

Целта е с най-малко вложени пари да има най-голяма печалба. За най-голяма възможна печалба моделите са определили следните възможности:

|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 10.03.2023 / 160.34$ | 25.01.2023 / 94.22$ |
| Дата на продажба / цена | 27.03.2023 / 168.69$ | 11.12.2023 / 253.85$ |
| Печалба / % печалба | 8.35$ / 5% | 159.63$ / 169% |

*Таблица 7: Предвидена печалба за акциитена Tesla.*

|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 16.06.2023 / 121.16$ | 03.01.2023 / 124.14$ |
| Дата на продажба / цена | 21.08.2023 / 132.28$ | 11.12.2023 / 152.92$ |
| Печалба / % печалба | 11.12$ / 9% | 28.78$ / 23% |

*Таблица 8: Предвидена печалба за акциитена Apple.*

|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 20.01.2023 / 213.13$ | 06.02.2023 / 228.82$ |
| Дата на продажба / цена | 08.02.2023 / 217.58$ | 11.12.2023 / 253.41$ |
| Печалба / % печалба | 4.45$ / 2% | 24.59$ / 10% |

*Таблица 9: Предвидена печалба за акциитена Microsoft.*

|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 10.04.2023 / 0.52$ | 03.01.2023 / 5.65$ |
| Дата на продажба / цена | 11.08.2023 / 3.60$ | 11.12.2023 / 8.40$ |
| Печалба / % печалба | 3.08$ / 592% | 2.75$ /48% |

*Таблица 10: Предвидена печалба за акциитена Carparts.*

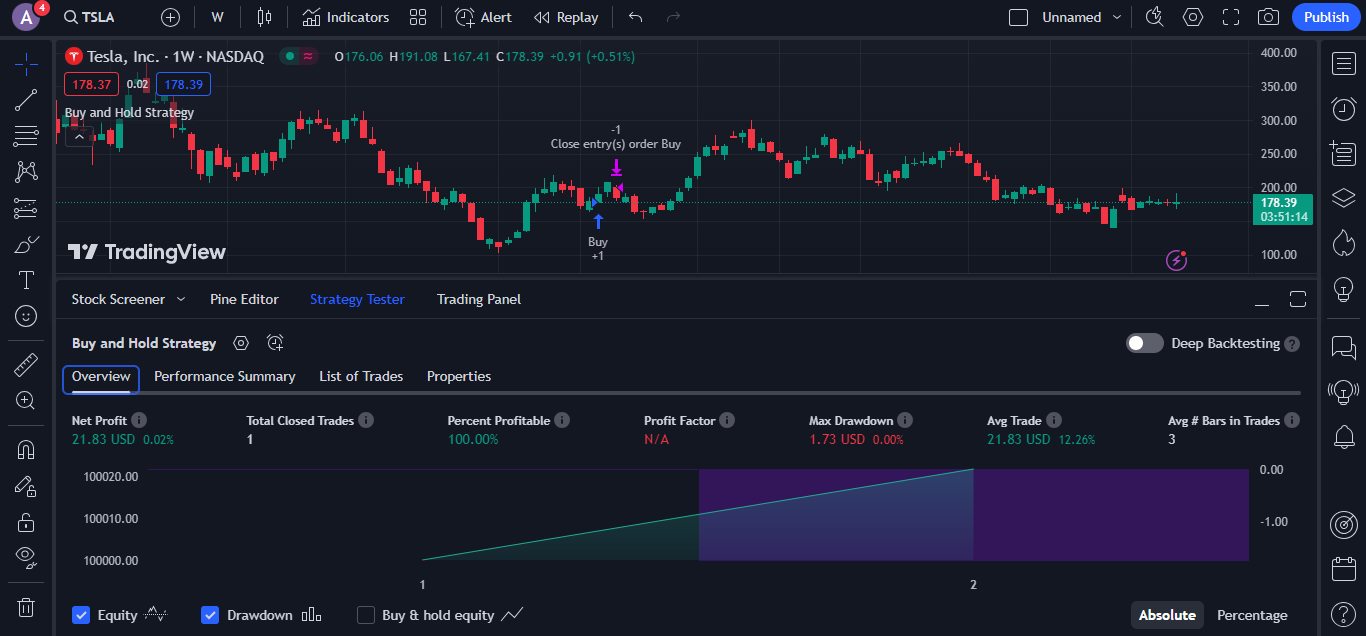
|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 28.04.2023 / 2.69$ | 27.02.2023 / 6.11$ |
| Дата на продажба / цена | 10.08.2023 / 6.95$ | 29.11.2023 / 7.77$ |
| Печалба / % печалба | 4.26$ / 158% | 1.66$ / 27% |

