Predicția întârzierii zborurilor

Duma Amalia Diana

Inginerie Software

Cuprins

1 Introducere

Predicția întârzierii zborurilor reprezintă un subiect de interes major datorită consecințelor asupra mediului, a economiei și a siguranței. În primul rând, întârzierile zborurilor pot contribui la creșterea emisiilor de carbon și la poluarea aerului, deoarece avioanele consumă mai mult combustibil în timpul așteptării pe pista de decolare sau în timpul zborului în cerc, așteptând permisiunea de aterizare.

Din punct de vedere economic, întârzierile zborurilor generează costuri suplimentare considerabile atât pentru companiile aeriene, cât și pentru aeroporturi. Aceste costuri includ cheltuieli cu personalul suplimentar, reprogramarea zborurilor, compensarea pasagerilor afectați și chiar pierderi financiare datorate pasagerilor care aleg alte companii aeriene sau modalități de transport. În plus, întârzierile frecvente pot afecta reputația companiei aeriene și a aeroportului, ducând la o scădere a încrederii și loialității clienților.

În ceea ce privește siguranța, întârzierile pot crea situații stresante pentru personalul de zbor și pentru pasageri, ceea ce poate conduce la o creștere a riscului de erori operaționale. De asemenea, gestionarea unui număr mare de zboruri întârziate poate pune presiune suplimentară pe controlorii de trafic aerian și pe personalul de la sol, ceea ce poate afecta negativ eficiența și siguranța operațiunilor aeriene.

Având în vedere aceste aspecte, dezvoltarea și implementarea unor sisteme eficiente de predicție a întârzierilor zborurilor devine esențială. Utilizarea tehnologiilor avansate, cum ar fi inteligența artificială și analiza datelor mari, poate ajuta la anticiparea întârzierilor și la luarea de măsuri preventive pentru a minimiza impactul acestora. Astfel, companiile aeriene și aeroporturile pot îmbunătăți calitatea serviciilor oferite, reducând în același timp impactul negativ asupra mediului, economiei și siguranței.

Companiile aeriene sunt printre cele mai afectate entități de întârzierile zborurilor. Acestea trebuie să gestioneze nu doar costurile financiare dar și impactul pe termen lung asupra reputației lor. Pasagerii, pe de altă parte, resimt direct consecințele întârzierilor zborurilor prin pierderea timpului, frustrarea și stresul asociat incertitudinii și disconfortului. Aeroporturile, la rândul lor, se confruntă cu provocări majore în gestionarea întârzierilor. Acestea trebuie să asigure funcționarea eficientă a infrastructurii și a serviciilor.

2 Studiul literaturii de specialitate

În prima lucrare analizată, autorii au realizat predicții privind anulările de zboruri folosind patru algoritmi de clasificare: regresia logistică, support vector machines (SVM), naive Bayes și arborele decizional, bazându-se pe informațiile provenite din 5 milioane de zboruri din Statele Unite în 2016. Au comparat timpul de antrenare al acestor algoritmi în funcție de numărul de noduri dintr-un cluster Spark, precum și acuratețea clasificării, AUC (Area Under Curve) și PR (Precision-Recall) [6].

Rezultatele au evidențiat următoarele:

- Timpul mediu de antrenare al algoritmilor a scăzut pe măsură ce numărul de noduri a crescut, dar acest declin a încetinit începând de la 7 noduri. De exemplu, cu un singur nod, timpul mediu de antrenare a fost de 460 de secunde, iar cu 11 noduri a fost de 43,5 secunde.

- Algoritmul naive Bayes a avut cel mai scurt timp de antrenare cu 1 și 3 noduri, respectiv 386 și 185 secunde. În schimb, SVM a avut cel mai scurt timp de antrenare cu 5, 7, 9 și 11 noduri, respectiv 72, 21, 19 și 18 secunde.
- Acuratețea clasificării a fost de aproximativ 90% pentru SVM și arborele de decizie, în timp ce naive Bayes a avut o acuratețe de doar 50,8%, iar regresia logistică de 62,4%. Comparația se poate observa în figura 1.
- Atât AUC cât și PR pentru algoritmul arborelui decizional au fost cele mai mari, AUC fiind 0.558 și PR 0.439.

