

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни
«Сучасні методи та моделі інтелектуальних систем»
для студентів спеціальності F7 Комп'ютерна інженерія
освітньої програми «Спеціалізовані комп'ютерні системи»
усіх форм навчання
Частина 1

Методичні вказівки до лабораторних робіт з дисципліни «Сучасні методи та моделі інтелектуальних систем» для студентів спеціальності F7 Комп'ютерна інженерія освітньої програми «Спеціалізовані комп'ютерні системи» усіх форм навчання. Частина 1 /Укл.: М.Ю. Тягунова, Т.В. Голуб, С.Ю. Скрупський. – Запоріжжя: НУ «Запорізька політехніка», 2025. – 19 с.

Укладачі: М.Ю. Тягунова, к.т.н., доцент
Т.В. Голуб, к.т.н., доцент
С.Ю. Скрупський, к.т.н., доцент

Рецензент: Р.К. Кудерметов, к.т.н., доцент

Відповідальний
за випуск Т.В. Голуб, к.т.н., доцент

Затверджено
на засіданні кафедри КСМ
Протокол № 2 від 29.08.2025

Рекомендовано до видання
на засіданні НМК факультету КНТ
Протокол 2 від 10.09.2025

ЗМІСТ

Вступ.....	4
Лабораторна робота № 1 Розробка експертних систем	5
1 Загальні відомості.....	5
2 Структура експертних систем	7
3 Етапи створення експертних систем	10
3.1 Ідентифікація.....	10
3.2 Концептуалізація	11
3.3 Формалізація	11
3.4 Реалізація прототипної версії	12
3.5 Тестування.....	13
4 Завдання до лабораторної роботи	15
5 Контрольні питання.....	17
6 Вимоги до оформлення письмових звітів	18
Перелік джерел посилань.....	19

ВСТУП

Сучасний етап розвитку штучного інтелекту характеризується швидким зростанням можливостей інтелектуальних систем, здатних аналізувати дані, формувати висновки, навчатися та приймати рішення в умовах невизначеності. Одним із ключових напрямів застосування таких технологій залишаються експертні системи – програмні комплекси, які моделюють процес мислення й ухвалення рішень фахівців у певній предметній галузі. Вони є основою для створення систем підтримки прийняття рішень, діагностичних і рекомендаційних платформ, що активно впроваджуються в медицині, транспорті, промисловості, енергетиці, ІТ та освіті.

Навчання майбутніх фахівців у галузі комп'ютерної інженерії передбачає не лише ознайомлення з теоретичними принципами штучного інтелекту, але й практичне формування навичок побудови інтелектуальних систем, здатних відтворювати логіку експертних суджень. Робота над створенням експертної системи дозволяє студентам засвоїти принципи побудови баз знань, алгоритмів логічного висновку, інтерфейсної взаємодії та структурування знань у формі, придатній для машинної обробки.

Метою лабораторної роботи є практичне ознайомлення студентів із процесом розробки експертних систем – від аналізу предметної області до побудови працюючого прототипу. Результатом виконання завдання стане створення колективного інтелектуального програмного продукту, який демонструє можливості експертних систем у вирішенні реальних практичних задач.

ЛАБОРАТОРНА РОБОТА № 1 РОЗРОБКА ЕКСПЕРТНИХ СИСТЕМ

Мета роботи: навчитися розробляти експертні системи, дотримуючись мінімально необхідних вимог.

1 ЗАГАЛЬНІ ВІДОМОСТІ

Експертні системи – це напрям у сфері штучного інтелекту, орієнтований на створення комп'ютерних систем, здатних приймати рішення, подібні до рішень фахівців-експертів у певній предметній галузі. Такі системи розробляються переважно для вирішення практичних задач у вузьких спеціалізованих областях, де головну роль відіграють глибокі професійні знання та досвід людини. Експертні системи стали одним із перших прикладних напрямів, що привернули значну увагу до досліджень у галузі штучного інтелекту.

Ключова відмінність експертних систем від більшості сучасних AI-моделей (наприклад, нейронних мереж або еволюційних алгоритмів) полягає в тому, що вони не намагаються бути універсальними. Їхнє завдання – забезпечити максимально якісне, обґрунтоване рішення конкретної проблеми в певній області, спираючись на заздалегідь формалізовані знання експертів [1-3].

