



ÓBUDAI EGYETEM  
ÓBUDA UNIVERSITY

**NEUMANN JÁNOS  
INFORMATIKAI KAR**



# SZAKDOLGOZAT

**OE-NIK  
2024**

Hallgató neve:  
Hallgató törzskönyvi száma:

**Hertelendy András  
T-008528/F112904/N**

Óbudai Egyetem  
Neumann János Informatikai Kar  
Szoftvertervezés és -fejlesztés Intézet

## SZAKDOLGOZAT FELADATLAP

Hallgató neve:	<b>Hertelendy András</b>
Törzskönyvi száma:	T/008528/FI12904/N
Neptun kódja:	W9T4EX

A dolgozat címe:

**Túra útvonalak menetidejének következtetése mesterséges intelligencia  
használatával**  
**Inference of hiking trail durations utilizing artificial intelligence  
applications**

Intézményi konzulens:	Dr. Tusor Balázs
Külső konzulens:	

Beadási határidő:	2024. december 15.
-------------------	--------------------

A záróvizsga tárgyai:	Számítógép architektúrák Mesterséges intelligencia specializáció
-----------------------	---

### **A feladat**

A hallgató feladata egy olyan alkalmazás elkészítése, amely turista útvonalakat tartalmazó bemenetből mesterséges intelligencia modell segítségével kikövetkezteti a beállított személy számára szükséges teljesítési időt. A modell figyelembe veszi a felhasználó fittségi szintjét, az útvonal szintváltozásait és azoknak sorrendjét.

### **A dolgozatnak tartalmaznia kell:**

- ☐ Szakirodalom összefoglalását
- ☐ Az applikáció felépítését és működését
- ☐ Tesztelések folyamatát és eredményeit
- ☐ Egyéb opcionális módszerek ismertetését és értékelését
- ☐ A felhasználói leírást
- ☐ Tovább fejlesztési lehetőségeket

Ph.

.....  
Dr. habil. Vámosy Zoltán  
intézetigazgató

A szakdolgozat elévülésének határideje: **2026. december 15.**  
(ÓE HKR 54.§ (10) bekezdés szerint)

A dolgozatot beadásra alkalmasnak tartom:

.....  
külső konzulens

.....  
intézményi konzulens



ÓBUDAI EGYETEM  
ÓBUDA UNIVERSITY

Neumann János Informatikai Kar

### KONZULTÁCIÓS NAPLÓ

Hallgató neve: ..... Neptun Kód: ..... Tagozat: .....  
Hertelendy András ..... W9T4EX..... nappali .....

Telefon: ..... Levelezési cím (pl.: lakcím): .....  
+36207704872 ..... 2500 Esztergom, Magyary László utca 10. ....

Szakedolgozat / Diplomamunka<sup>1</sup> címe magyarul:

Túra útvonalak menetidejének következtetése mesterséges intelligencia használatával .....

Szakedolgozat / Diplomamunka<sup>2</sup> címe angolul:

Inference of hiking trail durations utilizing artificial intelligence applications .....

Intézményi konzulens: ..... Külső konzulens: .....  
Dr. Tusor Balázs ..... ..

Kérjük, hogy az adatokat nyomtatott nagy betűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2024.03.06.	SZAKDOLGOZAT SZERKEZETÉNEK ÁTBESZÉLÉSE	Tusor Balázs
2.	2024.04.04.	EDDIGI HALADÁS BEMUTATÁSA	Tusor Balázs
3.	2024.04.25.	MEGVALÓSÍTÁSI KILÁTÁSOK ÁTBESZÉLÉSE	Tusor Balázs
4.	2024.05.02.	ÁLTALÁNOS KONZULTÁCIÓ	Tusor Balázs

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakedolgozat / Szakedolgozat I. / Szakedolgozat II. / Projektlabor 1 / Projektlabor 2 / Projektlabor 3 / Záródolgozati projekt / Diplomamunka I / Diplomamunka II / Diplomamunka III / Diplomamunka IV <sup>3</sup> tantárgy követelményét teljesítette, beszámolóra / védésre <sup>4</sup>bocsátható.

A konzulens által javasolt érdemjegy: .....5.....

Budapest, 2024.05.15.....

Tusor Balázs

Intézményi konzulens

<sup>1</sup> Megfelelő aláhúzendő!

<sup>2</sup> Megfelelő aláhúzendő!

<sup>3</sup> Megfelelő aláhúzendő!

<sup>4</sup> Megfelelő aláhúzendő!

# Tartalomjegyzék

<b>1</b>	<b>Bevezető</b>	<b>5</b>
<b>2</b>	<b>Irodalomkutatás</b>	<b>6</b>
<b>2.1</b>	<b>Túra útvonalak digitális formátuma</b>	<b>6</b>
2.1.1	GPX (GPS Exchange Format)	6
2.1.2	KML (Keyhole Markup Language)	6
2.1.3	TCX (Training Center XML)	6
2.1.4	FIT (Flexible and Interoperable Data Transfer)	6
2.1.5	Összegzés	7
<b>2.2</b>	<b>Hagyományos túraidő-becslési módszerek</b>	<b>7</b>
2.2.1	Naismith szabálya	7
2.2.2	Tobler függvény	8
2.2.3	Book Time formula	8
2.2.4	Munter módszer	9
<b>2.3</b>	<b>Mesterséges intelligencia alkalmazása a túraidő becslésben</b>	<b>9</b>
2.3.1	Mesterséges intelligenciát alkalmazó túra applikációk	9
2.3.2	Mesterséges intelligencia algoritmusok a menetidő meghatározásához	10
2.3.3	Random Forest algoritmus	10
2.3.4	XGBoost algoritmus	11
2.3.5	Recurrent Neural Networks (RNN)	12
2.3.6	Long Short-Term Memory (LSTM)	12
<b>2.4</b>	<b>Továbbfejlesztett mesterséges intelligencia modellek és technikák</b>	<b>13</b>
2.4.1	Mélytanulás és konvolúciós neurális hálózatok (CNN) alkalmazása térképes adatokon	13
2.4.2	Idősoros elemzések és azok speciális technikái, mint az ARIMA modellek túraidő előrejelzésre.	13
<b>2.5</b>	<b>Valós idejű adatfeldolgozás</b>	<b>14</b>
<b>2.6</b>	<b>Adaptív mesterséges intelligencia modellek és a csoportos túrázás</b>	<b>14</b>
<b>2.7</b>	<b>Adatvédelmi és etikai szempontok</b>	<b>16</b>
2.7.1	Titkosítás	16
2.7.2	Hozzáférés-vezérlés	16
2.7.3	Adatvédelmi szabályok betartása	16
2.7.4	Anonimitás	16
2.7.5	Biztonsági auditok	17
2.7.6	Adatvédelem összefoglalva	17
<b>2.8</b>	<b>Biztonsági megfontolások és hibakezelés</b>	<b>17</b>
2.8.1	Adatellenőrzés és -tisztítás:	17
2.8.2	Algoritmus robusztussága:	17
2.8.3	Valós idejű megfigyelés és riasztások:	18
2.8.4	Titkosítás:	18
2.8.5	Hozzáférés-vezérlés:	18
2.8.6	Rendszeres auditok és frissítések:	18
2.8.7	Anomáliaérzékelés:	18
<b>2.9</b>	<b>Adatforrások és adatgyűjtés</b>	<b>18</b>
2.9.1	Nyilvános adatbázisok és API-k használata	19

<b>3</b>	<b>Rendszerterv</b>	<b>20</b>
<b>3.1</b>	<b>Bevezetés a rendszertervbe</b>	<b>20</b>
3.1.1	A fejlesztendő alkalmazás célja és funkcionális követelményei	20
3.1.2	Fontos kérdés: Online szervereken vs Offline működés	21
3.1.3	Megoldás	21
<b>3.2</b>	<b>Alkalmazás architektúra</b>	<b>22</b>
3.2.1	Frontend technológiák	22
3.2.2	Backend technológiák	22
3.2.3	Framework és könyvtárak választása	23
3.2.4	API-k és integrációk	23
3.2.5	Cloud szolgáltatások és hosting	23
<b>3.3</b>	<b>Felhasználók kezelése</b>	<b>23</b>
<b>4</b>	<b>Következő félévi tervek</b>	<b>24</b>
<b>5</b>	<b>Összefoglalás</b>	<b>25</b>
<b>6</b>	<b>Irodalomjegyzék</b>	<b>26</b>
<b>7</b>	<b>Ábrajegyzék</b>	<b>29</b>

# 1 BEVEZETŐ

A túrázás népszerű szabadidős tevékenység, amely kihívások elé állítja a túrázókat, különösen a tervezés során. Az egyik legnagyobb kihívás az útvonalak pontos időtartamának becslése, ami kritikus lehet például a tömegközlekedéssel való hazajutás tervezésekor. A hagyományos módszerek, mint a Naismith-szabály (2.2.1) vagy a Book Time formula (2.2.3), gyakran nem veszik figyelembe a túrázók egyéni képességeit és az útvonal specifikus jellemzőit, például a szintkülönbséget és annak sorrendjét, valamint azt, hogy egyénileg vagy csoportban haladnak.

Ez a szakdolgozat egy innovatív megoldást mutat be, amely mesterséges intelligencia (MI) modellek segítségével becsüli meg a túraútvonalak teljesítési idejét. Az alkalmazás figyelembe veszi a felhasználó fizikai kondícióját és az útvonal jellegzetességeit, így pontosabb és személyre szabottabb időbecslést tesz lehetővé.

A modern technológia alkalmazása nemcsak pontosabbá teszi az időtartamok becslését, hanem növeli a túrázók biztonságát is, mivel realisabb képet kapnak az útvonal nehézségéről és időigényéről. Emellett az adatvezérelt megközelítés új perspektívákat nyit meg az útvonalak optimalizálásában és az egészségügyi hatások monitorozásában is. A dolgozat bemutatja, hogyan ötvözhető az AI technológia az aktív életmóddal, lépést tartva a technológiai fejlődéssel és a növekvő felhasználói igényekkel. Ezáltal nemcsak a jelenlegi, de a jövő túrázói számára is értékes eszközt kínál, megerősítve az MI létjogosultságát és hasznosságát a mindennapi élet számos területén.