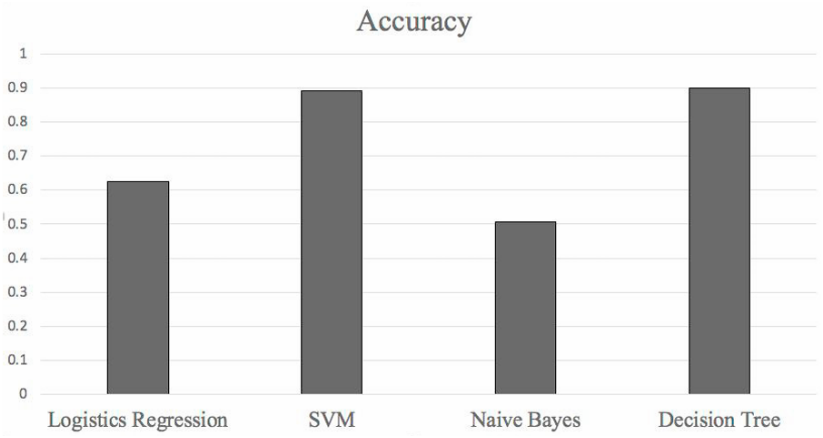


Figura 1: Comparație a preciziei clasificării sub 3 noduri [6]

A doua lucrare citită examinează combinarea cluster computing și data mining pentru a îmbunătăți predicțiile întârzierilor de zbor. Cluster computing a devenit esențial în ultimii ani datorită eficienței sale cost-beneficiu în comparație cu super-computerele. Această cercetare simulează un mediu virtual de cluster computing și implementează un algoritm de regresie logistică pentru a prezice întârzierile zborurilor. Rezultatele arată că utilizarea unui cluster format dintr-un nod master și trei noduri worker poate reduce timpul de calcul cu până la 27,03%. Adăugarea mai multor noduri îmbunătățește și mai mult performanța [5]. O viziune mai detaliată asupra timpului de calcul în funcție de mediu poate fi observată în tabelul 1. Cea mai mare acuratețe obținută a fost de 82.5%.

Nr.	**Mediu**	**Timp de calcul**					
		Experiment					**Media**
		1	**2**	**3**	**4**	**5**	
1	Standalone (3GB RAM)	169.4	166.2	169.3	161.9	167.3	166.82
2	Cluster_1 (2 workers) - @3GB	151.5	152.7	153	169.2	157.8	156.84
3	Cluster_2 (3 workers) - @3GB	129.9	114.8	128.6	118.2	117.1	121.72

Tabela 1: Rezultatele timpului de calcul în funcție de mediu

În a treia lucrare, autorii propun un model de predicție bazat pe Deep Learning (DL), capabil să extragă automat caracteristicile importante din date. Modelul utilizează un auto-encoder și algoritmul Levenberg-Marquart. Pentru evaluare, modelul

a fost aplicat pe un set de date dezechilibrat al zborurilor din SUA, utilizând metoda de undersampling pentru echilibrare. Performanța a fost măsurată prin precizie, acuratețe, sensibilitate, recall și F-measure, comparativ cu alte două structuri: SAE-LM și SDA. Rezultatele, observate în figura 2, au arătat că modelul propus, SDA-LM, oferă îmbunătățiri semnificative față de celelalte modele și față de metoda anterioară RNN, atât pe seturile de date dezechilibrate, cât și pe cele echilibrate [7].

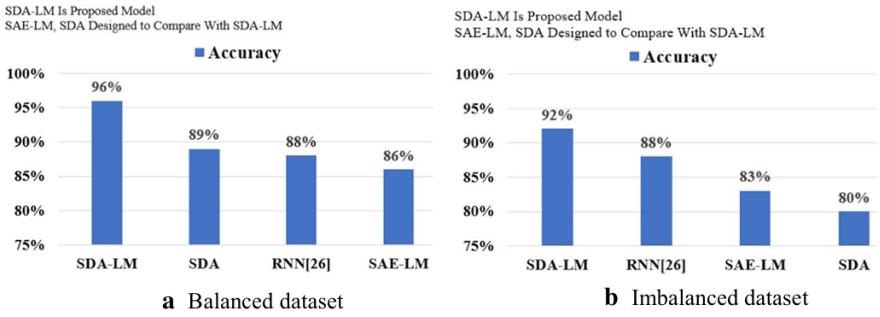


Figura 2: Compararea acurateții între SDA-LM, SAE-LM, SDA și RNN [7]