Експертне знання – це синтез теоретичного розуміння проблеми та перевірених на практиці навичок її вирішення. Фундаментом будь-якої експертної системи є база знань, сформована на основі досвіду експертів і структурована так, щоб комп'ютер міг використовувати її для логічних висновків. Від точності відбору експертів і якості формалізації їхніх знань безпосередньо залежить ефективність системи. Як і в реальному житті, де досвідчений лікар ставить точний діагноз завдяки поєднанню освіти й багаторічної практики, так і експертна система базує свої рішення на ретельно побудованій базі знань. Саме її якість визначає до 90% цінності кінцевого продукту.

Експертна система не є звичайною програмою, створеною виключно програмістами. Це результат спільної роботи експертів предметної області, інженерів знань (knowledge engineers) та розробників програмного забезпечення. Інколи самі експерти

виступають авторами логіки системи, якщо володіють достатніми технічними навичками.

Під час створення системи експерт передає інженеру знань структуровану інформацію про проблеми, типові ситуації та способи їх розв'язання. Наприклад, у випадку медичної діагностичної системи лікар описує характерні захворювання, симптоматику, діагностичні критерії та можливі схеми лікування. Інженер знань формалізує отриману інформацію, перетворюючи її у правила, факти та логічні зв'язки, які потім реалізуються у програмній частині системи.

Активний розвиток класичних експертних систем з часом сповільнився через низку об'єктивних причин:

- передача «глибоких» професійних знань у формалізований вигляд виявилася надзвичайно складним завданням, особливо щодо евристичних і контекстних знань;

- більшість експертних систем не здатні пояснити свої міркування так само гнучко й ґрунтовно, як людина; вони переважно відтворюють послідовність логічних кроків без інтерпретації;

- процес тестування й валідації експертних систем є складним і трудомістким, що особливо критично для сфер із підвищеними вимогами до надійності (авіація, транспорт, оборона, енергетика тощо);

- експертні системи не володіють механізмами самонавчання: для підтримки їх актуальності потрібне постійне оновлення бази знань інженерами. Без регулярної підтримки система швидко втрачає практичну цінність.

Сьогодні концепції експертних систем знаходять нове життя у поєднанні з сучасними методами машинного навчання, нейронними мережами та гібридними AI-підходами, які дозволяють автоматично оновлювати або уточнювати базу знань на основі аналізу великих обсягів даних.

2 СТРУКТУРА ЕКСПЕРТНИХ СИСТЕМ

На рисунку 1.1 зображена узагальнена структура експертної системи.

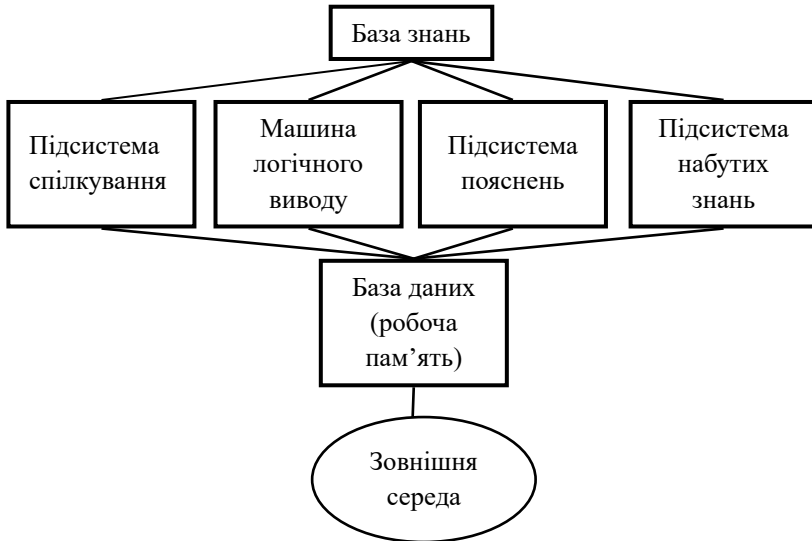


Рисунок 1.1 - Структура експертної системи

База знань (Knowledge Base) – це структуроване сховище експертних знань про певну предметну область, яке використовується системою для прийняття рішень. У сучасному розумінні база знань може містити як формалізовані правила, логічні зв'язки та фреймові структури, так і статистичні моделі або результати навчання машинного інтелекту, якщо система має гібридний характер.

