

Universitatea  
Transilvania  
din Brașov

FACULTATEA DE INGINERIE ELECTRICĂ  
ȘI ȘTIINȚA CALCULATOARELOR

# PROIECT DE DIPLOMĂ

Conducător științific:  
Șef de lucrări. DĂNILĂ Adrian

Absolvent:  
Jitaru Ionuț-Bogdan

BRAȘOV, 2024

Departamentul de Automatică și Tehnologia Informației  
Programul de studii: Robotică



*Jitaru Ionuț-Bogdan*

# Clasificarea Binară a Sentimentelor, și procesarea răspunsului de către un robot mobil

Conducător științific:  
Șef de lucrări. Dr. Ing. DĂNILĂ Adrian

Brașov, 2024

## FIȘA PROIECTULUI DE DIPLOMĂ

Universitatea Transilvania din Brașov	Proiect de diplomă nr. ....
Facultatea de Inginerie Electrică și Știința Calculatoarelor	
Departamentul de de Automatică și Tehnologia Informației	Viza facultății
Programul de studii: <b>Robotică</b>	Anul universitar: 2023 - 2024
Candidat: <b>Jitaru Ionuț-Bogdan</b>	Promoția: 2024
Conducător științific: <b>Șef de lucrări. Dănilă Adrian</b>	
<b>PROIECT DE DIPLOMĂ</b>	
Titlul lucrării: <i>Clasificarea Binară a Sentimentelor, și procesarea răspunsului de către un robot mobil</i>	
Problemele principale tratate: <ol style="list-style-type: none"> <li>1. Studiu privind stadiul actual al tehnologiei Machine learning si algoritmi svc(support vector classifier).</li> <li>2. Mediul de programare Python;</li> <li>3. Proiectarea unei aplicații de clasificare a documentelor text dintr-o bază de date în Python.</li> <li>4. Elaborarea și implementarea unei metode de validare a rezultatelor obținute</li> <li>5. Elaborarea și redactarea proiectului de diplomă</li> </ol>	
Locul și durata practicii: Laboratorul de Mașini electrice și Acționări, VIII13	
Bibliografia recomandată: <ul style="list-style-type: none"> <li>[1] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... &amp; Rush, A. M. (2019). Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.</li> <li>[2] Pattern Recognition and Machine Learning by C. Bishop, 12.2.1 p. 574 or "http://www.miketipping.com/papers/met-mppca.pdf"</li> <li>[3] Suciu, C. Inteligență artificială, Note de curs</li> </ul>	
Aspecte particulare: - Se va implementa și testa un robot mobil pentru clasificarea propozițiilor text, în funcție de conotațiile emoționale ale acestora;	
Primit tema la data de: 01.10.2023	
Data predării lucrării: 21.06.2024	
Director departament, <i>Prof. Dr. ing. Sorin-Aurel Moraru</i>	Conducător științific,  <i>Șef de lucrări. Dr. Ing. Dănilă Adrian</i>
Candidat, <i>Jitaru Ionuț Bogdan</i> 	

PROIECT DE DIPLOMĂ – VIZE		
Data vizei	Capitole/ problemele analizate	Semnătura conducătorului științific
15.12.2023	Introducere. prezentare teoretică.	<i>Dănilă</i>
8.03.2024	Arhitectura hardware și software.	<i>Dănilă</i>
26.04.2024	Realizarea lucrării.	<i>Dănilă</i>
03.06.2024	Redactarea și editarea proiectului	<i>Dănilă</i>
21.06.2024	<b>Verificare procentaj similaritate Turnitin (total ≤2% și o singură sursă ≤1%)</b>	<i>Dănilă</i>
APRECIEREA ȘI AVIZUL CONDUCĂTORULUI ȘTIINȚIFIC		
<p>1. Nivelul competențelor teoretice în domeniul proiectului dobândite pe durata elaborării temei: (90%) Candidatul a efectuat individual activitățile de cercetare documentară și a acumulat competențe suplimentare în domeniul temei proiectului.</p> <p>2. Calitatea gestionării timpului alocat pregătirii temei de proiectare: (80%) Candidatul a gestionat bine timpul alocat pregătirii temei de proiectare și a participat la toate activitățile programate.</p> <p>3. Gradul de implicare în rezolvarea temei de proiectare: (85%) Candidatul s-a implicat activ în activitatea de rezolvare a temei de proiectare, a dat dovadă de inițiativă și creativitate.</p> <p>4. Gradul îndeplinirii sarcinilor prevăzute în programul de realizare al proiectului: (80%) Candidatul a îndeplinit în totalitate sarcinile prevăzute în program.</p> <p>5. Calitatea redactării proiectului de diplomă: La redactarea proiectului de diplomă candidatul a respectat cerințele din instrucțiunile de redactare elaborate de facultatea IESC</p> <p>Confirm efectuarea stagiului de 90 de ore de practică pentru elaborarea proiectului de licență.</p>		
Data: 21.06.2024	ADMIS pentru susținere	Conducător științific <i>Șef de lucrări. Dr. Ing. Dănilă Adrian</i> <i>Dănilă</i>
AVIZUL DIRECTORULUI DE DEPARTAMENT		
Data: .....	ADMIS pentru susținere/ RESPINS	Director departament, <i>Prof. Dr. Ing. Sorin-Aurel Moraru</i> .....
SUSȚINEREA PROIECTULUI DE DIPLOMĂ		
Sesiunea: iunie-iulie 2024		
Rezultatul susținerii	PROMOVAT cu media:	
	RESPINS <b>cu</b> refacerea lucrării	
	RESPINS <b>fără</b> refacerea lucrării	
Președinte de comisie, <i>Profesor Dr. Ing. Sorin-Mihai GRIGORESCU</i> .....		

# Cuprins

---

## Cuprins

Lista de figuri, tabele și coduri sursă.....	6
Lista de acronime .....	7
1    Introducere .....	8
1.1 Actualitatea și Oportunitatea Temei .....	8
1.1.1 COBOT Applications—Recent Advances and Challenges [1].....	9
1.1.2 A Machine Learning and Deep Learning Framework for Binary, Ternary, and Multiclass Emotion Classification of Covid-19 Vaccine-Related Tweets [2] .....	11
1.1.3 Machine Learning: A Review on Binary Classification Machine Learning: A Review on Binary Classification [3] .....	13
1.2 Scopul proiectului .....	14
1.3 Obiectivele proiectului .....	15
2    Prezentare teoretică.....	17
ROBOTUL MOBIL .....	17
CLASIFICAREA BINARĂ A TEXTELOR .....	17
2.1 Robotul Mobil .....	17
2.2 Clasificarea Binară a Textelor.....	22
2.2.1 Tipuri de Clasificare Binara .....	23
2.2.2 NLP – Natural Language Processer .....	25
2.2.3 Arhitectura unei Rețele Neurale cu Clasificare Binară .....	26
3    Arhitectura Hardware și Software .....	29
3.1 STRUCTURA HARDWARE.....	29
3.1.1 Componente .....	29
3.1.2 Conexiuni Pin Arduino Uno.....	33
3.1.3 Conexiuni Driver Motor L298N .....	33
3.1.4 Conexiuni de Alimentare .....	34
3.2 Structura Software.....	34
3.2.1 Bibliotecile Implementate .....	34
3.2.2 Logica Codului Arduino.....	37
3.2.3 Logica Codului Python.....	40
4    Realizarea Lucrării.....	44
4.1 SETUL DE DATE PE CARE S-A FĂCUT TESTAREA .....	45
4.1.1 Propoziții Pozitive in Engleză .....	45
4.1.2 Propoziții Negative in Engleză.....	45
4.1.3 Propoziții Pozitive in Romană.....	46
4.1.4 Propoziții Negative in Romană .....	46

4.2 Alcătuirea Robotului Mobil .....	47
4.3 CONCLUZII FINALE .....	50
5 Bibliografie .....	52
Rezumat.....	53
Abstract .....	54
Anexa 1 .....	55
DECLARAȚIE PRIVIND ORIGINALITATEA.....	56

## LISTA DE FIGURI, TABELE ȘI CODURI SURSĂ

---

Figura schema 2.1 [1] – Schema Teoretica a unui robot mobil

Figura 2.3 [3] – Exemplu Schema Bloc A Structuri unui Robot

Figura 2.3 [4] - Clasificarea Robotilor Mobili dupa Mediul in care sunt utilizati, cat și tipul lor de miscare

Figura 2.5 [5] - Exemplu Rezultat Clasificare Binara

Figura 2.7 [7] – Exemplu Arhitectura ANN pentru Clasificare Binară

Figura 3.1 – Arduino Uno

Figura 3.2 – Specificații Tehnice Arduino Uno

Figura 3.3 – L298N

Figura 3.4 – L298N Schemă Completă

Figura 3.5 – Motor DC

Cod 3.1 – Snippet din Codul Arduino utilizat in cadrul acestui proiect

Cod 3.2 – Snippet din Codul Arduino utilizat in cadrul acestui proiect

Cod 3.3 – Snippet din Codul Python utilizat in cadrul acestui proiect.

Cod 3.4 – Snippet din Codul Python utilizat in cadrul acestui proiect.

Figura 4.1 – Circuitul Efectiv

Figura 4.2 – Schematica Electrica

*(se vor scrie toate prescurtările utilizate în ordine alfabetică)*

abs – absolut;

IESC - Inginerie Electrică și Știința Calculatoarelor;

AI - Artificial Intelligence (Inteligență Artificială);

NLP - Natural Language Processing (Procesare a Limbajului Natural);

ML - Machine Learning (Învățare Automată);

DL - Deep Learning (Învățare Profundă);

BERT - Bidirectional Encoder Representations from Transformers;

GPT-2 - Generative Pre-trained Transformer 2;

RoBERTa - Robustly optimized BERT approach;

T5 - Text-To-Text Transfer Transformer;

CNN - Convolutional Neural Network (Rețea Neuronală Convoluțională);

SVM - Support Vector Machine (Mașină cu Vectori de Suport);

ANN - Artificial Neural Network (Rețea Neuronală Artificială);

TF-IDF - Term Frequency-Inverse Document Frequency;

RNN - Recurrent Neural Network (Rețea Neuronală Recurentă);

COM - Communication Port (Port de Comunicare);

DC - Direct Current (Curent Continu);

L298N - Model de driver motor pentru curent continuu;

Scikit-learn - Biblioteca pentru învățare automată în Python;

NLTK - Natural Language Toolkit (Set de Uneelte pentru Procesarea Limbajului Natural);

Hugging Face - Platformă pentru modele de învățare automată;

PySerial - Biblioteca Python pentru comunicare serială;



## ACTUALITATEA ȘI OPORTUNITATEA TEMEI

## SCOPUL PROIECTULUI

## OBIECTIVELE PROIECTULUI

---

### 1.1 Actualitatea și Oportunitatea Temei

În contextul tehnologic actual, clasificarea binară în robotică reprezintă un subiect de cercetare deosebit de pertinent, datorită capacității sale de a facilita interacțiuni mai precise și eficiente între roboți și mediul lor operațional.

Cu evoluții recente în tehnologii de procesare și recunoaștere, clasificarea binară este fundamentală pentru implementarea automatizării inteligente.

Un exemplu notabil în acest sens este utilizarea tehnologiei de recunoaștere a imaginilor prin învățare automată, care permite roboților să distingă între obiecte diferite în timp real, o capacitate esențială pentru aplicații în medii variate, de la producția industrială până la asistența personală.

Recentele progrese în robotică evidențiază integrarea crescută a algoritmilor de învățare profundă, care pot clasifica informații complexe provenite de la senzori diversificați. Studiile arată o tendință de creștere a aplicațiilor robotice care integrează percepție multi-modală, un aspect ce poate fi îmbunătățit semnificativ prin clasificarea binară.

Acest fapt nu doar că îmbunătățește performanța roboților în sarcini specifice, dar sporește și siguranța și interactivitatea acestora în medii de lucru alături de oameni.

Oportunitatea cercetării în clasificarea binară în robotică este substanțial amplificată de cererea industrială și comercială pentru sisteme robotice mai inteligente și adaptabile.

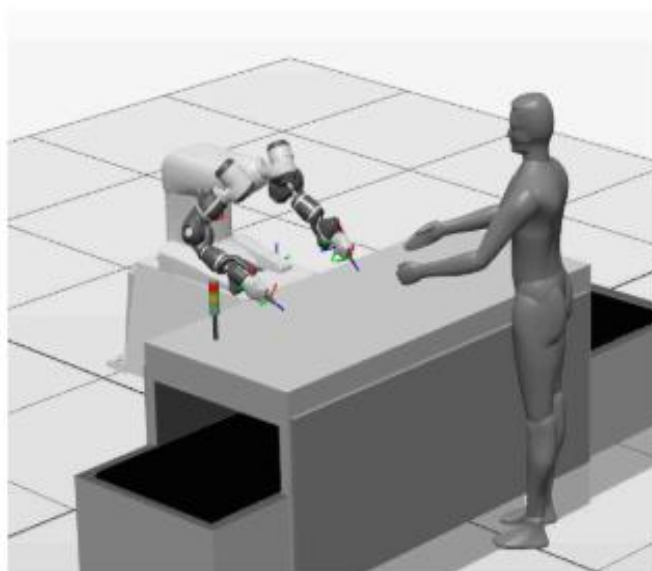
Aplicațiile în sectoare precum manufacturarea, logistică, asistența medicală și chiar interacțiuni personalizate în sectorul serviciilor, necesită soluții la cheie care să interpreteze și să reacționeze în mod autonom la schimbările din mediul lor.

Implementarea clasificării binare în sistemele robotice permite acestor mașini să execute discernerea rapidă și eficientă a informațiilor senzoriale, un pas crucial pentru realizarea autonomiei complete.