A 2. fejezetben irodalomkutatás keretein belül bemutatom a túraútvonalak digitális formátumait, majd ismertetem a hagyományos túraidő-becslési módszereket, mint a Naismith-szabály és a Tobler-függvény. Ezt követően a mesterséges intelligencia alkalmazását vizsgálom a túraidő becslésben, részletezve az olyan algoritmusokat, mint a Random Forest és az XGBoost. A 3. fejezetben az alkalmazás rendszertervét ismertetem, beleértve a frontend és backend technológiák kiválasztását, az API-k és integrációk használatát, valamint a felhasználók kezelését és az adatvédelmi szempontokat. A dolgozat végén pedig a megvalósítási tervet mutatom be, ahol részletesen leírom a fejlesztési mérföldköveket.

## 2 IRODALOMKUTATÁS

### 2.1 Túra útvonalak digitális formátuma

#### 2.1.1 GPX (GPS Exchange Format)

A GPX (1. ábra) egy XML adat formátum, amelyet GPS eszközök és telefonos navigációs alkalmazások adatainak veszteség mentes tárolására és átvitelére terveztek. Egy GPX fájl részletes adatokat képes tárolni, például GPS jel erősségét, útjelzőpontokat, útvonalakat, nyomvonalakat, időt és sebességet, sőt még szív ritmust is egy úgynevezett Garmin TrackPointExtension kiegészítő segítségével [1]. Ez a formátum széles körben használatos túra útvonalak és egyéb földrajzi adatok megosztására, mivel megőrzi az összes rögzített információt és megkönnyíti az adatok átvitelét különböző eszközök és alkalmazások között [2].

#### 2.1.2 KML (Keyhole Markup Language)

A KML egy prezentációs nyelv, amelyet a Google fejlesztett ki térképek annotálására. Földrajzi jellemzőket, például pontokat, vonalakat és poligonokat képes leírni, és gyakran használják a Google Earth alkalmazással. A KML fájlok globális hely adatokat és más fájlokra (jellemzően grafikákra) való hivatkozásokat tartalmazhatnak szöveg alapú formátumban. Például egy KML fájl tartalmazhat útjelzőpontokat és a hozzájuk kapcsolódó térkép képeket egy külön fájlban. A KML leginkább az útvonalak vizualizálására alkalmas olyan alkalmazásokban, mint a Google Earth [2].

#### 2.1.3 TCX (Training Center XML)

A TCX, vagy Training Center XML, egy adat formátum, amelyet elsősorban a Garmin eszközök használnak fitness adatok tárolására, beleértve az edzés részleteit és az útvonalakat. Ez egy XML-alapú formátum, amely olyan tevékenységeket rögzít, mint a futás, kerékpározás és túrázás. A TCX fájlok részletes adat pontokat tartalmaznak, például szív ritmust, kadenciát és magasságot, amelyek elengedhetetlenek a fitness tevékenységek elemzéséhez. A formátum támogatja az egyes körök és az összesített edzés összefoglalók rögzítését is, így alkalmas a fitness követésére és a teljesítmény elemzésére [3].

#### 2.1.4 FIT (Flexible and Interoperable Data Transfer)

A FIT, azaz Flexible and Interoperable Data Transfer, egy bináris fájl formátum, amelyet a Garmin fejlesztett ki fitness és egészségügyi adatok rögzítésére. Ez a formátum hatékonyan tárolja a fizikai tevékenységek részletes feljegyzéseit, mint például a szív ritmus, lépésszám és tengerszint feletti magasság, így ideális fitness tevékenységek teljes körű követésére. A FIT formátum rendkívül hatékony, lehetővé téve nagy mennyiségű adat tárolását kompakt formában, ezért széles körben használják Garmin eszközökben és kompatibilis fitness alkalmazásokban [4].



### 2.1.5 Összegzés

A formátumok kiválasztásánál fontos figyelembe venni a támogatott attribútumokat, és hogy a túra alkalmazáshoz mire lesz szükség. Mindegyik formátum számos attribútummal rendelkezik, amik hasznosak számunkra, ebből a szempontból bármelyik választható. Azonban van még egy szempont, amit figyelembe ajánlott venni, ez pedig a formátum elterjedtsége, és támogatottsága. Kutatásom szerint ebből a szempontból a GPX formátum a legelőnyösebb, mivel ezt támogatja a legtöbb alkalmazás és ez a leguniverzálisabb [5], tehát ezt fogom nagy valószínűséggel választani. Továbbá van egy olyan előnye is, hogy aktivitás rögzítő alkalmazásokból le lehet tölteni gpx formátumú aktivitásokat (1. ábra), még más felhasználóhoz tartozó publikus felvételeket is, például az egyik legnépszerűbb, a Strava alkalmazásból ezt meg lehet tenni [6].

```
<trkpt lat="47.6032530" lon="18.9515470">
  <ele>175.5</ele>
  <time>2024-04-19T20:40:48Z</time>
</trkpt>
<trkpt lat="47.6032670" lon="18.9515540">
  <ele>175.5</ele>
  <time>2024-04-19T20:40:49Z</time>
</trkpt>
<trkpt lat="47.6032800" lon="18.9515610">
  <ele>175.6</ele>
  <time>2024-04-19T20:40:50Z</time>
</trkpt>
```

1. ábra: Egy létező GPX formátumú felvétel részlete kiegészítők nélkül.

## 2.2 Hagyományos túraidő-becslési módszerek

A túrázás menetidejének kiszámítására számos módszer létezik, amelyeket a világ különböző részein alkalmaznak. Ezek a módszerek különböző tényezőket vesznek figyelembe, mint például a távolságot, az emelkedőket, és a túrázók fizikai állapotát. Az alábbiakban bemutatom a leggyakoribb és legismertebb módszereket:

### 2.2.1 Naismith szabálya

A Naismith-szabályt arra használják, hogy becsüljék a gyaloglási időt egyenetlen terepen. A szabályt William Naismith skót hegymászó dolgozta ki 1892-ben, ami 5 km/óra sebességet feltételez sík terepen, minden 300 méter emelkedésre plusz fél órát hozzáadva. Egy egyenlet is van hozzá, ahogy lentebb látható (1). A szabályt később finomították különböző empirikus tesztek alapján. Langmuir 1984-ben javított a szabályon, 10 percet levonva minden 300 méteres lejtőért 5 és 12 fok között, és hozzáadva 10 percet minden 300 méteres lejtőért, ha az nagyobb mint 12 fok. [7] [8]

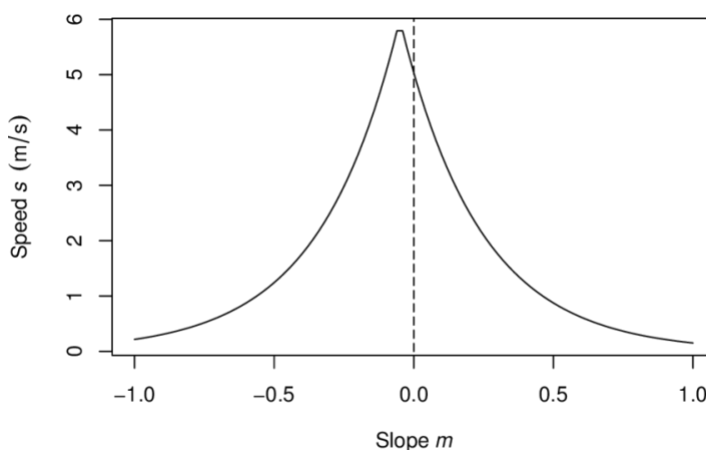
$$Idő = \frac{Vízszintes\ távolság}{3} + \frac{Szintemelkedés}{2000} \quad (1)$$

A Naismith szabály azonban nem vesz figyelembe több olyan tényezőt, mint a terep nehézsége, a túrázó fizikai állapota vagy az időjárási körülmények, amik jelentősen befolyásolhatják a túra időtartamát. Ezért gyakran szükséges módosításokat alkalmazni vagy modernizált változatokat használni, amelyek jobban megfelelnek a valós körülményeknek.

Ezeket a korlátokat felismerve különböző túrázási szakértők és szervezetek fejlesztettek ki kiegészítő szabályokat vagy teljesen új modelleket, amelyek részletesebben veszik figyelembe a különféle változókat, mint például a Langmuir-féle módosítás, amely az emelkedés mértékével arányosan növeli az időtartamot.

### 2.2.2 Tobler függvény

A Tobler gyaloglási függvény, amelyet Waldo Tobler dolgozott ki, exponenciális modellt használ a gyaloglási sebesség becslésére a terep lejtése alapján. A függvény a legnagyobb gyaloglási sebességet enyhe lejtőnél (-5 százalékos gradiens) jósolja meg, és ezeket a paramétereket használja: gyaloglási sebesség  $v$ , maximális sebesség  $\alpha$ , és együtthatók  $\beta_1$  és  $\beta_2$ , amelyek a lejtő változása függvényében a sebesség változásait számolják. (2. ábra) Ez a modell különösen hatékony a változatos topográfiai körülmények közötti mozgás modellezésére. [8] [9]



2. ábra: Tobler függvénye. [10]

### 2.2.3 Book Time formula

A "Book Time" képlet egy egyszerűsített módszer a túrák teljesítéséhez szükséges idő becslésére, amelyet gyakran használnak túratervezőkben. Általában minden megtett mérföldre 30 percet, és minden 1000 láb szint-emelkedésre további 30 percet számol. Ez a képlet hasonló a Naismith-szabályhoz, de valamivel kisebb tempót alkalmaz, ami különösen hasznos lehet hegyvidéki terepen, ahol a meredekebb és nehezebb utak miatt lassabb haladás várható. [8] [11]

Továbbá, a módszer elismeri, hogy a környezeti, logisztikai és személyes tényezők jelentősen befolyásolhatják a túrák időtartamát, ami azt jelenti, hogy a valós idő eltérhet a terepen.