База даних (Database / Working Memory) виконує функцію оперативного зберігання фактів, гіпотез, проміжних результатів обчислень або діалогових повідомлень. Вона відображає поточний стан задачі, яку розв'язує експертна система, і постійно оновлюється у процесі взаємодії з користувачем чи зовнішнім середовищем. У сучасних реалізаціях це може бути окрема база оперативних фактів

(working memory) або інтегрована пам'ять, що поєднує дані від користувача, сенсорів, зовнішніх API чи інших AI-модулів.

Машина логічного висновку (Inference Engine) – це ядро системи міркувань, яке оперує знаннями та даними для отримання нових фактів і рішень. Вона використовує різні підходи до логічного висновку: дедуктивні, індуктивні, абдуктивні або гібридні механізми. У класичних системах застосовуються алгоритми дедуктивного висновку на основі правил або пошуку рішень у мережах фреймів чи семантичних мережах. У сучасних AI-системах дедуктивна логіка часто поєднується з машинним навчанням і байєсівськими підходами [4, 5], що дозволяє отримувати більш гнучкі й адаптивні висновки.

Підсистема взаємодії з користувачем (User Interaction / Dialogue Subsystem) забезпечує діалог між експертною системою та користувачем. Вона відповідає за запит необхідних фактів, уточнення контексту задачі та представлення результатів у зручній формі. Сучасні експертні системи можуть реалізовувати цю підсистему через інтелектуальні інтерфейси — чатботи, голосових асистентів або інтерактивні веб-інтерфейси, здатні адаптуватися під стиль користувача.

Підсистема пояснень (Explanation Subsystem) призначена для інтерпретації логіки прийнятого рішення. Вона надає користувачеві можливість зрозуміти, яким чином система дійшла певного висновку, які правила чи дані при цьому використовувалися. Наявність зрозумілих пояснень критично важлива для довіри користувача. Відсутність прозорості робить систему «чорним ящиком», рішення якого важко перевірити чи прийняти. Сучасні AI-підходи активно використовують концепцію Explainable AI (XAI) — пояснюваного штучного інтелекту, який дозволяє користувачам бачити структуру логічних чи статистичних міркувань системи.

Підсистема набуття знань (Knowledge Acquisition Subsystem) відповідає за оновлення, розширення та уточнення бази знань. У базовому варіанті це інтелектуальний редактор, який дозволяє інженеру знань вносити зміни вручну. У більш складних системах ця підсистема автоматично добуває знання з різних джерел: баз даних, текстових і графічних матеріалів, наукових публікацій або великих неструктурованих масивів інформації. Сучасні методи включають machine learning-based knowledge acquisition, тобто здобуття нових

знань із даних за допомогою алгоритмів навчання, а також ontology learning – автоматичне формування онтологій для певної галузі.

Таким чином, експертна система – це інтелектуальна програмна система, призначена для моделювання процесу прийняття рішень експертом у певній предметній галузі. Вона виконує функції аналізу, діагностики, прогнозування, планування або рекомендацій, спираючись на формалізовані знання, досвід і логічні правила, які відображають експертну компетентність людини. Головна ідея експертної системи полягає в перенесенні експертного досвіду з людини у формалізовану, машинно-оброблювану форму. На відміну від універсальних моделей штучного інтелекту, які навчаються на великих даних, класичні експертні системи ґрунтуються на евристичних і логічних знаннях, отриманих від фахівців. Сучасний розвиток експертних систем спрямований на їх інтеграцію з технологіями машинного навчання, глибинних нейронних мереж і мовних моделей (LLM). Такі гібридні експертні системи поєднують формальну логіку з адаптивним навчанням із даних, що дозволяє їм: автоматично оновлювати базу знань, узагальнювати досвід з великих джерел інформації, формувати пояснювані висновки (explainable reasoning), взаємодіяти з користувачем у природній мові. Експертна система є поєднанням логічного інтелекту та доменної експертизи, яке забезпечує точність, обґрунтованість і відтворюваність рішень у конкретній області. У сучасному розумінні вона виступає ядром інтелектуальних систем підтримки прийняття рішень, інтегрованих у комплексні AI-екосистеми.

3 ЕТАПИ СТВОРЕННЯ ЕКСПЕРТНИХ СИСТЕМ

У проектуванні експертних систем можна виділити наступні етапи.

