Konspekt kursu: Python, LLM i analiza danych w notatnikach Jupyter

Cel kursu

Celem kursu jest rozwinięcie praktycznych umiejętności wykorzystania języka Python, notatników Jupyter oraz modeli językowych LLM w procesie analizy danych z naciskiem na wykorzystanie loklanyhc modeli LLM. Kurs prowadzi studenta przez kompletny, powtarzalny workflow analityczny: od wczytania danych, przez ich kontrolę jakości, czyszczenie, przekształcanie, eksploracyjną analizę danych i wizualizację, aż po interpretację wyników oraz przygotowanie raportu obliczeniowego.

Kurs nie jest kursem programowania w Pythonie od podstaw. Zakłada, że student zna podstawowe konstrukcje języka Python, ale potrzebuje uporządkowanego wprowadzenia do pracy analitycznej z bibliotekami Pandas, NumPy i Matplotlib. Szczególny nacisk położony jest na pracę w środowisku Jupyter Notebook, w którym kod, dane, wyniki, wykresy i komentarze tworzą spójny raport obliczeniowy.

Istotnym elementem kursu jest świadome wykorzystanie modeli językowych LLM jako narzędzi wspomagających pracę analityka danych. Studenci uczą się korzystać z LLM do wyjaśniania kodu, generowania hipotez, proponowania kroków analizy, przygotowywania opisów wyników, tworzenia checklist jakości oraz redagowania raportu. Jednocześnie kurs podkreśla ograniczenia modeli językowych: możliwość generowania błędnego kodu, niepoprawnych interpretacji, nadmiernie pewnych wniosków oraz odpowiedzi wymagających weryfikacji przez dane, kod, dokumentację i krytyczne myślenie.

Adresaci kursu

Kurs jest przeznaczony dla studentów uczelni technicznej, którzy znają podstawy programowania w Pythonie lub ukończyli wcześniejszy kurs wprowadzający do Pythona. Szczególnie dobrze wpisuje się w potrzeby studentów kierunków technicznych, informatycznych, inżynierskich, ekonomiczno-technicznych oraz wszystkich osób, które chcą nauczyć się praktycznej analizy danych w środowisku Jupyter Notebook.

Organizacja kursu

Kurs obejmuje 30 godzin zajęć dydaktycznych oraz do 60 godzin pracy własnej studenta. Zajęcia powinny mieć charakter praktyczny i warsztatowy. Każdy moduł powinien łączyć krótkie wprowadzenie pojęciowe, demonstrację kodu, ćwiczenia wykonywane przez studentów oraz element interpretacji wyników.

Proponowana forma zaliczenia to projekt analityczny realizowany w Jupyter Notebook oraz test samosprawdzający. W ramach projektu student przygotowuje raport obliczeniowy na podstawie wybranego lub otrzymanego zbioru danych. Raport powinien obejmować czyszczenie danych, eksploracyjną analizę danych, wizualizacje, interpretację wyników oraz udokumentowane wykorzystanie modelu LLM wraz z oceną jakości jego sugestii.

Podział kursu na rozdziały / moduły

Moduł 1. Powtarzalny workflow analizy danych w Jupyter Notebook

Pierwszy moduł wprowadza studenta w sposób pracy charakterystyczny dla raportu obliczeniowego. Student poznaje strukturę notatnika Jupyter jako dokumentu łączącego kod, opis, wyniki, wykresy i wnioski. Omawiane są dobre praktyki organizacji projektu analitycznego: podział katalogów, przechowywanie danych wejściowych, zapisywanie wyników, komentowanie decyzji analitycznych oraz dbałość o odtwarzalność.

Zakres modułu obejmuje uruchomienie środowiska Jupyter Notebook, podstawową strukturę notatnika, komórki Markdown i komórki kodu, zasady czytelnego dokumentowania pracy oraz przykładowy mini-workflow analizy danych. Student uczy się, że notatnik nie powinien być jedynie zbiorem przypadkowo uruchamianych komórek, ale uporządkowanym raportem pokazującym tok rozumowania.

W module pojawia się także pierwsze wykorzystanie LLM: model może pomóc uporządkować plan analizy, wyjaśnić fragment kodu, zaproponować strukturę raportu albo pomóc w redagowaniu komentarzy. Student uczy się jednak, że propozycje modelu należy traktować jako sugestie wymagające sprawdzenia.

Moduł 2. Wczytywanie, czyszczenie i przygotowanie danych

Drugi moduł koncentruje się na etapie przygotowania danych do analizy. Student pracuje z typowymi formatami plików, takimi jak CSV i XLSX, oraz poznaje najczęstsze problemy występujące w rzeczywistych zbiorach danych: braki danych, błędne typy kolumn, duplikaty, niespójne nazwy kategorii, nietypowe wartości, błędne jednostki oraz wartości odstające.

Zakres modułu obejmuje wczytywanie danych za pomocą Pandas, sprawdzanie struktury zbioru, analizę typów danych, obsługę braków, filtrowanie rekordów, zmianę nazw kolumn, konwersje typów, tworzenie nowych kolumn oraz proste agregacje. Przykłady mogą dotyczyć danych pomiarowych, zużycia energii, temperatury, frekwencji, wyników testów lub danych eksploatacyjnych.