#### 2.2.4 Munter módszer

A Munter módszer, amelyet Werner Munter svájci túravezető fejlesztett ki, egy gyakran használt eszköz a hegyekben történő utazási idő becslésére, figyelembe véve az utazott távolságot, a szintkülönbséget és az utazási módot. Ez a módszer különösen hasznos a hegyi túrázáshoz, síeléshez és hegymászáshoz. [12]

Az időt úgy számítja, hogy egységeket vesz figyelembe, ahol minden megtett kilométer vagy minden 100 méter szint-emelkedés egy-egy egységet jelent. Az összes egységet egy együtthatóval osztják el, amely változik, attól függően, hogy milyen tevékenységet végeznek (pl. gyaloglás, síelés, hegymászás) és hogy felfelé, vagy lefelé haladnak.

A Munter módszer egy egyszerű képletet alkalmaz, ahol az időt úgy számolja ki, hogy hozzáadja az utazott távolságot a szintemelkedéshez, osztva százzal, majd ezt az összeget elosztja egy meghatározott ráta értékkel. Lásd: (2)

$$Idő = \frac{Vízszintes\ távolság + \frac{Szintemelkedés}{100}}{Ráta} \quad (2)$$

Például: A gyalogos felfelé haladása esetén a *Ráta*-t általában 4-es értéken állapítják meg, sík vagy lefelé haladásnál gyalog 6-os értéket használnak, síelésnél lefelé pedig 10-es értéket.

A tervezéshez a GuidePace [12] – mobil alkalmazás is használja a Munter módszert más rendszerekkel együtt, hogy segítse a felhasználókat az időbecslések tervezésében és kiszámításában különböző terep típusokra. Ez az alkalmazás különösen az útikalauzok és komoly „backcountry” rajongók számára készült, akiknek pontos időbecsléseket kell készíteniük útjaikhoz.

### 2.3 Mesterséges intelligencia alkalmazása a túraidő becslésben

#### 2.3.1 Mesterséges intelligenciát alkalmazó túra applikációk

Az AllTrails kiemelkedik átfogó funkcióival, amelyek megfelelnek a hétvégi kirándulóknak és a tapasztalt hegymászóknak is. Mesterséges intelligencia felhasználásával az AllTrails valós idejű adatok és felhasználói visszajelzések alapján állítja be a túrák időbecsléseit, biztosítva, hogy a javaslatok mindig az aktuális útviszonyokhoz és az egyéni tempóhoz igazodjanak. Az alkalmazás képessége, hogy integrálja a személyes fittségi szintet és a túrázási preferenciákat az útvonaltervező algoritmusába, lehetővé teszi személyre szabott útiterv kialakítását, amely kihívást jelent, de mégis megvalósítható. További információk az AllTrails weboldalán találhatók [13].

A Komoot részletes útvonal tervezést kínál, fordulóról fordulóra történő hang alapú navigációval, amely különösen hasznos az ismeretlen helyszíneken. Az alkalmazás

mesterséges intelligencia alkalmazásával javítja az útvonal tervezést, figyelembe véve olyan tényezőket, mint a terep nehézsége, az időjárási viszonyok és a túrázó korábbi kirándulásai, annak érdekében, hogy a legalkalmasabb utakat javasolja. Ez a technológia biztosítja, hogy a túrázók újra tervezést valósíthassanak meg, offline állapotban is, így megbízható társ a vadonban történő kalandokhoz. További információk a Komoot weboldalán érhetők el [14].

A PeakVisor egy másik innovatív alkalmazás, amely mesterséges intelligenciát ötvöz kiterjesztett valósággal, hogy valós időben tudja azonosítani a hegyeket a túrázók számára. A kamera vizuális bemenetének elemzésével a PeakVisor képes megjeleníteni az éppen látható hegyek nevét és információit, javítva ezzel a túrázás informatív részét. Az AI vezérelte felület az információkat a felhasználó helyzetéhez és nézőpontjához igazítja, így személyre szabott és interaktív kultéri élményt nyújt. További információk a weboldalon találhatók [15].

### **2.3.2 Mesterséges intelligencia algoritmusok a menetidő meghatározásához**

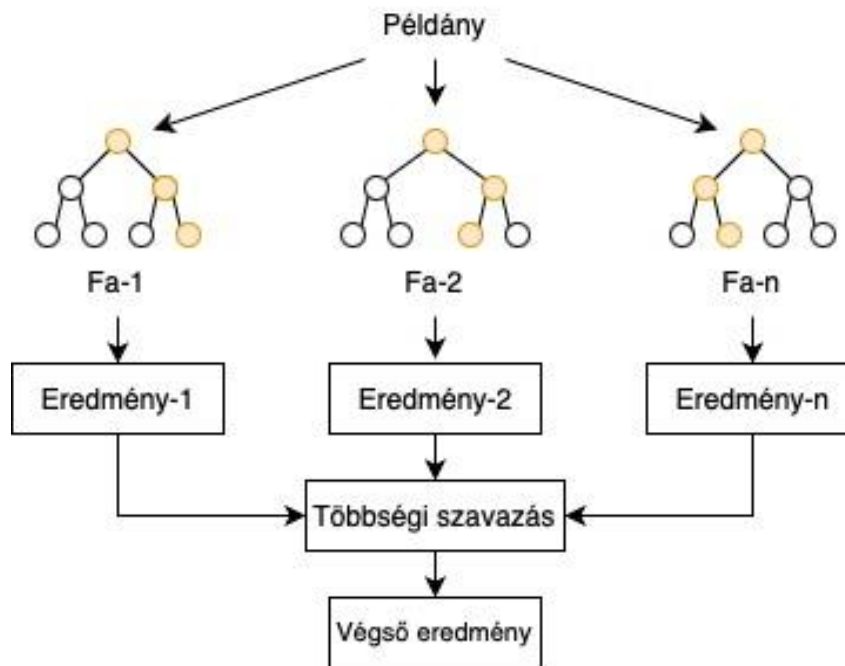
A túra útvonalak időtartamának becslésében kiemelkedő szerepet tudnak játszani a mesterséges intelligencia algoritmusai, különösen az összetett tanítási módszerek, beleértve a Random Forest (2.3.3) és az XGBoost-ot. Ezek a technikák azért hatékonyak, mert több gépi tanulási modell következtetéseit összesítik, növelve ezzel a pontosságot és megbízhatóságot. Ezenkívül neurális hálózatokat is alkalmazhatunk mély tanulási képességeik miatt, amelyek hatékonyan kezelik a bonyolult adatgyűjteményeket és az adat-változásokat, amelyek a különböző terep-viszonyokkal és körülményekkel járnak.

### **2.3.3 Random Forest algoritmus**

A Random Forest algoritmus egy összetett gépi tanulási technika, amelyet osztályozási és regressziós feladatokra egyaránt használnak. Ez a módszer a döntési fák koncepciójára épül, amelyek egyszerű modellek, melyek elágazási módszereket alkalmaznak a döntések meghozatalához a bemeneti jellemzők alapján. A Random Forest javítja a döntési fa teljesítményét úgy, hogy egy "erdőt" hoz létre fákból (3. ábra), melyeket a képzési adatok és jellemzők véletlenszerűen választott rész-halmazai alapján generálnak, majd átlagolják előrejelzéseiket a pontosság növelése és a túltanulás ellenőrzése érdekében [16].

Lényegében a Random Forest az egyes, egymással nem szorosan összefüggő döntési fák előnyét használja fel egy általánosíthatóbb és robusztusabb modell létrehozásához. Az erdő minden egyes fája „szavaz” egy adott előrejelzésre, és a végső kimenetet a többségi szavazat határozza meg osztályozási esetben, vagy az átlag regressziós esetben. Ez a módszer hatékonyan csökkenti a túltanulás kockázatát, amely az egyes döntési fákra jellemző, mivel több előrejelzés átlagolásával kiküszöböli a torzításokat (3. ábra).

Az algoritmus hatékonysága abból adódik, hogy képes nagy adat-halmazokat kezelni magas dimenzionalitással, és nem érzékeny a lényegtelen jellemzők bevonására, így sokoldalú és hatékony eszköz a gépi tanulási módszerek között.



3. ábra: A Random Forest algoritmus működése.

### 2.3.4 XGBoost algoritmus

Az XGBoost [17] egy rendkívül hatékony továbbfejlesztett változata a gradiens fa alapú algoritmusoknak, amely széles körben használt a gépi tanulási feladatoknál, különösen nagy vagy ritka adat-halmazok esetén. Ez az algoritmus több fejlett funkcióval finomítja az alap folyamatokat:

**Szabályozott Tanulási Cél:** Az XGBoost szabályozási kifejezéseket vezet be a cél-függvénybe a túltanulás kontrollálása érdekében, ami gyakori probléma a standard erősítési eljárásoknál. Ez a szabályozás nemcsak a modell teljesítményének javítására szolgál, hanem egyszerűsíti a kapott modellt is, gyorsabbá és hatékonyabbá téve azt.

**Gradiens Fa Erősítés:** Az XGBoost gradiens erősítési keretrendszereket használ az egymást követő fák felépítésére, ahol minden új fa korrigálja az előzőleg betanított fák által elkövetett hibákat. A modell additív módon képződik, vagyis az új fák egyesével adódnak hozzá, és a modellben már meglévő fák nem módosulnak.

**Szórványosság tudatosság:** Az algoritmus úgy van kialakítva, hogy kezelje a valós adatokban előforduló különböző adat-ritkasági mintákat. Ezáltal olyan fa-struktúrát épít, amely figyelembe veszi a hiányzó adatokat, a nullás bejegyzéseket és az úgynevezett „one-hot encoded” [18] jellemzőket, optimalizálva az osztó-pontok megtalálását és alkalmazását a tanítás során.

