National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"

Computer Systems Software Department

MACHINE LEARNING

Syllabus

Requisites of the Course	
Cycle of Higher Education	***Second cycle of higher education (Master's degree)***
Field of Study	*F Information Technologies*
Speciality	*F2 Software engineering*
Education Program	*Software engineering of multimedia systems and smart technologies,* *Software engineering of immersive systems and digital twin platforms*
Type of Course	*Selective*
Mode of Studies	*full-time*
Year of studies, semester	*1 year (2 semester)*
ECTS workload	*5 credits (ECTS). Time allotment - 150 hours, including 46 hours of classroom work (lectures – 32, lab. - 14), and 104 hours of self-study.*
Testing and assessment	*Exam, modular test , calendar control*
Course Schedule	*Classes by the timetable http://roz.kpi.ua/*
Language of Instruction	*English*
Course Instructors	*Lecturer: PhD, Associate Professor, Liubov Oleshchenko, oleshchenkoliubov@gmail.com* Laboratory work*: PhD, Associate Professor, Liubov Oleshchenko, oleshchenkoliubov@gmail.com*
Access to the course	*Google classroom: Access is given to registered students.*

Outline of the Course

1. Course description, goals, objectives, and learning outcomes

The study of the discipline "Machine Learning" allows students to form the competencies necessary to solve practical problems of professional activities related to data analysis using machine learning technologies. The purpose of studying the discipline "Machine Learning" is to develop students' ability to use machine learning technologies to develop software for clustering, data classification, image recognition and time series prediction. The subject of the discipline "Machine Learning" is machine learning technology for clustering, data classification, image recognition and time series prediction.

After mastering the discipline "Machine Learning" the learning outcomes are:

***knowledge**:*

- basic methods of data analysis using machine learning technologies.

***skills**:*

- create software for data analysis using machine learning technologies;

- use the capabilities of the Python programming language for clustering, data classification, image recognition and time series prediction.

***experience**:*

- development of software that allows to perform clustering, classification and forecasting to solve practical problems in various fields of human activity.

The study of the discipline "Machine learning" helps students to develop the competencies necessary to solve practical tasks of professional activity related to the use of machine learning methods:

GC01 *Ability to abstract thinking, analysis and synthesis.*

GC03 *Ability to conduct research at the appropriate level.*

PC02 *Ability to develop and implement scientific and / or applied projects in the field of software engineering.*

2. Prerequisites and post-requisites of the course

(the place of the course in the scheme of studies in accordance with curriculum)

Successful study of the discipline "Machine Learning" is preceded by the study of disciplines "Discrete Mathematics for Computer Science", "Mathematical Analysis", "Linear Algebra and Analytical Geometry", "Probability Theory", "Programming", the curriculum for bachelors in F2 "Software Engineering". The theoretical knowledge and practical skills obtained during the mastering of the discipline "Machine Learning" are necessary for the study of the discipline "Network Software Design and Development" of the curriculum for the preparation of masters in the specialty F2 "Software Engineering".

To successfully master the discipline requires a basic level of English not less than A2.

3. Content of the course

The discipline "Machine Learning" involves the study of the following topics:

Topic 1. Mathematical models and software tools for machine learning.

Topic 2. The use of neural networks.

Modular test.

Exam.

4. Coursebooks and teaching resources

Basic references:

1. Dipanjan Sarkar, Raghav Bali, Tushar Sharma. Practical Machine Learning with Python / https://link.springer.com/book/10.1007%2F978-1-4842-3207-1

2. Utilizing Python for Agent-Based Modeling: The Mesa Framework / https://www.researchgate.net/publication/344675633_Utilizing_Python_for_Agent-Based_Modeling_The_Mesa_Framework

3. Alex Smola and Vishwanathan S.V.N. Introduction to Machine Learning / https://alex.smola.org/drafts/thebook.pdf

4. Your First Machine Learning Project in Python Step-By-Step / https://machinelearningmastery.com/machine-learning-in-python-step-by-step/

5. Jurafsky, D., Martin, J. H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. (3rd ed., draft). Stanford University. https://web.stanford.edu/~jurafsky/slp3/

Additional references:

1. *Commonly used Machine Learning Algorithms (with Python and R Codes) / https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/*
2. *The 10 Best Machine Learning Algorithms for Data Science Beginners / https://www.dataquest.io/blog/top-10-machine-learning-algorithms-for-beginners/*
3. *Decision Tree / https://www.geeksforgeeks.org/decision-tree/*
4. *Explained: Neural networks / https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414*
5. *Using neural nets to recognize handwritten digits / http://neuralnetworksanddeeplearning.com/chap1.html*
6. *Dychka, I., Legeza, V., Oleshchenko, L., Bohutskyi, D. Method Simultaneous Using GAN and RNN for Generating Web Page Program Code from Input Image. Advances in Intelligent Systems and Computing. 2021.Vol. 1247. P. 338–349. DOI https://doi.org/10.1007/978-3-030-55506-1_31*
7. *Oleshchenko L.M. Intercity passenger flow forecasting and MTE buses optimal operation using LSTM neural network. 2024. https://doi.org/10.32782/2663-5941/2024.1.1/40*
8. *Oleshchenko L. Machine Learning Algorithms Comparison for Software Testing Errors Classification Automation. Lecture Notes on Data Engineering and Communications Technologies, 2023, 181, pp. 615–625. https://doi.org/10.1007/978-3-031-36118-0_55*
9. *Oleshchenko L.M., Zheng Jinsong. LSTM neural network implementation for the stock market operations predicting. 2024. https://doi.org/10.32782/2663-5941/2024.3.1/24*
10. *Oleshchenko L., Melnychuk O. Ensemble Classification Methods of Machine Learning for Analyzing News Texts for Falsity. Lecture Notes on Data Engineering and Communications Technologies, 2025, vol. 242. 349-361. Springer, Cham. https://doi.org/10.1007/978-3-031-84228-3_30*

Educational content

5. Methodology

№	Type of training session	Description of the lesson	Hours
Topic 1. Mathematical models and software tools for machine learning.			
1	*Lecture 1. History of machine learning.*	*History of machine learning. History of machine learning development. Predictive analysis and tasks of machine learning. Stages of scientific research. Errors in predictive analysis. Evaluation of the results of machine learning models. Types of machine learning.*	2
2	*Lecture 2. Analysis of statistical dependences and forecast models.*	*Linear correlation and linear regression. Estimation of statistical dependences and forecast models. Polynomial regression. Multicollinearity. Logistic regression.*	2
3	*Lecture 3. Using Python libraries for data analysis.*	*Modeling and forecasting of basic statistical dependences. Using Python libraries to analyze data.*	2
4	*Lecture 4. Software methods of machine learning.*	*Jupyter Notebook environment. Python tools for machine learning.*	2
5	*Lecture 5. Data preprocessing and feature selection for*	*Preparing data for training machine learning models. Data cleaning and missing value handling. Data scaling and normalization. Categorical feature encoding. Feature*	2

	machine learning.	*selection and feature engineering. Data dimensionality reduction (PCA). Splitting data into training, validation, and test sets. Preventing overfitting of models.*	
6	*Lecture 6. Data clustering methods.*	*Clustering methods. Examples of implementation of the K-means algorithm in Python.*	*2*
7	*Laboratory work 1. Using the K-Means clustering algorithm to analyze data in Python.*	*Lesson 1. Data preparation and construction of initial clustering.* *Task: load a dataset in CSV format (consumer segmentation task) into an environment using Python (Pandas). Conduct preliminary data analysis: review structure, occupancy, statistical characteristics. Perform data normalization (StandardScaler/MinMaxScaler). Implement the K-Means algorithm with a given number of clusters and visualize the initial distribution of points on the plane.*	*2*
		Lesson 2. Using K-Means++ and determining the optimal number of clusters. Analysis of results and visualization of clusters. *Task: implement the K-Means++ algorithm for centroid initialization. Construct an elbow curve (Elbow Method) to determine the optimal number of clusters for a given dataset. Compare clustering results at different values of the number of clusters and draw a conclusion about the optimal choice. Construct graphical visualizations of consumer segmentation (scatter plots, 2D/3D graphs). Compare clusters by key characteristics (mean attribute values). Interpret the results: describe consumer segments and explain the practical significance of clustering for business analytics.*	*2*
8	*Lecture 7. Methods of association and classification of data.*	*Methods of data association and classification. SVM method.*	*2*
9	*Lecture 8. Decision trees.*	*Decision trees. Random Forest.*	*2*
10	*Laboratory work 2. Forecasting based on the decision tree classifier.*	*Lesson 1. Data preparation and building an initial decision tree model.* *Task: load a dataset in CSV format (over 1300 passenger records) using the pandas library. Perform preliminary analysis and data preparation: check for gaps, encode categorical variables (LabelEncoder/OneHotEncoder), split into training and test samples. Create a basic decision tree classifier model using the sklearn library and evaluate its accuracy.*	*2*
		Lesson 2. Decision tree setup and visualization. Analysis of results and comparison with other models. *Task: optimize decision tree parameters (depth, number of leaves, split criterion). Use the Graphviz library to visualize the constructed decision tree. Explain the logic of decision-making in tree nodes using specific records from the dataset. Evaluate the effectiveness of the decision tree model using the accuracy, precision, recall, F1-score metrics. Construct a confusion matrix to visually display the results. Compare the performance of the decision tree with other classification algorithms (e.g., Random Forest or Logistic Regression). Draw*	*2*