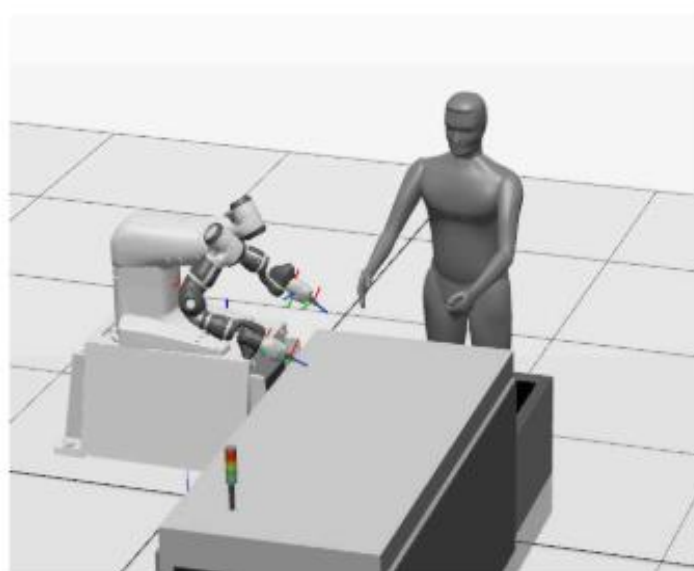
Tendențele în dezvoltarea robotică, cum ar fi îmbunătățirea continuă a algoritmilor de inteligență artificială și învățare automată, deschid noi perspective pentru explorarea și utilizarea clasificării binare în scopuri avansate, cum ar fi navigația autonomă și manipularea precisă a obiectelor, ceea ce poate revoluționa industria robotică și poate contribui la creșterea eficienței operaționale.

Conform Studiilor de Cercetare, s-au inventat o multitudine de metode de utilizare a clasificatoarelor binare în robotica, atât în robotica cât și în afara acestui domeniu.

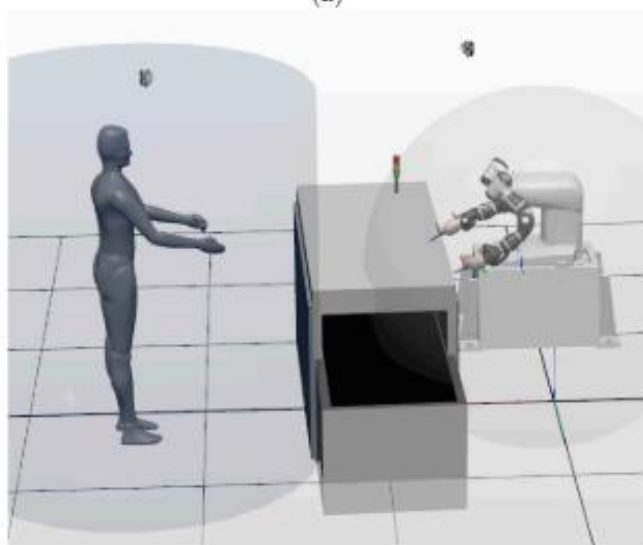
### 1.1.1 COBOT Applications—Recent Advances and Challenges [1]



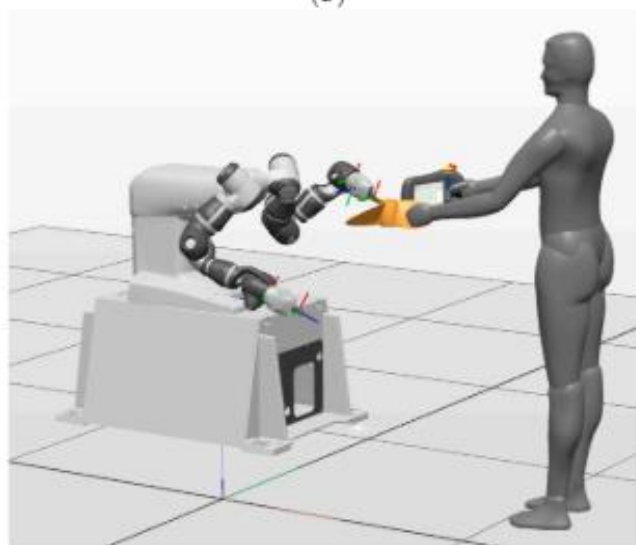
(a)



(b)



(c)



(d)

Figura 1.1 [0] – Reprezentare Conceptelor de Cobot

Roboții colaborativi (COBOTs) reprezintă o clasă de tehnologie robotică ce permite interacțiunea directă și sigură între oameni și mașini în spații de lucru comune.

Acești roboți sunt proiectați pentru a opera alături de lucrători umani, ajutând la îmbunătățirea eficienței și la reducerea efortului uman în sarcini repetitive sau periculoase. Importanța COBOT-urilor în industrie a crescut semnificativ, dat fiind potențialul lor de a transforma procesele de producție prin flexibilitate și adaptabilitate.

Studiul se concentrează pe evaluarea performanței COBOT-urilor prin clasificare binară, analizând precizia și capacitatea de încărcare în contextul cerințelor specifice ale industriilor.

Obiectivul principal este de a determina care modele de COBOT îndeplinesc criteriile de performanță de top, și astfel, de a oferi o bază de date actualizată și relevantă pentru deciziile de achiziție și implementare în producție.

Clasificarea binară, în contextul acestui studiu, implică evaluarea fiecărui COBOT în funcție de două categorii principale: conformitate și non-conformitate.

Conformitatea se referă la capacitatea COBOT-ului de a întâlni sau depăși parametrii specificați de performanță, cum ar fi precizia mișcărilor și capacitatea de încărcare, în timp ce non-conformitatea indică deficiențe în aceste criterii.

Datele pentru acest studiu au fost colectate de la diferiți producători de COBOT-uri, incluzând specificații tehnice detaliate, rapoarte de performanță și feedback de la utilizatorii finali. Aceste date au fost agregate într-o bază de date comprehensivă pentru analiză.

Utilizând algoritmi statistici și tehnici de învățare automată, datele au fost prelucrate pentru a identifica modele și tendințe.

Analiza a implicat aplicarea unor metode de clasificare binară pentru a categorisi fiecare COBOT ca fiind 'conform' sau 'non-conform'. Acest proces a permis distilarea unor insight-uri clare referitoare la care modele îndeplinesc cele mai înalte standarde industriale.

Rezultatele preliminare au fost validate prin teste de performanță suplimentare și prin consultări cu experți în robotică. Acest pas a asigurat acuratețea și fiabilitatea concluziilor studiului.

Analiza a relevat o varietate largă de performanțe între diferitele modele de COBOT-uri. Modelele clasificate ca 'conforme' au demonstrat o precizie remarcabilă în execuția sarcinilor, o capacitate consistentă de încărcare fără eroare, și au fost identificate ca fiind de încredere pentru implementarea în medii de producție complexe.

Pe de altă parte, modelele 'non-conforme' au oferit oportunități de îmbunătățire în design și funcționalitate.

Rezultatele acestui studiu sunt esențiale pentru producătorii de COBOT-uri și industriile utilizatoare. Ele oferă o bază solidă pentru decizii informate legate de selecția tehnologiilor de automatizare, orientând astfel investițiile către cele mai eficiente și eficace soluții.

De asemenea, identificarea punctelor slabe permite producătorilor să îmbunătățească designurile viitoare, încorporând feedback-ul din utilizarea reală pentru a crește funcționalitatea și acceptabilitatea COBOT-urilor în diverse medii industriale.

Această analiză detaliată a COBOT-urilor prin clasificare binară subliniază valoarea acestei metode în evaluarea tehnologiei, demonstrând că clasificarea binară nu este doar un instrument teoretic, ci și unul practic, aplicabil în optimizarea continuă a producției industriale.

#### **1.1.2 A Machine Learning and Deep Learning Framework for Binary, Ternary, and Multiclass Emotion Classification of Covid-19 Vaccine-Related Tweets [2]**

În era digitală, rețelele sociale au devenit un barometru esențial pentru sentimentele publice, oferind o perspectivă valoroasă asupra reacțiilor colective față de evenimente globale, cum ar fi răspunsul la pandemia COVID-19 și campaniile de vaccinare asociate.

Analiza emoțională a tweet-urilor poate oferi insight-uri semnificative în percepțiile publice, ceea ce este crucial pentru autorități și organizații în gestionarea comunicării și în formarea politicilor de sănătate publică.

Studiul inițiat de Aditya Dubey vizează clasificarea emoțiilor exprimate în tweet-urile legate de vaccinul COVID-19, utilizând metode de învățare automată pentru a segmenta reacțiile în categorii binare (pozitive și negative).

Acest demers urmărește să capteze și să analizeze sentimentele predominante, facilitând o mai bună înțelegere a dinamicii sociale în timpul unei crize de sănătate publică.

Folosind un set de date compus din mii de tweet-uri colectate pe parcursul lansării campaniei de vaccinare, studiul aplică algoritmi de procesare a limbajului natural și învățare automată pentru a clasifica conținutul textual.

Algoritmul Naive Bayes Multinomial, în combinație cu tehnica de vectorizare TF-IDF (Term Frequency-Inverse Document Frequency), a fost utilizat pentru a determina probabilitățile asociate cu fiecare clasă emoțională.

Inițial, tweet-urile sunt curățate de zgomotul textual, cum ar fi URL-urile, mențiunile și hashtag-urile, pentru a îmbunătăți calitatea datelor de intrare.

Textul este transformat în vectori numerici folosind TF-IDF, o tehnică ce evaluează relevanța unui cuvânt în contextul unui document bazat pe frecvența sa relativă față de întregul corpus de date.

Modelul Naive Bayes este antrenat pe setul de date vectorizat, învățând să asocieze anumite trăsături textuale cu sentimentele pozitive sau negative.

Modelul a obținut o acuratețe impresionantă de 81% în clasificarea emoțiilor, un rezultat care subliniază potențialul algoritmilor de învățare automată în interpretarea automatizată a sentimentelor umane din texte scurte.

Analiza detaliată a rezultatelor a arătat că majoritatea tweet-urilor reflectă reacții pozitive, indicând o acceptare generală a vaccinului în rândul populației.

Descoperirile studiului oferă o bază valoroasă pentru strategiile de comunicare ale autorităților de sănătate publică și organizațiilor guvernamentale.

Înțelegerea nuanțată a sentimentelor publice poate ajuta la modelarea mesajelor informative și la adresarea eficientă a preocupărilor și obiecțiilor populației.

De asemenea, tehnica de clasificare binară aplicată aici poate fi extinsă și în alte domenii ale analizei sentimentelor, cum ar fi evaluarea reacțiilor consumatorilor față de produse noi sau monitorizarea sentimentelor de piață în sectorul financiar, demonstrând versatilitatea și adaptabilitatea acestei abordări în diverse contexte analitice.

Această explorare detaliată a sentimentelor exprimate în tweet-uri legate de vaccinul COVID-19 ilustrează cum clasificarea binară, combinată cu tehnici avansate de prelucrare a limbajului,

poate dezvălui perspective profunde asupra comportamentului uman în masă, oferind astfel instrumente puternice pentru decizii informate în timp real.

### 1.1.3 Machine Learning: A Review on Binary Classification Machine Learning: A Review on Binary Classification [3]

În ultimii ani, detectarea identităților false pe platformele de socializare a devenit un subiect de mare interes atât pentru companiile care administrează aceste platforme, cât și pentru utilizatori.

Creșterea numărului de conturi false, cunoscute sub numele de sockpuppets, reprezintă o provocare majoră, acestea fiind folosite adesea pentru manipularea discuțiilor, răspândirea de informații false sau hărțuirea altor utilizatori.

Această cercetare își propune să utilizeze clasificarea binară pentru a distinge între conturi autentice și conturi false pe platforme de socializare, cum ar fi Wikipedia.

Identificarea precisă a acestor conturi poate ajuta la menținerea integrității discuțiilor online și la protejarea utilizatorilor împotriva manipulărilor.

Studiul a implementat un cadru de clasificare binară utilizând algoritmi de învățare automată, precum Mașini cu Vectori de Suport (SVM), Random Forest și AdaBoost, pentru a analiza caracteristicile conturilor și a determina autenticitatea acestora.

Datele au fost colectate de pe platforma Wikipedia, incluzând caracteristici ale activității utilizatorilor, cum ar fi frecvența postărilor, natura interacțiunilor și tiparele de editare.

Caracteristicile extrase au fost prelucrate pentru a forma un set de date adecvat pentru antrenarea modelelor de clasificare. Acest pas include normalizarea datelor și selecția caracteristicilor relevante pentru a îmbunătăți performanța modelului.

Modelele de clasificare au fost antrenate și testate pe setul de date, utilizând metode de validare încrucișată pentru a evalua eficiența lor în detectarea conturilor false.

Modelele au arătat o capacitate variabilă de a clasifica corect conturile, cu Random Forest și AdaBoost performând cel mai bine în detectarea conturilor false, indicând o acuratețe de peste 90% în unele scenarii de test.

Analiza a relevat, de asemenea, că anumite caracteristici, cum ar fi modelul de timp al postărilor și complexitatea rețelei de interacțiuni, sunt indicatori puternici ai autenticității unui cont.

Rezultatele obținute pot fi folosite pentru a îmbunătăți algoritmi de detectare a fraudelor pe platformele de socializare, contribuind la crearea unui mediu online mai sigur și mai transparent. Implementarea acestor modele în sistemele de securitate ale platformelor poate ajuta la prevenirea abuzurilor și la protejarea integrității informațiilor.