**Rendszer-tervezés:** Az XGBoost rugalmas és erőteljes rendszer-tervezéssel büszkélkedhet, amely támogatja mind a sűrű, mind a ritka adatokat. A rendszer egyetlen gépen futtatható, vagy több gépen is skálázható egy megosztott környezetben, kihasználva a nagyon nagy adat-halmazok hatékony kezelésének képességét. Összességében az XGBoost skálázhatósága, teljesítménye és a különböző adat-típusok

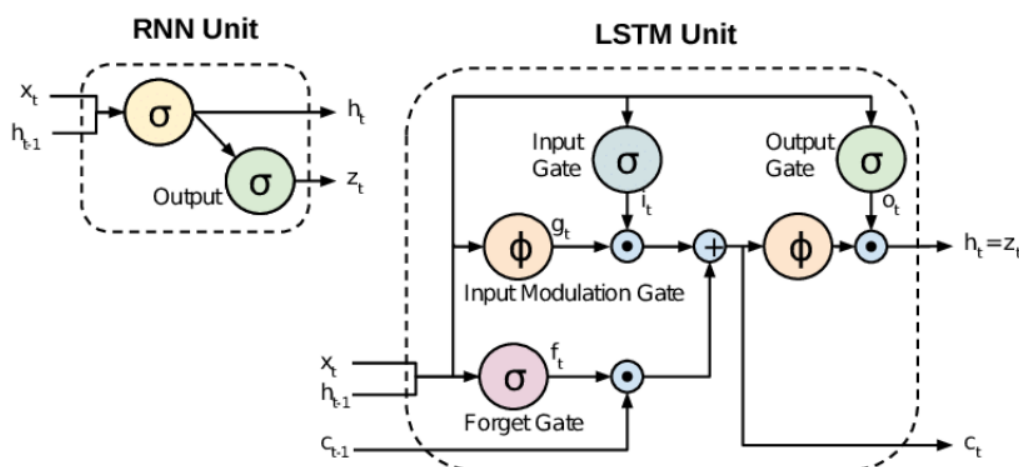
kezelésének képessége miatt népszerű választás a különböző területeken dolgozó adat-tudománnyal foglalkozók körében, amikor összetett prediktív modellezési kihívásokkal kell megbirkózniuk.

### 2.3.5 Recurrent Neural Networks (RNN)

Az ismétlődő neurális hálózatok (RNN-ek) olyan neurális hálózatok osztálya, amelyek sorozatos adatok feldolgozására lettek tervezve. [19] Különösen hatékonyak, mivel fenntartanak egyfajta memóriát azáltal, hogy kimenetüket bemenetként használják a következő lépéshez, ami lét-fontosságú azoknál a feladatoknál, ahol a sorozat korábbi részeiből származó kontextus releváns a későbbi részek feldolgozásához. A nagyvonalú felépítése a 4. ábraán látható.

### 2.3.6 Long Short-Term Memory (LSTM)

A hosszú rövid távú memória-hálózatok, (LSTM-ek) egy fejlettebb típusú RNN (2.3.5), amelyet az elhalványuló gradiens probléma kezelésére fejlesztettek ki, ami a standard RNN-ekben előfordulhat. Ez a probléma megnehezíti az RNN számára, hogy megtanuljon és megőrizzen mintákat hosszú intervallumokkal rendelkező adatokban. Az LSTM-ek ezt úgy oldják meg, hogy beépítenek úgynevezett kapukat: bemeneti, elfelejtési és kimeneti kapukat (4. ábra). Ezek a kapuk határozzák meg, hogy milyen információkat kell megtartani vagy eldobni a hálózaton keresztül, lehetővé téve az LSTM-ek számára, hogy megtanulják az adatok hosszú távú függőségeit [19].



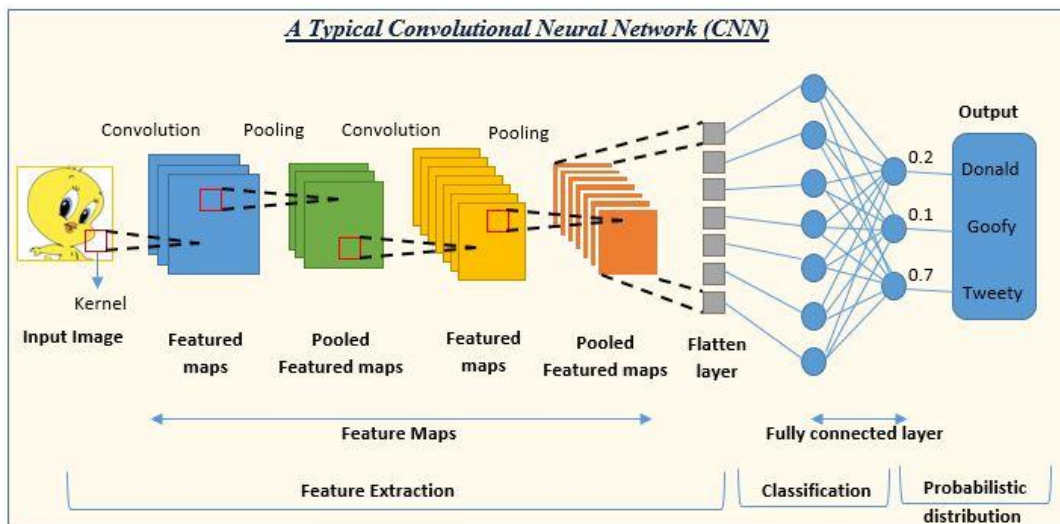
4. ábra: Az RNN és az LSTM architektúrák működése. [20]

Mindkét architektúra értékes a sorozatos adatok feldolgozásának és az idősoros adatokon alapuló előrejelzések készítésének képessége miatt, így alkalmasak olyan alkalmazásokra, mint a beszéd-felismerés, nyelv modellezés és még összetett feladatok, mint például a túra útvonalak időtartamának előrejelzése, ahol az időbeli minták kulcsfontosságúak.

## 2.4 Továbbfejlesztett mesterséges intelligencia modellek és technikák

### 2.4.1 Mélytanulás és konvolúciós neurális hálózatok (CNN) alkalmazása térképes adatokon

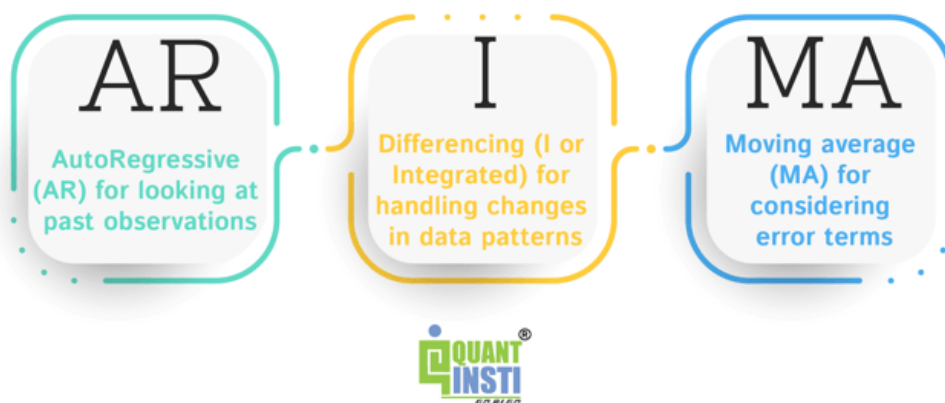
A konvolúciós neurális hálózat (CNN) egy mély-tanulási algoritmus típusa, amely elsősorban olyan adatok feldolgozására szolgál, amelyek rács-hálós topológiával rendelkeznek, mint például a képek. A CNN-ek különösen hatékonyak a képeken belüli mintázatok és szerkezetek azonosításában, ezáltal ideálisak olyan feladatokra, mint a kép-klasszifikáció és objektum-felismerés. Működése egy ábrával van szemléltetve lentebb (5. ábra). A földrajzi adatok esetében a CNN-ek képesek elemzést végezni és értelmezni a műholdképeket, felismerni a tájképi jellemzőket, és hozzájárulni a fejlett térképezési technikákhoz. Az adatok térbeli hierarchiáinak tanulási és felismerési képességük rendkívül hasznos a különböző alkalmazásokban megjelenő összetett kép-elemzési feladatokhoz [21].



5. ábra: A konvolúciós neurális hálózat felépítése. [22]

### 2.4.2 Idősoros elemzések és azok speciális technikái, mint az ARIMA modellek túraidő előrejelzésre.

Az ARIMA (AutoRegressive Integrated Moving Average – autoregresszív integrált mozgó-átlag) modell egy népszerű eszköz a statisztikai elemzésben, különösen az idősoros adatok előrejelzésében. Rendkívül hasznos a jövőbeli adat-pontok előrejelzésére a múltbeli trendek elemzésével. Az ARIMA modell egyedülállóan integrálja a három kulcs-aspektust: az autoregressziót, a differenciálást és a mozgóátlagot (6. ábra). A túraidő becslésében az ARIMA elemezheti a terepen eltöltött idők múltbeli adatait, figyelembe véve változókat, mint az évszak, időjárás és az ösvény állapota, hogy előrejelezze a várható túraidőt. Ez a prediktív képesség teszi az ARIMA-t értékes eszközzé a túraidő becslések pontosságának és megbízhatóságának javításában, segítve a gondosabb időgazdálkodást és a túrázók jobb felkészülését [23].



6. ábra: Az ARIMA modell elemei. [24]

## 2.5 Valós idejű adatfeldolgozás

A valós idejű adat-feldolgozás implementálása a túraidő előrejelzésében több kritikus komponenst igényel az pontos és időszerű információ-szolgáltatás biztosításához. A folyamat az adatok folyamatos gyűjtésével kezdődik érzékelőkön és viselhető eszközökön keresztül, amelyek nyomon követik a földrajzi helyzetet, a környezeti feltételeket és a túrázó biometriai adatait. Ezután az adatokat valós időben közvetíthetik, olyan technológiák segítségével, mint az Apache Kafka [25], amely hatékonyan kezeli a nagy adat-mennyiségeket, vagy egyből feldolgozásra kerülhetnek a készüléken magán.

