

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Программалық инженерия»

Жұмабаева Г.Ж.

Жазба бойынша танып білу жүйесіне арналған программалық камтама өңдеу

**ТҮСІНДІРМЕ ЖАЗБА**

дипломдық жобаға

5B070400 – «Есептеу техникасы және бағдарламалық камтамасыз ету»

мамандығы

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ


Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Программалық инженерия»

**ҚОРҒАУҒА ЖІБЕРІЛДІ**

ПИ кафедра меңгерушісі

техника ғылымдарының  
кандидаты, доцент, ассистент-  
профессор

 Р. Юнусов  
" 14 " мае 2019ж.

**ТҮСІНДІРМЕ ЖАЗБА**

дипломдық жобаға


Тақырыбы: “Жазба бойынша танып білу жүйесіне арналған программалық  
қамтама өңдеу ”

5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»  
мамандығы

Орындаған

Жұмабаева Г.Ж.

Ғылыми жетекші  
техникалық ғылымдар  
магистрі, лектор

 Ж.М. Алибиева  
" 02 " 05 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӨТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Программалық инженерия»

5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»

**БЕКІТЕМІН**

ПИ кафедра меңгерушісі,  
техника ғылымдарының  
кандидаты, доцент,  
ассистент-профессор

Р. Юнусов

" 14 " \_\_\_\_\_ 2019ж.

**Дипломдық жобаны орындауға  
ТАПСЫРМА**

Білім алушыға Жұмбаева Гулсезім Жұмабайқызы

Тақырыбы: Жазба бойынша танып білу жүйесіне арналған программалық қамтама өңдеу

Университет ректоры бұйрығының № 1162-б "16" қазан 2018 ж. шешімімен бекітілген.

Орындалған жобаның өткізу мерзімі " 14 " мамыр 2018 ж.

Дипломдық жобаның бастапқы мәліметтері: Жобаның төлқұжаты, технология бойынша техникалық құжаттама, техникалық тапсырма, диаграмма түрінде сипаттамасы.

Есеп – түсініктеме жазбаның талқылауға берілген сұрақтардың тізімі:

a) қолданылған әдістерді қарастыру;

b) тұжырымдамаға сәйкес веб-қосымшаның көп деңгейлі архитектурасын енгізу;

в) пайдаланушы интерфейсін жобалау және дамыту;

г) бағдарламаны құру, қателіктерді анықтау және түзету, тестілеу;



Графикалық материалдар тізімі (міндетті суреттердің нақты көрсетілуімен): презентацияның 20 слайды ұсынылған.

Ұсынылған негізгі әдебиеттер: 15 әдебиеттер тізімінен

Дипломдық жобаны орындау  
КЕСТЕСІ

Бөлімдердің атаулары, зерттелген мәселелердің тізімі	Ғылыми жетекшіге және кеңесшілерге ұсыну мерзімі	Ескерту
1. Кіріспе	10.01.2018	тоқ
2. Тапсырма қойылымы және бағдарламалау ортасын таңдау	18.01.2018	тоқ
3. Бағдарламалық қамтаманы жобалау.	01.02.2018	тоқ
4. Бағдарламаны әзірлеу	14.02.2018	тоқ
5. Бағдарламаның мүмкіндіктерін дамыту	12.03.2018	тоқ
6. Дипломдық жобаға түсіндірме жазба жазу	30.03.2018	тоқ

Дипломдық жұмыс бөлімдерінің кеңесшілерінің аяқталған жұмысқа қойған қолтаңбалары

Бөлімдер атауы	Кеңес берушілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қолтаңба қойылған мерзімі	Қолы
Нормалық бақылаушы	А.Б. Рамазан Тьютор	13.05.19	
Бағдарламалық бөлім	А.М. Байғаринов Техн. ғыл. магистр, лектор	08.05.19	

Ғылыми жетекші \_\_\_\_\_  Ж.М. Алибиева

Тапсырманы орындауға қабылдап алған студент \_\_\_\_\_  Г.Ж. Жұмабаева

Күні \_\_\_\_\_ «08» 01 2019 ж.

## АҢДАТПА

Ұсынылған дипломдық жобада жазбаны танып білу жүйесіне арналған бағдарламалық қамтама көрсетілген.

Бұл жобаның мақсаты – қағаз бетіне жазылған қолжазба суретін жасалынған қамтама көмегімен танып білу және жазбаны баспа әріптерімен жазылған текске ауыстыру болып табылады. Сонымен қоса, жазылған дипломдық жоба мақсаттарына машиналық оқыту және жасанды нейрондық желі тақытыптарын зерттеу де жатады.

Дипломдық жобада жасалынған бағдарламалық қамтаманың қолданыс табатын салалары, ұқсас қамтамалар тізімі, бағдарлама интерфейсінің сипаттамасы, бағдарламалық код бөліктері және бағдарламаның техникалық тапсырмасы көрсетілген.

## АННОТАЦИЯ

В представленном дипломном проекте представлено программное обеспечение для системы распознавания записи.

Целью данного проекта является распознавание рукописного текста на бумаге с помощью разработанного обеспечения и замена записи на текст, написанный печатными буквами. Кроме того, целью написанного дипломного проекта является изучение компонентов искусственной нейронной сети и машинного обучения.

В дипломном проекте представлены области применения программного обеспечения, перечень аналоговых программ, описание интерфейса программы, части программного кода и техническое задание программы.

## ANNOTATION

In the submitted thesis project presents software for the recording recognition system.

The purpose of this project is to recognize handwritten text on paper using the developed software and replace the record with text written in block letters. In addition, the purpose of the written graduation project is to study the components of an artificial neural network and machine learning.

In the thesis project presents the application of the software, the list of analog programs, a description of the program interface, parts of the program code and the program's technical task.

## МАЗМҰНЫ

	Кіріспе	8
1	Танып білуде қолданылатын негізгі түсініктер мен теориялар	10
1.1	Машиналық оқыту	10
1.1.1	Машиналық оқыту алгоритмдерінің түрлері	11
1.1.2	Машиналық оқытудың практикалық қолдану салалары	15
1.1.3	Оқыту үшін қолданылатын деректер	15
1.2	Жазбаны танып білуде жасанды нейрондық желіні қолдану	16
1.2.1	Жасанды нейрондық желіні оқыту	18
1.3	Ұқсас бағдарламаларға аналитикалық шолу жасау	19
1.3.1	ABBYY FineReader бағдарламасы	19
1.3.2	SimpleOCR бағдарламасы	20
1.3.3	FreeOCR бағдарламасы	20
1.3.4	Microsoft Office Document Imaging бағдарламасы	21
2	Құрылымдық-техникалық бөлім	22
2.1	Жазбаны танып білуде YOLO моделін қолдану	22
2.2	Жазбаны танып білу бағдарламасын пайдалану алгоритмі	25
3	Функционалдық талаптарды ұйымдастыру	27
3.1	Функционалдық диаграммалар	27
	Қорытынды	33
	Пайдаланылған әдебиеттер тізімі	34
	Қосымша А	35
	Қосымша Б	37
	Ерекшеліктері	46



## КІРІСПЕ

Жазбаны танып білу – компьютердің интеллектуалды жазбаны қабылдау және интерпретациялау қабілеті. Қазіргі уақытта баспа әріптерімен жазылған мәтіндерді жақсы сапамен сканерленген болса ғана нақты танып білу жүргізіледі. Бул жағдайда танып білу дәлдігі 99%-дан асады. Ал абсолютті танып білу тек қана адам көмегімен жүргізілген өзгертулерден кейін ғана болады. Қолмен жазылған мәтіндерді танып білу қазіргі таңда зерттеулер үстінде. Жазбаны танып білу кеңінен таралған және зерттеліп жатырған проблемалардың бірі болып табылады. Жазбаны танып білуде оптикалық танып білу (OCR-жүйесі) жүйесін қолдану кең қолданылады.

Жазбаны танып білу жүйесін дипломдық жоба ретінде таңдап алған мақсатым – қолмен жазылған жазбаны оптикалық танып білуді және жалпы танып білу жүйесін зерттеу болды. Зерттеу жұмысының объектісі ретінде суреттегі жазбаны танып білу және жазаны детекциялау процессі жатады.

Жазбаны танып білудің практикалық маңыздылығына тоқталып кететін болсам: дипломдық жобамда ұсынылған әдістер мен алгоритмдер кәсіпорындардың құжат айналымы жүйесінде практикалық қолдануға, халықтан сауалнама алу жұмысы кезінде, паспорттық-визалық қызмет және басқа да жүйелерде қолдануға арналған және кіріс деректері ретінде қолмен жазылған мәтіннің бейнесі болып табылады.

Жазбаны танып білудің екі әдісі бар: «оффлайн» танып білу және «онлайн» танып білу. Оффлайн танып білу әдісі қағазға жазылған жазба бойынша жүргізіледі. Онлайн танып білу әдісі графикалық планшет көмегімен қалам қозғалысын да ескеру арқылы жүргізіледі. Жазбаны оффлайн танып білу жүйесі көбінесе қолмен жазылған құжаттарды өңдеу салаларында сәтті қолданылады. Оған мысалға сақтандыру компанияларын алып қарауымызға болады. Танып білу сапасын құрылымдалған құжаттарды қолдану арқылы арттыруға болады. Сонымен қоса, танып білу сапасын енгізетін символдардың диапазонын азайту арқылы да арттыруымызға болады. Жазбаны оффлайн танып білу онлайн танып білу жүйесімен салыстырғанда әлдеқайда күрделі болып келеді.

Пернетақтаның көмегімен дәстүрлі енгізуге балама ретінде қолданылған қолжазба мәтінін енгізудің сериялық құрылғылары 1980 жылдың басында пайда болды. Бұл, мысалы, Pencept Penpad және Inforite point-of-sale terminal сияқты қолжазба жазбасы бар терминалдар болды. Дербес компьютерлер нарығын кеңейтуге байланысты жеке компьютерлерде пернетақтаны және тінтуірді ұсынатын жалғыз жүйемен ауыстыруға арналған PenCept, SIC сияқты бірнеше коммерциялық өнімдер пайда болды. Ашық сатылымда бірінші болып, 1989 жылы grid Systems-тен шығарылған GRiDPad планшетті компьютері болды. Оның операциялық жүйесі MS-DOS-қа негізделген болатын.

90-шы жылдардың басында аппараттық қамтамасыз етуді өндірушілер, яғни NCR, IBM және EO қоса алғанда, GO Corp әзірлеген penpoint операциялық

жүйесі бар планшеттік компьютерлерді шығара бастады. PenPoint барлық жерде қолжазба енгізуді пайдаланды және бөгде бағдарламалық қамтамасыз етулермен де үйлесімділікті қамтамасыз ете білді. IBM планшеттік компьютері бірінші ThinkPad және IBM қолжазба мәтінін тануды қолданды. Бұл тану жүйесі кейінірек Microsoft Windows for Pen Computing және IBM 's Pen OS/2 үшін көшірілді. Осы технологиялардың ешқайсысы коммерциялық табысқа жете алмады.

Электрониканың дамуы қолжазба енгізуді тану үшін қажетті есептеу қуатын планшеттік компьютерлерге қарағанда аз мөлшерге бөлуге және PDA үшін енгізу құралы ретінде қолжазба мәтінін тануды пайдалануға мүмкіндік берді. Жазбаша енгізуді қамтамасыз ететін бірінші PDA Apple Newton болды, ол көпшілікке пайдаланушы интерфейсінің артықшылықтарын көрсетті. Алайда, бұл құрылғы пайдаланушының хат мәнерін зерттеуге тырысқан бағдарламалық жасақтаманың коммерциялық жетілмеуінің себебінен сәтті болды.