3.1 Ідентифікація

Цей етап складається з кількох внутрішніх етапів:

а) визначення учасників та їх ролей під час створення і експлуатації експертної системи.

У процесі створення експертної системи можуть брати участь наступні фахівці: інженери зі знань, експерти, програмісти, керівник проекту, замовники (кінцеві користувачі). При реалізації порівняно простих експертних систем програмістів може не бути. Роль інженера зі знань - здобуття професійних знань з експертів і проектування бази знань експертної системи і її архітектури. Програміст необхідний при розробці спеціалізованого для цієї експертної системи програмного забезпечення, коли відповідного стандартного (наприклад, оболонки для створення експертних систем) не існує або його можливостей недостатньо і потрібні додаткові модулі.

На етапі експлуатації можуть брати участь кінцеві користувачі, експерти, адміністратор.

б) ідентифікація проблеми.

На цьому етапі розробники повинні відповісти на ряд питань, що визначають особливості завдань, які вирішують експерти, а отже, і майбутня експертна система. Ці особливості визначають і особливості архітектури експертної системи, що формується на наступних етапах. До цих питань відносяться наступні:

- 1) який клас задач повинна вирішувати ЕС;
- 2) як ці завдання можуть бути охарактеризовані або визначені;
- 3) які можна виділити підзадачі;
- 4) які вихідні дані повинні використовуватися для вирішення;
- 5) які поняття і взаємозв'язки між ними використовуються при вирішенні задачі експертами;

- б) який вигляд має рішення і які концепції використовуються в ньому;
- 7) які аспекти досвіду експерта істотні для вирішення завдання;
- 8) яка природа і обсяг знань, необхідних для вирішення задачі;
- 9) які перешкоди зустрічаються при вирішенні задач;
- 10) як ці перешкоди можуть впливати на рішення задачі;
- в) визначення необхідних ресурсів - часових, людських, матеріальних;
- г) визначення цілей.

Очікується, що створення експертної системи дозволить досягти ряду цілей, до яких можуть відноситися: підвищення швидкості прийняття рішення, підвищення якості рішень, тиражування досвіду експертів тощо.

3.2 Концептуалізація

На цьому етапі розробники повинні відповісти на наступні питання:

- які типи даних потрібно використовувати;
- які дані задано та що повинно бути виведено;
- чи мають підзадачі найменування;
- чи мають стратегії найменування;
- чи є ясні часткові гіпотези, які широко використовуються.

3.3 Формалізація

Стадія формалізації є важливою в процесі створення експертної системи, оскільки саме на цьому етапі відбувається перехід від концептуального розуміння предметної області до чітко визначених формальних структур, придатних для реалізації в комп'ютерній системі. Формалізація передбачає переклад основних концепцій, підзадач, об'єктів і характеристик інформаційного потоку, виділених на етапі концептуалізації, у формат, який можна обробляти за допомогою обчислювальних методів.

У межах цього етапу розробляються формальні моделі знань, обираються відповідні методи подання знань (продукційні правила, фрейми, семантичні мережі, онтології тощо), а також визначається структура взаємозв'язків між елементами знань.

На даному етапі виділяють три ключові компоненти:

— простір гіпотез – сукупність можливих відповідей, рішень або припущень, які система може сформулювати на основі вхідних даних. Цей простір повинен бути чітко окреслений, з урахуванням обмежень та логіки предметної області, щоб уникнути хибних висновків;

— модель процесу – опис логіки міркування або виведення, яка буде використовуватись у системі. Це можуть бути правила логічного висновку, алгоритми ухвалення рішень, логіка евристичних методів або інші підходи, що визначають, як система переходитиме від фактів до гіпотез;

— характеристики даних – опис типів, джерел, форматів і властивостей вхідної інформації, яка буде використовуватись експертною системою. Також визначається, які атрибути є ключовими, які значення можуть приймати ті чи інші параметри, і як обробляти неповні або нечіткі дані.

Формалізація забезпечує основу для подальшої реалізації системи, зокрема побудови бази знань і механізму логічного виведення. Вона вимагає тісної співпраці між предметним експертом і інженером знань, оскільки саме на цьому етапі знання фахівця мають бути максимально точно переведені у формальну мову системи.