Az áramoltatott adatok feldolgozásához és elemzéséhez olyan rendszereket is használhatunk, amelyek fejlett technológiával vannak felszerelve, mint az Apache Spark [26] vagy a Hadoop [27]. Ezek a platformok képesek kezelni a nagy adat-készletek valós idejű elemzéséhez szükséges számításokat, biztosítva a szükséges sebességet az azonnali adat-feldolgozáshoz.

A gépi tanulási modellek kulcs-fontosságú szerepet játszanak az adatok értelmezésében. Az RNN-ek vagy az LSTM-ek különösen alkalmasak az idősoros adatok elemzésére. Ezeket a modelleket dinamikusan lehet igazítani a valós idejű adatok alapján, hogy javítsák az előrejelzés pontosságát.

Ezen rendszerek integrálásához API-kat fejlesztettek ki, amelyek különböző forrásokból, mint az időjárás-frissítések (pl. Weather API [28]), vagy a topográfiai változások, amelyek jelentősen befolyásolhatják a túrázási feltételeket, szereznek be adatokat. Az eredmény egy felhasználói felület, amely valós idejű előrejelzéseket és frissítéseket jelenít meg a túrázóknak, javítva navigációs élményüket és biztonságukat.

## 2.6 Adaptív mesterséges intelligencia modellek és a csoportos túrázás

Az AI algoritmusok kulcs-szerepet játszanak a felhasználók változó viselkedéséhez való alkalmazkodásban és a túra útvonalak időtartamának pontos becslésében. Ahogy a túrázók különböző ösvényeken haladnak, tempójuk és állóképességük változhat számos tényező miatt, például fittségi szint, terep nehézsége és időjárási viszonyok. A gépi



tanulási modellek, mint a Random Forest és az XGBoost, múlt-béli adatok alapján képezhetők ki ezeknek a változásoknak a figyelembevételére. Ezek a modellek folyamatosan tanulnak az új adatokból, finomítva előrejelzéseiket a legfrissebb felhasználói viselkedés és környezeti feltételek alapján. Például, ha egy túrázó rendszerint lelassul a meredek emelkedőkön, de egyenletes tempót tart a sík terepen, a modell felismeri ezeket a mintákat a valós idejű adatokban, és ennek megfelelően igazítja az időtartam becsléseit.

Továbbá, a mély-tanulási technikák, beleértve az ismétlődő neurális hálózatokat (RNN) (2.3.5) és a konvolúciós neurális hálózatokat (CNN) (2.4.1), hasznos módszereket kínálnak az idősoros és földrajzi adatok feldolgozására. Az RNN-ek, mivel képesek a szekvenciális adatok kezelésére, képesek nyomon követni a túrázó sebességében bekövetkező változásokat, és előre jelezni a jövőbeni teljesítményt a múltbeli viselkedés alapján. A CNN-ek ezzel szemben kiválóan értelmezik a térbeli adatokat, mint például az ösvény-térképeket és a magassági profilokat, hogy megértsék, hogyan befolyásolják az ösvény különböző szakaszai a túra időtartamát. A gépi tanulás és a mély-tanulás kombinációja lehetővé teszi a túrázók teljesítményének árnyalt elemzését, növelve az időtartam-becslések pontosságát.

A csoportos túrázás további bonyodalmakat eredményez, amelyek befolyásolhatják a modell hatékonyságát. A csoport-dinamika, mint a különböző fittségi szintek és eltérő tempók, következtelen sebességi mintázatokat eredményezhet. Az AI algoritmusoknak figyelembe kell venniük ezeket a változatosságokat, több csoport-tag adatainak integrálásával. Ezt elérhetjük olyan összetett tanulási technikák alkalmazásával, ahol az egyes modellek előrejelzéseit összesítik, hogy pontosabb becslést készítsenek. Továbbá, a modell tanulhat a csoportos viselkedési mintákból az idő múlásával, úgy igazítva előrejelzéseit, hogy azok a csoport kollektív tempóját tükrözzék, nem pedig az egyéni teljesítményeket. Ez a megközelítés nagyban javítja a pontosságot.

A valós idejű felhasználói visszajelzések integrálása tovább finomíthatja a modelleket. A túrázók bevihetik az ösvények nehézségi szintjét, pihenőidőket és egyéb releváns információkat, lehetővé téve az algoritmusok számára, hogy dinamikusán igazítsák az előrejelzéseket. Ez a visszacsatolási kör biztosítja, hogy a modellek érzékenyek maradjanak a csoportos túrázás egyedi kihívásaira, mint például a pihenőidők koordinálása és a résztvevők eltérő energia-szintjének kezelése.

Összességében, az AI algoritmusok felhasználói viselkedés változásaihoz és a csoportos túrázás bonyolultságaihoz való alkalmazkodása elengedhetetlen a pontos időtartam-becsléshez. A gépi tanulási és mély-tanulási technikák kihasználásával ezek a modellek folyamatosan tanulhatnak és igazíthatók, megbízható és személyre szabott előrejelzéseket tudnak biztosítani a túrázók számára. A valós idejű adatok és a felhasználói visszajelzések integrációja tovább növeli a modellek hatékonyságát, biztosítva, hogy azok figyelemmel maradjanak a túrázási tevékenységek sokféleségére és dinamikájára.

## **2.7 Adatvédelmi és etikai szempontok**

Az adat-védelem és a felhasználói adat-védelem biztosítása kiemelten fontos az AI-alapú túrázási alkalmazások fejlesztése és bevezetése során. Ezek az alkalmazások gyakran gyűjtenek érzékeny információkat, beleértve a GPS hely-adatokat, a viselhető eszközök biometrikus adatait és a felhasználók által generált tartalmakat. Adatvédelmi intézkedések bevezetése elengedhetetlen ezeknek az információknak védelme és a felhasználói bizalom fenntartása érdekében.

### **2.7.1 Titkosítás**

Először is, az adatok titkosítása kritikus fontosságú. Az összes adatot, amelyet a felhasználó eszköze és az alkalmazás szerverei között továbbítanak, ipar-ági szabványú protokollokkal, például SSL/TLS (Secure Sockets Layer/Transport Layer Security) használatával kell titkosítani [29]. Ez biztosítja, hogy még ha az adatokat a továbbítás során elfogják is, azok olvashatatlanok maradjanak illetéktelen személyek számára. Ezenkívül a szervereken tárolt adatokat nyugalmi állapotban is kell titkosítani, további védelmet nyújtva az adatvédelmi incidensek ellen.

### **2.7.2 Hozzáférés-vezérlés**

Szigorú hozzáférés-vezérlés bevezetése szükséges az érzékeny adatokhoz való hozzáférés korlátozása érdekében. Ilyenek hitelesítési mechanizmusok, például a többletgyőző hitelesítés használatát a felhasználói identitások ellenőrzésére, valamint a szerepkör alapú hozzáférés-vezérlést (Role-based access control) (RBAC) [30], az egyes felhasználói szerepekhez rendelt hozzáférések korlátozására. Csak az engedélyezett személyek férhetnek hozzá a kritikus rendszerekhez és felhasználói adatokhoz, minimalizálva a belső adatszivárgások kockázatát

### **2.7.3 Adatvédelmi szabályok betartása**

További fontos szempont az adatvédelmi szabályozásoknak való megfelelés, mint például az Európai Unió Általános Adatvédelmi Rendelete (GDPR) [31] és az Egyesült Államokban a Kaliforniai Fogyasztói Adatvédelmi Törvény (CCPA) [32]. Ezek a szabályozások átláthatóságot követelnek meg az adatgyűjtési gyakorlatokban, és megkövetelik az alkalmazásoktól, hogy hozzájárulást szerezzenek a felhasználóktól adataik gyűjtése előtt. Emellett a felhasználóknak lehetőséget kell biztosítani arra, hogy megtekinthessék, módosíthassák és törölhessék adataikat, ezáltal biztosítva számukra az irányítást személyes adataik felett.

### **2.7.4 Anonimitás**

Az adat anonimizálása kulcsfontosságú módszer a felhasználói adatvédelem biztosítására azáltal, hogy eltávolítja vagy titkosítja a személyazonosításra alkalmas információkat (PII) az adatállományokból, így megelőzve az egyének azonosítását. Ez a folyamat biztosítja az adatok bizalmasságát és a GDPR (Általános Adatvédelmi Rendelet)

és HIPAA (Egészségbiztosítási Hordozhatósági és Elsődleges Számviteli Törvény) szabályozásoknak való megfelelést [33], és lehetővé teszi az adatok biztonságos megosztását kutatás és elemzés céljából az adatvédelem veszélyeztetése nélkül. Az adatmaszkolás, amely egy specifikus anonimizálási technika, az érzékeny információkat fiktív adatokkal helyettesíti, megőrizve az eredeti adat formátumát, de potenciálisan kockáztatva az újra-azonosítást, ha nem megfelelően hajtják végre. A generalizáció csökkenti az azonosítás kockázatát azáltal, hogy a konkrét értékeket általánosabb kategóriákkal helyettesíti, például korcsoportokkal, ami segít megvédeni a magánszférát, de csökkentheti az adatok hasznosságát a részletes elemzéshez [33].

#### **2.7.5 Biztonsági auditok**

A rendszeres biztonsági auditok és sérülékenységi értékelések elvégzése elengedhetetlen a potenciális biztonsági kockázatok azonosításához és mérsékléséhez. Ezeknek az értékeléseknek az elvégzése segít biztosítani, hogy az alkalmazás ellenálló maradjon a felmerülő fenyegetésekkel szemben, és megfeleljen az adatvédelem legjobb gyakorlati irányelveinek.

#### **2.7.6 Adatvédelem összefoglalva**

Az AI alapú tárházi alkalmazásokban a felhasználói adatok védelme rendkívüli odafigyelést igényel, amely magában foglalja a titkosítást, a hozzáférés-vezérlést, a szabályozási megfelelést, az anonimizálást és a rendszeres biztonsági auditokat. Ezeknek az intézkedéseknek a prioritásként kezelése révén a fejlesztők biztosíthatják a felhasználói adatvédelem megőrzését, miközben kihasználják az AI előnyeit a tárházi élmények javítására.