# 1 Жазбаны танып білуде қолданылатын негізгі түсініктер мен теориялар

## 1.1 Машиналық оқыту

Машиналық оқыту - бұл нақты міндеттерді тиімді орындау үшін компьютерлік жүйелермен қолданылатын алгоритмдер мен статистикалық үлгілерді нақты нұсқауларды пайдаланбай, оның орнына шаблондар мен ақыл-ойға сүйеніп ғылыми зерттеу. Бұл жасанды интеллект жиынтығы ретінде қарастырылады. Машиналық оқыту алгоритмдері нақты болжамдар немесе шешімдер жасау есептерін бағдарламалаусыз орындау үшін "оқыту деректері" ретінде белгілі іріктемелі деректер негізінде математикалық модель құрастырады.

Машиналық оқыту алгоритмдері электронды поштаны фильтрациялау процесінде және компьютерлік көру сияқты түрлі қосымшаларда қолданылады. Машиналық оқыту компьютерлерді пайдалана отырып болжауға бағытталған есептеуіш статистикамен тығыз байланысты. Математикалық оңтайландыруды зерттеу машиналық оқыту саласындағы әдістерді, теорияларды және қолдану салаларын жеткізеді. Деректерді интеллектуалды талдау машиналық оқытудағы зерттеу саласы болып табылады және мұғалімсіз оқыту арқылы деректерді зерттеу талдауына бағытталады.

Машиналық оқытудың міндеттері бірнеше санатқа бөлінеді. Бақыланатын оқыту кезінде алгоритм деректер жиынтығынан математикалық модель құрастырады. Деректер жиынтығының кіріс және қалаулы шығу жолдары болады. Мысалы, егер тапсырма белгілі бір объектінің бейнесі бар-жоғын анықтау керек болатын болса, онда оқытудың бақыланатын алгоритмі үшін оқыту деректері осы объектімен және объектісіз (кіру) бейнелерді қамтиды және әрбір сурет объектінің бар-жоғын білдіретін белгіге (шығу) ие болады. Ерекше жағдайларда, кіріс деректері тек ішінара немесе тек арнайы пікірлер болуы мүмкін. Жартылай басқарылатын оқыту алгоритмдері оқытудың толық емес мәліметтерінен алынған математикалық модельдерді әзірлейді, мұнда іріктеменің кіріс мәліметтерінің бір бөлігі белгілері жоқ болады.

Классификация алгоритмдері шектеулі мәндер жиынтығымен шектелген кезде қолданылады. Электрондық хаттарды сүзетін жіктеу алгоритмі үшін кіріс электрондық хат, ал шығыс-электрондық хат жіберу қажет қалтаның аты болады. Спам хабарларды сәйкестендіретін алгоритм үшін қорытынды true және false логикалық мәндерімен ұсынылған "спам" немесе "спам емес" деп болжауға болады. Регрессия алгоритмдері үздіксіз шығулардың болу себебінен аталады, бұл олар диапазон шегінде кез келген мәнге ие болуы мүмкін дегенді білдіреді. Үздіксіз мәнді мысалдары объектінің температурасы, ұзындығы немесе бағасы болып табылады.

Классификация түсінігін айтып кететін болсақ, машиналық оқыту мен статистикада жіктеу көптеген санаттардың (кіші топтардың) қайсысына жаңа бақылау жататынын анықтау проблемасын білдіреді. Мысал ретінде "спам" немесе "спам емес" класына осы хатты беру, сондай-ақ пациенттің байқалатын сипаттамалары негізінде осы пациентке диагноз қою (жынысы, артериялық қысым, белгілі бір симптомдардың болуы немесе болмауы және т.б.) жатады. Жіктеу бейнелерді тану үлгісі болып табылады.

Бақылаусыз оқытуда алгоритм деректер жиынтығынан математикалық модель құрастырады, ол тек кіріс деректерінен тұрады және қажетті шығыс белгілерін қамтымайды. Бақыланбайтын оқыту алгоритмдері деректер нүктелерін топтастыру немесе кластерлеу сияқты деректердің құрылымын іздеу үшін пайдаланылады. Бақыланбайтын оқыту деректердегі заңдылықтарды анықтай алады және функцияны оқытудағы сияқты санаттар бойынша кіріс деректерін топтастыра алады. Өлшемдікті кішірейту – бұл деректер жиынтығындағы "объектілер" немесе кіріс деректерінің санын азайту процесі болып табылады.

Белсенді оқыту алгоритмдері бюджет негізінде кірулердің шектеулі жиынтығы үшін қажетті шығуларға (оқыту белгілеріне) қол жеткізеді және ол оқыту белгілерін сатып алатын кірулердің таңдауын оңтайландырады. Интерактивті пайдалану кезінде олар таңбалау үшін пайдаланушыға ұсынылуы мүмкін. Нығайтуды оқыту алгоритмдері динамикалық ортада оң немесе теріс бекіту түрінде кері байланыс алады және автономды көлік құралдарында немесе адам қарсыласына қарсы ойында пайдаланылады.

Машиналық оқытудың басқа мамандандырылған алгоритмдеріне тақырыптық модельдеу жатады, онда компьютерлік бағдарлама табиғи тілде құжаттар жиынтығын алады және ұқсас тақырыптарды қамтитын басқа да құжаттарды табады. Машиналық оқыту алгоритмдері тығыздықты бағалау есептеріндегі ықтималдық тығыздығының қадағаланбайтын функциясын іздеу үшін пайдаланылуы мүмкін. Мета-оқыту алгоритмдері алдыңғы тәжірибеге негізделген өзінің индуктивті ығысуларын зерттейді. Дамытушы робототехникада роботтарды оқыту алгоритмдері дербес зерттеу және адамдармен әлеуметтік қарым-қатынас жасау арқылы жаңа дағдыларды алу үшін оқу бағдарламасы ретінде белгілі оқу тәжірибесінің өз кезектілігін жасайды. Бұл роботтар белсенді оқыту, жетілу, моторлы синергия және имитация сияқты басқару механизмдерін пайдаланады.

### **1.1.1 Машиналық оқыту алгоритмдерінің түрлері**

Машиналық оқыту алгоритмдерінің түрлері олар енгізетін және шығаратын деректер типі және олар шешуге арналған есептер немесе проблемалар типі бойынша ерекшеленеді. Машиналық оқыту нейрожүйелер туралы ғылымды негізге алып, желілерді оқыту әдістерін және олардың архитектурасының топологияларының түрлеріне бөлу нәтижесінде және

математикалық статистика әдістерін қолдану арқылы пайда болды. Перцептрон және көп қабатты перцептрон (сондай-ақ олардың модификациялары) сияқты нейрожүйелердің базалық түрлері мұғаліммен де, мұғалімсіз де, өзін-өзі ұйымдастырумен де оқытылуы мүмкін. Машиналық оқытудың негізгі түрлеріне келесілер жатады: мұғаліммен оқыту, мұғалімсіз оқыту, нығайтумен оқыту, белсенді оқыту, мұғалімнің жартылай қатысуымен оқыту, трансдуктивті оқыту, көпесепті оқыту, көп нұсқалы оқыту, Байесов желісі және бустинг.

Мұғаліммен оқыту кезінде әрбір прецедент үшін талап етілетін шешімі бар жағдай беріледі. Мұғаліммен оқыту біршене түрлі алгоритмдермен жүргізілуі мүмкін, оларға: тірек векторлар әдісі, қателерді жөндеу әдісі, қатені кері тарату әдісі, жасанды нейрондық желі жатады. Бұл эксперимент кері байланысы бар кибернетикалық эксперименттің жеке жағдайы болып табылады. Осы экспериментті қою эксперименталды жүйенің, оқыту әдісінің және жүйені сынау әдісінің немесе сипаттамаларды өлшеудің болуын болжайды. Эксперименттік жүйе өз кезегінде сыналатын (пайдаланылатын) жүйеден, сыртқы ортадан алынатын ынталандыру кеңістігінен және нығайтуды басқару жүйесінен (ішкі параметрлердің реттеуішінен) тұрады. Бекітуді басқару жүйесі ретінде автоматты реттеуші құрылғы (мысалы, термостат) немесе жүйе жадының жағдайын өзгертетін нығайтудың ерекше ережелерін қолдану арқылы сыналатын жүйенің реакциясына және сыртқы ортаның стимулына ден қоюға қабілетті адам-оператор (мұғалім) пайдаланылуы мүмкін.

Мұғалімсіз оқыту кезінде әрбір прецедент үшін тек "жағдай" ғана беріледі, нысандарды кластерге топтастыру, объектілердің қос ұқсастығы туралы деректерді пайдалана отырып, және/немесе деректердің өлшемін төмендету талап етіледі. Мұғалімсіз оқыту алгоритмдеріне мыналарды жатқызамыз: нығайтудың альфа-жүйесі, нығайтудың гамма-жүйесі, жақын көршілер әдісі. Әрбір оқыту объектісі үшін "дұрыс жауап" мәжбүрлеп қойылса және жүйенің стимулдары мен реакцияларының арасында тәуелділікті табу талап етілсе, мұғалімсіз оқыту мұғаліммен оқуға жиі қарсы қойылады. Көптеген қолданбалы жетістіктерге қарамастан, мұғаліммен білім алу өзінің биологиялық дұрыс еместігі үшін сыналды. Кері байланыс арқылы түзету жасай отырып, қажетті және нақты шығу мәндерін салыстыра алатын мидағы оқыту механизмін көрсету қиын. Егер мидағы осындай механизм рұқсат етілсе, онда қалаған шығу қайдан пайда болады? Мұғалімсіз оқыту - биологиялық жүйеде оқытудың шыншыл үлгісі. Кохонен және басқа да көптеген дамыған, ол шығу үшін мақсатты векторды қажет етпейді, демек, алдын ала анықталған мінсіз жауаптармен салыстыруды талап етпейді.

Нығайтумен оқыту кезінде әр прецедент үшін жұп қабылданған шешім бар жағдай беріледі. Нығайтумен оқыту алгоритміне генетикалық алгоритм жатады. Ал белсенді оқыту кезінде оқытылып жатырған алгоритм келесі зерттелетін жағдайды өз бетінше тағайындау мүмкіндігіне ие болады. Мұғалімнің жартылай қатысуымен оқыту кезінде прецеденттердің бір бөлігі үшін "талап етілетін шешім" жұбы, ал бір бөлігі үшін "жағдай" ғана беріледі.

Нығайтумен оқыту - машиналық оқыту тәсілдерінің бірі, оның барысында сыналатын жүйе (агент) кейбір ортамен өзара іс-қимыл жасай отырып оқиды. Кибернетика тұрғысынан алғанда, кибернетикалық эксперимент түрлерінің бірі болып табылады. Қабылданған шешімдерге бекітілген сигналдар болып табылады, сондықтан мұндай оқыту мұғаліммен оқытудың жеке жағдайы болып табылады, бірақ мұғалім орта немесе оның моделі болып табылады. Сондай-ақ, кейбір бекіту ережелері, мысалы, жасанды нейрондық орта жағдайында, формальды нейрондардың бір мезгілде белсенділігіне негізделген, себебі оларды мұғалімсіз оқуға жатқызуға болады. Агент ортаға әсер етеді, ал сәрсенбі агентке әсер етеді. Мұндай жүйе туралы ол кері байланыс бар дейді. Мұндай жүйені біртұтас ретінде қарастыру қажет, сондықтан орта мен агент арасындағы бөлім сызығы жеткілікті түрде шартты. Әрине, анатомиялық немесе физикалық тұрғыдан орта мен агент (организм) арасында белгілі бір шекара бар, бірақ егер бұл жүйені функционалдық тұрғыдан қарастырса, онда бөлу анық емес болады. Мысалы, мүсінші қолындағы кескіш мәрмәр кесегіне пішін беретін күрделі биофизикалық механизмнің бөлігі немесе жүйке жүйесін басқаруға тырысатын материалдың бөлігі деп санауға болады. Алғашқы рет кері байланыс арқылы оқыту 1961 жылы белгілі кеңес математигі Михаил Львович Цетлиннің жұмысында ұсынылды және зерттелді. М. Л. Цетлин сол кезде белгілі бір конструкцияның соңғы автоматын сыртқы ортаға салды, ол автоматты жазалады немесе ынталандырды. Орта реакциясына сәйкес автомат өзінің ішкі жағдайын өз бетінше өзгертті, бұл жазалау санын біртіндеп азайтуға, яғни оқытуға әкеп соқты. Бұл автоматтың мінез-құлқын талдау үшін алғаш рет А. А. Марков жасаған Марков тізбектерінің аппараты пайдаланылды, ол дәл және дәлелді нәтиже алуға мүмкіндік берді.