A patra lucrare propune două modele de predicție a întârzierilor de zbor folosind rețele neuronale convoluționale adânci (DCNN și SE-DenseNet) bazate pe fuziunea datelor meteorologice și de zbor. Modelul DCNN (Dual-channel Convolutional Neural Network) se bazează pe structura rețelei ResNet și utilizează două canale pentru a asigura transmiterea fără pierderi a matricii de caracteristici și pentru a îmbunătăți fluxul rețelei adânci. Modelul SE-DenseNet (Squeeze and Excitation-Densely Connected Convolutional Network) combină avantajele DenseNet și SENet, adăugând un modul SE după fiecare strat de convoluție al blocului DenseNet pentru a îmbunătăți transmiterea informațiilor adânci și recalibrarea caracteristicilor. ezultatele cercetării arată că, prin includerea informațiilor meteorologice, acuratețea modelului a crescut cu 1% comparativ cu utilizarea exclusivă a datelor de zbor. Modelele DCNN și SE-DenseNet au atins o acuratețe de 92.1% și respectiv 93.19% [4]. O viziune detaliată a acurateții modelelor înainte și după fusionarea datelor poate fi observată în figura 3.

Layers	DCNN		SE-DenseNet	
	Flight data	Flight data and meteorological data	Flight data	Flight data and meteorological data
16	89.70	90.89	92.06	92.33
22	90.13	91.97	92.15	92.69
34	90.65	92.02	92.29	92.80
40	90.87	92.15	92.33	93.14
52	91.07	92.24	92.35	93.17
58	91.10	92.26	92.38	93.19

Figura 3: Compararea acurateții modelelor înainte și după fusionarea datelor[4]

Autorii lucrării a cincea propun un model predictiv utilizând algoritmi de învățare supravegheată. Pentru dezvoltarea modelului predictiv, au fost folosite date despre zborurile interne din SUA și date meteorologice din perioada iulie 2019 - decembrie 2019. Algoritmii XGBoost și regresia liniară au fost aplicați pentru a crea modelul. Datele despre zboruri și condițiile meteorologice au fost introduse în model, unde XGBoost a fost utilizat pentru clasificarea binară a întârzierilor și regresia liniară pentru a prezice durata întârzierii. Performanța fiecărui algoritm a fost analizată pentru a determina eficiența în predicția întârzierilor la sosirea zborurilor. Modelul a obținut o acuratețe de 94.2% [1].

A șasea lucrare examinează performanța modelului SVM distribuit folosind Apache Spark, cu accent pe timpul de rulare și acuratețea clasificării. Evaluările au fost realizate utilizând diverse dimensiuni de date și numere de noduri worker în cluster. Rezultatele au arătat că utilizarea a patru noduri worker a fost cea mai eficientă, cu cel mai scurt timp de rulare. În ceea ce privește acuratețea, numărul de noduri worker nu a avut un impact semnificativ, variațiile fiind cauzate de procesul de oversampling aleatoriu aplicat datelor. Algoritmul SVM utilizat cu Spark a demonstrat o performanță bună în procesul de clasificare, atingând o acuratețe maximă de 93,98% [2]. Compararea timpului de rulare în funcție de numărul de noduri și volumul de date se poate observa în figura 4.

Dataset Size	**Running time (min)**			
	1 Worker	**2 Workers**	**3 Workers**	**4 Workers**
1 GB	5.1	5.7	4.2	3.4
1.5 GB	9.5	9.7	6.4	6.3
2 GB	16.3	13.6	11.5	10.2

Figura 4: Timpul de rulare în funcție de numărul de noduri și volumul de date

Ultima lucrare analizează performanța algoritmului pădure aleatorie. Experimentul utilizează un nod master și trei noduri worker cu specificații hardware identice. Rezultatele arată că această simulare poate accelera performanța algoritmului random forest cu până la 35,8%. Performanța poate fi îmbunătățită și mai mult prin adăugarea de noduri suplimentare cu specificații hardware superioare [3]. O viziune mai detaliată asupra timpului de calcul în funcție de mediu poate fi observată în tabelul 2. Cea mai mare acuratețe obținută a fost de 92.7%.