### **2.8 Biztonsági megfontolások és hibakezelés**

Az AI alapú alkalmazásokban, különösen a túraútvonal időtartamának előrejelzésére szolgáló rendszerekben, a hibakezelés és a biztonsági protokollok kritikusak a felhasználók védelme érdekében. A hamis előrejelzések ugyanis kellemetlen helyzetekhez vezethetnek. A jól kidolgozott hibakezelési mechanizmusok és biztonsági protokollok bevezetése biztosítani tudja az előrejelzések megbízhatóságát és pontosságát, egyúttal megőrizve a felhasználók bizalmát és biztonságát.

#### **2.8.1 Adatellenőrzés és -tisztítás:**

Az adatok integritásának biztosítása kulcsfontosságú. Az adatellenőrzési folyamatok anomáliákat, ellentmondásokat és hibákat keresnek a valós idejű adatbemenetekben, például a GPS, időjárás API-k és felhasználói eszközök adataiban. Az automatizált tisztítási rutinok korrigálják vagy jelzik a pontatlan adatokat, mielőtt azok befolyásolhatnák az előrejelzéseket.

#### **2.8.2 Algoritmus robusztussága:**

Elengedhetetlen olyan robusztus gépi tanulási modellek alkalmazása, amelyek képesek kezelni a zajos vagy hiányos adatokat. Az összetett tanulási technikák, mint

például a Random Forest (2.3.3), javíthatják az előrejelzések pontosságát azáltal, hogy több modell erősségeit kombinálják és csökkentik a hibás adatok hatását.

### **2.8.3 Valós idejű megfigyelés és riasztások:**

Valós idejű megfigyelő rendszerek bevezetése, amelyek nyomon követik az előrejelzések kimeneteit, segítheti az anomáliák azonosítását és korrekcióját. Riasztások indíthatók, ha az előrejelzések jelentősen eltérnek a várható tartományoktól, további vizsgálatot vagy korrekciós intézkedéseket kezdeményezve.

### **2.8.4 Titkosítás:**

Az összes adatot, amelyet a felhasználói eszközök és az alkalmazáserverek között továbbítanak, SSL/TLS protokollokkal kell titkosítani. Ez megakadályozza az illetéktelen hozzáférést és az adatok manipulálását, amely hamis előrejelzésekhez vezethetne (2.7.1).

### **2.8.5 Hozzáférés-vezérlés:**

Szigorú hozzáférés-vezérlési intézkedések bevezetése biztosítja, hogy csak az engedélyezett személyek és rendszerek férjenek hozzá és módosíthassák az előrejelzési algoritmusokat és az adatokat. A több lépcsős hitelesítés (MFA) és a szerepkör-alapú hozzáférés-vezérlés (RBAC) hatékony stratégiák a biztonság növelésére (2.7.2).

### **2.8.6 Rendszeres auditok és frissítések:**

A rendszeres biztonsági auditok és frissítések elvégzése elengedhetetlen a sebezhetőségek azonosításához és annak biztosításához, hogy a rendszer megfeleljen a legújabb biztonsági szabványoknak. Ez magában foglalja a szoftverek frissítését, ismert sebezhetőségek javítását és a biztonsági protokollok naprakészen tartását.

### **2.8.7 Anomáliaérzékelés:**

Fejlett anomáliaérzékelő rendszerek bevezetése az adatokban lévő szokatlan mintázatok azonosítására, amelyek hibákat vagy potenciális biztonsági megsértéseket jelezhetnek. Gépi tanulási modellek képezhetők az ilyen anomáliák felismerésére és a megfelelő intézkedések megtételére azok előrejelzésekre gyakorolt hatásának megelőzése érdekében.

## **2.9 Adatforrások és adatgyűjtés**

A túra hátralévő idejének pontosabb becsléséhez különböző forrásokból és adattípusokból gyűjthetők információk, amelyek figyelembe veszik a túrázás fizikai és környezeti aspektusait.

Egy lista a lehetséges adatforrásokról:

GPS-adatok: Követi a túrázó aktuális pozícióját, a megtett útvonalat és az időbeni mozgásmintákat.

Topográfiai térképek: Elemzi a magassági profilokat és a tereptípusokat a hátralévő ösvény szakaszok nehézségének felmérésére.

**Időjárás-jelentések:** Szerez valós idejű időjárás-frissítéseket az időbecslések olyan feltételek alapján történő módosításához, mint az eső, hó vagy a hőmérsékleti szélsőségek.

**Fitness okos eszközök:** Használ adatokat olyan viselhető eszközökről, amelyek mérhetik a túrázó szívritmusát, tempóját és általános fizikai megterhelését.

**Mobil túraalkalmazások:** Használ alkalmazásokat, amelyek útvonal információkat, felhasználói véleményeket és valós idejű frissítéseket nyújtanak az ösvény állapotáról.

**Korábbi túraadatok:** Gyűjt adatokat már megtett túrákról ugyanazon az ösvényen az idő becslésére a múltbéli összesített átlagok alapján.

**Környezeti tulajdonságok kézi bevitele:** Fogad információkat a felhasználótól, az olyan feltételekről, mint az ösvény nedvessége, a hó mélysége vagy a hőmérséklet.

**Tervezett pihenők:** Fogadja a felhasználó által előre beütemezett pihenő információit, például útvonalon kiválasztva (pl. A Hármashatár hegy tetején 30 perc), vagy időben kiválasztva (pl. Délben 1 óra), ezeket nemcsak hozzáadja, hanem figyelembe veszi a túrázó kipihentsége és tempóváltozása szempontjából is.

**Műholdas képek:** Elemzi a képeket a terep és a táj változásainak megértésére, amelyek befolyásolhatják a túrázás sebességét.

**Népszerű helyeken kihelyezett webkamerák képe:** Férjen hozzá webkamerák felvételeihez valós idejű vagy a legutóbbi információkért a természet állapotáról.

### **2.9.1 Nyilvános adatbázisok és API-k használata**

Számos nyilvános adatbázis és egyéb eszköz áll rendelkezésünkre, ahonnan információt szerezhetünk. Például a következők:

A Google Maps API egy használat alapú árazási modellt kínál. A Google biztosít korlátozott ingyenes használatot is, amely havonta 200 dollár értékű kreditre fordítható, ami jelentős számú API-hívást fedezhet, attól függően, hogy milyen szolgáltatásokat vesz igénybe a felhasználó [34].

Az OpenWeatherMap API rendelkezik ingyenes és fizetős szolgáltatásokkal is. Az ingyenes csomag elegendő funkciót kínál az alapvető alkalmazások számára, de korlátozott percenkénti lekérdezésszámmal. A fizetős tervek több lekérdezést és további adatszolgáltatásokat is kínálnak [35].

A Mapbox szintén kínál ingyenes és fizetős szolgáltatásokat. Az ingyenes csomag jelentős mennyiségű API-kérést tartalmaz, de ha magasabb keretekre vagy vállalati szolgáltatásokra van szüksége a felhasználónak, akkor fizetős csomag előfizetésére lehet szükség [36].

A Strava API általában ingyenesen használható a nyilvános információkhoz vagy a felhasználó által birtokolt adatokhoz való hozzáférésre, de kiterjedt használat vagy kereskedelmi célok esetén kapcsolatba kell lépni velük további részletekért [37].

## 3 RENDSZERTERV

### 3.1 Bevezetés a rendszertervbe

A túraútvonal időtartam-becslő szoftver fejlesztéséhez olyan frontend és backend eszközöket és módszereket kell használni, amelyek biztosítják a funkcionalitást, skálázhatóságot és felhasználóbarát működést.

#### 3.1.1 A fejlesztendő alkalmazás célja és funkcionális követelményei

Az elsődleges célja a túraútvonal időtartam-becslő alkalmazásnak, hogy pontos, valós idejű előrejelzéseket nyújtson a túrák időtartamáról a felhasználói adatok és dinamikus információk alapján. Az alkalmazás célja, hogy javítsa a túrázási élményt megbízható becslésekkel és személyre szabott kalkulációkkal, biztosítva, hogy a felhasználók hatékonyan és biztonságosan tervezhessék meg túraikat.

Ennek elérése érdekében az alkalmazás átfogó funkciókat kínál, kezdve a GPX fájlok importálásának és kezelésének lehetőségével, amelyek részletes útvonal-információkat tartalmaznak. Az alkalmazás két túraidő-számítási módot támogat: egy egyszeri becslést az előzetes tervezéshez, és valós idejű frissítéseket, amelyek nyomon követik a haladást és dinamikusan állítják be a becsült érkezési időt a túra során.

A felhasználók ütemezhetik a tervezett pihenőket és megállókat, akár időpontok, akár helyszínek megadásával, amelyeket az alkalmazás beépít az összesített időtartam-számításba.

Emellett az alkalmazás lehetővé teszi több személyiség (persona) létrehozását és tárolását, mindegyik saját túrázási preferenciákkal és képességekkel, biztosítva a személyre szabott időtartam-becsléseket. Az alkalmazás támogatja a csoportos túrázást is, lehetővé téve, hogy a felhasználók több személyiséget adjanak hozzá egyetlen túrához, az előrejelzéseket a csoportdinamika alapján igazítva.

A valós idejű nyomon követési funkciók figyelik a felhasználó előrehaladását az ösvényen, értesítéseket küldve a tervezett ütemezéstől való jelentős eltérésekről vagy biztonsági aggályokról.

A felhasználói felület (UI) intuitív és felhasználóbarát lesz, megkönnyítve a fájlok importálását, a beállítások kezelését, valamint az útvonalak, becsült idők és előrehaladási frissítések egyértelmű vizuális megjelenítését.

Ezeknek a funkcionális céloknak az elérésével az alkalmazás átfogó eszközként szolgál a túrázók számára, javítva azon képességüket, hogy magabiztosan és pontosan tervezzék és élvezzék szabadtéri kalandjaikat. A frontend és backend technológiák, mint például a React vagy Vue.js az UI-hoz, és a Node.js Express-szel vagy a Django a szerveroldali logikához, robusztus kombinációja biztosítja a skálázható, biztonságos és hatékony rendszert, amely megfelel a modern túrázók igényeinek.