Трансдуктивтік оқыту - тек тестілік іріктеуден прецеденттер үшін болжанған кезде мұғалімді ішінара тарту арқылы оқыту. Логикада және статистикалық әдісте-трансдуктивтік ақыл-ой немесе трансдуктивтік әдіс жеке тестілік жағдайлар (оқыту деректері) негізінде байқалатын жеке жағдайлар (тестілік деректер) туралы қорытындылар болып табылады. Керісінше, индуктивті ой-пікір байқалатын жеке оқыту жағдайларын жалпы ережелерге әкеледі, содан кейін тест жағдайларына қолданылады. Трансдуктивтік модельдің болжамдары бірде-бір индуктивтік модельге қол жеткізілмеген жағдайларда мұндай жағдайлар әртүрлі сынақ үлгілерінде трансдуктивтік қорытынды нәтижесінде өзара қарама-қайшы болжамдар алынатын кезде орын алады. Трансдукция 1990-ші жылдары Владимир Вапникпен енгізілді. Оқытудың индуктивті үлгісі болып табылмайтын екілік жіктеу жағдайы болуы мүмкін, онда кіріс деректері әдетте екі топқа топтастырылады. Жеке, тестілік кіріс деректерінің үлкен жиынтығы жіктеме белгілері туралы қажетті ақпаратты бере отырып, дербес бірлік ретінде қарастырылатын бірнеше біртекті элементтерді іздестіруге көмектесе алады. Сол болжамдар тек оқыту жағдайларына негізделген функцияны индукциялайтын модельден алынбайтын болады, бұл бақыланатын оқытудың еденіне мысал бола алады, бірақ Вапник мотивациясы басқа болды. Бұл

категориядағы алгоритмнің үлгісі тірек векторлардың трансдуктивтік әдісі болып табылады (Transductive Support Vector Machine TSVM).

Трансдукцияға әкелетін үшінші себеп аппроксимация, жақындау қажеттігінен туындайды. Егер нақты қорытынды есептеу мүмкін болмаса, жеңілдету жеке жағдайларға сәйкес келетініне көз жеткізуге болады. Сонда жеке жағдайлар ерікті түрде бөлінуден (оқу материалдарын бөлуге міндетті емес) түседі, бұл оқытудың бақыланатын моделінің еденінде рұқсат етілмейді. Осы категорияға жататын алгоритмнің мысалы-байесовка машинасы (Bayesian Committee Machine BCM).

Көпесепті оқыту - өзара байланысты есептер тобын бір мезгілде оқыту және олардың әрқайсысы үшін өз жұптары, яғни "жағдай, талап етілетін шешім» болады. Көп нұсқалы оқыту - прецеденттер топтарға біріктірілуі мүмкін, олардың әрқайсысында барлық прецеденттер үшін "жағдай" бар, бірақ олардың біреуі үшін ғана (және қайсысы белгісіз) "талап етілетін шешім" жұбы бар оқыту. Ал бустинг - бұл әрбір келесі алгоритм барлық алдыңғы алгоритмдер композициясының кемшіліктерін өтеуге ұмтылатын машиналық оқыту алгоритмдерінің композициясын дәйекті құру процедурасы.

Байесов желісі - Байес бойынша көптеген айнымалы және олардың ықтималдық тәуелділіктерін білдіретін графикалық ықтималдық модель. Мысалы, байесов желісі симптомдар мен аурулар арасындағы тәуелділік туралы деректерді негізге ала отырып, бірқатар симптомдардың болуы немесе болмауы бойынша пациенттің немен ауырғанын есептеу үшін пайдаланылуы мүмкін. Байес желілерінің математикалық аппаратын американдық ғалым Джуда Перел, Тьюринг сыйлығының лауреаты жасаған. Формальды, байесовская желісі — бұл бағытталған ациклический баған, әр қияға сәйкес келетін кездейсоқ айнымалы, ал доға баған кодируют қарым-қатынас шартты тәуелсіздік осы айнымалы. Шыңдар кез келген түрдегі айнымалыларды, өлшенген параметрлермен, жасырын айнымалылармен немесе гипотезалармен болуы мүмкін. Байесов желілерін есептеу және оқыту үшін қолданылатын тиімді әдістер бар. Егер байесов желісінің айнымалылары дискретті кездейсоқ шамалар болса, онда мұндай желі дискретті байесов желісі деп аталады. Айнымалы тізбектерді моделдейтін байесов желілері динамикалық байесов желілері деп аталады. Байесовские желі, онда қатыса алады ретінде дискретті айнымалылар, сондай-ақ үздіксіз деп аталады гибридік байесовскими желілері. Доғалар шартты Тәуелсіздік қатынастарынан басқа, себеп - салдарлық Байес желілері деп те атайды.

Байесов желісі-айнымалы және олардың қатынастары үшін толық үлгі болғандықтан, ол ықтимал сұрақтарға жауап беру үшін пайдаланылуы мүмкін. Мысалы, басқа айнымалыларды (айнымалылар-куәліктер) бақылап, айнымалы жиынының жағдайы туралы жаңа білім алу үшін желіні пайдалануға болады. Бұл айнымалыларды айнымалы-куәліктер бойынша апостериорлік бөлуді есептеу процесі ықтимал қорытынды деп аталады. Бұл нәтиже бізге қосымшалар үшін әмбебап баға береді, онда айнымалы жиынының мәнін таңдау керек, ол шығын функциясын азайтады, мысалы,

қате шешім ықтималдығы. Байесов желісі сондай-ақ күрделі есептер үшін Байес теоремасын кеңейтуді автоматты түрде құруға арналған механизм болып саналады.

### **1.1.2 Машиналық оқытудың практикалық қолдану салалары**

Машиналық оқытудың мақсаты адам қызметінің түрлі салаларында күрделі кәсіби міндеттерді шешуді ішінара немесе толық автоматтандыру болып табылады. Машина оқыту қосымшаларының келесідей түрлері бар:

- а) сөйлеуді тану;
- б) қимылдарды тану;
- в) қолжазба енгізуін тану;
- г) бейнелерді тану;
- д) техникалық диагностика;
- е) медициналық диагностика;
- ж) уақытша қатарларды болжау;
- з) биоинформатика;
- и) алаяқтықты анықтау;
- к) спамды анықтау;
- л) құжаттарды санаттау;
- м) биржалық техникалық талдау;
- н) қаржылық қадағалау;
- п) кредиттік скоринг;
- р) клиенттердің күтуін болжау;
- с) хемоинформатика;
- т) ақпараттық іздестіруде ранжирлеуге оқыту.

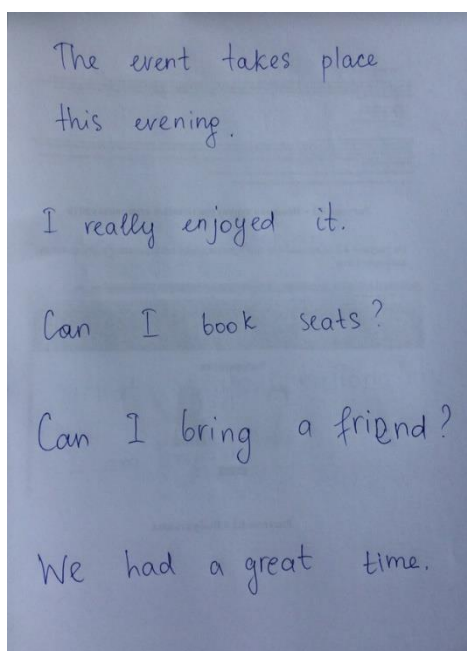
Машиналық оқытуды қолдану саласы үнемі кеңеюде. Жаппай ақпараттандыру ғылымда, өндірісте, бизнесте, көлікте, денсаулық сақтауда үлкен көлемнің жинақталуына алып келеді. Бұл ретте пайда болатын болжау, басқару және шешімдер қабылдау міндеттері жиі прецеденттер бойынша оқуға түседі. Бұрын мұндай деректер болмаған жағдайда, бұл міндеттер мүлдем қойылмаған немесе мүлдем басқа әдістермен шешілген болатын.

### **1.1.3 Оқыту үшін қолданылатын деректер**

Желіні оқыту үшін деректерді таңдау және оларды өңдеу тапсырманы шешудің ең маңызды бөліктерінің бірі болып табылады. Сонымен қатар, кіріс және шығыс деректерді ұсыну үлкен рөл атқарады. Жазбаны танып білуге арналған қамтама жасау кезінде қағаз бетіне бірнеше сөздер немесе кішігірім мәтін жазылған деректер жиналады. 1.1-суретте мысал көрсетілген.



Берілген осы деректер бойынша машиналық оқыту жүргізіледі. Оқытылуға арналған сөздер мен мәтіндер ағылшын әліпбиімен жазылды.



1.1-сурет – Оқыту үшін қолданылған жазба

## 1.2 Жазбаны танып білуде жасанды нейрондық желіні қолдану

Жасанды нейрондық желі — математикалық модель, сондай-ақ биологиялық нейрондық желілерді - тірі ағзаның жүйке жасушаларының желілерін ұйымдастыру және қызмет ету принципі бойынша құрылған оның бағдарламалық немесе аппараттық іске асырылуы болып табылады. Бұл ұғым мидағы процестерді зерттеу кезінде және осы процестерді модельдеу кезінде пайда болды. Мұндай бірінші әрекет У. Маккалок пен У. Питтс нейрондық желілер болды. Оқыту алгоритмдерін әзірлегеннен кейін алынған модельдерді практикалық мақсаттарда қолдана бастады: болжамдау міндеттерінде, бейнелерді тану үшін, басқару міндеттерінде және тағы басқа салаларда.

Жасанды нейрондық желі - біріктірілген және өзара әрекеттесетін қарапайым процессорлар (жасанды нейрондар) жүйесі. Мұндай процессорлар әдетте қарапайым (әсіресе жеке компьютерлерде қолданылатын процессорлармен салыстырғанда). Мұндай желінің әрбір процессоры тек қана мезгіл кезеңінде алатын сигналдармен және қандайда бір мезгіл кезінде басқа процессорларға жіберетін сигналдармен айналысады. Сонымен бірге, басқарылатын өзара іс-қимылмен айтарлықтай үлкен желіге қосылғанда, мұндай қарапайым процессорлар бірге өте күрделі тапсырмаларды орындауға қабілетті.

Нейрондық желілер осы сөздің әдеттегі мағынасында бағдарламаланбайды, нейрондық желілер оқытылады. Нейрондық желілердің дәстүрлі алгоритмдер алдындағы басты артықшылықтарының бірі - оқыту мүмкіндігі болып табылады. Техникалық оқыту нейрондар арасындағы байланыс коэффициенттерін табу болып табылады. Оқыту барысында нейрондық желі кіріс деректері мен шығу арасындағы күрделі тәуелділікті анықтауға, сондай-ақ қорытуды орындауға қабілетті. Бұл дегеніміз, табысты оқыту жағдайында желі оқыту үлгісінде болмаған деректердің, сондай-ақ толық емес және/немесе "ойластырылған", ішінара бұрмаланған деректердің негізінде дұрыс нәтижені қайтара алады.