*Таблица 11: Предвидена печалба за акциитена ACM Research.*

|  |  |  |
| --- | --- | --- |
|  | Prophet | TimeGPT |
| Дата на купуване / цена | 12.05.2023 / 20.88$ | 17.01.2023 / 24.31$ |
| Дата на продажба / цена | 17.11.2023 / 24.27 | 08.12.2023 / 43.66$ |
| Печалба / % печалба | 3.39$ / 16% | 19.35$ / 79% |

*Таблица 12: Предвидена печалба за акциитена Peri Network.*

Проверката на данните се осъществява чрез Pine Script. Това е инструмент, разработен от TradingView за създаване на персонализирани технически анализи и търговски стратегии. Той позволява на потребителите да създават собствени индикатори, скриптове и автоматизирани търговски стратегии, които могат да бъдат използвани директно в платформата на TradingView.

  
*Фигура 37: Изглед от сайта на TradingView.*

Информацията от прогнозите за проверка се въвеждат по следния начин:

//**@version=**4

strategy("Buy and Hold Strategy", overlay=true)

entryDate = timestamp(2023, 03, 10, 0, 0)

exitDate = timestamp(2023, 03, 23, 0, 0)

numShares = 1

entryCondition = time >= entryDate and time[1] < entryDate

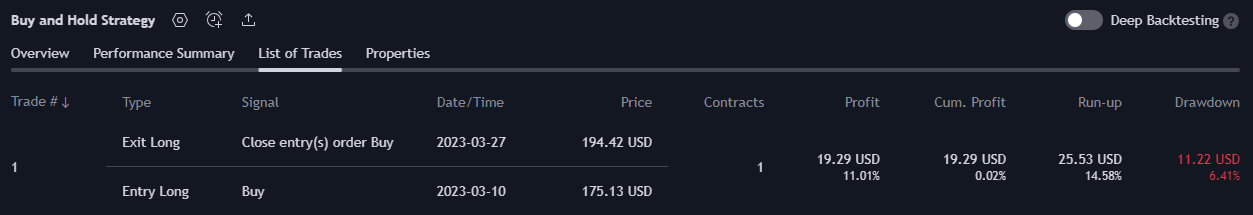
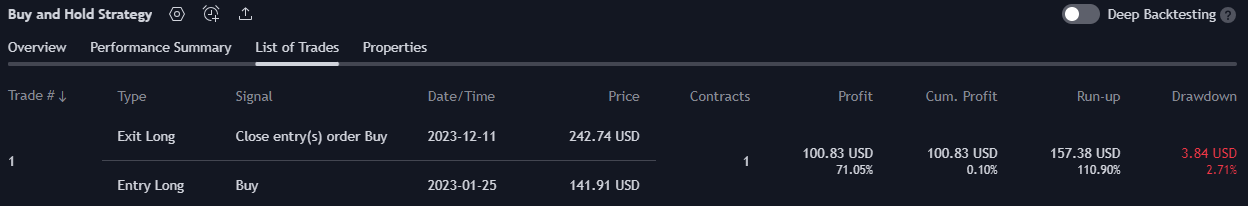
exitCondition = time >= exitDate and time[1] < exitDate

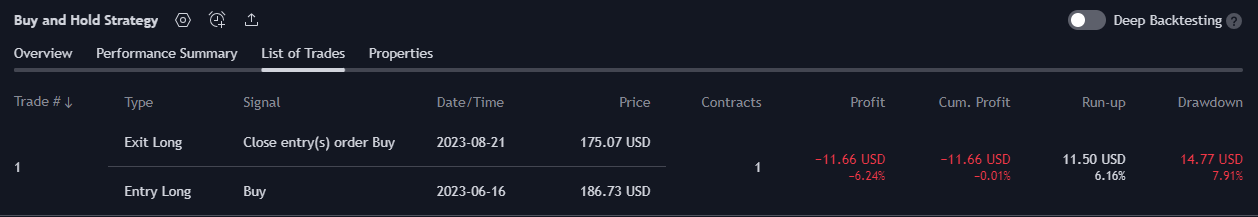
if (entryCondition)