Nr.	**Mediu**	**Timp de calcul**					
		Experiment					**Media**
		1	**2**	**3**	**4**	**5**	
1	Standalone (3GB RAM)	177.5	183.3	178.5	178.3	184.5	180.4
2	Cluster_1 (2 workers) - @3GB	140.0	143.2	145.5	141.4	144.3	142.9
3	Cluster_2 (3 workers) - @3GB	115.9	120.6	117.2	120.9	118.3	118.6

Tabela 2: Rezultatele timpului de calcul în funcție de mediu

3 Metodologie

În această lucrare, au fost utilizate două seturi de date: Flight Delay Dataset și US Weather Events Dataset. Flight Delay Dataset conține informații despre întârzierile zborurilor din perioada 2018-2022, incluzând detalii precum data zborului, aeroportul de origine și destinație, timpul de plecare, distanța, timpul de zbor, timpul de taxi (TaxiIn și TaxiOut), dacă zborul a avut o întârziere de cel puțin 15 minute (ArrDel15), precum și multe alte informații legate de diferite id-uri. În total sunt 61 de coloane. US Weather Events Dataset include evenimente meteorologice din SUA, specificând data de început (StartDate), codul aeroportului (AirportCode), tipul evenimentului meteorologic (Type), severitatea acestuia (Severity) și detalii legate de latitudinea și longitudinea stației meteo care a raportat evenimentul, în total fiind 14 coloane. Pentru această lucrare au fost considerate zborurile și datele meteorologice aferente anului 2021.

3.1 Pre-procesarea datelor

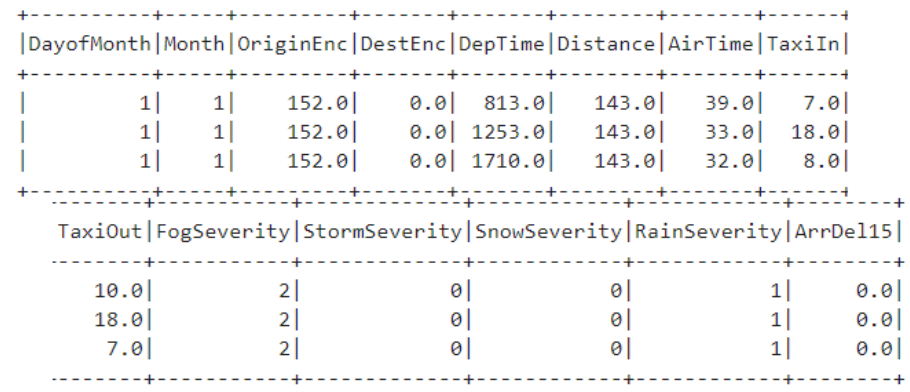
În primul pas, s-a pre-procesat setul de date meteo. Coloana StartDate a fost modificată pentru a păstra doar data, eliminând ora. Au fost păstrate doar coloanele relevante: StartDate, AirportCode, Type și Severity. S-a observat că codul aeroportului avea un 'K' în față, care a fost eliminat pentru consistență. S-a realizat o mapare a severității evenimentului meteorologic astfel: Other și UNK au fost mapate la 0, Light și Moderate la 1, iar Heavy și Severe la 2. Au fost efectuate operații pentru a determina cea mai mare severitate a fiecărui tip de eveniment, precum ceață, furtună, ninsoare și ploaie, pentru fiecare zi și aeroport. Valorile lipsă au fost imputate cu 0, semnificând necunoscut. Versiunea finală a acestui set de date se poate observa în figura 5.