Абстракцияның әр түрлі деңгейлерінде анықталатын және нейрондық жүйелердің әр түрлі аспектілерін модельдейтін үлгілердің көптеген типтері қолданылады. Олар жекелеген нейрондардың қысқа мерзімді мінез-құлқының үлгілерінен, нейрондық схемалардың динамикасы жекелеген нейрондардың арасындағы өзара әрекеттестерден пайда болатындай модельдерден және, ақырында, мінез-құлық толық жүйе ұсынатын абстрактілі нейрондық модульдерден пайда болуы мүмкін модельдерге дейін түрленеді. Оларға нейрондық жүйелердің ұзақ мерзімді және қысқа мерзімді икемділік модельдері және олардың оқуға және жеке нейроннан жүйелі деңгейге қатысты жады жатады.

Бірінші және екінші санаттағы әдістерге қарағанда, нейрожелілік әдістер бейнелерді тану міндеттерін шешуге өзге амал ұсынады. Нейрондық желілердің архитектурасы мен жұмыс істеуінің биологиялық бейнелері бар. Нейрондық желінің салмағы аналитикалық теңдеулерді шешу жолымен есептелмейді, ал оқыту кезінде әр түрлі оптимизациялық әдістермен құрылады. Оқыту мысалдары жиынтығында нейрондық желілер оқытылады. Нейрондық желілер оқыту процесінде негізгі белгілерді автоматты түрде шығарып алу, олардың маңыздылығын анықтау және олардың арасындағы өзара байланысты құру жүреді. Оқытылған нейрондық желілер оқыту процесінде алынған тәжірибені жақсы жалпылама қабілеттер есебінен белгісіз бейнелерге табысты қолдана алады.

Нейрондық желілер дегеніміз қарапайым биологиялық процестерді моделдейтін, әдетте адам миының процестерімен байланысты есептеуіш құрылымдар. Бейімделетін және оқитын адамдар оң және теріс әсерлерді талдау арқылы оқуға қабілетті параллельді жүйелер болып табылады. Бұл желілердегі қарапайым түрлендіргіш жасанды нейрон немесе биологиялық прототиппен ұқсас деп аталатын жай нейрон болып табылады.

Математикалық тұрғыдан жасанды нейрон-бұл барлық кіріс сигналдарының сумматоры, алынған өлшенген сомаға кейбір қарапайым, жалпы жағдайда, сызықтық емес функцияны қолданатын, барлық анықтау аймағында үздіксіз. Әдетте бұл функция біркелкі өседі. Алынған нәтиже жалғыз шығуға жіберіледі.

Жасанды нейрондар жасанды нейрондық желіні құру арқылы өзара белгілі бір жолмен біріктіріледі. Әрбір нейрон мидың жүйке жасушаларына ұқсас өзінің ағымдағы жағдайымен сипатталады, олар қозғалуы немесе

тежеуі мүмкін. Ол синапстардың тобы – басқа нейрондардың шығуымен біріктірілген бір бағыттағы кіріс байланыстары бар, сондай – ақ сигнал келесі нейрондардың синапстарына түсетін осы нейронның аксон-шығыс байланысы бар. Әдетте, объектілерді тану есептерінде нейрондық желілердің келесі архитектуралары қолданылады: көп қабатты персептрон; ықтималдықты нейрондық желілер; Кохонен желілері; үю нейрондық желілер. Бұл ретте белгілі бір архитектураны таңдау танылатын объектінің ерекшелігіне байланысты.

### **1.2.1 Жасанды нейрондық желіні оқыту**

Оқыту барысында желі белгілі бір тәртіппен оқыту іріктемесін қарайды. Мұғалімдерсіз оқытын кейбір желілер (мысалы, Хопфилд желілері) іріктеуді тек бір рет қарайды. Басқалары (мысалы, Кохонен желілері), сондай-ақ мұғаліммен бірге оқытын желілер іріктеуді бірнеше рет қарайды, сонымен қатар таңдау бойынша бір толық өту оқу кезеңі деп аталады. Мұғаліммен оқу кезінде бастапқы мәліметтер жиынтығын екі бөлімге бөледі — оқу іріктемесі мен тест мәліметтері; бөлу принципі ерікті болуы мүмкін. Оқыту деректері оқыту үшін желілер беріледі, ал тексеру желі қателерін есептеу үшін пайдаланылады (тексеру деректері желіні оқыту үшін ешқашан қолданылмайды). Осылайша, егер тексеру деректерінде қате азайса, онда желі шын мәнінде жалпылауды орындайды. Егер оқыту мәліметтеріндегі қате азайып, ал тест мәліметтеріндегі қате көбейсе, онда желі қорытуды орындауды тоқтатты және оқыту деректерін жай ғана "есте сақтайды". Бұл құбылыс желіні қайта оқыту немесе оверфиттинг деп аталады. Мұндай жағдайларда оқыту әдетте тоқтатылады. Оқыту барысында басқа да проблемалар пайда болуы мүмкін, мысалы сал немесе желінің қате бетінің жергілікті минимумына түсуі. Қандай да бір проблеманың көрінуін алдын ала болжау мүмкін емес, сол сияқты оларды шешуге бір мағыналы ұсыныстар беру мүмкін емес.

Жоғарыда айтылғандардың барлығы нейрожелілік шешімдерді іздеудің итерациялық алгоритмдеріне ғана қатысты. Олар үшін ешнәрсе кепілдендіруге болмайды және нейрондық желілерді оқытуды толығымен автоматтандыруға болмайды. Алайда, оқытудың итерациялық алгоритмдерімен қатар, өте жоғары төзімділікке ие және оқыту процесін толығымен автоматтандыруға мүмкіндік беретін итерациялық емес алгоритмдер бар.

Бір қарағанда оқыту процесі табысты болған жағдайда да желі әрдайым құрушы қалаған нәрсеге үйренбейді. Нейрожеліні оқыту сапасын тестілеуді оның оқуына қатыспаған мысалдарда жүргізу қажет. Бұл ретте тестілік мысалдар Саны оқыту сапасы жоғары болған сайын соғұрлым көп болуы тиіс. Егер нейрондық желінің қателіктері бір миллиардқа жақын болса, онда

бұл ықтималдықты растау үшін миллиард тестілік мысалдар қажет. Жақсы оқытылған нейрондық желілерді тестілеу өте қиынға соғады.

Белгілерге оқыту немесе ұсыныстарды оқыту-бұл жүйеге бастапқы (шикі) деректердің белгілерін анықтау немесе жіктелуі үшін қажетті түсініктерді автоматты түрде анықтауға мүмкіндік беретін техникалардың жиынтығы. Бұл белгілердің қолмен құрастырылуын алмастырады және машинаға белгілерді зерттеуге және оларды арнайы есептерді шешу үшін пайдалануға мүмкіндік береді. Сигналдарды оқыту машиналық оқыту есептерінде, мысалы, жіктеу, жиі математикалық және есептеуішпен өңдеуге ыңғайлы кіріс талап етіледі. Алайда, бейнелер, бейнелер және датчиктердің деректері сияқты нақты деректер ерекше белгілерді алгоритмдік анықтамаға берілмейді. Альтернатива белгілі бір алгоритмдерге тіреусіз зерттеу жолымен осындай белгілерді немесе көріністерді анықтау болып табылады. Белгілерге оқыту мұғаліммен немесе онсыз болуы мүмкін.

Тірек векторлар әдісі-жіктеу және регрессиялық талдау есептері үшін қолданылатын оқытушымен ұқсас оқыту алгоритмдерінің жиынтығы. Желілік классификаторлар тобына жатады және Тихонов бойынша реттеудің арнайы жағдайы ретінде қарастырылуы мүмкін. Тірек векторлар әдісінің ерекше қасиеті жіктеудің эмпирикалық қатесінің үздіксіз азаюы және саңылаудың ұлғаюы болып табылады, сондықтан да әдіс ең жоғары саңылауы бар жіктеуіш әдісі ретінде белгілі. Әдістің негізгі идеясы-бастапқы векторларды жоғары өлшемдік кеңістікке көшіру және осы кеңістіктегі ең жоғары саңылауы бар бөлетін гиперплоскосты іздеу. Екі параллель гиперплоскость кластарды бөлетін гиперплоскость екі жағынан құрылады. Гиперплоскостпен бөлінетін гиперплоскосты екі параллель гиперплоскостқа дейін максималдайтын гиперплоскосты болады. Алгоритм осы параллель гиперплоскостардың арасындағы айырмашылық немесе қашықтық неғұрлым көп болса, классификатордың орташа қатесі соғұрлым аз болады деген болжаммен жұмыс істейді.

### **1.3 Ұқсас бағдарламаларға аналитикалық шолу жасау**

Төменгі тақырыпшаларда көрсетілген ұқсас бағдарламалардың менің дипломдық жобамнан басты ерекшелігі және артықшылықтарының бірі – адам қолымен жазылған жазбаны, яғни қолжазбаны танып білу болып табылады. Соңғы уақыттарда, әрине, машиналық оқыту және танып білу жүйесі саласы үлкен қарқынмен дамып келе жатыр. Соған байланысты қолжазбаны танып білетін бағдарламалар да пайда болуда. Қолжазбаны танып білу баспа әріптерімен жазылған тексттерді танып білуге қарағанда әлдеқайда күрделі болып табылады. Осыған байланысты танып білу пайызы да біршама аз болады. Менің ойымша қолжазбаны танып білуді 100 пайызға жеткізу өте қиын жұмыстардың бірі. Қолжазбаны, әрине, толықтай биометрикалық дерекке жатқыза алмаймыз. Бірақ, әрбір адамның

қолжазбасы бір бірінен ерекшеленеді. Осыған орай танып білу үшін қажетті деректер қоры да өте үлкен мөлшерде болуы керек.

### **1.3.1 ABBYY FineReader бағдарламасы**

ABBYY FineReader – бұл құжат суреттерін және PDF форматындағы файлдарды тез әрі дәл танып білетін және танып білген ақпаратты электронды тузету болатын форматқа ауыстыратын бағдарламалардың бірі болып табылады. Бағдарлама дәлдігі 98 пайызға шамалас. Бағдарламаның 11-ші версиясынан бастап файлдарды djvu форматында сақтауға болатын болды. Ал бағдарламаның 12-ші версиясы тексттерді 190 тілде танып біле алады. Сонымен қоса, соның ішіндегі 48 тілде қосымша орфографиялық қателерді тексеруге мүмкіндігі бар.

### **1.3.2 SimpleOCR бағдарламасы**

SimpleOCR - ағылшын және француз тілдеріндегі мәтінді тану үшін танымал тегін бағдарламалық қамтаманың бірі. Айта кету керек, егер құжаттарда бірнеше бағаналы белгілер, стандартты емес қаріптер, түсті суреттер немесе төмен суреттер болса бұл жағдайда тану бағдарламасына арналған төрт ақылы қосымшалардың бірін жүктеу қажет. Бағдарлама ағылшын және француз тілдеріндегі құжаттармен жұмыс істей алады, бірақ тек басқа тілдерде қолданылады әр түрлі жол үсті және жол асты белгілері бар таңбаларды тани алмайдымысалы (ö, ä, é және т. б.).

SimpleOCR мәтінді тану үшін қарапайым бағдарлама, бірақ өкінішке орай, бағдарлама орыс және қазақ тілдерінде жұмыс істеуге арналмаған. Кеңейтілген функционалды алу қажет болған жағдайда ақылы жұмыстарды сатып алу қажет болады.