		conclusions about the suitability of the decision tree for prediction on this data set.	
Topic 2. The use of neural networks.			
11	*Lecture 9. History of neural networks.*	*Neural networks. History of neural networks development.*	*2*
12	*Lecture 10. Architecture and algorithm of neural network.*	*Models and components of an artificial neuron. Transfer function (activation function). Types and characteristics of neural network activation functions. Architectures of connections of artificial neurons.* *The process of learning the neural network. Algorithm of neural network operation. Gradient descent method.*	*2*
13	*Lecture 11. Types of neural networks. TensorFlow and Keras machine learning software library.*	*Types of neural networks and their applications. Features of the TensorFlow library. Features of the Keras library.*	*2*
14	*Lecture 12. Convolutional neural network (CNN).*	*Convolutional neural network (CNN). Examples of application of convolutional neural networks.*	*2*
15	*Laboratory work 3. Using the convolutional neural network for image classification tasks.*	*Lesson 1. Data preparation and preprocessing.* *Task: load and explore an image dataset (CIFAR-10, MNIST). Perform data preprocessing: resize and scale images, normalize pixel values, convert labels to one-hot encoding, split data into training, validation, and test sets. Visualize several example images and classes for better understanding of the data.*	*2*
		Lesson 2. Building and training a CNN model. *Task: create a basic architecture of a convolutional neural network using TensorFlow/Keras libraries (Conv2D, MaxPooling2D, Flatten, Dense, Dropout layers). Train the model on the prepared dataset, tracking accuracy and loss over epochs. Plot accuracy and loss graphs for the training and validation sets. Evaluate the model's performance on the test set.*	*2*
		Lesson 3. Evaluation, improvement and visualization of results. *Tasks: optimize the model (add layers, change parameters, apply regularization or data augmentation). Make predictions for test samples and convert probabilities into classes. Build and analyze a confusion matrix and classification report. Visualize examples of images that the model classifies correctly and incorrectly. Compare the results of the basic and improved models.*	*2*
16	*Lecture 13. Recurrent neural network.*	*Recurrent neural network (RNN). Examples of application of recurrent neural networks.*	*2*
17	*Lecture 14. Long short-term memory.*	*Long short-term memory (LSTM). Examples of the use of long short-term memory.*	*2*
18	*Lecture 15. Natural*	*Fundamentals of Natural Language Processing (NLP).*	*2*

	Language Processing (NLP) methods.	*Preprocessing of text data: tokenization, lemmatization, stemming, removal of stop words. Representation of text as numerical features: Bag of Words, TF-IDF, Word Embeddings. Using NLTK, spaCy and TensorFlow libraries for text data analysis. Sentiment Analysis. Classification of text documents using neural networks. Language models and Transformer architecture. Using BERT, GPT and other large language models (LLM) for text data analysis.*	
19	*Lecture 16. Final lesson.*	*Review of the studied material. Modular test.*	*2*

6. Self-study

The discipline "Machine Learning" is based on independent preparation for classroom classes on theoretical and practical topics.

№	*The name of the topic that is submitted for independent study*	*Hours of study*	*References*
1	*Preparing for Laboratory work 1-3.*	*12*	*[1], pp. 119-304, [4]*
2	*Preparing for Topic 1.*	*32*	*[1], pp. 3-304.*
3	*Preparing for Topic 2.*	*28*	*[1], pp. 305-520, [2-5]*
4	*Preparing for modular test.*	*12*	*[1], pp. 3-520, [2-5]*
5	*Preparing for Exam.*	*20*	*[1], pp. 3-520, [2-5], [1-10 add.]*

Policy and Assessment

7. Course policy

Forms of organizing the educational process, types of training sessions and assessment of learning outcomes are regulated by the Regulations on the Organization of the Educational Process at the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".