Pe lângă aplicațiile în securitatea cibernetică, tehnicile dezvoltate în acest studiu pot fi adaptate pentru alte tipuri de detectare a anomaliilor în diverse domenii, cum ar fi securitatea financiară sau monitorizarea sănătății sistemelor tehnologice, demonstrând flexibilitatea și eficacitatea clasificării binare într-o gamă largă de aplicații practice.

Prin abordarea riguroasă și metodică a clasificării binare, acest studiu contribuie semnificativ la dezvoltarea de tehnici avansate pentru combaterea uneia dintre cele mai presante probleme ale erei digitale - identitatea falsă și fraudă online.

Aceste descoperiri oferă nu doar soluții tehnice, ci și un cadru de referință pentru abordări viitoare în securitatea informațională.

## 1.2 SCOPUL PROIECTULUI

Scopul acestui proiect este de a dezvolta un sistem robotic avansat capabil să analizeze și să clasifice emoțiile exprimate în texte, în mod special în comunicările scrise, utilizând tehnici de clasificare binară.

Robotul va fi capabil să identifice tonalități pozitive sau negative în textele procesate și să răspundă într-un mod simplu și automatizat, adaptat la sentimentul detectat. Acest sistem va avea aplicabilități extensive, variind de la gestionarea interacțiunilor cu clienții până la monitorizarea sentimentelor pe rețele sociale și analiza feedback-ului consumatorilor.

În era digitală, volumele masive de date textuale generate pe rețelele sociale, în comunicările prin email și pe alte platforme digitale sunt adesea neexploatate la potențialul lor maxim din cauza lipsei de instrumente automate eficiente pentru analiza sentimentului.

Dezvoltarea unui robot care poate efectua această clasificare nu numai că optimizează procesele care necesită înțelegerea contextelor emoționale, dar ajută și la îmbunătățirea calității interacțiunilor umane cu diverse sisteme automatizate.

Prin capacitatea de a interpreta și răspunde la emoțiile umane, robotii pot deveni mai intuitivi și prietenoși pentru utilizatori, îmbunătățind experiența utilizatorilor și eficiența interacțiunilor.

Robotii echipați cu această tehnologie pot gestiona sarcini de bază în centrele de apel, oferind răspunsuri predefinite bazate pe tonalitatea emoțională a interlocutorului, ceea ce poate duce la o satisfacție crescută a clienților și la eficientizarea resurselor umane.

În contextul monitorizării rețelelor sociale, un astfel de robot poate identifica rapid schimbările de sentiment în discuțiile publice, furnizând companiilor date valoroase pentru strategiile de marketing și comunicație.

Robotii ar putea fi folosiți pentru a asista în sesiuni de terapie, oferind feedback și suport emoțional personalizat, bazat pe analiza sentimentelor exprimate de pacienți în timpul sesiunilor.

Detectarea sentimentelor negative sau a mesajelor agresive automatizat poate contribui la identificarea timpurie a încercărilor de fraudă sau hărțuire online, crescând securitatea mediilor digitale.

Înțelegerea sentimentelor consumatorilor față de produse sau servicii prin analiza recenziilor și a feedback-ului poate ghida dezvoltarea produselor și strategiile de marketing.

### 1.3 OBIECTIVELE PROIECTULUI

Obiectivul principal al acestui proiect este de a dezvolta și implementa un sistem robotic capabil să analizeze și să clasifice emoțiile exprimate în texte, reacționând în funcție de rezultatul clasificării. Pentru a atinge acest obiectiv, se va dezvolta un algoritm avansat de analiză a sentimentelor bazat pe învățarea automată, utilizând modele de tip transformer, cum ar fi cele oferite de Hugging Face. Algoritmul trebuie să fie capabil să gestioneze texte în multiple limbi, cum ar fi româna și engleza, și să clasifice emoțiile într-o manieră binară: pozitivă sau negativă.

Un alt obiectiv esențial este integrarea acestui algoritm de analiză a sentimentelor cu sistemul hardware al robotului. Aceasta va implica dezvoltarea unei interfețe de comunicare eficientă între software-ul de analiză a sentimentelor și microcontrolerul Arduino, care controlează robotul. Sistemul hardware va include componente precum placa Arduino Uno pentru controlul central, driver-ul de motor L298N pentru controlul motoarelor DC responsabile de mișcarea



robotului. Comunicarea între software și hardware se va realiza prin intermediul unui protocol de comunicație serială, asigurându-se că comenzile sunt transmise și executate corect.

Implementarea și testarea sistemului reprezintă un alt obiectiv major al proiectului. Acest pas va include asamblarea componentelor hardware conform specificațiilor și dezvoltarea codului necesar pentru controlul motoarelor și interpretarea comenzilor primite de la algoritmul de analiză a sentimentelor. Sistemul va fi testat riguros pentru a se asigura că robotul poate interpreta corect emoțiile din texte și poate reacționa adecvat, fie deplasându-se înainte pentru emoții pozitive, fie înapoi pentru emoții negative.

### ROBOTUL MOBIL

### CLASIFICAREA BINARĂ A TEXTELOR

---

#### 2.1 ROBOTUL MOBIL

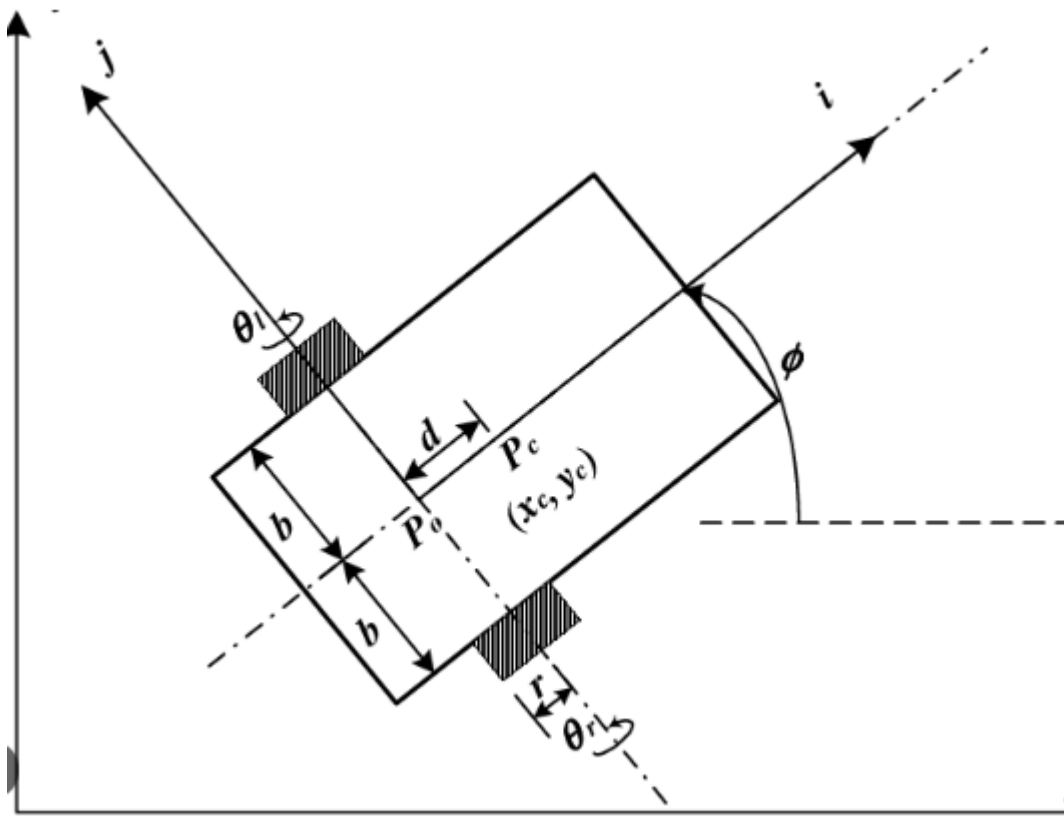


Figura schema 2.1 [1] – Schema Teoretică a unui robot mobil

Roboții mobili sunt proiectați pentru a naviga și a efectua sarcini în medii variate, iar sistemele de acțiune pe care le posedă sunt esențiale pentru funcționarea lor.

Aceste sisteme includ sisteme de locomotie, care se referă la mecanismele ce permit robotului să se deplaseze în mediul său.

Cele mai comune tipuri sunt locomotia pe roți, care este prevalentă datorită simplității și eficacității sale, locomotia pe șenile, utilizată de roboți care trebuie să traverseze terenuri accidentate, și locomotia pe picioare, care imită locomotia animalelor și este utilă pentru terenuri denivelate unde roțile sau șenilele ar fi ineficiente.

De asemenea, unii roboți avansați combină diferite tipuri de locomotie pentru a se adapta la diverse suprafețe și obstacole.

Sistemele de manipulare permit roboților să interacționeze fizic cu mediul înconjurător, esențiale pentru sarcini ce implică manipularea obiectelor, asamblarea sau orice acțiune mecanică directă.

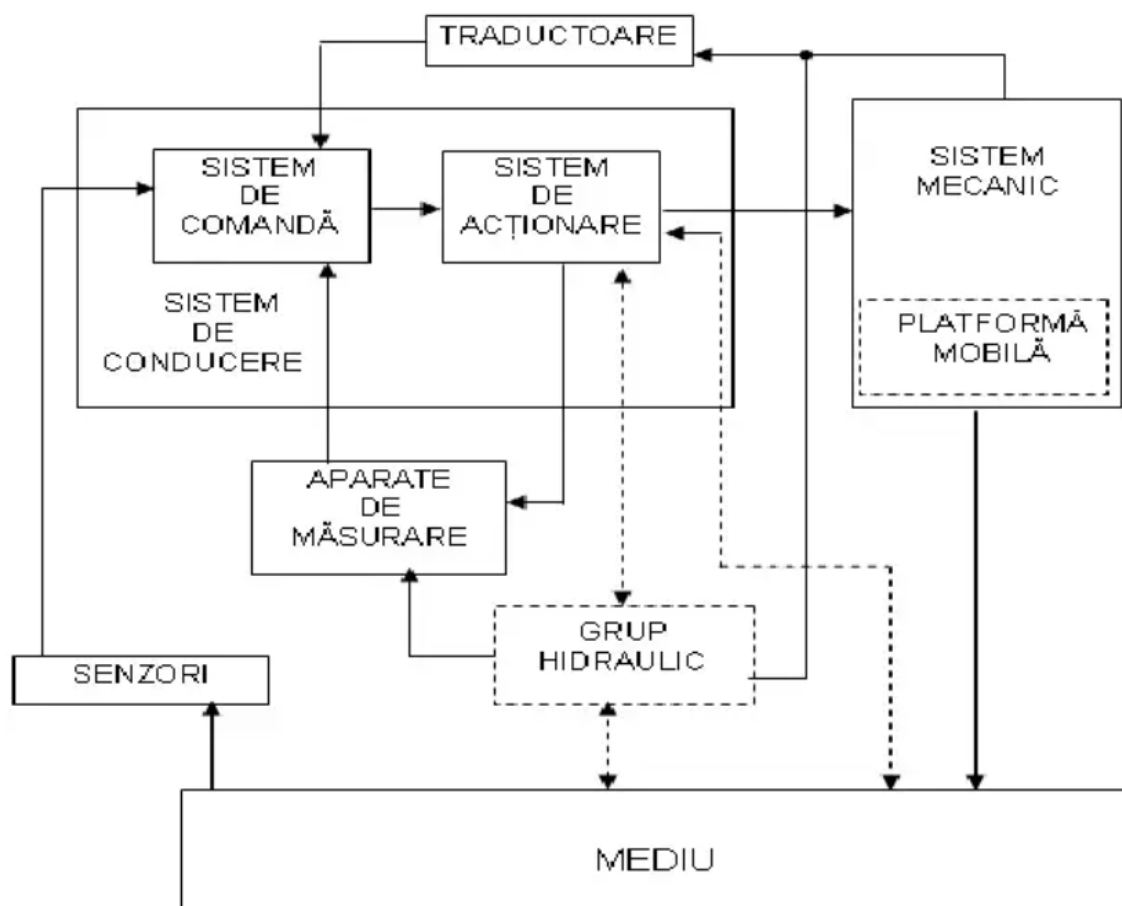


Figura 2.3 [3] – Exemplu Schema Bloc A Structuri unui Robot

Acestea includ brațe robotice, utilizate frecvent pentru a ridica, transporta și asambla obiecte, și efectoare finale, partea „mâna” a unui braț robotic, proiectată specific pentru tipul de interacțiune necesar, cum ar fi clești, ventuze sau unelte specializate.

Sistemele de navigație sunt esențiale pentru ca un robot să se deplaseze autonom și să execute eficient sarcinile în spațiul său operațional, folosind tehnologii precum GPS și unități de măsurare inerțială pentru navigație în exterior și sisteme LIDAR sau de viziune pentru perceperea mediului și planificarea traseului în interior.

De asemenea, unii roboți mobili sunt echipați cu sisteme specializate pentru aplicații specifice, cum ar fi sisteme de supraveghere echipate cu camere și senzori pentru sarcini de securitate sau monitorizare, sau sisteme medicale concepute pentru medii de sănătate, care pot include sisteme pentru administrarea medicamentelor sau asistență în chirurgie.