### **1.3.3 FreeOCR бағдарламасы**

FreeOCR - сканерленген мәтінді тану үшін тегін бағдарлама. Ол тек сурет файлдарымен ғана емес, pdf файлдарымен де және сканермен тікелей жұмыс істейді. Сканерлеу үшін тек қосылған сканер қажет. Бұл утилитаны орнату үшін интернет желісіне қосылу қажет, өйткені іске қосылатын қосымша салмағы 150 Кб және қосымша базалар және сөздіктер интернет желісінен жұмыс істеу кезінде жүктеледі. Жалпы алғанда, бағдарлама қатты дискіде 11 Мб артық емес орын алады. Бағдарламаның негізгі артықшылығы оның толық автоматтандырылуы және қандай да бір жөнделуді жасауды

қажет етпеуі болып табылады. Сонымен қоса JPEG, GIF, TIFF BMP және PDF форматтары қолдайды (тек бірінші бет, жақын арада алғашқы 10 бетті қолдауға уәде етілген). Сондай-ақ, сағатына 10 сурет лимиті бар. Бұл жүйе Шығыс Еуропа тілдерінің көпшілігін тануға қабілетті, оның ішінде орыс және украин тілдерін.

### **1.3.4 Microsoft Office Document Imaging бағдарламасы**

Microsoft Office Document Imaging (MODI) - Microsoft Office пакетінің құрамына кіретін және Microsoft Office Document Scanning сканерленген құжаттарды тану және өңдеу үшін қызмет ететін компонент. Алғаш рет Microsoft Office XP-де пайда болды және келесі буманың нұсқасы. Office 2010 бастап қол жетімді емес. Office 2010 бағдарламасына Document Imaging орнату үшін SharePoint Designer 2007 бумасын шығарды.

MODI шағын TIFF форматындағы кескіндерін оқи және жаза алады. Ол сондай-ақ танылған мәтінді бастапқы TIFF файлына сақтай алады. Әдепкі бойынша OCR қозғалтқышы тану кезінде дұрыс бет бағдарын талап етеді. Егер object name әдісін шақырса.save (), ол түзетілген бетті бастапқы файлға жаза алады. Таңбаларды тану дәлдігі жоғары емес, сонымен қатар кіріс бейнелерінің сапасына жоғары талаптар қояды.

## 2 Құрылымдық-техникалық бөлім

### 2.1 Жазбаны танып білуде YOLO моделін қолдану

You only look once (YOLO) - бұл нақты уақытта объектілерді анықтаудың заманауи жүйесі. Алдыңғы анықтау жүйелерінде анықтауды орындау үшін классификаторларды немесе локализаторларды қайта бағдарлауды орындайтын еді. Олар модельді бірнеше жерде және масштабта қолданады. Суреттің жоғары бағалау аймақтары анықтау деп саналады.

YOLO моделінде басқа тәсілді пайдаланады. Толық бейнеге бірыңғай нейрондық желіні қолданады. Бұл желі кескінді аймаққа бөледі және әр аймақ үшін шектеу шеңбері мен ықтималдығын болжайды. Бұл шектеу шеңберлері болжанған ықтималдықтармен өлшенеді. YOLO моделінің артықшылықтарына келесітер жатады.

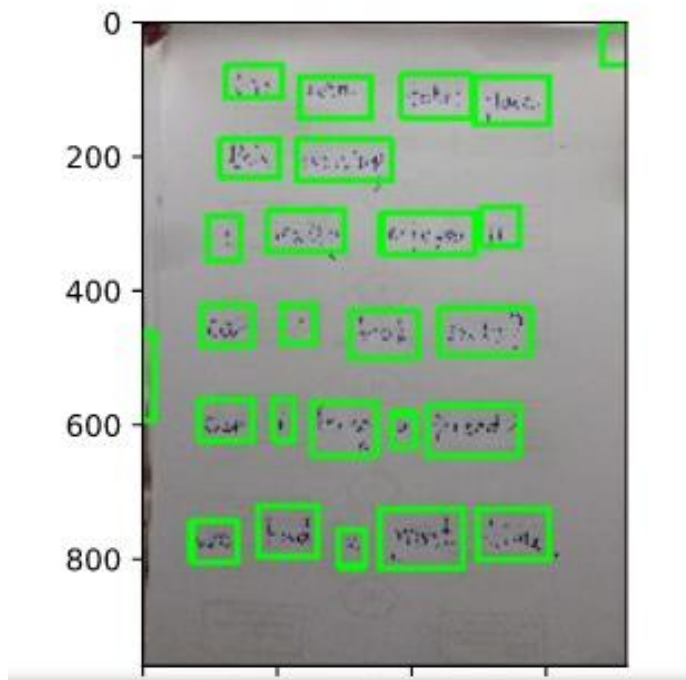
а) нақты уақытта өңдеу кезінде жылдам жұмыс жасайды;

б) болжамдар (объектілердің орналасқан жері және сыныптар) бір желіден орындалады;

в) YOLO жалпыланған және басқа салаларда табиғи бейнелерді қорытуда басқа әдістерден асып түседі;

г) YOLO топ ұяшығында бір нысанды анықтайды. Ол болжамдарды жасау кезінде кеңістіктік әртүрлілікті қамтамасыз етеді;

Бұл модель классификатор жағынан артықшылықтарының бірі - негізделген жүйелер. Ол тестілеу кезінде барлық бейнеге қарайды, сондықтан оның болжамдары бейнедегі жаһандық контекстпен хабарланады. Бұл 1000X R-CN және 100x fast R-CN қарағанда өте жылдам. YOLOv2 оқыту мен өнімділікті жақсарту үшін бірнеше трюктерді пайдаланады. Overheat және SSD сияқты біз толығымен бүктелген модельді пайдаланып, бірақ біз әлі де қатты негативтерде емес, тұтас кескіндерде жаттығады. Жылдам RCN сияқты, біз ені мен биіктігін тікелей болжау орнына шектеу қораптарында қатаң түзету жүргізіледі. Алайда, әлі де X және y координаттарын тікелей болжамайды. YOLO бір объект үшін табу дубликаттарын жасай алады. Оны түзету үшін, YOLO аз сенімділікпен дубликаттарды жою үшін ең көп басуды қолданады. Талқылау үшін біз түпнұсқа фотосуретімізді кесеміз. Әрбір топ ұяшығы тек бір нысанды болжайды. Мысалы, сары топ ұяшығы төменгі топ ұяшығының ішіне кіретін "адам" нысанын болжауға тырысады. Әрбір топ ұяшығы шекаралық өрістердің тіркелген санын болжайды. Бұл мысалда сары топ ұяшығы адамның қайда екенін табу үшін шекаралық өрістердің (көк өрістің) екі болжамын жасайды. 2.1-суретте нақты уақытта объектілерді анықтау көрсетілген.



2.1-сурет – Нақты уақытта объектілерді анықтау

YOLO негізгі тұжырымдамасы тензорды болжау үшін жасанды нейрондық желісін құру болып табылады (7, 7, 30). Ол нейрондық желісін әрбір орналасқан жердегі 1024 шығыс арналарымен  $7 \times 7$  дейін кеңістіктік өлшеуді азайту үшін пайдаланады. YOLO  $7 \times 7 \times 2$  шекаралық болжам жасау үшін екі толық байланысқан қабат арқылы сызықтық регрессияны орындайды (төмен орташа сурет). Соңғы болжам жасау үшін, біз сенімділіктің жоғары бағасын (0,25-тен көп), біздің соңғы болжамдар сияқты (дұрыс сурет) сақтаймыз. YOLO 24 орама қабаты бар, одан кейін 2 толық қосылған қабат (FC). Кейбір қабаттар  $1 \times 1$  азайту қабаттарын объектілер карталарының тереңдігін азайту үшін балама ретінде пайдаланады. Ораудың соңғы қабаты үшін ол нысаны бар тензорды шығарады (7, 7, 1024). Содан кейін тензор жабысады. Сызықтық регрессия нысаны ретінде 2 толық байланысқан қабатты пайдалана отырып, ол  $7 \times 7 \times 30$  параметрлерін шығарады, содан кейін (7, 7, 30), яғни 2 шекаралық өрісті болжау орнына айналады.

YOLO тор ұяшығына бірнеше шектейтін тіктөртбұрыштарды болжайды. Шынайы оң үшін жоғалтуды есептеу үшін, біз тек бір ғана объект үшін жауап беруді қалаймыз. Осы мақсат үшін біз ең жоғары IoU (бірлестік үстінен қиылысу) негізгі ақиқатпен таңдаймыз. Бұл стратегия шектейтін тіктөртбұрыштың болжамдары арасында мамандануға әкеледі. Әрбір болжам тараптардың белгілі бір өлшемдері мен арақатынасын болжау кезінде жақсы болады.

YOLO шығындарды есептеу үшін болжамдар мен негізгі шындық арасындағы квадраттар сомасының қателігін пайдаланады. Жоғалту функцияларына келесілер жатады:



- а) классификациялық шығындар;
- б) локализацияның жоғалуы (болжамды шекара өрісі мен жердің шынайылығы арасындағы қателер);
- в) сенімнің жоғалуы (қораптың мәні).

SSD YOLO үшін күшті бәсекелес болып табылады, ол бір сәтте нақты уақытта өңдеудің жоғары дәлдігін көрсетеді. Yolo аймағы негізінде детекторлармен салыстырғанда жоғары орын қателері бар, ал шолу (барлық объектілерді қаншалықты жақсы табу) төмен. Жылдамдату кезінде дәлдікті айтарлықтай арттыру мақсатында Yolo-ның екінші нұсқасы YOLOv2 жасалды.

YOLO 160 дәуірде ImageNet 1000 сыныптарының жіктеме деректерін жинаудың көмегімен оқиды: бастапқы оқу жылдамдығы 0,1 стохастикалық градиентті төмендеу, 4 қуаты бар полиномиальді құлдырау, 0,0005 салмағы төмендеуі және 0,9 импульс. YOLO бастапқы дайындық  $224 \times 224$  суреттерді пайдаланады, содан кейін оны  $448 \times 448$  суретпен 10 дәуірінде 10-3 оқу жылдамдығымен қайта құрастырады. Оқытудан кейін классификатор топ-1 дәлдікке 76,5% және топ-5 дәлдікке 93,3% жетеді.

Осыдан кейін толық қосылған қабаттар және соңғы қабаттар детектор үшін алынады. YOLO әрбір 1024 сүзгілермен  $3 \times 3$  үш ұю қабатын, содан кейін  $1 \times 1$  Соңғы ұю қабатын 125 Шығыс арналарымен қосады. YOLO сондай-ақ passthrough қабатын қосады. YOLO 10 60 және 90 дәуірде оны бөлу арқылы бастапқы жылдамдығы 10-3 арқылы 160 дәуірде желіні жаттықтырады. YOLO салмағы 0.0005 және 0.9 импульсін пайдаланады.

Объектілерді табу үшін деректер жинағы классификацияға қарағанда класстардың санаттарынан әлдеқайда аз. YOLO таба алатын кластарды кеңейту үшін, YOLO оқыту кезінде анықтау және жіктеу деректер жиынтықтарынан суреттерді араластыру әдісін ұсынады. Ол толассыз желіні объектілерді табу үлгілеріне үйретеді, ал кері сыныптау жолын оқыту үшін классификация үлгілерінен классификациялау шығындарын таратады.

YOLO классификациялау суретін көргенде, ол классификацияны оқыту үшін классификация шығындарын қайтарады. YOLO осы класс үшін ең үлкен ықтималдықты болжайтын шектеуші шеңберді табады және сыныптаманың жоғалуын, сондай-ақ ата-аналарынан айырылуды есептейді. (Егер нысан биплан ретінде белгіленсе, ол ұшақ, ауа, көлік құралы ретінде белгіленеді... ) бұл модель олар үшін ортақ функцияларды алуға итермелейді. Осылайша, тіпті біз объектілерді табу үшін белгілі бір объектілер класын ешқашан оқытпадық,біз әлі де осындай болжамдарды жасай аламыз. Объект табылған кезде біз PR (физикалық нысан) қораптың объектінің бар-жоғын өлшейтін сенім балына тең етіп орнатамыз. YOLO ол кейбір шегіне жеткенше әрбір раскольде ең жоғары сенім жолын қабылдай отырып, ағаш кесіп, және YOLO осы объектілер класын болжайды.