### 3.1.2 Fontos kérdés: Online szervereken vs Offline működés

Több igen kritikus érv szól amellett, hogy az online szervereken futó számítások kényelme és teljesítménybeli lehetőségei helyett offline, a készülék határain belül legyenek a számítások futtatva. Ilyen érv például a megbízhatóság, különösen a mérsékelt térerővel rendelkező területeken, a felhasználó adatbiztonsága, ami jelentős kockázatoktól nyer szabadulást, ha az információ ki sem jut a készülékből, és nem utolsósorban a szerver költségek és szerver infrastruktúra implementálásának elhagyhatósága. Negatívumként természetesen szerepel az, hogy a készülék akkumulátora nagy valószínűséggel magasabb megterhelést kap a számítások elvégzése közben, mint ha a készüléknek csak közvetíteni volna szükséges az adatokat az interneten keresztül.

Szintén egy negatívumként elmondható lenne az is, hogy az offline módszernek köszönhetően, korlátozott mértékű visszajelzés értékű információ állna rendelkezésére a fejlesztőknek az alkalmazás továbbfejlesztéséhez és optimalizálásához, ám ez egyszerűen kiküszöbölhető azzal, hogy utólag szinkronizálja a készülék a megfelelő adatokat, ugyanis nincsen szüksége ezeknek az információknak azonnali feldolgozásra. Figyelni kell viszont a megfelelő titkosításra, de a kockázat legnagyobb része el van hárítva azzal, hogy a készülék élőben nem osztja meg a földrajzi helyzetét a szerverrel, ami ugyanis az egyik legérzékenyebb adat.

### 3.1.3 Megoldás

Kutatásom alapján valószínű, hogy a túraútvonal időtartamának valós idejű kiszámításához szükséges és elegendően összetett gépi tanulási algoritmusok mobil eszközökön offline is tudjanak futni. A mobil és „edge computing” technológiák, mint például a TensorFlow Lite [38] és a PyTorch [39] Mobile, lehetővé tették, hogy kifinomult modelleket közvetlenül mobil eszközökön telepítsünk.

A TensorFlow Lite kifejezetten mobil és edge eszközökre lett tervezve, lehetővé téve a hatékony eszközön történő gépi tanulást olyan optimalizációkkal, mint a kvantálás és a hardveres gyorsítás. Támogatja a modellek offline futtatását, ami elengedhetetlen az olyan alkalmazások számára, amelyek távoli túraútvonalakon működnek, ahol korlátozott az internet-hozzáférés.

A PyTorch Mobile lehetővé teszi a gépi tanulási modellek futtatását mobil eszközökön, támogatva az offline funkcionalitást. Teljesítményre optimalizált, biztosítva a komplex modellek hatékony következtetését, ami elengedhetetlen a valós idejű útvonal követéshez és előrejelzéshez.

A modern mobil eszközök erőteljes hardveres gyorsítókkal vannak felszerelve, amelyek képesek kezelni a valós idejű gépi tanulási algoritmusok számítási igényeit. Ezeknek az algoritmusoknak az offline futtatása fokozza a felhasználói adatvédelmet, csökkenti a kapcsolatfüggőséget, és csökkenti a szerver költségeket. A TensorFlow Lite és a PyTorch Mobile használatával az alkalmazás robusztus, valós idejű előrejelzéseket tud nyújtani, miközben fenntartja a felhasználói adatvédelmet és működési hatékonyságot.

## 3.2 Alkalmazás architektúra

### 3.2.1 Frontend technológiák

**React Native:** A React Native lehetővé teszi mobilalkalmazások építését JavaScript és React használatával, támogatva mind az iOS, mind az Android platformokat. Natív kinézetet és érzést biztosít, széles körű ökoszisztémával és közösségi támogatással, amely biztosítja a szükséges kiegészítők és könyvtárak elérhetőségét. Helyi tárolási megoldásokat, például SQLite vagy Realm használ, az offline adat tároláshoz. Olyan könyvtárak, mint a Redux Offline, kezelhetik az alkalmazás állapotát és a szinkronizálást, amikor az internet kapcsolat helyreáll. A React Native támogatja a hot-reload funkciót, amely lehetővé teszi a fejlesztők számára, hogy valós időben lássák a változtatásokat az alkalmazás teljes újraindítása nélkül, növelve ezzel a fejlesztési sebességet és hatékonyságot [40] [41].

**Flutter:** A Google által fejlesztett Flutter a Dart programozási nyelvet használja, és lehetővé teszi nagy teljesítményű alkalmazások készítését natív érzettel. Hot-reload funkciója és kiterjedt widget könyvtára egyszerűsíti a UI tervezést és gyorsítja a fejlesztést. Támogatja az offline képességeket helyi adatbázisok, például az SQLite használatával, és képes kezelni a valós idejű adatszinkronizálást, amikor a készülék újra csatlakozik. Teljes natív teljesítmény, rugalmas felhasználói felület (UI), és erős widget támogatás a komplex UI tervezésekhez [40].

**Ionic:** Az Ionic egy hibrid mobilalkalmazás keretrendszer, amely lehetővé teszi a platformok közötti alkalmazások fejlesztését HTML, CSS és JavaScript használatával. Az Angularra épül, támogatva a magas minőségű alkalmazásfejlesztést gyorsan. Szervizmunkásokat és helyi tárolási lehetőségeket, például IndexedDB vagy PouchDB használ, amelyek szinkronizálják az adatokat, amikor a készülék online lesz. Gyorsabb alkalmazásfejlesztés, platformfüggetlenség, és egy hatalmas kiegészítő könyvtár más technológiákkal való interfészekhez [40].

### 3.2.2 Backend technológiák

**Node.js és Couchbase Mobile:** A Node.js ideális a skálázható és nagy teljesítményű backend szolgáltatások építéséhez, hatékonyan kezelve a valós idejű adatfeldolgozást. **Offline-First adatbázis:** A Couchbase Mobile magában foglalja a Couchbase Lite-ot helyi tárolásra és a Couchbase Sync Gateway-t a szinkronizálásra, biztosítva az adatok konzisztenciáját internetkapcsolat nélkül is. **Adatszinkronizálás:** Beépített konfliktuskezelést és biztonságos adatszinkronizálási képességeket biztosít [42].

**Firebase:** A Firebase átfogó backend szolgáltatási csomagot kínál, beleértve a valós idejű adatbázisokat, hitelesítést és felhő tárolást. Zökkenőmentesen integrálható mind a React Native, mind a Flutter alkalmazásokkal. A Firestore támogatja az offline adatelérést és szinkronizálást, így kiváló választás az offline-first alkalmazásokhoz. Valós idejű frissítések, könnyű integráció és minimális beállítás szükséges [43].



### **3.2.3 Framework és könyvtárak választása**

A választott frontend framework a React Native lett, bőségesen támogatott felhasználói bázisa, kiforrott rendszere és fejlett képességei miatt (3.2.1).

A választott backend a Node.js lett, kiváló skálázhatósága, backend szolgáltatásai miatt, mely az Express.js keretrendszerrel kombinálva robusztus megoldásokat kínál a szerveroldali műveletek hatékony kezeléséhez (3.2.1).

### **3.2.4 API-k és integrációk**

Az időjárás lekérdezéséhez az OpenWeather-től a Weather API-t fogom használni, mert egy ígéretes megoldásnak tűnik, és napi 1000 lekérdezésig ingyenes a használata [44].

A térképet tekintve amennyiben implementálni fogok térképes megjelenítést, a Google Maps API-t fogom használni [45], az Apple Maps API-val az IOS-es változatban [46].

### **3.2.5 Cloud szolgáltatások és hosting**

Felhő szolgáltatások intézésére a Google Cloud Platformot (GCP) fogom használni. Skálázható számítási teljesítménye, egységesített objektumtárolása és fejlett felhasználói szerepkörök és jogosultságok kezelése miatt [47].

## **3.3 Felhasználók kezelése**

A felhasználók kezelésére és hitelesítésére a Clerk szolgáltatását fogom használni, modern kialakítása és a pozitív vele kapcsolatos tapasztalataim alapján [48].

## 4 KÖVETKEZŐ FÉLÉVI TERVEK

A következő félévben a megvalósításra fogok koncentrálni. Első lépésként elkészítek egy prototípust, amely képes importálni és kezelni a GPX formátumú fájlokat, személyeket (persona) kezelni, és előre megbecsülni az importált túra útvonal idejét. Ezután tesztelni fogom a rendszert úgy, hogy teljesített idősoros GPX formátumú útvonalakból minél többet összegyűjtök. Ezeket a rögzített idő attribútumaiktól elválasztva használok fel a mesterséges intelligencia modellek tanítására.

Amikor ez a prototípus már működik, megkezdem a valós idejű időbecslő rendszer implementálását hasonló stratégiával. Miután ez a funkció is elkészült, alaposan tesztelni és pontosítani fogom a rendszert. Ezt követően megírom a szakdolgozat végleges dokumentációját, amelyben ismertetem az applikáció felépítését és működését, a tesztelések folyamatát és eredményeit, valamint azokat az opcionális módszereket, amelyeket a fejlesztés közben találtam. Végül a felhasználói leírást is tartalmazni fogja, amely segítséget nyújt az applikáció kezeléséhez, valamint a továbbfejlesztési lehetőségeket is bemutatom.

## 5 ÖSSZEFOGLALÁS

Ebben a dolgozatban a célom egy olyan alkalmazás kifejlesztése, amely mesterséges intelligencia (MI) segítségével pontosabb és személyre szabottabb túraidő becsléseket nyújt a túrázók számára. Az alkalmazás figyelembe veszi a felhasználók fizikai kondícióját, az útvonal specifikus jellemzőit, és különböző egyéni és csoportos túrázási helyzeteket.