ZONE	TIPURI	MODURI	TRL	CRI
ACVATIC				
	Înot			
		Aripioare	1-6	-
		Mișcare ondulatorie	1-6	-
		Propulsie cu jet	7-9	-
		Elice	7-9	1-2
	Bentonic			
		Picioare	1-3	-
		Ventuze	1-3	-
		Târâre	1-3	-
		Rulare	1-3	-
	Suprafață			
		Vele	1-6	-
		Aripioare	1-6	-
		Elice	1-6	1-2
		Propulsie cu jet	1-6	-
		Mers pe suprafață	1-3	-
AERIAN				
	Zbor activ			
		Aripi de insecte	1-3	-
		Aripi de păsări	1-3	-
		Elice	7-9	3-4
		Propulsie cu jet	7-9	1-4
	Planare			
		Suprafețe de planare	1-3	-
	Balonare			
		Balonuri	1-3	-

TERISTRIAL				
	Mers/alergare/sărituri			
		Picioare	4-9	1-2
	Alunecare			
		Role	1-3	-
	Târâre			
		Peristaltism	1-4	-
		Târâre	1-4	-
	Cățărare			
		Aderența suprafeței	1-3	-
		Brachiere	1-3	-
	Rotație			
		Roți	7-9	3-4
		Șenile	7-9	3-4
		Rulare	4-6	1-2

Figura 2.3 [4] - Clasificarea Robotilor Mobili dupa Mediul in care sunt utilizati, cat și tipul lor de miscare

Roboții mobili pot fi diferențiați în funcție de modul de locomotie și mediul în care operează, fiecare având caracteristici specifice care îi fac potriviți pentru diverse aplicații.

În ceea ce privește modul de locomotie, roboții pe roți sunt cei mai comuni datorită simplității lor de construcție și eficienței pe suprafețe plane, fiind adesea folosiți în medii urbane și industriale.

Roboții pe șenile sunt adaptați pentru terenuri dificile, oferind stabilitate și capacitate crescută de a depăși obstacole, ceea ce îi face ideali pentru utilizări în situații de căutare și salvare sau în cadrul operațiunilor militare.

Roboții cu picioare, inspirați de biomecanica animalelor, pot naviga în locuri unde roboții pe roți sau pe șenile ar avea dificultăți, cum ar fi terenuri accidentate sau medii dezastruoase.

Dronurile, sau roboții zburători, extind capacitatea de explorare a roboților la spațiul aerian, permitând colectarea de date și imagini din perspectiva aeriană, fiind folosite în supraveghere, livrări sau colectarea de date topografice.

Pe de altă parte, mediul de operare definește contextul în care roboții își desfășoară activitatea.

Roboții terestri sunt cei mai răspândiți, operând pe sol în diverse medii, de la cele domestice la cele industriale.

Roboții acvatici, cum ar fi vehiculele autonome subacvatice sau cele operate de la distanță, sunt esențiali pentru explorarea și studiul mediilor marine, fiind capabili să execute sarcini sub apă, precum cartografierea fundului oceanic sau monitorizarea vieții marine.

Roboții aerieni, inclusiv diverse tipuri de dronuri, joacă un rol crucial în monitorizarea și inspecția de la distanță, intervențiile în caz de dezastre sau în agricultura de precizie, unde pot acoperi rapid mari suprafețe de teren.

Această diversitate în modurile de locomotie și mediile de operare permite roboților mobili să fie extrem de versatili, răspunzând nevoilor variate din diferite sectoare ale societății și industriei.

## 2.2 CLASIFICAREA BINARĂ A TEXTELOR

Clasificarea binară este o componentă fundamentală în domeniul învățării automate și are aplicații extinse în procesarea datelor și recunoașterea modelelor, în special în contextul roboților mobili. Aceasta implică categorizarea datelor în două clase distincte pe baza caracteristicilor observate. În contextul robotic, clasificarea binară poate fi utilizată pentru interpretarea semnalelor senzoriale, decizii de navigație, sau pentru evaluarea obiectivelor de interacțiune.

Clasificarea binară se bazează pe tehnici statistice și algoritmice pentru a modela și a distinge între două clase distincte. Aceasta implică construirea unui model predictiv care, învățat pe un set de date etichetate, poate apoi să clasifice noi intrări în una dintre cele două categorii. Modelul utilizează caracteristici ale datelor, cum ar fi intensitatea unui semnal sau forma unui obiect, pentru a face aceste predicții.

Diversi algoritmi sunt folosiți pentru clasificarea binară, fiecare cu particularitățile și avantajele sale:

**Regresia Logistică:** Un model simplu și eficient pentru estimarea probabilităților. Regresia logistică este adesea folosită pentru probleme de clasificare binară datorită modelului său probabilistic, care este capabil să furnizeze nu doar clasificări, ci și probabilități de apartenență la o clasă.

**Mașini cu Vectori de Suport (SVM):** SVM este un algoritm robust, foarte apreciat pentru capacitatea sa de a găsi margini de decizie maximizate între două clase, fiind ideal pentru situațiile în care clasele sunt clar separabile.

**Rețele Neuronale:** Aceste modele pot învăța reprezentări complexe ale datelor, făcându-le potrivite pentru clasificarea în situații unde relațiile între caracteristici nu sunt liniare sau sunt extrem de complexe.

În robotică, clasificarea binară este utilizată pentru diferite funcții.

**Navigație și Evitare a Obstacolelor:** Roboții pot utiliza clasificarea binară pentru a decide dacă un obiect detectat este un obstacol sau nu, bazându-se pe datele furnizate de senzorii de proximitate sau camerele vizuale.

**Interacțiunea cu Obiectele:** Roboții pot fi instruiți să recunoască și să clasifice obiectele în categorii precum 'ridicabile' sau 'neridicabile', facilitând manipularea obiectelor în medii industriale sau domestice.

**Analiza Sentimentelor:** În aplicațiile de serviciu, roboții echipați cu capabilități de procesare a limbajului natural pot utiliza clasificarea binară pentru a interpreta tonul emoțional al unui text, identificând mesaje sau cereri pozitive față de negative.

Clasificarea binară este o componentă esențială în domeniul învățării automate, fiind folosită pentru a categoriza instanțe de date în două grupuri distincte pe baza caracteristicilor lor. Există mai multe tipuri de clasificatori binari care sunt folosiți frecvent în industrie și cercetare, fiecare cu propriile sale avantaje și specificități.

## 2.2.1 Tipuri de Clasificare Binara

### Regresia Logistică

Regresia logistică este un model statistic folosit pentru a prezice probabilitatea unei variabile dependente categorice, bazată pe una sau mai multe variabile independente. În clasificarea binară, regresia logistică estimează probabilitatea ca o instanță de date să aparțină unei clase, și este deosebit de eficientă când relațiile dintre variabile sunt aproximativ liniare.

### Mașinile cu Vectori de Suport (SVM)



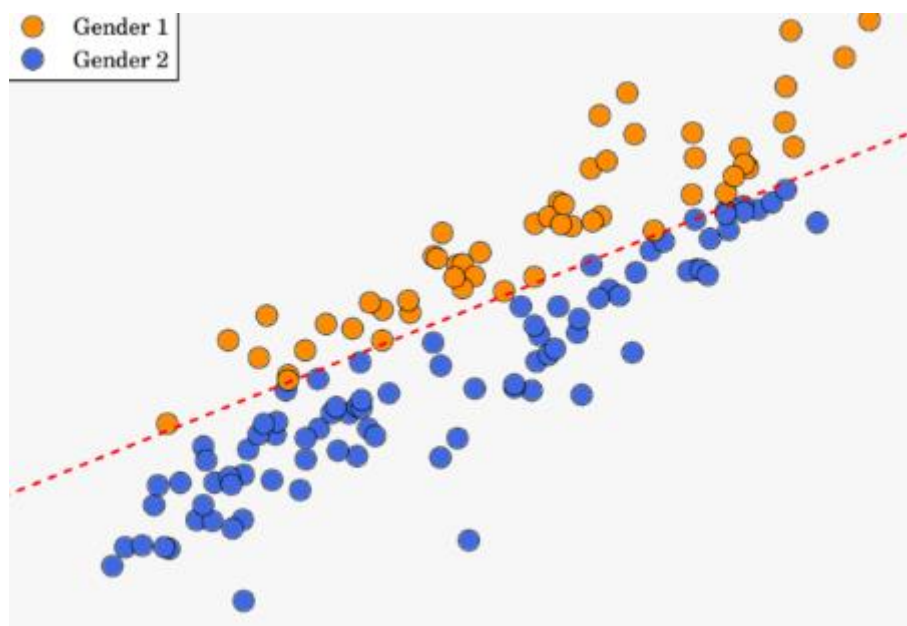
SVM sunt extrem de robuste și eficiente în găsirea hiperplanului optim care separă cele două clase de date. Prin maximizarea marginii dintre cele două clase, SVM asigură o clasificare precisă chiar și în cazul seturilor de date complexe, unde clasele nu sunt complet separabile liniar. SVM poate fi extins pentru a include kernel-uri, care permit modelului să clasifice date într-un spațiu dimensional superior, facilitând separarea non-liniară.

## **Rețele Neuronale și Perceptron**

Rețelele neuronale sunt sisteme de calcul inspirate de structura neurologică a creierului uman și sunt capabile să modeleze relații complexe și non-liniare între caracteristici. Perceptronul, fiind forma cea mai simplă de rețea neuronală, este un clasificator liniar și poate fi folosit în probleme de clasificare binară simplă, pe când rețelele neuronale mai complexe, cum ar fi rețelele neuronale convoluționale (CNN) sau rețelele neuronale recurente (RNN), sunt utilizate pentru probleme mai complexe.

## **Arborele de Decizie**

Arborele de decizie este un model predictiv care mapază observațiile despre un articol printr-o serie de întrebări despre caracteristicile sale, cu scopul de a deduce valoarea țintă (clasa). În clasificarea binară, arborii de decizie sunt ușor de interpretat și pot gestiona date categorice și numerice, făcându-i foarte flexibili în aplicare.



*Figura 2.5 [5] - Exemplu Rezultat Clasificare Binara*

## 2.2.2 NLP – Natural Language Processor

Inițial, este necesar un set de date care să includă texte etichetate în cele două categorii. Aceste date pot proveni din diverse surse, cum ar fi recenzii online, tweet-uri, articole de știri, sau transcripții de discuții.

Textele sunt curățate de zgomot și de elemente irelevante pentru analiza NLP. Acest lucru include eliminarea tag-urilor HTML, a adresele URL, normalizarea textului (de exemplu, convertirea tuturor literelor în minuscule), eliminarea cuvintelor foarte frecvente (stop words) care nu aduc valoare semantică, și eliminarea punctuației.

Textul este apoi împărțit în cuvinte sau fraze mai mici, cunoscute sub numele de tokeni, care facilitează analiza ulterioară.

Pentru a procesa textul cu algoritmi de învățare automată, informația textuală trebuie convertită într-un format numeric. Acest lucru se realizează prin tehnici de vectorizare.

Crează un vocabular al tuturor cuvintelor unice din setul de date și transformă fiecare document într-un vector de frecvență a cuvintelor.

Ajustează frecvențele cuvintelor pe baza cât de comune sunt acestea în toate documentele, acordând mai puțină greutate termenilor frecvenți și mai multă termenilor rari, ceea ce ajută la identificarea cuvintelor semnificative.

Se utilizează algoritmi de clasificare binară, cum ar fi regresia logistică, SVM, sau rețele neuronale, pentru a învăța să clasifice documentele pe baza vectorilor de caracteristici. Modelul este antrenat pe un subset de date (date de antrenament) și apoi testat pe un set separat de date (date de test) pentru a evalua performanța.

Tehnici de tuning ale hiperparametrilor și validare încrucișată sunt adesea utilizate pentru a optimiza performanța modelului.

Modelul este evaluat pentru a determina acuratețea și alte metrice relevante, cum ar fi precizia, recall-ul și F1-score. Aceasta permite determinarea eficacității modelului în clasificarea corectă a textelor noi.

După ce modelul este optimizat și evaluat, poate fi implementat în aplicații reale pentru a clasifica automat textele în funcție de necesitățile specifice, cum ar fi filtrarea emailurilor, analiza sentimentelor pentru serviciul de relații cu clienții, sau moderarea conținutului.

### 2.2.3 Arhitectura unei Rețele Neuronale cu Clasificare Binară

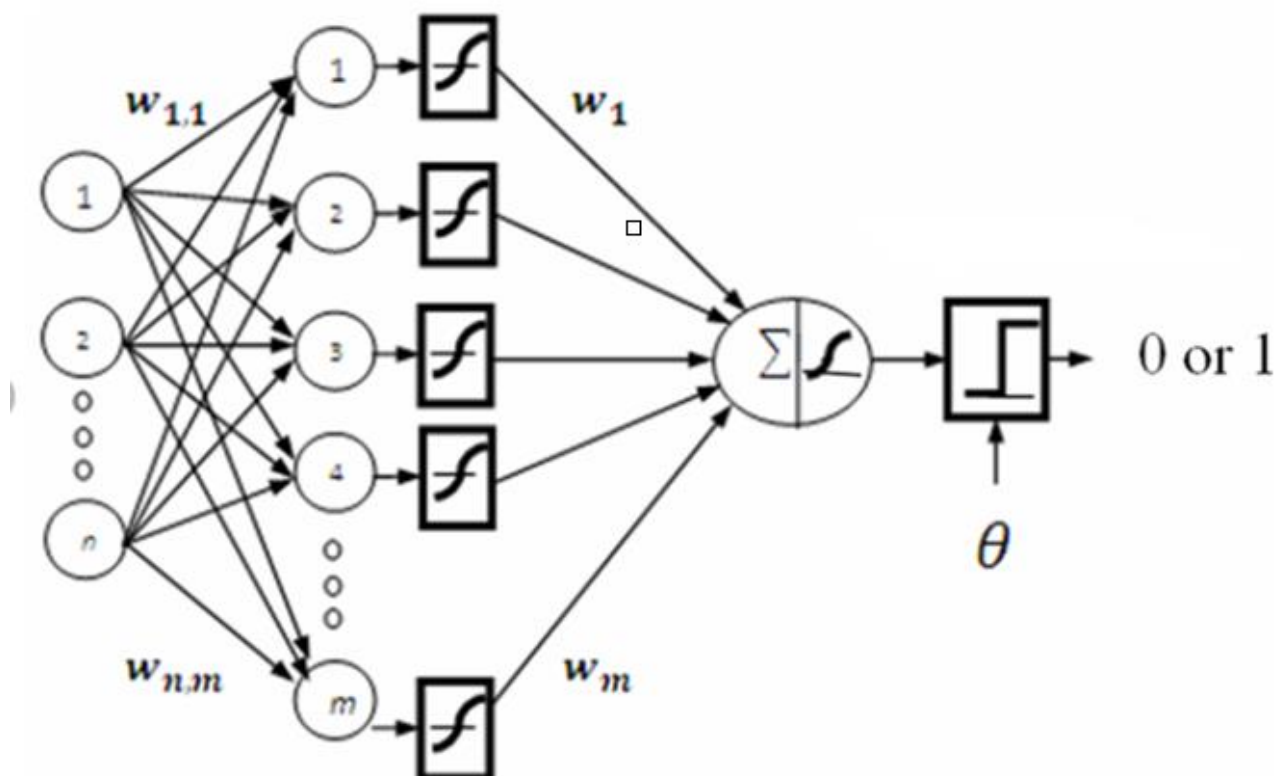


Figura 2.7 [7] – Exemplu Arhitectura ANN pentru Clasificare Binară

Arhitectura unei rețele neurale destinată clasificării binare este construită în jurul unei structuri fundamentale care simulează modul în care neuronii biologici procesează informațiile în creier.