YOLO9000 9000-нан астам кластағы нысандарды табу үшін 9418 WordTree түйіні бар иерархиялық жіктемені қолданады. Ол COCO үлгілерін және ImageNet үздік 9000 сыныптарды біріктіреді. Yolo үлгілері әрбір Коко үшін Деректер базасына төрт шексіз қол жеткізу. Ол COCO-да табылған

деректерді пайдалана отырып, нысандарды тауып, ImageNet үлгілерімен осы нысандарды жіктеуді үйренеді.

YOLO бағалау кезінде ол қалай жіктеуге болатынын білетін, бірақ оларды тікелей табуға үйретілмеген, яғни COCO жоқ санаттар бойынша суреттерді тестілейді. YOLO9000 ImageNet объектілерін анықтау деректер жиынтығынан өз нәтижесін бағалайды, ол 200 санаты бар. Ол COCO бар 44 санатты бөледі. Осылайша, деректер жинағы оларды қалай табу туралы тікелей оқытылған ешқашан 156 санаттарды қамтиды. YOLO байланысқан Нысандар түрлері үшін ұқсас функцияларды алады. Демек, біз бұл 156 категорияны объектілердің мәні бойынша анықтай аламыз.

YOLO9000 осы 156 санат бойынша 16.0 картамен жалпы 19.7 картасын алады. YOLO9000 COCO табылған жоқ жануарлардың жаңа түрлерімен жақсы жұмыс істейді, себебі олардың нысандары оңай олардың ата-аналық сыныптарынан қорытылуы мүмкін. Дегенмен, COCO киімнің кез келген түріне шектеу жапсырмасы жоқ, Сондықтан тест "күннен қорғайтын көзілдірік" сияқты санаттармен күреседі.

YOLOv3 туралы айтып кететін болсақ, ол логистикалық регрессия арқылы әрбір шектеуші тіктөртбұрыш үшін объектілік бағалауды болжайды. YOLOv3 шығындар функциясын есептеу тәсілін өзгертеді. Егер шектеуші тіктөртбұрыш (Зәкір) жер ақиқатының объектісін басқаларынан артық жапса, объективтіліктің тиісті бағасы 1 болуы тиіс. Белгіленген шектен асатын (әдепкі 0,5) жабыны бар басқа порттар үшін олар шығындарды талап етпейді. Жер ақиқатының әрбір объектісі тек бір шекаралық өріспен байланысты. Егер шектеуші рамка тағайындалмаса, ол жіктелуі мен орналасуын жоғалтпайды, тек объективтілікке деген сенімнің жоғалуы. Біз шығындарды есептеу үшін  $t_x$  және  $t_y$  (орнына  $b_x$  және  $b_y$ ) пайдаланамыз.

Coco AP YOLOv3 метрикасы SSD бар бір деңгейде, бірақ 3 есе жылдам. Бірақ AP YOLOv3 әлі де торлы торға. Атап айтқанда,  $AP@IoU=0.75$  тамшы тор көз торымен салыстырғанда айтарлықтай, бұл YOLOv3 жоғары оқшаулау қатесі бар деп болжайды. YOLOv3 сондай-ақ шағын нысандарды анықтауда айтарлықтай жақсаруды көрсетеді.

## 2.2 Жазбаны танып білу бағдарламасын пайдалану алгоритмі

Жазбаны танып білуге арналған бағдарламам ubuntu операциялық жүйесінде жұмыс жасайды. Бағдарламамен жұмысты бастау үшін ең алдымен терминал қосымшасын ашамыз және келесі суретте көрсетілгендей сөздерді жазамыз. 2.2-суреттегі бірінші команда бізге керекті ортаға өтуді орындайды.

```
gulsezim@gulsezim: ~
(base) gulsezim@gulsezim:~$ conda activate text_d
(text_d) gulsezim@gulsezim:~$ jupyter notebook
```

2.2-сурет – Бағдарламамен жұмыс жасау үшін терминалды қолдану

Терминалдағы екінші команда орындалған соң интерактивті есептеулер жүргізетін орта ашылады. Терезедегі бірнеше папкалар ішінен жұмыс устелі папкасын тандаймыз. Кейін осы папка ішінен text\_detection папкасын таңдап, main файлына кіреміз. Бізге танып білуге керекті алдын ала суретке тусірілген суреттің жолын код бөлігінде көрсетеміз. Бұл келесі суретте көрсетілген.

```
img = cv2.imread('bUeY4bfXlJg.jpg', cv2.IMREAD_COLOR)
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

words = word_predict.return_predict(img)

print(len(words))
...
n = len(words)
for i in range(n):
    for j in range(0, n-i-1):
        if words[j]['topleft']['x'] > words[j+1]['topleft']['x']:
            words[j], words[j+1] = words[j+1], words[j]
    ...
print(words)
```

2.3-сурет – Сурет жолын көрсету

Кейін код бөліктерін орындалуға жібереміз. Танып білу орындалғаннан кейін бағдарлама жауабы txt форматындағы файлға жазылады. Бұл келесі суретте көрсетілген.

```
Result.txt (~/Рабочий стол/text_detection) - gedit
Открыть ▾ [икона] Сохранить
the d event takes place this evening x really enjoyed it canc book seats a
can id bring dad riend chad htime wed dad greaty
```

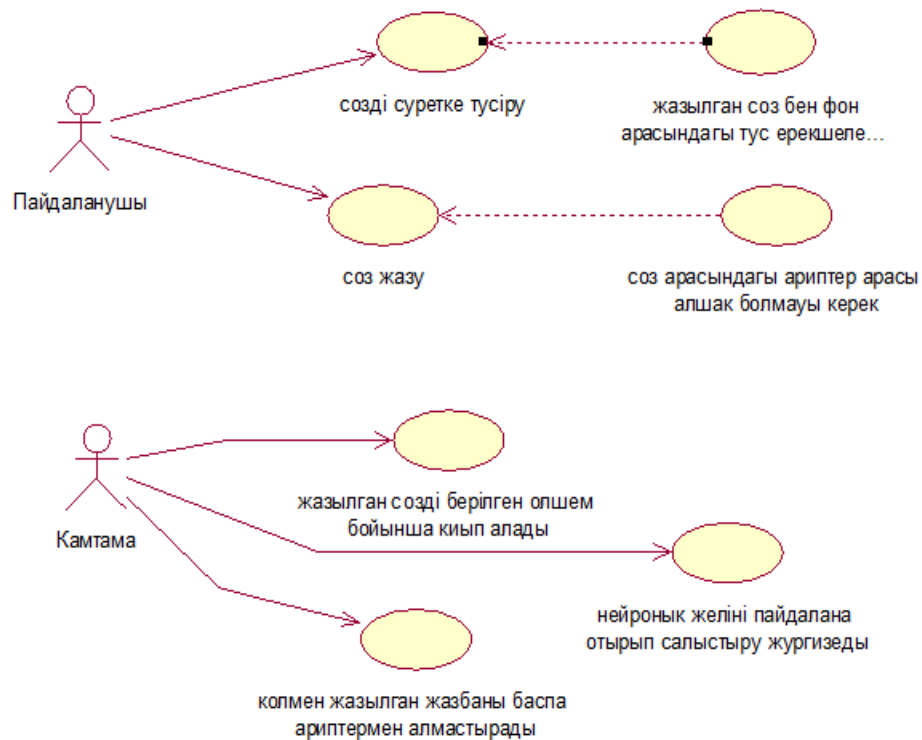
2.4-сурет – Жауаптың файлда сақталуы

## 3 Функционалдык талаптарды ұйымдастыру

### 3.1 Функционалды диаграммалар

Жобаға байланысты диаграммалар қуру кезінде UML, яғни модельдеудің унифицирленген тілін пайдалану және Rational Rose құралын пайдалану арқылы жасадым. UML тілі қандай да бір бағдарламалардың графикалық сипаттамасын көрсету шін қолданылды. UML тілі бағдарламау тіліне жатпайды, бірақ UML моделі негізінде кодты генерациялауға болады. Rational Rose – CASE ақпараттық жүйелер мен кәсіпорындарды басқару үшін бағдарламалық қамтамасыз етуді жобалау және әзірлеу құралы. Басқа CASE құралдары сияқты (ARIS, BPwin, ERwin) оны бизнес процестерді талдау және модельдеу үшін қолдануға болады. Бұл өнімнің бірінші нұсқасын Rational Software компаниясы шығарды. Rational Rose басқа құралдардан принципті айырмашылығы объектілі-бағытталған тәсілде. Осы құралдың көмегімен жасалған графикалық модельдер объектілі-бағытталған принциптерге және UML тіліне негізделген. Rational Rose моделдеу құралдары әзірлеушілерге иерархияның әртүрлі деңгейлері арасындағы барлық өзара байланыс пен басқару әсерлерін сақтай отырып, кәсіпорын процестерінің тұтас архитектурасын жасауға мүмкіндік береді. UML тілінің көмегімен көптеген диаграммалар салуға болады, оларға: класстар диаграммасы, компоненттер диаграммасы, композиция диаграммасы, объекттер диаграммасы, пакеттер диаграммасы, ашып қарау диаграммасы, прецеденттер диаграммасы, кй диаграммасы, тізбек диаграммалары жатады.

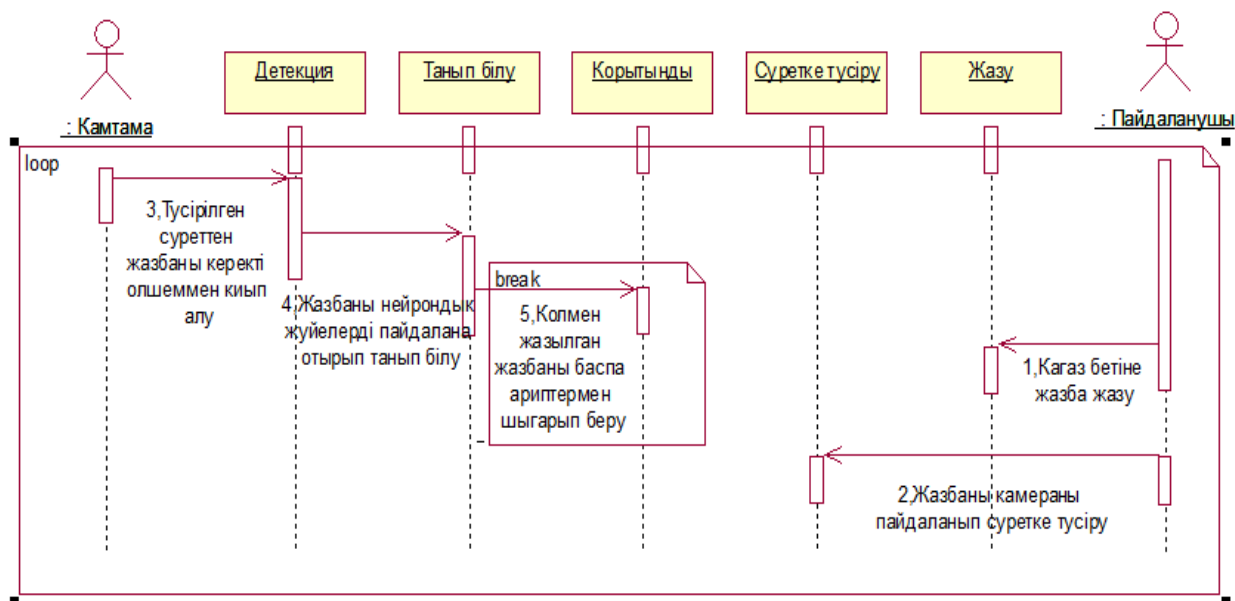
Прецеденттер диаграммасы тұлғалар мен прецеденттер арасындағы қатынастарды анықтайды және жүйені концептуалды деңгейде сипаттау кезінде қолданылады. Прецеденттер диаграммасын әдетте қолдану варианттар диаграммасы деп те атайды. Прецеденттер диаграммасы тұлғалардан, прецеденттерден және олардың арасындағы байланыстардан трады. Тұлғалар мен прецеденттер бірнеше трлі байланыстармен байланысуы мүмкін. Прецеденттер диаграммасы 3.1-суретте көрсетілген.