StartDate	AirportCode	FogSeverity	StormSeverity	SnowSeverity	RainSeverity
2021-06-02	NUQ	2	0	0	0
2021-08-02	AMA	2	0	0	0
2021-02-19	BLM	2	0	1	1

Figura 5: Setul de date meteo la finalul pre-procesării

Ulterior, s-a trecut la preprocesarea setului de date despre zboruri. Au fost selectate coloanele esențiale: DayofMonth, Month, FlightDate, Origin, Dest, DepTime, Distance, AirTime, TaxiIn, TaxiOut și ArrDel15. Setul de date despre zboruri a fost îmbinat cu setul de date meteo preprocesat anterior, pe baza coloanelor FlightDate și AirportCode. Valorile lipsă din noul set de date combinat au fost imputate cu 0, iar valorile lipsă din setul de date original despre zboruri au fost eliminate.

Transformarea datelor a inclus conversia atributelor categorice în atribute numerice folosind tehnici de indexare, asigurând astfel că modelul de învățare automată poate interpreta și utiliza aceste date eficient. Versiunea finală a acestui set de date se poate observa în figura 6. În final, setul de date a fost împărțit în 70% pentru antrenare și 30% pentru testare.



DayofMonth	Month	OriginEnc	DestEnc	DepTime	Distance	AirTime	TaxiIn	TaxiOut	FogSeverity	StormSeverity	SnowSeverity	RainSeverity	ArrDel15
1	1	152.0	0.0	813.0	143.0	39.0	7.0	10.0	2	0	0	1	0.0
1	1	152.0	0.0	1253.0	143.0	33.0	18.0	18.0	2	0	0	1	0.0
1	1	152.0	0.0	1710.0	143.0	32.0	8.0	7.0	2	0	0	1	0.0

Figura 6: Setul de date la finalul pre-procesării și îmbinării

3.2 Algoritmi

Pentru predicția întârzierilor zborurilor, au fost utilizați trei algoritmi principali: regresia logistică, arborii de decizie și mașina de suport vectorial (SVM). Acești algoritmi au fost aleși datorită capacității lor de a gestiona date complexe și voluminoase și de a oferi predicții precise în diverse scenarii de clasificare. În plus, s-a dorit compararea cu rezultatele identificate în literatura de specialitate.

Regresia logistică a fost utilizată pentru a modela probabilitatea unei întârzieri binare, arborii de decizie au oferit un model interpretabil prin structura lor ierarhică de decizii, iar SVM a fost folosit pentru capacitatea sa de a găsi hiperplanuri optime în spațiul de caracteristici pentru a separa clasele. Aceste modele au fost antrenate și testate pe seturile de date preprocesate, iar performanța lor a fost evaluată pentru a determina acuratețea și eficiența în predicția întârzierilor zborurilor.

În vederea configurării regresiei logistice, a fost setat parametru **maxBins**=400 pentru a seta numărul maxim de intervale permise pentru caracteristicile continue. Valoarea inițială de 40 era prea mică.

Pentru algoritmul de arbore de decizie, au fost explorate și testate mai multe combinații de parametri pentru a optimiza performanța modelului. S-au experimentat diferite valori pentru adâncimea maximă a arborelui (**maxDepth**), anume 3, 5, 10 și 15, precum și două criterii de impuritate: 'entropy' și 'gini'. După evaluarea performanței fiecărei combinații, s-a constatat că parametrii optimi pentru acest model sunt **maxDepth**=15 și **impurity**='gini', această configurație oferind cele mai bune rezultate în termeni de acuratețe a predicțiilor.

Setarea algoritmul SVM a implicat utilizarea parametrilor **maxIter**=10 și **regParam**=0.1. Parametrul **maxIter**, care specifică numărul maxim de iterații pentru algoritm, a fost setat la 10 pentru a asigura un timp de execuție rezonabil, menținând în același timp o performanță bună a modelului. Parametrul **regParam**, care reprezintă coeficientul de regularizare, a fost stabilit la 0.1 pentru a preveni supraînvățarea modelului, menținând un echilibru între complexitatea modelului și acuratețea predicțiilor. Alegerea acestor valori a fost ghidată de necesitatea de a obține un model eficient și robust, capabil să generalizeze bine pe seturile de date de testare.