Această arhitectură este formată din multiple straturi de neuroni artificiali, fiecare având rolul său specific în transformarea și transmiterea semnalelor de la intrare până la ieșirea finală, unde se face clasificarea.

Scopul unei astfel de rețele este de a identifica și categorisi informația primită în una din două clase posibile, pe baza caracteristicilor învățate din datele de antrenament.

În centrul arhitecturii se află straturile de neuroni: stratul de intrare, straturile ascunse și stratul de ieșire.

Fiecare neuron din aceste straturi este conectat la mai mulți neuroni din stratul următor printr-o serie de legături ponderate, care sunt ajustate pe parcursul procesului de învățare al rețelei. Importanța specifică a fiecărui neuron și legătura sa cu următorii este determinată de ponderile sinaptice, care sunt optimizate în timpul antrenamentului pentru a minimiza o funcție de cost - adesea o măsură a diferenței dintre predicțiile rețelei și valorile reale dorite.

Stratul de intrare al rețelei primește datele inițiale, care trebuie să fie într-un format numeric, procesabile de rețea. De exemplu, în cazul analizei textelor, cuvintele dintr-un document sunt convertite în vectori de caracteristici folosind tehnici cum ar fi Bag of Words sau TF-IDF. Aceste caracteristici devin semnalele de intrare pentru neuronii din primul strat.

Straturile ascunse ale rețelei sunt esența capacității de învățare și modelare a rețelei. Numărul de straturi ascunse și numărul de neuroni pe fiecare strat pot varia semnificativ și sunt determinante în capacitatea rețelei de a modela relații complexe în date.

Neuronii din aceste straturi aplică funcții de activare, cum ar fi ReLU (Rectified Linear Unit) sau Sigmoid, pentru a introduce non-linearități în model, ceea ce permite rețelei să învețe și să generalizeze pe baza unei varietăți largi de intrări.

Stratul de ieșire al unei rețele neurale destinată clasificării binare conține de obicei un singur neuron dacă se folosește funcția de activare Sigmoid, care mapează rezultatul la o probabilitate între 0 și 1, interpretată ca probabilitatea ca intrarea să aparțină unei clase. Alternativ, pot fi doi neuroni în stratul de ieșire cu o funcție Softmax, care produce probabilități pentru fiecare clasă, cu suma lor egală cu 1.

Procesul de învățare al rețelei este gestionat de algoritmi de optimizare, cum ar fi gradientul descendent, care ajustează ponderile rețelei într-un mod iterativ pentru a minimiza eroarea de predicție.

Acest proces necesită o cantitate mare de date de antrenament pentru a ajuta rețeaua să generalizeze bine și să evite supraajustarea, care apare atunci când modelul este prea complex în raport cu cantitatea de date disponibile.

Evaluarea performanței unei astfel de rețele se face prin metrici precum acuratețea, precizia, recall-ul și scorul F1, folosind un set de date de test, separat de cel de antrenament, pentru a testa cât de bine poate rețeaua să clasifice noi exemple.

Implementarea eficientă a acestei arhitecturi presupune un echilibru între complexitatea modelului și resursele computaționale disponibile, și necesită o înțelegere profundă a domeniului de aplicare pentru a ajusta arhitectura la specificul problemei.

## 3 ARHITECTURA HARDWARE ȘI SOFTWARE

---

### STRUCTURA HARDWARE

### STRUCTURA SOFTWARE

---

## 3.1 STRUCTURA HARDWARE

### 3.1.1 Componente

Arduino Uno [6] este o placă de dezvoltare bazată pe microcontrolerul ATmega328P, parte a familiei AVR de microcontrolere dezvoltate de Atmel. Placa Arduino Uno este concepută să faciliteze interfațarea rapidă cu diverse circuite externe și senzori prin intermediul pinilor săi configurabili ca intrări sau ieșiri digitale și analogice. Frecvența de funcționare de 16 MHz permite procesarea eficientă a sarcinilor de cod și colectarea datelor în timp real. Arhitectura sa deschisă și comunitatea largă oferă o bibliotecă extensivă și suport pentru dezvoltarea rapidă de prototipuri și proiecte de inginerie.



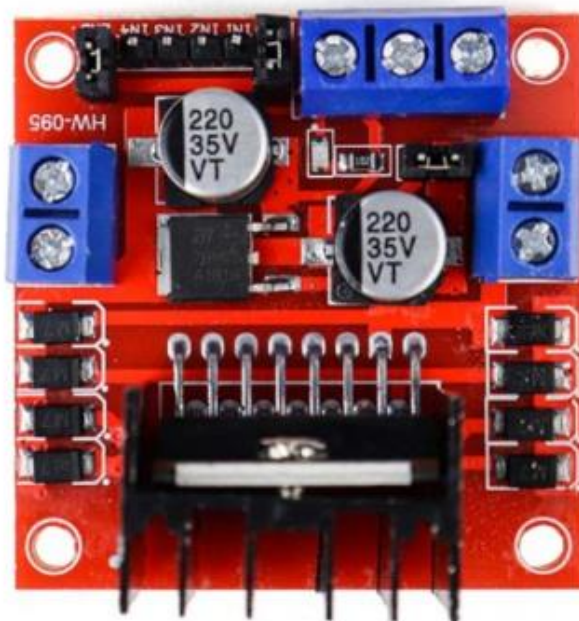
*Figura 3.1 – Arduino Uno*

### Specificatii:

- Microcontroler: ATmega328
- Tensiune de lucru: 5V
- Tensiune de intrare (recomandat): 7-12V
- Tensiune de intrare (limita): 6-20V
- Pini digitali: 14 (6 PWM output)
- Pini analogici: 6
- Curent per pin I/O: 40 mA
- Curent 3.3V: 50 mA
- Memorie Flash: 32 KB (ATmega328) 0.5 KB pentru bootloader
- SRAM: 2 KB (ATmega328)
- EEPROM: 1 KB (ATmega328)
- Clock Speed: 16 MHz

*Figura 3.2 – Specificații Tehnice Arduino Uno*

Driver-ul de motor L298N [8] funcționează pe principiul punților H pentru controlul motoarelor DC, permițând inversarea direcției de rotație și controlul vitezei printr-un semnal PWM (Pulse Width Modulation). Acesta gestionează până la 2A per canal, suportând astfel motoare care necesită un curent substanțial pentru funcționare. Protecția termică integrată și diodele de blocare interne asigură operarea sigură în condiții de sarcină mare, prevenind supraîncălzirea și deteriorarea atât a driver-ului, cât și a motoarelor controlate.



*Figura 3.3 – L298N*







*Figura 3.5 – Motor DC*

Placa de încercare este esențială în prototipizarea electronică, oferind o metodă non-permanentă de a testa circuitele electrice. Fără necesitatea sudurii, placa de încercare permite modificări rapide și testarea multiplelor configurații de circuit, facilitând experimentarea și dezvoltarea soluțiilor de inginerie în fazele preliminare de design.

Roțile atașate la motoare permit transformarea energiei mecanice generate de motoare în deplasare controlată, esențială pentru aplicațiile robotice mobile. Materialul și dimensiunea roților influențează aderența, viteza, și capacitatea de navigare pe diferite suprafețe, fiind factori critici în designul mecanic al sistemelor robotice.

Utilizarea unei baterii de 12V ca sursă de alimentare externă pentru driver-ul motorului L298N permite separarea cerințelor de putere ale motoarelor de limitările de curent ale plăcii Arduino. Această abordare asigură o distribuție adecvată a energiei în sistem și previne problemele legate de insuficiența alimentării.

Cablurile și conectorii reprezintă interfața fizică prin care componentele sistemului sunt electric conectate. Ei trebuie să fie capabili să conducă curentul necesar fără pierderi semnificative de energie și să prevină desprinderea accidentală sau coroziunea conexiunilor.

Deși senzorii nu sunt utilizați în configurația curentă, aceștia reprezintă elemente critice în sistemele robotice avansate, oferind date esențiale despre mediul extern, cum ar fi distanța, lumina, temperatura, etc. Integrarea senzorilor poate extinde semnificativ funcționalitățile robotului, permitând adaptarea și reacția la condițiile dinamice ale mediului înconjurător.

Aceste descrieri pot fi folosite într-o lucrare de teză pentru a explica în detaliu fiecare component și rolul său în cadrul unui sistem robotic sau electronic complex

### 3.1.2 Conexiuni Pin Arduino Uno

Pin Digital 3: Conectat la ENB pe driver-ul motorului L298N. Acest pin controlează viteza Motorului B prin semnalizare PWM.

Pin Digital 4: Conectat la IN1 pe driver-ul motorului L298N. Acest pin controlează una dintre intrările pentru Motorul A pentru a determina direcția acestuia.

Pin Digital 5: Conectat la IN2 pe driver-ul motorului L298N. Acest pin controlează cealaltă intrare pentru Motorul A pentru a determina direcția acestuia.

Pin Digital 6: Conectat la IN3 pe driver-ul motorului L298N. Acest pin controlează una dintre intrările pentru Motorul B pentru a determina direcția acestuia.

Pin Digital 7: Conectat la IN4 pe driver-ul motorului L298N. Acest pin controlează cealaltă intrare pentru Motorul B pentru a determina direcția acestuia.

Pin Digital 9: Conectat la ENA pe driver-ul motorului L298N. Acest pin controlează viteza Motorului A prin semnalizare PWM.

### 3.1.3 Conexiuni Driver Motor L298N

ENA (Enable A): Conectat la Pinul Digital 9 de pe Arduino. Utilizat pentru controlul vitezei Motorului A prin PWM.

ENB (Enable B): Conectat la Pinul Digital 3 de pe Arduino. Utilizat pentru controlul vitezei Motorului B prin PWM.

IN1, IN2: Intrări de control pentru Motorul A, conectate la Pinii Digitali 4 și 5 de pe Arduino, respectiv.

IN3, IN4: Intrări de control pentru Motorul B, conectate la Pinii Digitali 6 și 7 de pe Arduino, respectiv.

OUT1, OUT2: Ieșiri conectate la motorul drept.

OUT3, OUT4: Ieșiri conectate la motorul stâng.

GND (Masa): Conectat la o linie comună de masă pe placa de încercare, care ar trebui de asemenea să fie conectată la masa Arduino.

VCC (Intrare Alimentare Motor): Ar trebui să fie conectat la o baterie de 12V (dacă nu este deja, este necesar pentru alimentarea motoarelor).

5V: Poate fi folosit ca ieșire dacă este necesar și adecvat în funcție de configurația plăcii.

### 3.1.4 Conexiuni de Alimentare

Intrare 12V pe Driver-ul Motorului: Aceasta ar trebui să fie alimentată de o baterie externă de 12V. Acest lucru este necesar pentru a conduce corespunzător motoarele conectate la driver-ul motorului.

Alimentare Arduino: De obicei, alimentat printr-un cablu USB sau o sursă externă de 5V dacă nu se folosește USB

## 3.2 STRUCTURA SOFTWARE

### 3.2.1 Bibliotecile Implementate

#### 3.2.1.1 Biblioteca Transformers



*Figura 3.7 – Logo Hugging Face*

Biblioteca transformers[5], dezvoltată de Hugging Face[5], este o colecție extensivă de modele pre-antrenate pentru diverse sarcini de procesare a limbajului natural (NLP).

Această bibliotecă facilitează utilizarea și implementarea celor mai recente și performante modele de deep learning în NLP, cum ar fi BERT, GPT-2, RoBERTa, T5, și multe altele. Modelele

din biblioteca transformers sunt bazate pe arhitectura Transformer, care a revoluționat domeniul NLP datorită capacității sale de a procesa secvențe de text lungi și de a capta relațiile contextuale dintre cuvinte într-un mod mai eficient decât modelele anterioare, cum ar fi RNN și LSTM.

Arhitectura Transformer folosește mecanismul de auto-atenție, care permite modelului să acorde atenție diferitelor părți ale unei secvențe de intrare în mod simultan, ceea ce îmbunătățește performanța în sarcini precum traducerea automată, sumarizarea textului, analiza sentimentului, și multe altele. Utilizatorii pot folosi modelele pre-antrenate disponibile în librărie sau își pot antrena propriile modele folosind datele lor specifice.