***Class attendance policy**. Attendance at lecture classes is mandatory. Attendance at laboratory classes may be occasional and required for the defense of laboratory work. The presence or absence of a student at a class is not assessed by awarding or deducting points. If a student cannot attend classes, he or she is still responsible for studying the theoretical material and completing practical assignments.*

***Policy on ethical norms in the classroom**: discipline; compliance with subordination; honesty; responsibility; respect for those present, turning off phones.*

***Policy on assessing learning outcomes**. The policy on assessing learning outcomes is regulated by the Regulations on the system of assessing learning outcomes at Igor Sikorsky Kyiv Polytechnic Institute. According to the Regulations, each grade is given in accordance with the criteria developed by the teacher and announced to students in advance. If a student fails to complete all four laboratory tests, he/she will not be allowed to take the test. Failure to pass the current control measure (modular test) without good reason is assessed as 0 points.*

The policy and principles of academic integrity are regulated by the norms set of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" (https://kpi.ua/files/honorcode_en.pdf); Regulations on the Organization of the Educational Process, Regulations on the System for Preventing Academic Plagiarism, Regulations on the Commission on Ethics and Academic Integrity. Plagiarism and other forms of violation of the principles of academic integrity

are unacceptable. The student must complete all laboratory practical tasks independently using open sources of information and acquired knowledge and skills.

Plagiarism and other forms of violation of the principles of academic integrity are unacceptable. All work for current and semester tests must be completed independently by the student using open sources of information and the acquired knowledge and skills.

All works that violate the principles of academic integrity (the program code does not match the assignment option, the identity of the program code among different works, etc.) are evaluated at 0 points. To gain access to the test, the student must independently complete the laboratory work (without changing the current rating). In the case of semester control work that violates the principles of academic integrity, the semester control report is marked "Eliminate".

Policy on appealing the results of assessment of control measures. According to the "Regulations on resolving conflict situations at Igor Sikorsky Kyiv Polytechnic Institute" https://osvita.kpi.ua/sites/default/files/downloads/regulations_resolving_conflict_situations_2020.pdf, students have the right to appeal the results of control measures with arguments, explaining which criterion they disagree with according to the assessment. A student can raise any issue related to the procedure of control measures and expect that it will be considered in accordance with predetermined procedures. Policy on assigning incentive points. According to the "Regulations on the system of assessing learning outcomes at Igor Sikorsky Kyiv Polytechnic Institute", incentive points are not included in the main RSO scale, and their sum cannot exceed 10 points.

Incentive points are awarded for a creative approach in performing laboratory work (the maximum number of points for all work is 10 points), as well as for participation in scientific projects and conferences related to the topic of this course

8. Monitoring and grading policy

During the semester, students complete 6 computer workshops. Maximum number of points for each Laboratory work: 10 points.

Points are awarded for:

- quality of Laboratory work: 0-4 points;

- answer during the defense of the Laboratory work: 0-4 points;

- timely submission of work to the defense: 0-2 points.

Maximum number of points for performing and defending Laboratory work:

10 points × 3 comp. work. = 30 points.

The task for the modular test consists of 5 questions - 3 theoretical and 2 practical. The answer to each theoretical question is evaluated by 3 points, the answer to the practical question is evaluated by 3 points.

Criteria for evaluating each theoretical question of the module test:

3 points - the answer is correct, complete, well-argued;

2 points - there are minor errors in the answer;

1 points - there are significant errors in the answer;

0 points - no answer or the answer is incorrect.

Criteria for evaluating the practical question of modular control work:

3 points - the answer is correct, complete, well-argued;

2 points - there are minor errors in the answer;

1 points - there are significant errors in the answer;

0 points - no answer or the answer is incorrect.

Maximum number of points for each modular test:

3 points × 3 theoretical questions + 3 points × 2 practical questions = 15 points.

The rating scale for the discipline is equal to:

R = Rs = 30 points + 30 points = 60 points.

According to the description: $R = R_{comp.work} + R_{tests} + R_{exam} = 30 + 30 + 40$ points = 100 points

Calendar control: conducted twice a semester as a monitoring of the current state of compliance with the requirements of the syllabus.

At the first attestation (8th week) the student receives "credited" if his current rating is not less than 50% of the maximum number of points that the student can receive before the first attestation.