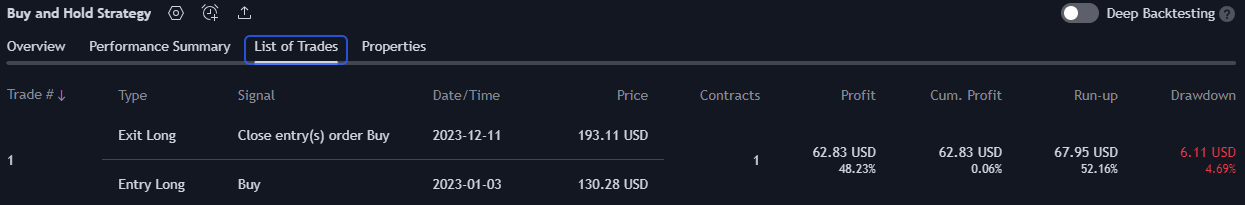
    strategy.entry("Buy", strategy.long, qty=numShares)

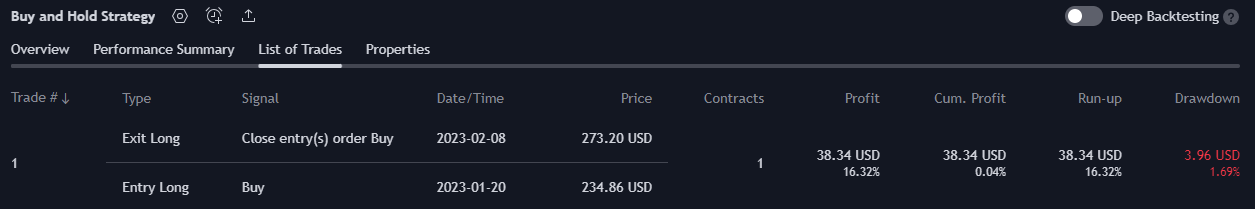
if (exitCondition)

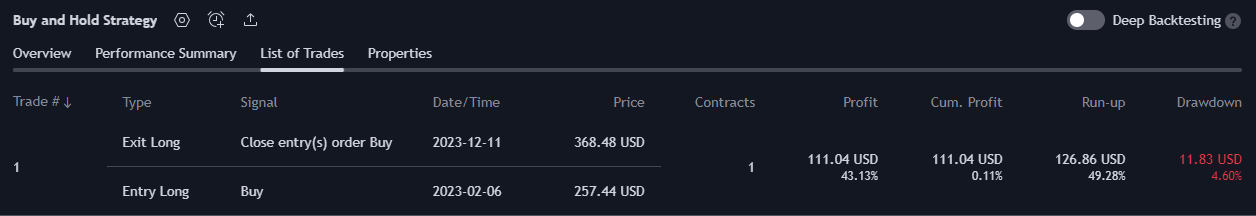
    strategy.close("Buy")

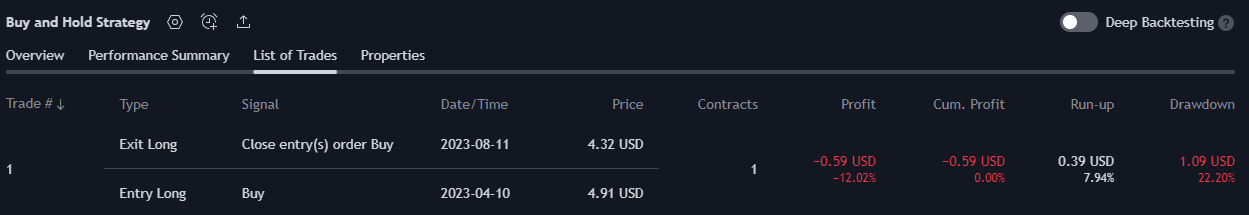
  
  
 *Фигура 38: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на Tesla.*

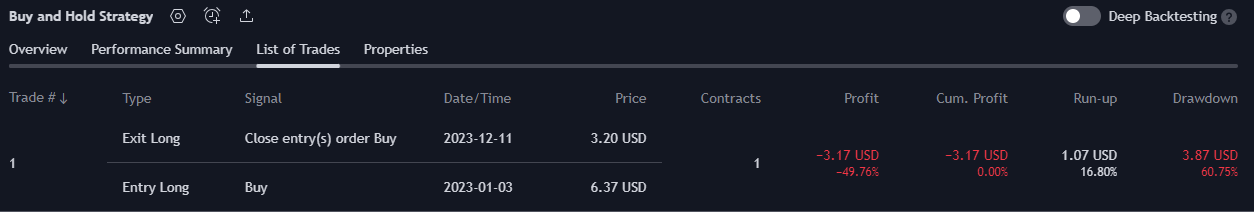


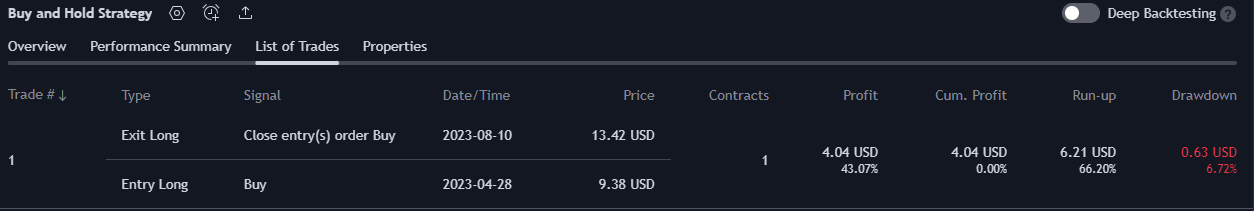
 *Фигура 39: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на Apple.*