3.4 Реалізація прототипної версії

Стадія реалізації включає в себе переклад знань, формалізованих на попередній стадії, в схему уявлення, яка визначається обраною мовою. Як тільки представлені за цією схемою знання стають узгодженими і організованими так, щоб визначити потоки інформації і управління, вони стають робочою програмою.

Конструктор розвиває схему подання знань і використовує її для отримання прототипного варіанту ЕС. Предметно-орієнтовані знання, виявлені і сформульовані на етапі формалізації, визначають зміст

структури даних, правил введення і стратегій управління. Схема подання визначає їх форму. Локальна несуперечливість використовуваних для вирішення елементів, яка була досягнута на попередніх стадіях, не гарантує працездатності програми так, як можуть бути глобальні невідповідності між структурами даних і будь-якими правилами або стратегіями управління.

Такі протиріччя повинні бути усунені, щоб забезпечити швидкий розвиток прототипного варіанту ЕС. Створення прототипного варіанту є виключно важливим кроком при побудови ЕС. Окремі фрагменти цієї програми, що, в кінцевому рахунку, викидаються, можуть бути збережені і використані в більш пізніх версіях. Однак головною метою діяльності на цьому етапі стає перевірка відповідності формальної схеми основним використовуваним ідеям.

Прототипна база знань створюється за допомогою різних програмних засобів, наявних для обраної схеми подання знань (текстові редактори, інтелектуальні редактори, програми для здобуття знань). Якщо існуючі засоби виявляються невідповідними, то інженер знань повинен створити новий засіб. Може навіть трапитися так, що потрібно розробити нову систему, призначену для побудови ЕС або нову мову.

3.5 Тестування

Стадія тестування передбачає перевірку прототипного варіанту системи і схем представлення знань, використаних для створення цього варіанту. Як тільки прототипний варіант може виконати процес від початку до кінця на 2-3 прикладах, він повинен бути перевірений на багатьох прикладах для того, щоб виявити дефекти в базі знань і механізмі логічного висновку або дедуктивної машини.

Головними характеристиками введення-виведення є отримання даних і представлення результатів. Конкретний метод отримання даних може виявитися помилковим або невідповідним через те, що по ходу роботи задаються невірні питання або збирається недостатньо інформації. Наприклад, запитання можуть бути важкими для розуміння, двозначними або погано сформульованими для користувача системи. Сама система може бути незручною для роботи, в ній може бути відсутня можливість для користувача перервати хід обчислень, щоб ввести дані відповідно до обраних порядку і форми.

Найбільш очевидне місце для пошуку помилок в міркуваннях - множина правил виведення. Вони рідко бувають незалежними один від одного. Крім усього іншого, правила можуть бути некоректними, суперечливими, неповними або відсутніми взагалі. Якщо посилання в правилі визначені невірно, воно може додаватися в неправильному контексті, порушуючи, таким чином, всю логіку. У таких випадках дуже часто не вдається помічати підвипадки. Висновок правила може бути неправильним часто за своїм змістом або за можливістю розрізнити підвипадки. Навіть за умови правильності посилань і висновків, зв'язки між ними можуть бути некоректними. Так як правила не є незалежними, то має враховуватися їх вплив на інші правила.

Оцінки, ступені впевненості, ваги зв'язків грають важливу роль, і вони не можуть присвоюватися незалежно від того, як групи правил будуть використовуватися в контексті конкретних випадків.

Часто помилки в прототипній версії виникають через особливості використаних стратегій управління. Якщо система розглядає гіпотезу в порядку, відмінному від відповідного порядку, якого дотримується експерт, то конструктор повинен проаналізувати дедуктивну машину.

4 ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ

Колективно, у складі навчальної групи, під керівництвом викладача, розробити інтелектуальну експертну систему нового покоління, призначену для підтримки прийняття рішень користувачем-неспеціалістом у вибраній предметній області.

Мета – створити систему, здатну на основі формалізованих знань експертів, правил логічного висновку та (за бажанням) елементів машинного навчання, надавати обґрунтовані рекомендації або рішення.

Приклад теми: експертна система для допомоги користувачу у виборі автомобіля, ноутбука, маршруту подорожі, параметрів інвестицій, комп'ютерних комплектуючих, системи охорони тощо.