LLM jest w tym module wykorzystywany jako pomoc w diagnozowaniu problemów jakości danych. Student może poprosić model o listę możliwych problemów w zbiorze, propozycję kroków czyszczenia lub wyjaśnienie błędu w kodzie. Każda sugestia modelu musi jednak zostać zweryfikowana przez uruchomiony kod i kontrolę wyników.

Moduł 3. Eksploracyjna analiza danych z użyciem Pandas

Trzeci moduł wprowadza eksploracyjną analizę danych jako etap poszukiwania wzorców, zależności i potencjalnych problemów w danych. Student uczy się formułować pytania analityczne, obliczać podstawowe statystyki opisowe, grupować dane, porównywać kategorie i analizować zmienność wyników.

Zakres modułu obejmuje statystyki opisowe, filtrowanie danych, sortowanie, grupowanie, agregacje, tabele przestawne, analizę liczebności, porównywanie podzbiorów danych oraz proste badanie

zależności między zmiennymi. Nacisk położony jest na interpretację wyników, a nie tylko na uzyskanie poprawnego kodu.

LLM może wspierać studenta w tworzeniu pytań analitycznych, generowaniu hipotez, wyjaśnianiu działania metod Pandas oraz proponowaniu kolejnych kroków analizy. Student uczy się odróżniać obserwację od wniosku: obserwacja wynika bezpośrednio z danych, natomiast wniosek wymaga uzasadnienia i ostrożności interpretacyjnej.

Moduł 4. Wizualizacja danych i narracja analityczna

Czwarty moduł pokazuje, jak przekształcać wyniki analizy w czytelne wykresy i komunikaty analityczne. Student poznaje zasady doboru wykresu do typu danych i celu komunikacyjnego. Omawiane są wykresy liniowe, słupkowe, punktowe, histogramy, wykresy pudełkowe oraz proste mapy ciepła.

Zakres modułu obejmuje tworzenie wykresów z użyciem Matplotlib, opisywanie osi, stosowanie jednostek, nadawanie tytułów, kontrolę czytelności, unikanie nadmiarowych ozdobników oraz dopasowanie wykresu do odbiorcy. Student uczy się, że wizualizacja nie jest dodatkiem do analizy, ale jednym z głównych narzędzi rozumienia danych.

LLM jest wykorzystywany do proponowania typów wykresów, redagowania podpisów, streszczania obserwacji oraz wskazywania potencjalnych błędów wizualizacji. Student sprawdza, czy opis wygenerowany przez model rzeczywiście odpowiada temu, co widać na wykresie i co wynika z danych.

Moduł 5. LLM jako asystent analityka danych

Piąty moduł jest poświęcony świadomemu i krytycznemu wykorzystaniu modeli językowych w pracy analitycznej. Student poznaje dobre praktyki promptowania w kontekście analizy danych, generowania kodu, refaktoryzacji notatnika, wyjaśniania błędów, dokumentowania decyzji oraz tworzenia checklist jakości.

Zakres modułu obejmuje przykłady promptów do analizy danych, iteracyjne doprecyzowywanie poleceń, porównywanie odpowiedzi modelu, testowanie wygenerowanego kodu, wykrywanie błędnych założeń oraz dokumentowanie użycia LLM w raporcie. Szczególną uwagę należy poświęcić bezpieczeństwu danych i zasadzie nieprzekazywania danych poufnych do narzędzi, które nie są do tego przeznaczone.

Student uczy się, że LLM nie zastępuje analityka. Model może przyspieszyć pracę, podpowiedzieć możliwe kierunki analizy, pomóc zrozumieć kod lub przygotować wstępny opis wyników, ale nie gwarantuje poprawności. Ostateczna odpowiedzialność za jakość analizy, poprawność kodu i zasadność wniosków pozostaje po stronie człowieka.

Moduł 6. Raport obliczeniowy i prezentacja wyników

Szósty moduł integruje wszystkie wcześniejsze elementy kursu. Student przygotowuje końcowy raport obliczeniowy w Jupyter Notebook, który zawiera opis problemu, dane wejściowe, kontrolę jakości danych, czyszczenie, analizę eksploracyjną, wizualizacje, interpretację wyników oraz końcowe wnioski.

Zakres modułu obejmuje strukturę raportu technicznego, sposób opisywania metod, prezentowanie tabel i wykresów, formułowanie wniosków, wskazywanie ograniczeń analizy oraz przygotowanie wersji

raportu dla różnych odbiorców. Student uczy się, jak uporządkować notatnik, usunąć zbędne eksperymenty, opisać najważniejsze decyzje analityczne i zadbać o odtwarzalność wyników.

W module końcowym student dokumentuje również wykorzystanie LLM: wskazuje, w których miejscach korzystał z modelu, jakie prompty były użyte, jakie sugestie przyjął, które odrzucił i w jaki sposób zweryfikował odpowiedzi modelu. Dzięki temu LLM staje się jawnym elementem procesu analitycznego, a nie ukrytym źródłem gotowych odpowiedzi.