3.1-сурет – Прецеденттер диаграммасы

Менің бағдарламам бойынша жасалған прецеденттер диаграммасында «пайдаланушы» және «қамтама» атты екі тлға берілді. «Пайдаланушы» тлғасы «сөзді суретке тусіру» және «сөз жазу» атты прецеденттермен байланысқан. Ал «сөзді суретке тусіру» прецедентіне қосымша ретінде «жазылған сөз бен фон арасындағы тусті ерекшелеу» прецеденті байланысқан. «Сөз жазу» прецеденті қосымша шартты білдіретін «сөз арасындағы әріптер арасы алшақ болмауы керек» прецедентімен байланысқан. Ал «қамтама» прецеденті «жазылған сөзді берілген өлшем бойынша қиып алады», «нейрондық желіні пайдалана отырып салыстыру жүргізеді» және «қолмен жазылған жазбаны баспа әріптерімен алмастырады» атты үш прецедентпен байланысқан.

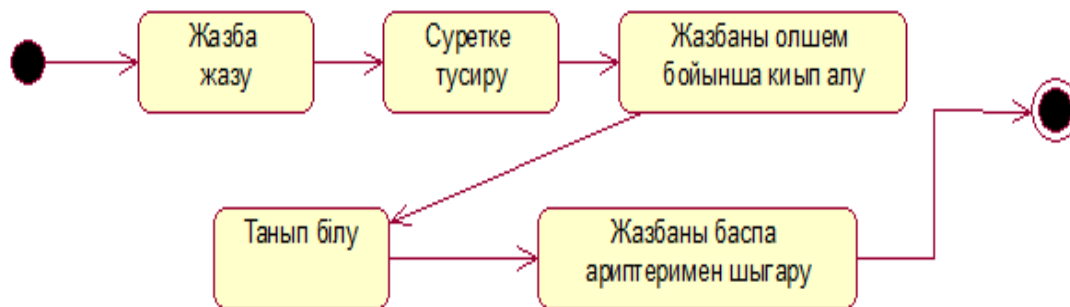
Тізбек диаграммасы - басты акцент хабарламаның уақытқа бірыңғайланып жасалған әрекеттесу диаграммасы болып табылады. Яғни бағдарламада болатын оқиғаларың орындалу реті көрсетіледі. Тізбек диаграммасы тулғалардан, төртбурыштармен көрсетілген объектілерден, объекті астына жалғасқан сызық ретінде көрсетілген «өмір сызықтарынан» және байланыс сызықтарынан тұрады. Тізбек диаграммасы 3.2-суретте көрсетілген.



3.2-сурет – Тізбек диаграммасы

Менің тақырыбым бойынша жасалған тізбек диаграммасы «пайдаланушы» және «қамтама» тулғаларынан және «детекция», «танып білу», «қорытынды», «суретке тусіру» және «жазу» атты объектілерден тұрады. Тулғалар мен объектілерді байланыстыратын нөмірленген, яғни белгілі ретпен жүретін, байланыстар көрсетілген. Бірінші байланыс «қағаз бетіне жазба жазу», екінші байланыс «жазбаны камераны пайдаланып суретке тусіру», үшінші «тусірілген суреттен жазбаны керекті өлшеммен кыып алу», төртінші «жазбаны нейрондық жүйелерді пайдалана отырып танып білу», бесінші «қолмен жазылған жазбаны баспа әріптерімен шығарып беру» байланыстары көрсетілген.

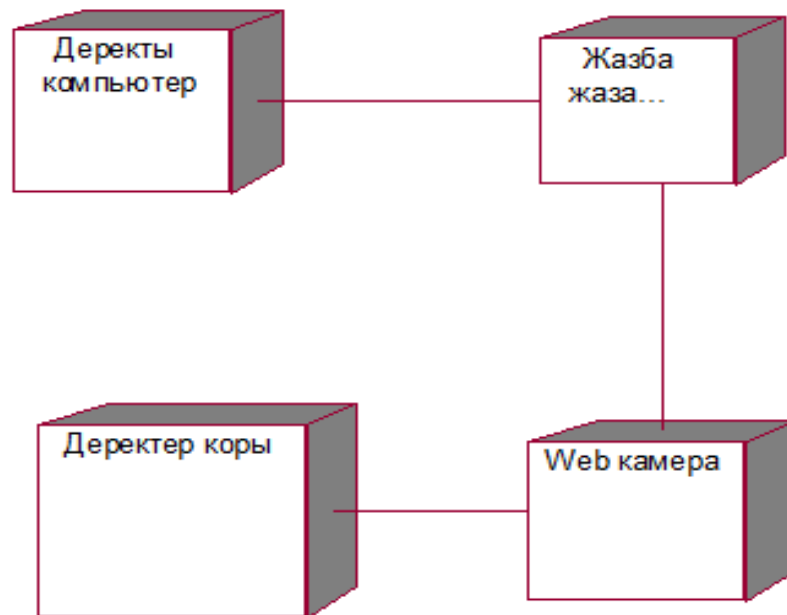
Күй диаграммасында объектілер тәртіппен және күймен сипатталады. Күй диаграммалары бизнес-процестерді, технологиялық процестерді, тізбекті және параллельді есептеулерді моделдеу кезінде қолданылады. Күй диаграммасы бағдаршалармен байланысқан бірнеше фигуралар байланысынан тұрады. Негізгі фигураларға тоқталатын болсақ, жан-жағы иілген төртбрыштар, бул фигура әрекеттерді білдіреді. Ал ромбы – шешімді, қалың сызықтар таралған әрекеттердің басын немесе соңын білдіреді. Қара дөңгелек процестің басын, ал қоршалған қара дөңгелек процестің соңын білдіреді. Бағыттамалар процестің басынан аяғына дейін жүреді және басқару ағындарын немесе объектілер ағындарын көрсетеді. Күй диаграммасы 3.3-суретте көрсетілген.



3.3-сурет – Күй диаграммасы.

Менің күй диаграммамда бес әрекет көрсетілген. Олар: жазба жазу, суретке тусіру, жазбаны өлшем бойынша қиып алу, танып білу, жазбаны баспа әріптермен шығару.

Ашып қарау диаграммасы. UML-да ашып қарау диаграммасы артефактілерді тораптарда физикалық ашып қарауды модельдейді. Мысалы, веб-сайтты сипаттау үшін ашып қарау диаграммасы қандай аппараттық компоненттердің бар екенін (мысалы, веб-сервер, деректер қоры сервері, қосымша сервері), әрбір торапта қандай бағдарламалық компоненттер жұмыс істейтінін (мысалы, веб-қосымша, деректер қоры) және осы кешеннің әр түрлі бөліктері бір-бірімен қалай қосылатынын (мысалы, JDBC, REST, RMI) көрсетуі тиіс. Түйіндер тікбұрышты параллелепипедтер ретінде, оларда орналасқан, тікбұрыштар түрінде бейнеленген артефактілер ретінде ұсынылады. Түйіндер тікбұрышты параллелепипедтер ретінде ұсынылатын төменгі тораптарға ие болуы мүмкін. Деректер қорының серверлері кластері сияқты көптеген физикалық тораптарды концептуалды түрде ұсына алады. Ашып қарау диаграммасы 3.4-суретте көрсетілген.

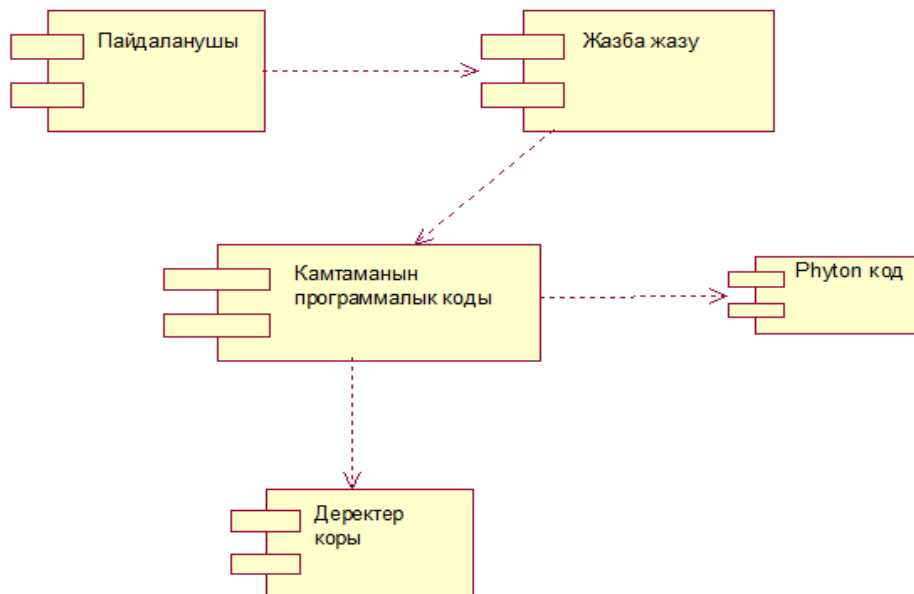


3.4-сурет – Ашып қарау диаграммасы.

Менің тақырыбым бойынша жасалынған ашып қарау диаграммасы төрт бір-бірімен байланысқан тораптан тұрады. Оларға: деректі компьютер, пайдаланушы интерфейсі, деректер қоры және суретке түсіру құралы жатады.

Компоненттер диаграммасы - UML моделдеу тілінің элементі, статикалық құрылымдық диаграмма, ол бағдарламалық жүйенің құрылымдық компоненттерге және компоненттер арасындағы байланыстарға (тәуелділікке) бөлінуін көрсетеді. Физикалық компоненттер ретінде файлдар, кітапханалар, модульдер, орындалатын файлдар, пакеттер және т. б. болуы мүмкін. Компоненттер диаграммасының көмегімен инкапсулирленген сыныптар олардың интерфейстік қабықшаларымен, порттармен және ішкі құрылымдармен бірге ұсынылады (олар да компоненттер мен коннекторлардан тұруы мүмкін). Компоненттер бір компоненттің қажетті интерфейсі басқа компоненттің қолда бар интерфейсімен қосылған кезде тәуелділіктер арқылы байланысады. Осылайша, клиент-екі компонент арасындағы дереккөз қарым-қатынасы суреттеледі. Тәуелділік бір компонент басқа компонентке қажетті сервисті ұсынады. Тәуелділік клиент интерфейсінен немесе портынан импортталатын интерфейске деген көрсеткімен бейнеленеді. Компоненттер диаграммасы компоненттердің ішкі құрылымын көрсету үшін пайдаланылса, құрамдас компоненттің қажетті интерфейстері ішкі компоненттердің тиісті интерфейстеріне жіберілуі мүмкін. Делегация компоненттің сыртқы келісім-шартының осы мінез-құлықты ішкі компоненттермен іске асырумен байланысы көрсетіледі. Компоненттер диаграммасы 3.5-суретте көрсетілген.





3.5-сурет – Компоненттер диаграммасы

Менің тақырыбым бойынша жасалынған компоненттер диаграммасы бір-бірімен байланысқан бес компоненттен тұрады. Оларға: пайдаланушы, жазба жазу, қамтаманың программалық коды, python код және деректер қоры жатады.