At the second attestation (14th week) the student receives "credited" if his current rating is not less than 50% of the maximum number of points that the student can receive before the second attestation.

Semester control: ***exam.***

Conditions of admission to semester control:

with a semester rating (Rs) of at least 60 points and enrollment in all computer workshops, the student receives the exam "automatically" according to the table (Table of correspondence of rating points to grades on the university scale).

Prerequisite for admission to the exam is the implementation and defense of a computer workshop.

The final performance score or the results of the Exam Fail/ Pass are adopted by university grading system as follows:

Score	Grade
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactory
64-60	Sufficient
Below 60	Fail
Course requirements are not met	Not Graded

9. Additional information about the course

According to the Regulations on the recognition of learning outcomes acquired in non-formal/informal education at Igor Sikorsky Kyiv Polytechnic Institute https://osvita.kpi.ua/node/179, during the study of the discipline, it is possible to enroll in an online course or other element of non-formal education in the discipline.

Having a certificate of completion of a similar course in machine learning is estimated as 10 points (if the course topic corresponds to the topic of the laboratory workshop), writing articles or participating in conferences/projects on the relevant topic is also estimated as an additional 5 points.

For example, laboratory work 3 can be automatically credited as 10 points by sending certificates of completion of such courses to the classroom as:

Convolutional Neural Networks in TensorFlow *(https://www.coursera.org/learn/convolutional-neural-networks-tensorflow),*

Advanced Computer Vision with TensorFlow *(https://www.coursera.org/learn/advanced-computer-vision-with-tensorflow) etc.*

In particular, individual labs may be automatically credited upon submission of a certificate of successful completion of the following online courses or their equivalents:

***Machine Learning Specialization (DeepLearning.AI & Stanford University, Coursera)** – 95 hours (consists of the courses Supervised Machine Learning: Regression and Classification, Advanced Learning Algorithms, Unsupervised Learning, Recommenders, Reinforcement Learning). The course covers the basics of machine learning, regression, classification, decision trees, neural networks, clustering, recommender systems, and reinforcement learning. It can be credited as labs 1 and 2 or scored up to 10 points. https://www.coursera.org/specializations/machine-learning-introduction*

***Supervised Machine Learning: Regression and Classification (DeepLearning.AI & Stanford University, Coursera)** – 33 hours. It can be credited as lab 2 or scored up to 10 points. https://www.coursera.org/learn/machine-learning*

***Advanced Learning Algorithms (DeepLearning.AI & Stanford University, Coursera)** – 34 hours. Can be counted as lab #2 or scored up to 10 points.*

https://www.coursera.org/learn/advanced-learning-algorithms

***Custom Models, Layers, and Loss Functions with TensorFlow (DeepLearning.AI, Coursera) – 30 hours. The course is relevant to the topic of convolutional neural networks and TensorFlow, can** be counted as lab #3 or scored up to 10 points.*

https://www.coursera.org/learn/custom-models-layers-loss-functions-with-tensorflow

***Advanced Computer Vision with TensorFlow (DeepLearning.AI, Coursera)** – 19 hours. Can be automatically credited as lab #3.*

https://www.coursera.org/learn/advanced-computer-vision-with-tensorflow

***Machine Learning (Prometheus)** – 30 hours. The course covers basic machine learning methods and can be credited as lab #1 or #2 or be graded up to 10 points.*

https://prometheus.org.ua/prometheus-free/machine-learning

***Artificial Intelligence Tools and Techniques (NEXT Digital Transformations for Supporting Next-Generation Labour, Erasmus+)** – 8 hours. The course can be graded as an additional 5 points.*

https://drive.google.com/file/d/1dsVRXORqFr7sJlSCYvJk2U1fZBA-xJ_s/view

Students may also submit certificates of completion of other online courses (Coursera, edX, Udemy, Google, Microsoft Learn, Kaggle Learn, Prometheus, DataCamp, Fast.ai, etc.) if their learning outcomes correspond to the content of the discipline. The decision on enrollment is made by the instructor after analyzing the course program and the acquired learning outcomes.

The student must inform the teacher about the course completed or planned to be studied and clarify the results and prospects for crediting the learning outcomes acquired in non-formal/informal education.

Syllabus of the course

Is designed by teacher PhD, Associate Professor, Liubov Oleshchenko

Adopted by Computer Systems Software Department (protocol № 12, 11.02.2026)

Approved by the Faculty Board of Methodology (protocol № 7, 23.02.2026)