4 Configurare

Pentru configurarea locală a Spark-ului, a fost utilizată configurația din figura 7, care asigură alocarea corespunzătoare a resurselor necesare pentru procesarea datelor.

```
spark = SparkSession.builder.appName('FlightsProject')
                    .config("spark.executor.memory", "4g")
                    .config("spark.driver.memory", "4g")
                    .getOrCreate()
```

Figura 7: Configurare Spark local

Pentru implementarea în cloud, s-a folosit Google Cloud Platform. A fost creat un nou proiect și un bucket pentru stocarea datelor. Configurarea clusterului a inclus următoarele specificații: limitat la 8 vCPUs și 500 GB SSD, utilizând instanțele **n2-highmem-2** (2vCPU, 16GB memorie) pentru nodurile Master și Worker. Configurarea Spark-ului în acest mediu s-a realizat exact ca în figura 8.

```
spark = SparkSession.builder.appName('Project')
                    .master("yarn")
                    .config("spark.executor.memory", "6g")
                    .config("spark.executor.cores", "2")
                    .config("spark.executor.instances", "2")
                    .config("spark.driver.memory", "6g")
                    .getOrCreate()
```

Figura 8: Configurare Spark local

5 Rezultate

Rezultatele prezentate în tabelul 3 oferă o imagine clară asupra performanței celor trei modele: regresia logistică, arbori de decizie și svm, analizate în funcție de timpul de rulare, acurateţe atât în mediu local, cât și în cloud.

În ceea ce privește timpul de rulare, toate cele trei modele au înregistrat timpi semnificativ mai buni în mediul local comparativ cu mediul cloud. Regresia logistică a avut un timp de rulare de 62.5 secunde local, față de 153.8 secunde în cloud. Arborii de decizie au avut un timp de rulare de 67.6 secunde local, comparativ cu 157.4 secunde în cloud. SVM a înregistrat 59.9 secunde local, comparativ cu 149.5 secunde în cloud. Această diferență poate fi atribuită mai multor factori. Latenta rețelei joacă un rol important, deoarece comunicarea între componentele sistemului și serverele cloud introduce o întârziere suplimentară. De asemenea, configurația hardware poate diferi semnificativ între sistemele locale și cele din cloud, iar resursele alocate în cloud pot varia în funcție de încărcarea serverelor și alți utilizatori care împart aceleași resurse.

În termeni de acurateţe, arborii de decizie au performat cel mai bine, cu o acurateţe de 85.2%, urmați de regresia logistică cu 84.13% și SVM cu 82.71%. Analizând

ROC_AUC, arborii de decizie au obținut din nou cele mai bune rezultate, cu un scor de 60.90%. ROC_AUC este un indicator important al performanței unui model de clasificare, reflectând capacitatea acestuia de a diferenția corect între clase. În ceea ce privește PR_AUC, Arborii de decizie au avut cel mai bun scor, de 51%.

Model	**Timp de rulare (sec)**		**Acuratețe**	**ROC_AUC**	**PR_AUC**
	Local	**Cloud**			
Regresie logistică	62.5	153.8	84.13%	56.09%	48.67%
Arbori de decizie	67.6	157.4	85.2%	60.90%	51%
SVM	59.9	149.5	82.71%	50%	17.28%

Tabela 3: Rezultate modele

Rezultatele din tabelul 5 oferă o comparație detaliată între modelele proprii și cele studiate în literatura de specialitate. Timpul de rulare este indicat în secunde, iar pentru unele modele este prezentat ca un timp mediu. Este important de menționat că setul de date folosit în această lucrare a fost diferit de cele din lucrări, unele fiind nespecificate. Astfelr rezultatele nu pot oferi o comparație justă a performanței algoritmilor.

Model	**Timp de rulare (sec)**		**Acuratețe**
	Local	**Cloud**	
Regresie logistică_model propriu	*62.5*	153.8	84.13%
Regresie logistică_m1[6]		∼207.25	62.4%
Regresie logistică_m2[5]		156.84	82.5%
Arbori de decizie_model propriu	*67.6*	157.4	85.2%
Arbori de decizie_m1[6]		∼207.25	∼90%
SVM_model propriu	*59.9*	149.5	82.71%
SVM_m1[6]		∼207.25	∼90%
SVM_m2[2]		816	93.98%

Tabela 4: Comparația modelelor proprii cu modelele studiate

6 Concluzie

Această lucrare a avut ca obiectiv analiza și predicția întârzierilor zborurilor utilizând două seturi de date esențiale: Flight Delay Dataset și US Weather Events Dataset. Metodologia a inclus o preprocesare minuțioasă a datelor și aplicarea unor algoritmi de învățare automată pentru a obține rezultate precise.