## ҚОРЫТЫНДЫ

Қорытындылай келгенде, жазбаны танып білу мақсатында жазылған бағдарлама, менің ойымша, қазіргі таңда сұранысқа ие және қажетті бағдарламалардың бірі. Дипломдық жобаға жазбаны танып білуді, яғни машиналық оқытуды таңдап алған мақсаттарымның бірі – осы салаға байланысты білімдерімді жетілдіріп, зерттеу жүргізі болатын. Жобаны жасау кезінде жасанды нейрондық желілер, оларды оқыту жұмыстары, машиналық оқыту және оның әдістері мен алгоритмдері сияқты көптеген ақпараттармен таныстым. Дипломдық жобамның тақырыбы қазіргі уақытта өзекті және сұранысқа ие салаларды қамтиды.

Жазбаны танып білу қазіргі уақытқа дейінгі зерттеліп келе жатырған мәселелердің бірі болып табылады. Дипломдық жоба нәтижесінде жасалған бағдарлама күнделікті өмірде қолданысқа керекті бағдарламалардың бірі. Қолмен жазылған құжаттармен жұмыс жасайтын мекемелерде, сақтандыру мекемелерінде бағдарлама қолданыс табады.

## ПАЙДАЛАНЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Потапов А.С. Распознавание образов и машинное восприятие. - СПб.: Политехника, 2007. - 548 с
- 2 Горбань А., Россиев Д. Нейронные сети на персональном компьютере. //Новосибирск, Наука, 1996. – С 114 – 119.
- 3 Вороновский Г. К., Махотило К. В., Петрашев С. Н., Сергеев С. А. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. — Харьков: Основа, 1997. — 112 с.
- 4 Ясницкий Л. Н. Введение в искусственный интеллект. — М.: Издат. центр «Академия», 2005. — 176 с.
- 5 Терехов В. А., Ефимов Д. В., Тюкин И. Ю. Нейросетевые системы управления. — М.: Высшая школа, 2002. — 184 с.
- 6 Mitchell T. Machine Learning. — McGraw-Hill Science/Engineering/Math, 1997.
- 7 Журавлев Ю. И., Рязанов В. В., Сенько О. В. «Распознавание». Математические методы. Программная система. Практические применения. — М.: Фазис, 2006.
- 8 Айвазян С. А., Бухштабер В. М., Енюков И. С., Мешалкин Л. Д. Прикладная статистика: классификация и снижение размерности. — М.: Финансы и статистика, 1989.
- 9 Сигеру Омату, Марзуки Халид, Рубия Юсоф. Нейроуправление и его приложения = Neuro-Control and its Applications. 2-е изд. — М.: ИПРЖР, 2000. — 272 с.
- 10 Савельев А. В. На пути к общей теории нейросетей. К вопросу о сложности // Нейрокомпьютеры: разработка, применение. — 2006.
- 11 Николенко С., Кадурын А., Архангельская Е. Глубокое обучение. — СПб.: Питер, 2018. — 480 с.
- 12 Миркес Е. М. Нейрокомпьютер. Проект стандарта. — Новосибирск: Наука, 1999. — 337 с.
- 13 Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика. — М.: Горячая линия - Телеком, 2001. — 382 с.
- 14 Каллан Р. Основные концепции нейронных сетей = The Essence of Neural Networks First Edition. — М.: Вильямс, 2001. — 288 с.
- 15 Еремин Д. М., Гарцеев И. Б. Искусственные нейронные сети в интеллектуальных системах управления. — М.: МИРЭА, 2004. — 75 с.

## **А қосымшасы (міндетті)**

### **Техникалық тапсырма**

#### **А.1 Жазба бойынша танып білу жүйесіне арналған қамтаманың мақсаты мен міндеттері.**

Қазіргі уақытта машиналық оқытудың және жасанды нейрондық жүйелердің пайда болуы және дамуы бізге қолжазба бойынша танып білу жүйесіне арналған қамтаманы жасауға мүмкіндік береді. Жазбаны танып білу жүйесін дипломдық жоба ретінде таңдап алған мақсатым – қолмен жазылған жазбаны о танып білуді және жалпы танып білу жүйесін зерттеу болды. Зерттеу жұмысының объектісі ретінде суреттегі жазбаны танып білу және жазаны детекциялау процесі жатады. Жазба бойынша танып білу жүйесіне арналған қамтаманың мақсаты – қағаз бетіне жазылған қолжазбаны қамтама арқылы танып біліп, оны баспа әріптеріне ауыстыру болып табылады.

#### **А.1.2 Қолдану саласы**

Дипломдық жоба нәтижесінде жасалған бағдарлама күнделікті өмірде қолданысқа керекті бағдарламалардың бірі. Қолмен жазылған құжаттармен жұмыс жасайтын мекемелерде, сақтандыру мекемелерінде бағдарлама қолданыс табады. Сонымен қоса жазбаны танып білуге арналған қамтама кәсіпорындардың құжат айналымы жүйесінде практикалық қолдануға, халықтан сауалнама алу жұмысы кезінде, паспорттық-визалық қызмет және кіріс деректері ретінде қолмен жазылған мәтін бейнесі болып табылатын басқа да жүйелерде қолдануға арналған.

#### **А.1.3 Анықтамалар, терминдер және қысқартулар**

Дипломдық жоба мақсатында жазылған жазба бойынша танып білу жүйесіне арналған қамтаманы жасау процесі кезінде бірнеше анықтамалар мен терминдер қолданылды. Оларға:

а) машиналық оқыту - бұл нақты міндеттерді тиімді орындау үшін компьютерлік жүйелермен қолданылатын алгоритмдер мен статистикалық үлгілерді нақты нұсқауларды пайдаланбай, оның орнына шаблондар мен ақыл-ойға сүйеніп ғылыми зерттеу;

## **А қосымшасының жалғасы**

б) жасанды нейрондық желі — математикалық модель, сондай-ақ биологиялық нейрондық желілерді - тірі ағзаның жүйке жасушаларының желілерін ұйымдастыру және қызмет ету принципі бойынша құрылған оның бағдарламалық немесе аппараттық іске асырылуы болып табылады;

в) перцептрон, машинада оқытуда, екілік классификаторларды бақыланатын оқытудың алгоритмі болып табылады. Бинарлық классификатор - бұл сандардың векторын ұсынатын кіріс немесе жоқ болса, белгілі бір сыныпқа жататындығын шеше алатын функция. Бұл желілік классификатордың түрі, яғни оның болжамын сызықша болжамды функцияға негізделген жіктеудің алгоритмі, салмақ жиынтығын функция векторымен біріктіреді.

### **А.2 Программалық интерфейстер**

Жазба бойынша танып білу жүйесіне арналған қамтаманың интерфейсі әлі де өзгертулер мен жетілдірулерді қажет етеді. Қазіргі уақытта пайдаланушы қамтамамен жұмыс жасау үшін терминал қосымшасын пайдаланады. Қамтаманың код бөлігіне танып білуге керекті жазба суретінің атын көрсетеді. Қамтама қорытындысы txt файл ішіне жазылады.

**Б қосымшасы**  
(міндетті)

Бағдарламалық код үлгісі

```
import cv2
from darkflow.net.build import TFNet
import matplotlib.pyplot as plt
%config InlineBackend.figure_format = 'svg'
options_word = {
    'model': 'word/yolo-config.cfg',
    'load': 'word/yolo-words_3400.weights',
    'labels': 'word/labels.txt',
    'threshold': 0.3
}

word_predict = TFNet(options_word)
options_char = {
    'model': 'char/yolo-config.cfg',
    'load': 'char/yolo-chars_final.weights',
    'labels': 'char/labels.txt',
    'threshold': 0.3
}

char_predict = TFNet(options_char)
img = cv2.imread('v5E5YuUs4FI.jpg', cv2.IMREAD_COLOR)
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

words = word_predict.return_predict(img)

print(len(words))
'''
n = len(words)
for i in range(n):
    for j in range(0, n-i-1):
        if words[j]['topleft']['x'] > words[j+1]['topleft']['x']:
            words[j], words[j+1] = words[j+1], words[j]
'''

print(words)
def text_d(image):

    chars = char_predict.return_predict(image)
```

## Б қосымшасының жалғасы

```
#print(chars)
print()
n = len(chars)
for i in range(n):
    for j in range(0, n-i-1):
        if chars[j]['topleft']['x'] > chars[j+1]['topleft']['x']:
            chars[j], chars[j+1] = chars[j+1], chars[j]
print(chars)
return(chars)
sum1=0
result = ""
for i in range(len(words)):
    tl = (words[i]['topleft']['x'], words[i]['topleft']['y'])
    br = (words[i]['bottomright']['x'], words[i]['bottomright']['y'])
    (x1, y1) = (words[i]['topleft']['x'], words[i]['topleft']['y'])
    (x2, y2) = (words[i]['bottomright']['x'], words[i]['bottomright']['y'])
    img = cv2.rectangle(img, tl, br, (0, 255, 0), 7)

    crop = img[y1:y2,x1:x2]
    crop = cv2.cvtColor(crop,cv2.COLOR_BGR2RGB)
    cv2.imwrite('words_data/crop-'+str(i)+'.jpg', crop)
    texts = text_d(crop)
    str1=""
    for i in range(len(texts)):
        word = texts[i]['label']
        str1 = str1 + word
    print('Resultat: '+str1)
    result = result + str1 + ' '
    cv2.imwrite('resu.jpg', img)

with open("Result.txt", "w") as text_file:
    text_file.write(result)
print('FILNAL Resultat: '+result)
plt.imshow(img)
plt.show()
yolo-config.cfg
[net]
# Testing
batch=1
subdivisions=1
# Training
```

## Б қосымшасының жалғасы

```
#batch=32
#subdivisions=8
height=128
width=256
channels=3
momentum=0.9
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1
```

```
learning_rate=0.001
burn_in=1000
max_batches = 80200
policy=steps
steps=40000,60000
scales=.1,.1
```

```
[convolutional]
batch_normalize=1
filters=32
size=3
stride=1
pad=1
activation=leaky
```

```
[maxpool]
size=2
stride=2
```

```
[convolutional]
batch_normalize=1
filters=64
size=3
stride=1
pad=1
activation=leaky
```

```
[maxpool]
size=2
```



## Б қосымшасының жалғасы

stride=2

[convolutional]  
batch\_normalize=1  
filters=128  
size=3  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=64  
size=1  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=128  
size=3  
stride=1  
pad=1  
activation=leaky

[maxpool]  
size=2  
stride=2

[convolutional]  
batch\_normalize=1  
filters=256  
size=3  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=128

## Б қосымшасының жалғасы

size=1  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=256  
size=3  
stride=1  
pad=1  
activation=leaky

[maxpool]  
size=2  
stride=2

[convolutional]  
batch\_normalize=1  
filters=512  
size=3  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=256  
size=1  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=512  
size=3  
stride=1  
pad=1  
activation=leaky

## Б қосымшасының жалғасы

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[maxpool]
size=2
stride=2
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
```

## Б қосымшасының жалғасы

pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=512  
size=1  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
filters=1024  
size=3  
stride=1  
pad=1  
activation=leaky

#####

[convolutional]  
batch\_normalize=1  
size=3  
stride=1  
pad=1  
filters=1024  
activation=leaky

[convolutional]  
batch\_normalize=1  
size=3  
stride=1  
pad=1  
filters=1024  
activation=leaky

[route]  
layers=-9

## Б қосымшасының жалғасы

```
[convolutional]
batch_normalize=1
size=1
stride=1
pad=1
filters=64
activation=leaky
```

```
[reorg]
stride=2
```

```
[route]
layers=-1,-4
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=1024
activation=leaky
```

```
[convolutional]
size=1
stride=1
pad=1
filters=155
activation=linear
```

```
[region]
anchors = 1.3221, 1.73145, 3.19275, 4.00944, 5.05587, 8.09892, 9.47112,
4.84053, 11.2364, 10.0071
bias_match=1
classes=26
coords=4
num=5
softmax=1
jitter=.3
rescore=1
```

## **Б қосымшасының жалғасы**

object\_scale=5  
noobject\_scale=1  
class\_scale=1  
coord\_scale=1

absolute=1  
thresh = .6  
random=0