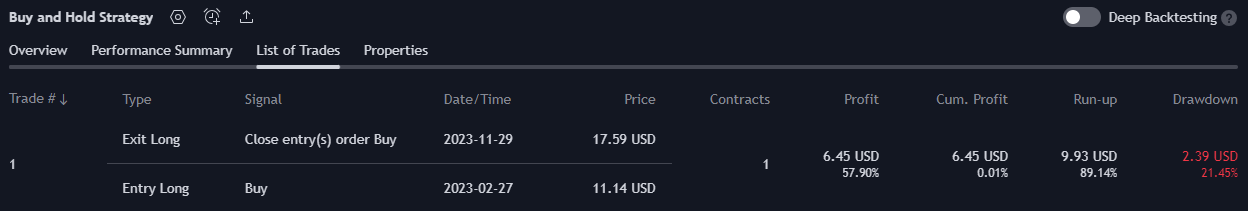


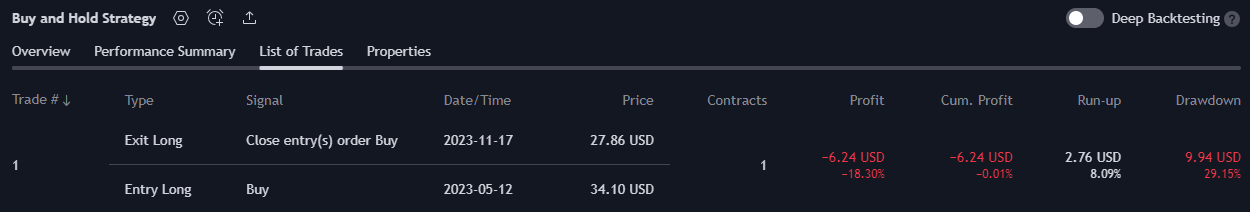
 *Фигура 40: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на Microsoft.*

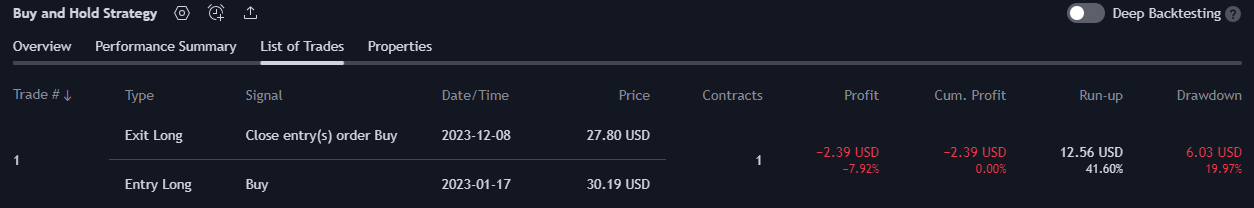


 *Фигура 41: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на Carparts.*



 *Фигура 42: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на ACM Research.*



 *Фигура 43: Историческа справка за посочените за търговия дати от Prophet (горе) и TimeGPT (долу) за акциите на Peri Network.*

# Глава.7 Заключение

Експериментът с проверката дали и колко доходоносни могат да бъдат алгоритмите от времеви редове показаха интересни резултати. Прогнозите за закупуване, за продажба или за задържане на акциите за максимална печалба в подходящо време е сложна задача, с която двата модела се справиха по следния начин:

* Моделът Prophet при своите прогнози за най-висока печалба успя да предскаже правилно три от разглежданите акции, като за две от тях (Tesla и Microsoft) действителната печалба е по-висока от прогнозираната, а за ACM Research е познат трендът на повишаване, но действителната печалба е по-ниска от предсказаната. За акциите на Apple, Carparts и Peri Network е сгрешен трендът и вместо печалба, при тях се наблюдава загуба.
* Моделът TimeGPT се справя по-добре за тази задача, като за акциите на Tesla, Apple, Microsoft и ACM Research се постига печалба и само при Tesla реалната печалба е по-ниска от прогнозираната. При акциите на Carparts и Peri Network се отчита загуба. При TimeGPT се наблюдава правилна прогноза за mega-cap компаниите.