Очікуваний результат: завершеним результатом роботи має бути працюючий прототип експертної системи, що реалізує всі основні функціональні компоненти, а також документація, яка описує її структуру, логіку, базу знань і приклади використання.

Обов'язкові структурні компоненти системи:

— база знань (Knowledge Base). містить формалізовані правила, факти, фрейми або онтології, що описують предметну область. Джерелом знань є опитування експертів, аналіз літератури або відкритих джерел. Бажано забезпечити можливість оновлення бази знань через окремий інтерфейс або механізм автоматичного поповнення;

— база даних (Working / Fact Base). Використовується для збереження проміжних результатів, введених користувачем фактів та гіпотез. Має бути реалізована у вигляді структурованої моделі даних (SQL, NoSQL або у вигляді об'єктної моделі в коді);

— механізм логічного висновку (Inference Engine). Реалізує алгоритм міркування над базою знань і фактів. Може використовувати класичні методи (forward chaining, backward chaining) або сучасні підходи (rule-based + ML гібрид). Має забезпечувати пояснення прийнятого рішення (explanation trace);

— користувацький інтерфейс (User Interface). Забезпечує взаємодію користувача із системою у зрозумілій формі. Допускається текстовий (CLI), графічний (GUI) або веб-інтерфейс. Бажано передбачити пояснення системи своїх рішень у вигляді тексту, таблиці або графічної схеми;

— інтерфейс експерта (Expert Interface). Призначений для введення, редагування та перевірки правил і знань. Повинен мати можливість поповнення бази знань без необхідності змінювати код програми.

Рекомендовані додаткові елементи (за бажанням):

— підсистема пояснень (Explanation Subsystem / XAI-модуль) – пояснення логіки прийнятих рішень;

— модуль навчання або адаптації (Learning Module) – автоматичне оновлення знань на основі нових прикладів чи даних користувачів;

— онтологічна модель предметної області (Ontology Layer) – графова структура понять і зв'язків між ними;

— API або інтеграція з зовнішніми джерелами даних (наприклад, отримання технічних характеристик товарів чи статистики з відкритих API);

— модуль валідації рішень – перевірка правильності результатів за тестовими сценаріями.

Рекомендації до реалізації. Проєкт виконується у малих підгрупах (3–5 осіб) із розподілом ролей: аналітик знань, інженер знань, розробник логіки, інтерфейсний розробник, тестувальник. Архітектура системи має бути модульною, з чітким розділенням логіки, знань та інтерфейсів. Мова реалізації – за вибором групи (Python, Java, C#, JavaScript, Prolog тощо). Обов'язково підготувати коротку презентацію системи з демонстрацією прикладу роботи.

5 КОНТРОЛЬНІ ПИТАННЯ

1. Що собою являють експертні системи?
2. Призначення експертних систем.
3. Які обов'язкові складові частини експертної системи?
4. Переваги використання експертних систем.
5. Відмінності експертної системи та великої мовної моделі.

6 ВИМОГИ ДО ОФОРМЛЕННЯ ПИСЬМОВИХ ЗВІТІВ

При оформленні кожного звіту необхідно дотримуватися наступних вимог.

Звіт повинен містити:

1. Титульний аркуш із зазначенням теми, дисципліни, студента, який виконав звіт, викладача, який приймає звіт;
2. Результати виконання завдань лабораторної роботи;
3. Відповіді на контрольні питання;
4. Висновки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Fan, F. L., Xiong, J., Li, M., & Wang, G. (2021). On interpretability of artificial neural networks: A survey. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 5(6), 741-760.
2. Нейронні мережі : теорія та практика: навч. посіб. / С. О. Субботін. – Житомир : Вид. О. О. Євенок, 2020. – 184 с
3. Aggarwal С.С. *Neural Networks and Deep Learning*: Springer, 2018. – 512 p. – ISBN 3319944622
4. Mariia Tiahunova, Olesia Tronkina, Galina Kirichek, Stepan Skrupsky. The Neural Network for Emotions Recognition under Special Conditions. In *CMIS-2020: The Third International Workshop on Computer Modeling and Intelligent Systems*, Zaporizhzhia, Ukraine, April 27, 2021. p. 121-134, ISSN 1616-0073
5. Jospin, L. V., Laga, H., Boussaid, F., Buntine, W., & Bennamoun, M. (2022). Hands-on Bayesian neural networks—A tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2), 29-48.