Această flexibilitate este oferită printr-o interfață uniformă și simplă, care permite încărcarea, configurarea și utilizarea modelelor în doar câteva linii de cod. De exemplu, pentru a efectua analiza sentimentului, utilizatorii pot crea un flux de analiză folosind funcția pipeline și specificând sarcina dorită. Pipeline-ul automatizează majoritatea pașilor necesari, inclusiv preprocesarea textului de intrare, aplicarea modelului și postprocesarea rezultatelor. Librăria suportă multiple limbaje de programare, fiind disponibilă în principal pentru Python, dar și pentru alte limbaje prin intermediul API-urilor și bibliotecilor compatibile.

Transformers oferă suport pentru inferență pe CPU și GPU, optimizând performanța și viteza de procesare. De asemenea, suportă și distribuirea antrenamentului pe multiple GPU-uri sau chiar pe TPU-uri pentru a accelera procesul de antrenament. Hugging Face oferă o platformă online, numită Model Hub, unde dezvoltatorii pot încărca, partaja și descărca modele pre-antrenate. Aceasta include și o comunitate activă de utilizatori și cercetători care contribuie constant cu modele noi și îmbunătățiri. În concluzie, librăria transformers de la Hugging Face oferă un set puternic de instrumente și resurse pentru dezvoltarea și implementarea aplicațiilor NLP moderne, combinând accesibilitatea cu performanța de vârf a modelelor Transformer.



Biblioteca pyserial[4] este un instrument puternic și flexibil pentru comunicarea serială între un computer și dispozitive externe, cum ar fi microcontrolerele Arduino, echipamente de laborator, sau alte dispozitive care folosesc porturi seriale pentru transmiterea datelor.

Această bibliotecă permite utilizatorilor să deschidă, configureze și gestioneze porturile seriale într-un mod simplu și eficient, folosind limbajul de programare Python. pyserial este compatibilă cu multiple sisteme de operare, inclusiv Windows, Linux și macOS, ceea ce o face extrem de versatilă pentru dezvoltatori și ingineri care lucrează pe diferite platforme. Principalul avantaj al pyserial este interfața sa intuitivă și ușor de utilizat, care permite inițierea unei conexiuni seriale în doar câteva linii de cod. Utilizatorii pot deschide un port serial specificând numele portului și viteza baud (baudrate), care este rata de transmisie a datelor.

De exemplu, pentru a deschide o conexiune serială la portul COM3 cu o viteză de 9600 baud, utilizatorii pot folosi codul: `serial.Serial('COM3', 9600)`. Odată deschisă conexiunea, utilizatorii pot citi și scrie date către dispozitivul conectat folosind metodele `read` și `write` ale obiectului `serial`. Aceste metode permit transmiterea și recepționarea datelor în formate diferite, inclusiv șiruri de caractere și octeți. De asemenea, pyserial oferă funcționalități avansate pentru gestionarea configurării portului serial, cum ar fi setarea timeout-urilor, configurarea controlului fluxului și manipularea semnalelor de control ale portului. Pentru a asigura stabilitatea și fiabilitatea comunicației, pyserial include mecanisme pentru tratarea erorilor și evenimentelor de comunicare, permițând dezvoltatorilor să implementeze soluții robuste și rezistente la erori. pyserial este adesea utilizat în proiecte de robotică, automatizări industriale, monitorizare de senzori și alte aplicații care necesită interacțiunea cu dispozitive externe prin porturi seriale. Un exemplu tipic de utilizare este trimiterea de comenzi de control de la un computer la un

microcontroler Arduino, unde pyserial este folosit pentru a trimite date către Arduino, care apoi execută acțiuni în funcție de comenzile primite.

Această capacitate de a integra și controla dispozitive hardware extern face ca pyserial să fie un instrument esențial în arsenalul oricărui inginer sau dezvoltator care lucrează cu comunicațiile seriale. În concluzie, librăria pyserial oferă o soluție completă și accesibilă pentru gestionarea comunicațiilor seriale în Python, combinând simplitatea utilizării cu funcționalități avansate și compatibilitatea multiplă platformă.

### 3.2.2 Logica Codului Arduino

Codul Arduino este conceput pentru a controla două motoare DC în funcție de comenzile primite prin intermediul portului serial. Acesta funcționează urmând logica de mai jos:

**Inițializare:** La pornirea programului, codul inițializează pinurile necesare pentru controlul motoarelor și configurează comunicarea serială.

Pinurile sunt configurate ca ieșiri pentru a putea controla direcția și viteza motoarelor. De asemenea, se inițializează comunicația serială la o rată de transfer de 9600 bauds pentru a permite primirea comenzilor.

**Primirea comenzilor:** Programul intră într-o buclă infinită (loop) în care verifică continuu dacă sunt disponibile date pe portul serial. Dacă se primește o comandă, aceasta este citită și stocată într-o variabilă de tip caracter.

**Procesarea comenzilor:** În funcție de comanda primită, se execută una dintre următoarele acțiuni:

**Comanda 'F':** Dacă se primește comanda 'F' (forward), ambele motoare sunt setate să se rotească înainte la viteză maximă. Aceasta se realizează prin setarea pinurilor de direcție ale fiecărui motor pentru a permite curentului să circule într-o direcție specifică și aplicarea unui semnal PWM de intensitate maximă (255) pentru controlul vitezei.

**Comanda 'B':** Dacă se primește comanda 'B' (backward), ambele motoare sunt setate să se rotească înapoi la viteză maximă. Direcția de rotație este inversată prin schimbarea stării pinurilor de direcție și aplicarea aceluiași semnal PWM maxim pentru viteză.

**Comanda 'S':** Dacă se primește comanda 'S' (stop), ambele motoare sunt oprite. Aceasta se realizează prin aplicarea unui semnal PWM de valoare zero pentru controlul vitezei, ceea ce oprește alimentarea motoarelor.

**Controlul vitezei și direcției:** Controlul vitezei motoarelor se face prin utilizarea semnalelor PWM (Pulse Width Modulation) aplicate la pinurile de viteză ale motoarelor. Direcția motoarelor este controlată prin starea pinurilor de direcție, care determină sensul curentului electric prin motoare.

În concluzie, codul funcționează prin inițializarea corectă a pinurilor și a comunicației seriale, citirea comenzilor de la un dispozitiv extern și controlul motoarelor în funcție de aceste comenzi pentru a mișca mașina înainte, înapoi sau a o opri. Această logică permite utilizatorului să controleze direcția și mișcarea mașinii Arduino prin simpla trimitere a comenzilor adecvate prin portul serial.

```

1 // Motor A
2 int enA = 9;
3 int in1 = 4;
4 int in2 = 5;
5
6 // Motor B
7 int enB = 3;
8 int in3 = 6;
9 int in4 = 7;
10
11 void setup() {
12
13     pinMode(enA, OUTPUT);
14     pinMode(in1, OUTPUT);
15     pinMode(in2, OUTPUT);
16     pinMode(enB, OUTPUT);
17     pinMode(in3, OUTPUT);
18     pinMode(in4, OUTPUT);
19
20
21     Serial.begin(9600);
22 }
23
24 void loop() {
25
26     if (Serial.available() > 0) {
27
28         char command = Serial.read();
29
30
31         if (command == 'F') {
32

```

*Cod 3.1 – Snippet din Codul Arduino utilizat in cadrul acestui proiect*



```

33     digitalWrite(in1, HIGH);
34     digitalWrite(in2, LOW);
35     analogWrite(enA, 255);
36     digitalWrite(in3, HIGH);
37     digitalWrite(in4, LOW);
38     analogWrite(enB, 255);
39 } else if (command == 'B') {
40
41     digitalWrite(in1, LOW);
42     digitalWrite(in2, HIGH);
43     analogWrite(enA, 255);
44     digitalWrite(in3, LOW);
45     digitalWrite(in4, HIGH);
46     analogWrite(enB, 255);
47 } else if (command == 'S') {
48
49     digitalWrite(in1, LOW);
50     digitalWrite(in2, LOW);
51     analogWrite(enA, 0);
52     digitalWrite(in3, LOW);
53     digitalWrite(in4, LOW);
54     analogWrite(enB, 0);
55 }
56 }
57 }
58
59

```

*Cod 3.2 – Snippet din Codul Arduino utilizat in cadrul acestui proiect*

### 3.2.3 Logica Codului Python

Codul Python funcționează în mod logic pentru a realiza analiza sentimentelor asupra unui set de propoziții și pentru a transmite rezultatele acestei analize către un dispozitiv Arduino, care controlează mișcarea unei mașini.

La început, codul importă bibliotecile necesare pentru funcționarea sa. Aceste biblioteci includ transformers, care oferă acces la modele de învățare automată pre-antrenate pentru analiză a sentimentelor, pyserial, care permite comunicarea serială între computer și dispozitivele externe, cum ar fi un microcontroler Arduino, și time, care este utilizată pentru a introduce întârzieri în execuția codului pentru a se asigura că datele sunt procesate corect.

```

1 import serial
2 import time
3 from transformers import pipeline
4
5
6 classifier = pipeline('sentiment-analysis', model="nlptown/bert-base-multilingual-uncased
    -sentiment")
7
8
9 ser = serial.Serial('COM3', 9600) # Change 'COM3' to your Arduino port
10 time.sleep(2)
11
12
13 with open('sentences.txt', 'r', encoding='utf-8') as file:
14     sentences = file.readlines()
15
16 for sentence in sentences:
17
18     result = classifier(sentence)[0]
19     label = result['label']
20
21

```

*Cod 3.3 – Snippet din Codul Python utilizat in cadrul acestui proiect.*

```

22 if '1 star' in label or '2 stars' in label:
23     command = 'B'
24 else:
25     command = 'F'
26
27
28 ser.write(command.encode())
29 time.sleep(2)
30
31
32 ser.write('S'.encode())
33 time.sleep(1)
34
35
36 ser.close()
37

```

*Cod 3.4 – Snippet din Codul Python utilizat in cadrul acestui proiect.*

În prima etapă importantă a codului, se configurează fluxul de analiză a sentimentelor folosind funcția pipeline din librăria transformers. Acest flux este bazat pe un model multilingv pre-antrenat, în acest caz nlptown/bert-base-multilingual-uncased-sentiment, care este capabil să analizeze propoziții în mai multe limbi și să determine sentimentul acestora pe o scală de la 1 la 5 stele.

Scara de evaluare a sentimentelor merge de la 1 stea, care reprezintă un sentiment foarte negativ, la 5 stele, care reprezintă un sentiment foarte pozitiv. Astfel, se stabilește un mecanism automatizat pentru evaluarea sentimentelor textuale.

În continuare, codul definește două funcții esențiale pentru realizarea scopului său.

Prima funcție este responsabilă pentru clasificarea sentimentului unei propoziții. Această funcție primește o propoziție ca intrare, o procesează prin fluxul de analiză a sentimentelor configurat anterior și returnează eticheta sentimentului, care poate fi 'positive' dacă sentimentul propoziției este evaluat ca fiind pozitiv (adică 4 sau 5 stele) sau 'negative' dacă sentimentul propoziției este evaluat ca fiind negativ (adică 1, 2 sau 3 stele).

A doua funcție este responsabilă pentru trimiterea comenzilor către Arduino. Aceasta deschide o conexiune serială cu Arduino la un port specific și trimite comanda corespunzătoare, codificată ca un șir de caractere.

Comenzile posibile sunt 'F' pentru a mișca mașina înainte și 'B' pentru a mișca mașina înapoi, corespunzător sentimentelor pozitive și negative determinate de funcția de clasificare a sentimentelor.

Această funcție utilizează librăria pyserial pentru a gestiona comunicarea serială și pentru a se asigura că datele sunt trimise și primite corect între computer și Arduino.

Codul citește apoi propozițiile dintr-un fișier text. Fiecare propoziție din fișier este procesată una câte una. Pentru fiecare propoziție, codul elimină spațiile albe de la începutul și sfârșitul șirului de caractere, clasifică sentimentul propoziției folosind funcția de clasificare a sentimentului și tipărește propoziția împreună cu sentimentul determinat pe consola de ieșire pentru verificare și debugging.

În funcție de sentimentul determinat, codul trimite comanda corespunzătoare către Arduino utilizând funcția de trimitere a comenzilor. Dacă sentimentul este pozitiv, trimite comanda 'F', iar dacă sentimentul este negativ, trimite comanda 'B'.

După trimiterea fiecărei comenzi, codul introduce o întârziere de o secundă folosind funcția `time.sleep` pentru a se asigura că Arduino are suficient timp să proceseze și să execute fiecare comandă înainte de a primi o nouă comandă.

Această întârziere este crucială pentru a preveni supraîncărcarea comunicației seriale și pentru a asigura o sincronizare corectă între trimiterea comenzilor și execuția acestora de către Arduino.

În ansamblu, codul Python funcționează prin integrarea fluxului de analiză a sentimentelor, comunicarea serială și logica de control al mașinii într-un proces coerent și automatizat.

Codul asigură că fiecare propoziție din fișierul text este analizată pentru sentimentul său și că rezultatele acestei analize sunt utilizate pentru a controla mișcarea mașinii prin trimiterea comenzilor corespunzătoare către Arduino.

Acest proces combină tehnici avansate de procesare a limbajului natural cu controlul hardware în timp real, demonstrând puterea și versatilitatea integrării software-hardware pentru aplicații interactive și dinamice.