Спрямо метриките за оценка на представянето на моделите TimeGPT се справя по-убидително от Prophet. TimeGPT има по-добри показтели за всички прогнози, без тази за ACM Research. Също така за прогнозата на тази акция са отчетени най-малки грешки сравнение с на останалите акции. Заедно с това, показателите за оценка на small-cap компаниите се оказват по-правилни от тези на mega-cap компаниите. Prophet и TimeGPT имат близки резултати при прогнозите за small-cap компаниите, докато при mega-cap компаниите TimeGPT се представя по-каачествено.

За бъдещи разработки могат да се приложат оптимизация и подобряване на алгоритмите за прогноза на цените на акциите, за да се постигне по-добро представяне с различни методи. Като подобряване произведителността са възможни следните техники:

1. **Feature engineering:** е процес за създаване на нови признаци от съществуващите данни с цел подобряване на моделите. Такива признаци могат да бъдат:
   1. Създаване на лагове.
   2. Подвижни средни (moving averages).
   3. Външни регресори.
   4. Агрегиране.
2. **Hyperparameter Tuning:** е процес за настройка на хиперпараметрите на моделите с цел подобряване на тяхната ефективност. Това може да се направи чрез различни техники:
   1. Grid Search .
   2. Random Search.
   3. Бейсов подход.
   4. Крос валидация.
3. **Ensemble methods:** са подход в машинното обучение, при който се комбинират повече от един модел, за да се постигне по-добра производителност от тази на отделните модели. Идеята зад ансамбловите методи е, че комбинирането на прогнозите на няколко модела може да компенсира слабостите на отделните модели и да предложи по-точни прогнози. Основните ансамблови методи са:
   1. Bagging (Bootstrap Aggregating).
   2. Boosting.
   3. Stacking.

Допълнително могат да се използват различни индикатори за акциите освен историческите данни, като по-обширните набори от данни могат да помогнат на моделите да уловят по-добре зависимости в цените на акциите.Такива могат да бъдат:

* **Финансови показатели:** Това могат да бъдат финансови показатели като печалба на акция, оборот на активи, задължения, показатели на ликвидност и други, които да дадат представа за финансовото здраве на компанията.
* **Технически анализи:** Това включва технически показатели като подвижни средни, индикатори за статистическа волатилност и други, които да помогнат за идентифициране на тенденции и сигнали за търговия.
* **Макроикономически данни:** Това включва данни като данни за безработица, инфлация и лихвени проценти.
* **Новини и социални медии:** Анализ на новини и социални медии за събития, които могат да влияят на пазарните настроение и цените на акциите.

# Глава 8. Литература

TimeGPT in Load Forecasting: A Large Time Series Model PerspectiveWenlong Liao, Fernando Porte-Agel, Jiannong Fang, Christian Rehtanz, Shouxiang Wang,Dechang Yang, Zhe Yang[3]

* Официална документация Python - <https://www.python.org/doc>
* Официална документация Jupyter Notebook - <https://docs.jupyter.org/en/latest/>
* Официална документация Yahoo Finance- [https://python-yahoofinance.readthedocs.io/en/latest](https://python-yahoofinance.readthedocs.io/en/latest/api.html)
* Официална документация Prophet - [https://facebook.github.io/prophet/docs](https://facebook.github.io/prophet/docs/quick_start.html)
* Официална документация TimeGPT - <https://docs.nixtla.io/docs>
* Официална страница на TradingView - <https://www.tradingview.com/>
* [1] Components of Time Series - <https://www.toppr.com/guides/business-mathematics-and-statistics/time-series-analysis/components-of-time-series/>
* [2] Time Series Model for Stock Market Prediction Utilising Prophet - [https://www.proquest.com/docview/](https://www.proquest.com/docview/2640416313?fromopenview=true&pq-origsite=gscholar&sourcetype=Scholarly%20Journals)
* [3] Forecasting at scale - [https://peerj.com/preprints/](https://peerj.com/preprints/3190/)
* [4] TimeGPT in Load Forecasting: A Large Time Series Model Perspective - <https://arxiv.org/pdf/2404.04885>
* [5] TimeGPT: Revolutionizing Time Series Forecasting - <https://datadance.ai/deep-learning/timegpt-revolutionizing-time-series-forecasting/>
* [6] Metrics for Time Series Forecasting Models - <https://analyticsindiamag.com/a-guide-to-different-evaluation-metrics-for-time-series-forecasting-models/>