A dolgozat elején bemutattam a túraútvonalak digitális formátumait, mint a GPX, KML és TCX, valamint ismertettem a hagyományos túraidő-becslési módszereket, például a Naismith-szabályt, a Tobler-függvényt, a Book Time formulát és a Munter-módszert. Ezeket összehasonlítottam, hogy megértsük erősségeiket és korlátaikat különböző túrázási körülmények között.

Ezt követően tárgyaltam az MI alkalmazását a túraidő becslésében, különös tekintettel az olyan algoritmusokra, mint a Random Forest és az XGBoost, valamint a mély-tanulási technikákra, mint az RNN-ek és CNN-ek. Ezek az algoritmusok képesek komplex térbeli és időbeli adatok feldolgozására, így pontosabb becsléseket adnak.

Az alkalmazás rendszertervében kiválasztottam a számomra legmegfelelőbb frontend és backend technológiákat. A frontend fejlesztéséhez a React Native keretrendszert részesítettem előnyben, amely lehetővé teszi a gyors és hatékony mobilalkalmazás-fejlesztést iOS és Android platformokra. A backend-hez a Node.js és a Couchbase Mobile kombinációját választottam, mivel ezek biztosítják a skálázhatóságot és a valós idejű adatfeldolgozást, valamint az offline működést.

Bemutattam az API-k és integrációk felhozatalát is, például a Google Maps API-t a térképi szolgáltatásokhoz és az OpenWeatherMap API-t az időjárási adatokhoz.

Kiemeltem a felhasználók kezelését és az adatvédelmi szempontokat, beleértve a hitelesítési módszereket (Clerk), valamint a GDPR megfelelést.

A dolgozat végén a megvalósítási tervet részleteztem, beleértve a fejlesztési ciklus fázisait és mérföldköveit.

Jelenleg a projekt a fejlesztési fázis kezdetén van, ahol az alapvető funkciók implementálása zajlik. A következő lépések között szerepel a kezdetleges rendszer tesztelése, majd a teljes rendszer integrációja, alapos tesztelése és finomítása, hogy a végleges verzió megfeleljen a felhasználói igényeknek.

## 6 IRODALOMJEGYZÉK

- [1] "gps-wizard - GPX - GPS Exchange Format," [Online]. Available: <https://logiqx.github.io/gps-wizard/formats/gpx.html>.
- [2] "TrailHiking - Using GPX & KML files for hiking," [Online]. Available: <https://www.trailhiking.com.au/navigation/gpx-kml-files-for-hiking/>.
- [3] "Fileformat - What is a TCX file?," [Online]. Available: <https://docs.fileformat.com/gis/tcx/>.
- [4] "Garmin Developer - FIT SDK," [Online]. Available: <https://developer.garmin.com/fit/protocol/>.
- [5] "Footpath - File types," [Online]. Available: <https://footpathapp.com/user-guide/file-types/>.
- [6] "Strava support - Downloading a GPX Route from other Athlete's Activities," 2023. [Online]. Available: <https://support.strava.com/hc/en-us/articles/216918447-Downloading-a-GPX-Route-from-other-Athlete-s-Activities>.
- [7] Y. Mingyu, F. van Coillie, M. Liu, R. de Wulf, L. Hens and X. Ou, "BioOne Digital Library," *A GIS Approach to Estimating Tourists' Off-road Use in a Mountainous Protected Area of Northwest Yunnan, China*, 2014.
- [8] "Adventure Nerds," [Online]. Available: <https://adventurenerds.com/article/hiking-time-calculator-and-how-to-estimate-hiking-time/>.
- [9] C. D. Higgins, "Findings," 2021. [Online]. Available: <https://findingspress.org/article/28107-hiking-with-tobler-tracking-movement-and-calibrating-a-cost-function-for-personalized-3d-accessibility>.
- [10] J. van Etten, "ResearchGate," [Online]. Available: [https://www.researchgate.net/figure/Toblers-hiking-function\\_fig3\\_305656946](https://www.researchgate.net/figure/Toblers-hiking-function_fig3_305656946).
- [11] "The Hiking Adventure - MASTERING HIKING SPEED: HOW TO CALCULATE YOUR PACE AND TIME," [Online]. Available: <https://thehikingadventure.com/hiking-speed-calculator>.
- [12] C. Burk, "Applied Alpinism," [Online]. Available: <https://www.appliedalpinism.com/guide-pace>.
- [13] "All Trails," [Online]. Available: <https://www.alltrails.com>.
- [14] "Komoot," [Online]. Available: <https://www.komoot.com>.
- [15] "PeakVisor," [Online]. Available: <https://peakvisor.com>.
- [16] G. Louppe, "Cornell University arXiv," 2015. [Online]. Available: <https://arxiv.org/abs/1407.7502>.
- [17] T. Chen and C. Guestrin, "Cornell University - arXiv," 2016. [Online]. Available: <https://arxiv.org/abs/1603.02754>.
- [18] "Codefinity - One-hot encoded," [Online]. Available: <https://codefinity.com/courses/v2/a65bbc96-309e-4df9-a790-a1eb8c815a1c/1fce4aa9-710f-4bc9-ad66-16b4b2d30929/a6d33d0d-3057-4a2f-b8df->

- 4ecd00ffd598?utm\_source=google&utm\_medium=cpc&utm\_campaign=21193856569&utm\_content=&utm\_term=&gad\_source=1&gclid=CjwKCAjwrv.
- [19] A. Sherstinsky, "Cornell University - arXiv," [Online]. Available: <https://arxiv.org/abs/1808.03314>.
  - [20] A. T. NLP, "Data Science Duniya," 2021. [Online]. Available: <https://ashutoshtripathi.com/2021/07/02/what-is-the-main-difference-between-rnn-and-lstm-nlp-rnn-vs-lstm/>.
  - [21] "RunAI - Deep Convolutional Neural Networks," [Online]. Available: <https://www.run.ai/guides/deep-learning-for-computer-vision/deep-convolutional-neural-networks>.
  - [22] S. Shah, "Analytics Vidhya," [Online]. Available: <https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/>.
  - [23] D. Abugaber-Bowman, "Using ARIMA for Time Series Analytics," [Online]. Available: <https://ademos.people.uic.edu/Chapter23.html>.
  - [24] M. Paradkar and C. Thakar, "Forecasting Stock Prices Using ARIMA Model," Quantinsti, 2023. [Online]. Available: <https://blog.quantinsti.com/forecasting-stock-returns-using-arima-model/>.
  - [25] "Kafka," [Online]. Available: <https://kafka.apache.org>.
  - [26] "Apache Spark," [Online]. Available: <https://spark.apache.org>.
  - [27] "Apache Hadoop," [Online]. Available: <https://hadoop.apache.org>.
  - [28] "Weather API," [Online]. Available: <https://www.weatherapi.com>.
  - [29] "Cloudflare - What is SSL?," [Online]. Available: <https://www.cloudflare.com/en-gb/learning/ssl/what-is-ssl/>.
  - [30] A. S. Gillis, "What is role-based access control (RBAC)?," *TechTarget*, 2024.
  - [31] "European Commission - Data Protection," [Online]. Available: [https://commission.europa.eu/law/law-topic/data-protection\\_en](https://commission.europa.eu/law/law-topic/data-protection_en).
  - [32] "California Consumer Privacy Act (CCPA)," 2024. [Online]. Available: <https://oag.ca.gov/privacy/ccpa>.
  - [33] "MonoVM - What Is Data Anonymization & De-Identification in 2024: Is It Truly Anonymous?," 2024. [Online]. Available: <https://monovm.com/blog/data-anonymization-de-identification/>.
  - [34] "Google Maps Platform," 2024. [Online]. Available: <https://mapsplatform.google.com/pricing/>.
  - [35] "meteomatics Weather API," [Online]. Available: <https://www.meteomatics.com/en/weather-api/>.
  - [36] "Mapbox pricing," [Online]. Available: <https://www.mapbox.com/pricing>.
  - [37] "Strava Developers - Rate Limits," [Online]. Available: <https://developers.strava.com/docs/rate-limits/>.
  - [38] "TensorFlow - Mobile & Edge," [Online]. Available: <https://www.tensorflow.org/lite>.

- [39] steerdeep, "Hexmous Journal - Point, Shoot, and Detect: Object Detection with PyTorch Mobile," 2023. [Online]. Available: <https://journal.hexmos.com/pytorch-mobile/>.
- [40] "Technostacks - Most Popular Mobile App Development Frameworks For App Developers," [Online]. Available: <https://technostacks.com/blog/mobile-app-development-frameworks/>.
- [41] A. Gomolinski, "Orangesoft - Realm vs SQLite: Which database to choose in 2024," 2024. [Online]. Available: <https://orangesoft.co/blog/realm-vs-sqlite>.
- [42] "Couchbase Developer - Using Node.js, Couchbase, and Express," [Online]. Available: <https://developer.couchbase.com/tutorial-quickstart-nodejs>.
- [43] "Firebase Documentation - Firestore," [Online]. Available: <https://firebase.google.com/docs/firestore/manage-data/enable-offline>.
- [44] "OpenWeather - WeatherAPI," [Online]. Available: <https://openweathermap.org/api>.
- [45] "Google Maps API," [Online]. Available: <https://developers.google.com/maps/documentation>.
- [46] "Apple Developer - Apple Maps Server API," [Online]. Available: <https://developer.apple.com/documentation/applemapsserverapi/>.
- [47] "Google Cloud - Documentation," [Online]. Available: <https://cloud.google.com/docs>.
- [48] "Clerk," [Online]. Available: <https://clerk.com>.

## 7 ÁBRAJEGYZÉK

1. ábra: Egy létező GPX formátumú felvétel részlete kiegészítők nélkül. _____	7
2. ábra: Tobler függvénye. [10] _____	8
3. ábra: A Random Forest algoritmus működése. _____	11
4. ábra: Az RNN és az LSTM architektúrák működése. [20] _____	12
5. ábra: A konvolúciós neurális hálózat felépítése. [22] _____	13
6. ábra: Az ARIMA modell elemei. [24] _____	14

(1) _____	7
(2) _____	9