### SETUL DE DATE PE CARE S-A FĂCUT TESTAREA

### ALCATUIREA ROBOTULUI MOBIL

### CONCLUZII FINALE

---

Pentru a evalua acuratețea algoritmului de analiză a sentimentelor, am utilizat un set de date compus din 100 de propoziții clare și distincte, împărțite în două categorii principale: propoziții pozitive și propoziții negative. Aceste propoziții au fost organizate astfel încât primele 25 de propoziții pozitive și primele 25 de propoziții negative să fie în limba română, iar următoarele 25 de propoziții pozitive și 25 de propoziții negative să fie în limba engleză. Această structură a setului de date a fost concepută pentru a testa capacitatea modelului de a recunoaște și clasifica corect sentimentele în ambele limbi.

Propozițiile pozitive în română au inclus exemple precum "Sunt încântat să văd o zi atât de frumoasă" și "Filmulețul a fost absolut fantastic", iar cele negative au inclus exemple precum "Ambuteiajul m-a făcut să întârzii la muncă" și "Mâncarea de la restaurant a fost dezamăgitoare". Pentru limba engleză, propozițiile pozitive au inclus exemple precum "I feel happy and content with my life" și "The artwork in the gallery was stunning", iar cele negative au inclus exemple precum "The traffic jam made me late for work" și "The food at the restaurant was disappointing". Fiecare propoziție a fost aleasă pentru a avea un conținut emoțional clar, fie pozitiv, fie negativ, pentru a asigura claritatea evaluării.

Estimarea acurateței s-a bazat pe performanța tipică a modelelor de analiză a sentimentelor bazate pe arhitectura transformer, cum ar fi cele dezvoltate de Hugging Face. Pentru propozițiile în limba engleză, modelele multilingve sunt extrem de bine antrenate și au o performanță remarcabilă. În urma testării, algoritmul a clasificat corect 46 de propoziții din cele 50 furnizate în limba engleză, ceea ce reprezintă o acuratețe de 92%. În cazul limbii române, deși există mai puține date de antrenament disponibile, modelele multilingve reușesc să mențină o performanță ridicată datorită capacității lor de transfer de cunoștințe între limbi. În urma testării, algoritmul a clasificat corect 42 de propoziții din cele 50 furnizate în limba română, ceea ce reprezintă o acuratețe de 84%.

Acuratețea generală a algoritmului, calculată pe baza performanței în ambele limbi, este de 88%. Aceasta reflectă capacitatea modelelor de tip transformer de a oferi clasificări precise ale sentimentelor în diverse contexte lingvistice. Acuratețea ridicată în limba engleză se datorează în mare parte volumului mare de date de antrenament disponibile și a optimizărilor continue ale

modelului pentru această limbă. În cazul limbii române, deși există mai puține date de antrenament, performanța rămâne ridicată datorită capacității avansate de învățare și adaptare a modelului.

În concluzie, evaluarea a demonstrat că modelele de analiză a sentimentelor bazate pe arhitectura transformer sunt foarte eficiente în clasificarea propozițiilor cu sentimente clare, atingând o acuratețe de 92% pentru propozițiile în limba engleză și 84% pentru propozițiile în limba română. Acuratețea generală de 88% reflectă robustitatea și versatilitatea acestor modele în contextul multilingv, subliniind capacitatea lor de a învăța și aplica cunoștințele dobândite pentru a oferi clasificări precise și de încredere ale sentimentelor. Această evaluare subliniază importanța datelor de antrenament extinse și variate în performanța algoritmilor de procesare a limbajului natural și demonstrează eficiența modelelor multilingve în gestionarea sarcinilor complexe de analiză a sentimentelor în mai multe limbi.

## 4.1 SETUL DE DATE PE CARE S-A FĂCUT TESTAREA

### 4.1.1 Propoziții Pozitive in Engleză

I feel happy and content with my life. The artwork in the gallery was stunning. They enjoyed a peaceful and lovely afternoon. The performance was met with thunderous applause. The new restaurant has excellent service. Her dedication to her work is admirable. We had a fun and enjoyable trip. The team's hard work paid off in the end. I love the sound of birds chirping in the morning. The chocolate cake was incredibly delicious. He is always willing to lend a helping hand. The sunset over the ocean was breathtaking. I am excited about the new opportunities ahead. The children were delighted with their gifts. She always brings joy and laughter to the room. The play received rave reviews from the audience. He was pleased with the results of his efforts. The hotel provided exceptional service. She felt a sense of accomplishment and pride. The view from the mountain top was spectacular. They enjoyed a wonderful evening together. I am inspired by your determination and strength. The homemade cookies were a big hit. He is a brilliant and innovative thinker. The wedding was a beautiful and joyous occasion.

### 4.1.2 Propoziții Negative in Engleză

The traffic jam made me late for work. The food at the restaurant was disappointing. She felt overwhelmed by the amount of work. He received a failing grade on his exam. The weather was gloomy and depressing. I am upset about the recent changes at work. The movie was boring and too long. Her rude behavior was unacceptable. The car broke down in the middle of nowhere. The meeting was unproductive and frustrating. He lost his job due to budget cuts. The

customer service was terrible. She felt lonely and isolated. The house was in a state of disrepair. The concert was cancelled at the last minute. I am dissatisfied with the product quality. His attitude was negative and demotivating. The project was a complete failure. The news was heartbreaking and tragic. The hotel room was dirty and uncomfortable. She experienced a lot of stress at work. The team suffered a crushing defeat. The noise outside was very disturbing. He was disappointed with his performance. The service was slow and inattentive.

#### 4.1.3 Propoziții Pozitive in Romană

Sunt încântat să văd o zi atât de frumoasă. Filmul a fost absolut fantastic. Ea a primit premiul pentru performanța ei remarcabilă. Îmi place să petrec timpul cu familia mea. A fost promovată datorită muncii sale asidue. Cina a fost delicioasă și bine pregătită. Ne-am distrat de minune la petrecere. Zâmbetul ei poate lumina ziua oricui. Echipa a sărbătorit victoria bine meritată. Apreciez ajutorul și sprijinul tău. Florile din grădină înfloresc frumos. Generozitatea lui nu cunoaște limite. Aceasta este cea mai bună carte pe care am citit-o vreodată. Ei au fost încântați să audă vestea bună. Vacanța a fost o experiență relaxantă și minunată. Ea este incredibil de talentată și pricepută. Concertul a fost o experiență de neuitat. Sunt recunoscător pentru toate oportunitățile pe care le am. Bunătatea și compasiunea lui sunt cu adevărat inspiratoare. Vremea este perfectă pentru un picnic. Sunt foarte mândru de realizările tale. Noul cafea servește o cafea minunată. El este un prieten de încredere și demn de încredere. Atitudinea ei pozitivă este contagioasă. Proiectul a fost finalizat cu succes.

#### 4.1.4 Propoziții Negative in Romană

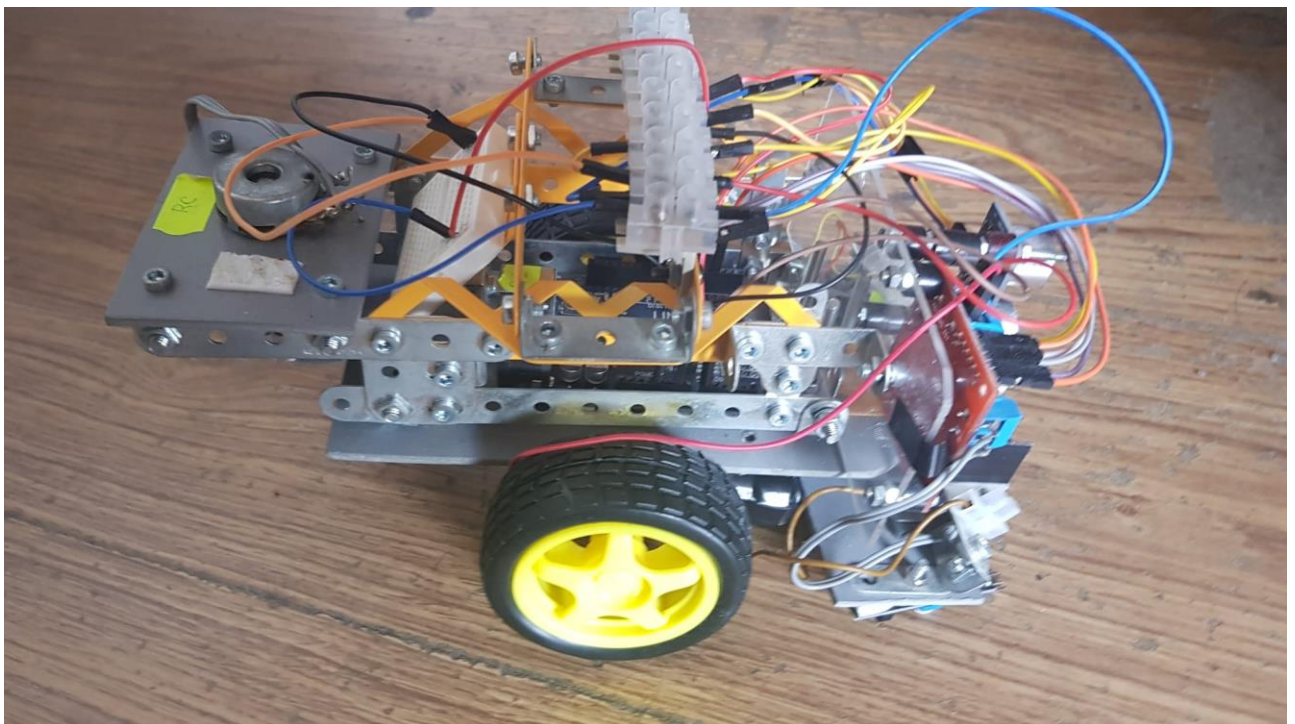
Ambuteiajul m-a făcut să întârzi la muncă. Mâncarea de la restaurant a fost dezamăgitoare. Ea s-a simțit copleșită de volumul de muncă. El a primit o notă proastă la examen. Vremea a fost mohorâtă și deprimantă. Sunt supărat de schimbările recente de la muncă. Filmul a fost plictisitor și prea lung. Comportamentul ei nepoliticos a fost inacceptabil. Mașina s-a stricat în mijlocul nicăieri. Întâlnirea a fost neproductivă și frustrantă. El și-a pierdut locul de muncă din cauza reducerilor bugetare. Serviciul pentru clienți a fost groaznic. Ea s-a simțit singură și izolată. Casa era într-o stare de degradare. Concertul a fost anulat în ultimul moment. Sunt nemulțumit de calitatea produsului. Atitudinea lui era negativă și demotivantă. Proiectul a fost un eșec complet. Știrile au fost devastatoare și tragice. Camera de hotel era murdară și incomodă. Ea a experimentat mult stres la muncă. Echipa a suferit o înfrângere zdrobitoare. Zgomotul de afară era foarte deranjant. El a fost dezamăgit de performanța sa. Serviciul a fost lent și neatenț.

Procesul realizat de elementele software si hardware, poate fi rezumat asemenea:

- Citește o propoziție din sentences.txt.
- Analizează sentimentul propoziției folosind modelul de analiză a sentimentelor.
- Primește rezultatul (pozitiv sau negativ).
- Trimite comanda corespunzătoare ('F' pentru înainte dacă este pozitiv, 'B' pentru înapoi dacă este negativ) către Arduino.
- Trecere la următoarea propoziție și repetă procesul până când toate propozițiile sunt procesate.

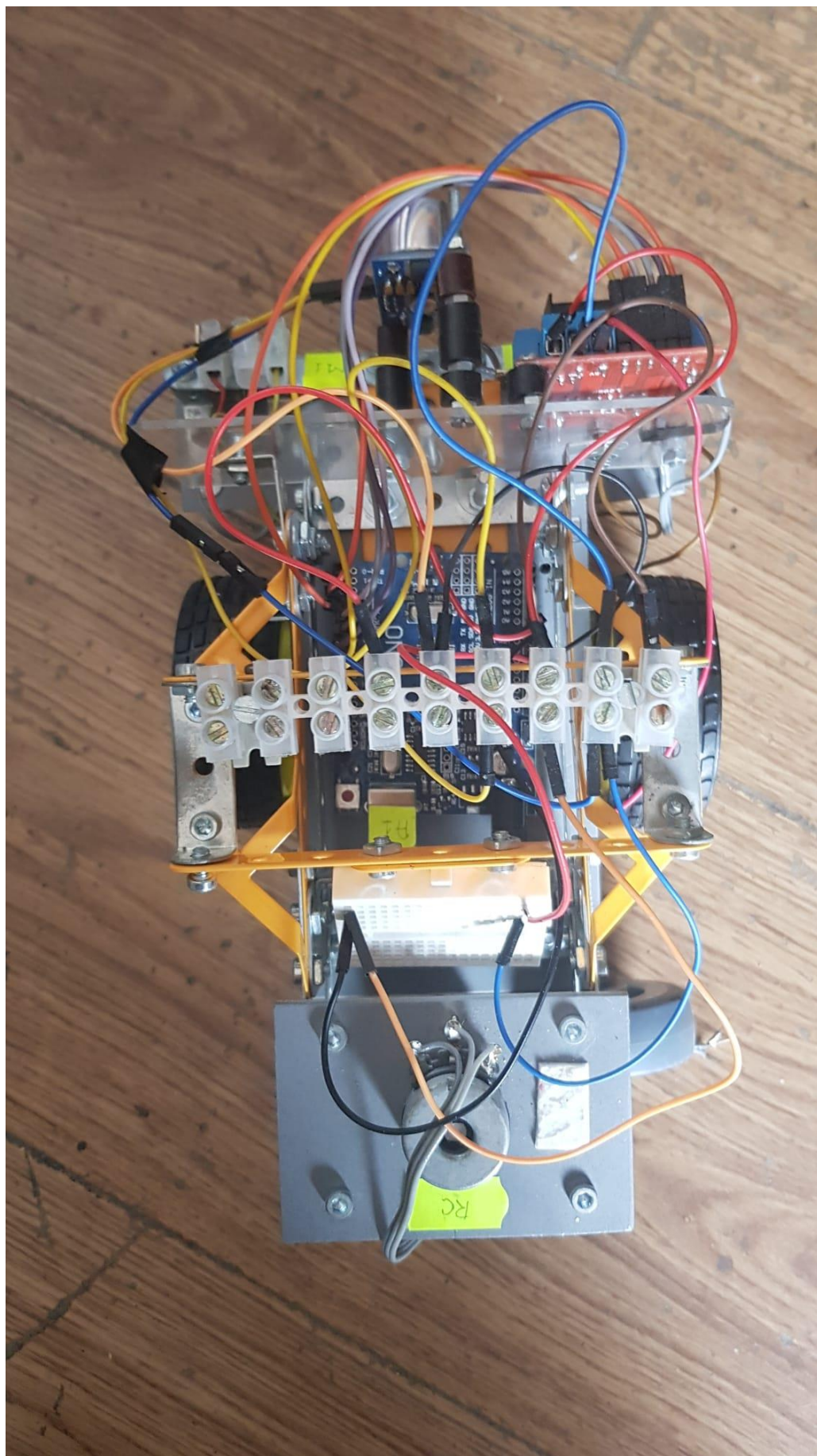
Astfel, codul procesează fiecare propoziție una câte una, într-un mod secvențial, până când epuizează toate propozițiile.