Preprocesarea datelor a fost un pas crucial în acest studiu, implicând transformarea și combinarea seturilor de date pentru a asigura consistență și relevanță. Datele meteorologice au fost ajustate pentru a reflecta doar informațiile esențiale, iar severitatea evenimentelor meteorologice a fost mapată corespunzător pentru a facilita integrarea cu setul de date al zborurilor. Această integrare a permis crearea unui set de date robust, pregătit pentru aplicarea algoritmilor de predicție.

Trei algoritmi principali au fost utilizați în acest studiu: regresia logistică, arborii de decizie și mașina de suport vectorial (SVM). Aceștia au fost aleși datorită capacității lor de a gestiona date complexe și voluminoase, oferind predicții precise. Configurarea și optimizarea parametrilor fiecărui model au fost realizate cu atenție pentru a maximiza performanța.

Rezultatele obținute au arătat că arborii de decizie au avut cea mai bună performanță în termeni de acuratețe (85.2%) și ROC_AUC (60.90%), fiind urmați de regresia logistică și SVM. În termeni de timp de rulare, toate modelele au avut performanțe mai bune în mediul local comparativ cu mediul cloud, acest lucru fiind atribuit latenței rețelei și configurației hardware.

Compararea cu modelele din literatura de specialitate a evidențiat diferențe semnificative, datorită seturilor de date diferite utilizate și configurațiilor variate ale modelelor. Totuși, modelele dezvoltate în această lucrare au demonstrat o performanță competitivă, evidențiind eficiența metodologiei adoptate.

În continuare, ar trebui dedicat mai mult timp colectării datelor pentru a construi un set de date și mai robust, ceea ce ar putea duce la îmbunătățirea și mai semnificativă a acurateței și eficienței modelelor predictive utilizate. Această lucrare demonstrează că integrarea datelor meteorologice cu datele de zbor și aplicarea algoritmilor de învățare automată poate oferi predicții precise ale întârzierilor zborurilor. Metodologia utilizată și analiza detaliată a performanței algoritmilor oferă o bază solidă pentru cercetări viitoare și aplicații practice în predicția întârzierilor zborurilor.

Bibliografie

[1] N Lakshmi Kalyani, G Jeshmitha, M Samanvitha, J Mahesh, BV Kiranmayee, et al. Machine learning model-based prediction of flight delay. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 577–581. IEEE, 2020.

[2] Husnul Khotimah, Baiq Wilda Al Aluf, Muhammad Ari Rifqi, Ari Hernawan, and Gibran Satya Nugraha. Performance analysis of the distributed support vector machine algorithm using spark for predicting flight delays. In *E3S Web of Conferences*, volume 465, page 02037. EDP Sciences, 2023.

[3] Cinantya Paramita, Catur Supriyanto, Luqman Afi Syarifuddin, and Fauzi Adi Rafrastara. The use of cluster computing and random forest algoritm for flight delay prediction. *International Journal of Computer Science and Information Security (IJCSIS)*, 20(2), 2022.

[4] Jingyi Qu, Ting Zhao, Meng Ye, Jiayi Li, and Chao Liu. Flight delay prediction using deep convolutional neural network based on fusion of meteorological data. *Neural Processing Letters*, 52(2):1461–1484, 2020.

[5] Catur Supriyanto, Fauzi Adi Rafrastara, Yani Parti Astuti, and Lisdi Inu Kencana. Clustered logistic regression algorithm for flight delay prediction. *International Journal of Computer Science and Information Security (IJCSIS)*, 19(2), 2021.

[6] Yu Yanying, Hai Mo, and Li Haifeng. A classification prediction analysis of flight cancellation based on spark. *Procedia Computer Science*, 162:480–486, 2019.

[7] Maryam Farshchian Yazdi, Seyed Reza Kamel, Seyyed Javad Mahdavi Chabok, and Maryam Kheirabadi. Flight delay prediction based on deep learning and levenberg-marquart algorithm. *Journal of Big Data*, 7(1):106, 2020.