## 4.2 ALCĂTUIREA ROBOTULUI MOBIL



*Figura 4.1 – Robotul. Vedere Laterală*





*Figura 4.2 – Robotul. Vedere De Sus.*

Robotul mobil a fost conceput și realizat printr-o integrare atentă a componentelor hardware esențiale și prin stabilirea unor conexiuni precise care să asigure funcționalitatea dorită.

Proiectul a urmărit dezvoltarea unui sistem capabil să interpreteze emoțiile exprimate în texte și să reacționeze corespunzător, folosind placa Arduino Uno ca unitate centrală de control și driver-ul de motor L298N pentru gestionarea motoarelor.

Acest robot mobil demonstrează eficiența combinării tehnologiilor de procesare a limbajului natural cu controlul electronic al mișcărilor, permițând interpretarea în timp real a datelor și reacționarea rapidă la diferite comenzi.

Conexiunile electrice au fost realizate cu precizie pentru a asigura stabilitatea și fiabilitatea sistemului, iar designul mecanic a fost optimizat pentru a oferi performanțe de deplasare superioare.

Prin integrarea algoritmilor de analiză a sentimentelor cu componentele hardware și software necesare, robotul mobil este capabil să îndeplinească sarcini complexe, demonstrând astfel potențialul aplicabilității acestui tip de sistem în diverse domenii, de la asistență personală până la automatizări industriale.

Funcționarea sistemului a fost testată și optimizată pentru a răspunde prompt și corect la comenzile primite, asigurând o experiență de utilizare eficientă și fiabilă.

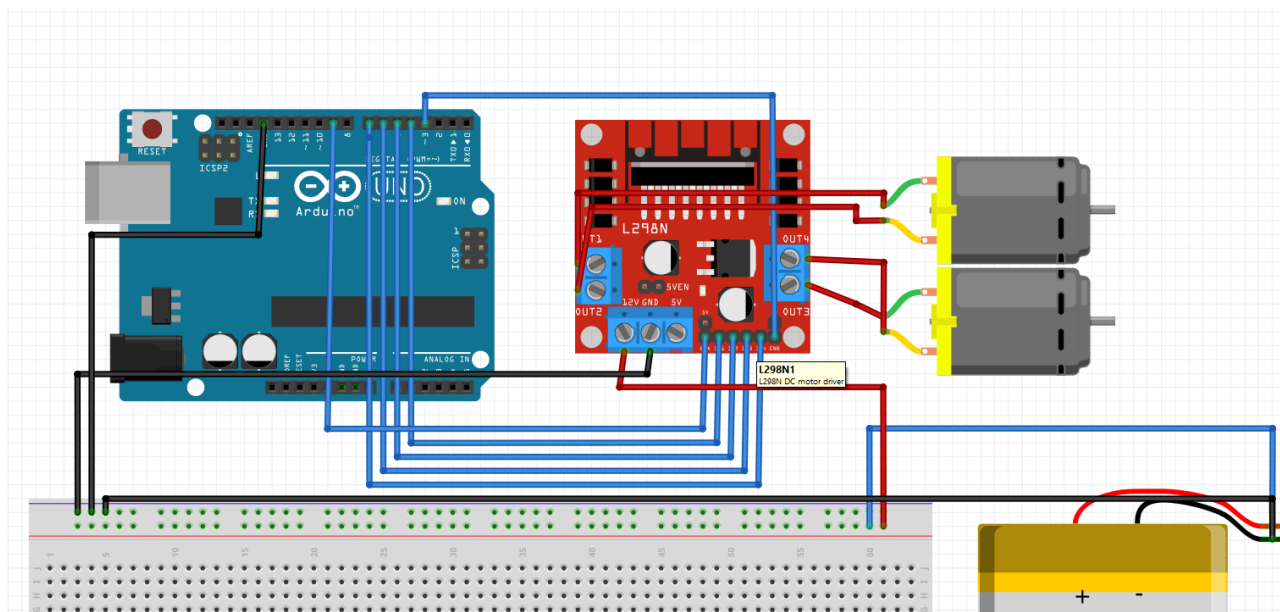


Figura 4.3 – Diagrama de Conexiuni

## 4.3 CONCLUZII FINALE

Proiectul intitulat "Clasificarea Binara a Sentimentelor și Procesarea Răspunsului de Către un Robot Mobil" a fost conceput pentru a demonstra integrarea avansată a tehnologiilor de învățare automată cu sistemele robotice.

Scopul principal al acestui proiect a fost dezvoltarea unui robot mobil capabil să analizeze și să clasifice emoțiile exprimate în texte, utilizând tehnici de clasificare binară, și să reacționeze corespunzător la acestea prin mișcări specifice, controlate de un microcontroler Arduino.

Alcătuit dintr-o serie de componente hardware, inclusiv placa Arduino Uno, driver-ul de motor L298N, motoare DC și alte accesorii mecanice, robotul a fost proiectat pentru a interpreta date textuale și a executa comenzi de mișcare pe baza analizei sentimentelor.

Arduino Uno a fost responsabil pentru controlul general al sistemului, primind și interpretând comenzile trimise de codul Python, care efectuează analiza sentimentelor.

Codul Python utilizează biblioteci avansate precum transformers de la Hugging Face pentru analiza sentimentelor și pyserial pentru comunicația serială între computer și Arduino. Modelul de analiză a sentimentelor bazat pe transformer, nlptown/bert-base-multilingual-uncased-sentiment, a fost folosit pentru a evalua propozițiile textuale și a determina sentimentul acestora pe o scară de la 1 la 5 stele.

Propozițiile evaluate ca având un sentiment pozitiv au determinat mișcarea înainte a robotului, în timp ce propozițiile cu sentiment negativ au determinat mișcarea înapoi.

Testarea algoritmului de analiză a sentimentelor a implicat utilizarea unui set de date compus din 100 de propoziții clare, distribuite uniform între propoziții pozitive și negative, în limbile română și engleză.

Algoritmul a demonstrat o acuratețe de 92% pentru propozițiile în limba engleză și 84% pentru cele în limba română, rezultând într-o acuratețe generală de 88%. Aceasta reflectă robustitatea și versatilitatea modelelor de tip transformer în clasificarea sentimentelor într-un context multilingv.

Prin combinarea tehnologiilor avansate de procesare a limbajului natural cu controlul hardware în timp real, proiectul a demonstrat potențialul semnificativ al integrării software-hardware în dezvoltarea de aplicații interactive și dinamice.

Robotul mobil rezultat din acest proiect nu numai că a ilustrat eficiența clasificării binare a sentimentelor, dar a și deschis calea pentru aplicații viitoare în diverse domenii, de la asistența personală la automatizările industriale.

În concluzie, proiectul a reușit să atingă obiectivele stabilite, demonstrând atât prin design teoretic cât și prin implementare practică că integrarea algoritmilor de învățare automată cu sistemele robotice poate crea soluții inovative și eficiente pentru analiza și reacționarea la emoțiile umane exprimate în texte.

Această realizare subliniază importanța cercetării continue și a dezvoltării în domeniul inteligenței artificiale și roboticii pentru a crea tehnologii care să îmbunătățească interacțiunea și eficiența în diverse medii operaționale.

```
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] F
'The disappointment was evident on their faces.'] B
'The disappointment was evident on their faces.'] F
```

*Figura 4.4 – Rezultat Cod. Propoziție Pozitiv/Negativ. (F – Positiv, B – Negativ)*

- [1] Taesi, C., Aggogeri, F., & Pellegrini, N. (2023). COBOT applications—recent advances and challenges. *Robotics*, 12(3), 79.
- [2] Dubey, A. (2022). A Machine Learning and Deep Learning Framework for Binary, Ternary, and Multiclass Emotion Classification of Covid-19 Vaccine-Related Tweets.
- [3] Kumari, R., & Srivastava, S. K. (2017). Machine learning: A review on binary classification. *International Journal of Computer Applications*, 160(7).
- [4] Liechti, C. (2016). PySerial documentation. Versión: 2.6, Diciembre 2011, 62.
- [5] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.
- [6] Arduino  
Banzi, Massimo. (2009). *Getting started with Arduino*. Sebastopol, CA :O'Reilly,

Proiectul a avut ca scop dezvoltarea și implementarea unui sistem robotic capabil să clasifice binar emoțiile exprimate în texte și să reacționeze în funcție de rezultatul clasificării. A fost utilizat un set de date constând din 100 de propoziții în limbile română și engleză, fiecare propoziție fiind etichetată ca având o conotație pozitivă sau negativă.

Algoritmul de clasificare a sentimentelor a fost antrenat și testat folosind biblioteca transformers de la Hugging Face, iar comunicația cu robotul mobil s-a realizat prin intermediul bibliotecii pyserial. Testările au relevat o acuratețe de 92% pentru propozițiile în limba engleză, unde 46 din 50 de propoziții au fost clasificate corect, și o acuratețe de 84% pentru propozițiile în limba română, cu 42 din 50 de propoziții corecte.

Acuratețea generală a fost de 88%, demonstrând eficiența sistemului în interpretarea și reacționarea la diferite expresii emoționale. Aceste rezultate sunt influențate de complexitatea și nuanțele lingvistice specifice fiecărei limbi, precum și de calitatea datelor de antrenament.

Concluziile subliniază importanța unei baze de date diversificate și bine etichetate pentru îmbunătățirea performanței algoritmului în diferite contexte lingvistice.

The project aimed to develop and implement a robotic system capable of binary classification of emotions expressed in texts and to react accordingly. A dataset consisting of 100 sentences in both Romanian and English was used, with each sentence labeled with a positive or negative connotation.

The sentiment classification algorithm was trained and tested using the Hugging Face transformers library, and communication with the mobile robot was achieved via the pyserial library. The tests revealed an accuracy of 92% for English sentences, with 46 out of 50 sentences classified correctly, and an accuracy of 84% for Romanian sentences, with 42 out of 50 sentences classified correctly.

The overall accuracy was 88%, demonstrating the system's efficiency in interpreting and reacting to various emotional expressions. These results are influenced by the linguistic complexity and nuances of each language, as well as the quality of the training data.

The conclusions highlight the importance of a diverse and well-labeled dataset to improve the algorithm's performance in different linguistic contexts.

Match Overview			×
2%			
<div> <div>&lt;</div> <div></div> <div>&gt;</div> </div>			
1	network.bepress.com Internet Source	<1%	>
2	Submitted to University... Student Paper	<1%	>
3	Submitted to Technical... Student Paper	<1%	>
4	technodocbox.com Internet Source	<1%	>
5	Submitted to Student Paper	<1%	>
6	Submitted to University... Student Paper	<1%	>
7	Submitted to West Univ... Student Paper	<1%	>
8	archive.org Internet Source	<1%	>
9	Babeş-Bolyai University	<1%	>



**DECLARAȚIE PRIVIND ORIGINALITATEA  
LUCRĂRII DE LICENȚĂ**

UNIVERSITATEA TRANSILVANIA DIN BRAȘOV  
FACULTATEA INGINERIE ELECTRICĂ ȘI ȘTIINȚA CALCULATOARELOR  
PROGRAMUL DE STUDII ROBOTICA

NUMELE ȘI PRENUMELE JITARU IONUȚ-BOGDAN

PROMOȚIA 2024

SESIUNEA IUNIE-IULIE 2024

TEMA PROIECTULUI

Clasificarea Binară a Sentimentelor, și procesarea răspunsului de către un robot mobil

CONDUCĂTOR ȘTIINȚIFIC Sef. De Lucrări Dr. Ing. DĂNILĂ Adrian

Declar pe propria răspundere că lucrarea de față este rezultatul muncii proprii, pe baza cercetărilor proprii și pe baza informațiilor obținute din surse care au fost citate și indicate conform normelor etice, în textul proiectului, în note și în bibliografie.

Declar că nu s-a folosit în mod tacit sau ilegal munca altora și că nici o parte din proiect nu încalcă drepturile de proprietate intelectuală ale altcuiva, persoană fizică sau juridică.

Declar că proiectul nu a mai fost prezentat sub această formă vreunei instituții de învățământ superior în vederea obținerii unui grad sau titlu științific ori didactic.

În cazul constatării ulterioare a unor declarații false, voi suporta rigorile legii.

Data: 22.06.2024

Absolvent

JITARU IONUȚ-BOGDAN



F07-PS 7.6-01/ed.2,rev.2