Sylwetka absolwenta

Absolwent kursu potrafi samodzielnie przygotować podstawową, ale kompletną analizę danych w środowisku Jupyter Notebook. Umie zaplanować strukturę pracy, wczytać dane, sprawdzić ich jakość, oczyścić zbiór, przeprowadzić eksploracyjną analizę danych, przygotować odpowiednie wizualizacje oraz sformułować wnioski wynikające z analizy.

Absolwent rozumie rolę notatnika Jupyter jako raportu obliczeniowego. Potrafi łączyć kod, wyniki, komentarze i wykresy w jeden spójny dokument. Dba o czytelność kodu, logiczny układ komórek, opisywanie założeń oraz możliwość ponownego uruchomienia analizy przez inną osobę.

Absolwent potrafi korzystać z podstawowych narzędzi analitycznych ekosystemu Pythona, w szczególności z bibliotek Pandas, NumPy i Matplotlib. Umie stosować je do typowych zadań analitycznych: filtrowania danych, agregacji, obliczania statystyk opisowych, porównywania grup, przekształcania danych i przygotowywania wykresów.

Absolwent potrafi wykorzystywać modele językowe LLM jako wsparcie w pracy analitycznej. Umie formułować prompty dotyczące kodu, danych, interpretacji wyników i struktury raportu. Potrafi korzystać z LLM do wyjaśniania błędów, proponowania kroków analizy, generowania roboczych opisów oraz tworzenia checklist kontrolnych.

Absolwent zna ograniczenia modeli językowych i potrafi krytycznie oceniać ich odpowiedzi. Wie, że model może wygenerować błędny kod, niepoprawnie zinterpretować dane lub zaproponować wnioski, które nie wynikają z analizy. Potrafi weryfikować sugestie modelu przez uruchomienie kodu, sprawdzenie wyników, porównanie z dokumentacją i logiczną ocenę założeń.

Absolwent jest przygotowany do realizacji prostych projektów analitycznych w kontekście technicznym, edukacyjnym, organizacyjnym lub inżynierskim. Potrafi pracować z danymi pomiarowymi, eksploatacyjnymi, laboratoryjnymi, ankietowymi lub organizacyjnymi. Umie przedstawić wyniki w sposób zrozumiały dla odbiorcy oraz wskazać ograniczenia przeprowadzonej analizy.

Proponowane efekty uczenia się

Po ukończeniu kursu student:

1. przygotowuje uporządkowany notatnik Jupyter zawierający kod, opis, wyniki i wnioski;
2. wczytuje dane z plików CSV i XLSX oraz ocenia ich podstawową jakość;
3. identyfikuje i koryguje typowe problemy w danych, takie jak braki, duplikaty, błędne typy i niespójne kategorie;
4. wykonuje eksploracyjną analizę danych z użyciem biblioteki Pandas;

5. oblicza i interpretuje podstawowe statystyki opisowe;
6. tworzy czytelne wizualizacje z użyciem Matplotlib;
7. dobiera typ wykresu do rodzaju danych i celu analizy;
8. wykorzystuje LLM do wspomagania pracy z kodem, analizą i raportem;
9. weryfikuje odpowiedzi LLM na podstawie danych, kodu i dokumentacji;
10. przygotowuje końcowy raport obliczeniowy opisujący proces analizy, wyniki, wnioski i ograniczenia.

Forma zaliczenia

Zaliczenie kursu ma formę projektu analitycznego oraz weryfikację nabytej wiedzy poprzez test samosprawdzający. W ramach projektu student wybiera lub otrzymuje zbiór danych, a następnie przygotowuje kompletny raport w Jupyter Notebook. Raport powinien zawierać opis problemu, charakterystykę danych, kontrolę jakości, czyszczenie i przygotowanie danych, eksploracyjną analizę danych, wizualizacje, interpretację wyników oraz końcowe wnioski.

Elementem obowiązkowym projektu powinno być udokumentowane wykorzystanie LLM. Student powinien wskazać, do czego używał modelu, podać przykłady promptów, opisać uzyskane odpowiedzi oraz wyjaśnić, w jaki sposób sprawdził ich poprawność. Ocenie nie powinno podlegać samo użycie LLM, lecz umiejętność odpowiedzialnego i krytycznego włączenia modelu w proces analityczny.

Przykładowy temat projektu końcowego

Analiza danych eksperymentalnych lub eksploatacyjnych z wykorzystaniem Pythona i LLM.

Student przygotowuje raport obliczeniowy w Jupyter Notebook na podstawie zbioru danych dotyczącego na przykład pomiarów temperatury, zużycia energii, frekwencji, wyników testów, czasu realizacji zadań, aktywności użytkowników albo danych laboratoryjnych. W raporcie dokumentuje proces przygotowania danych, wykonuje analizę eksploracyjną, tworzy wizualizacje, formułuje wnioski oraz opisuje, jak korzystał z modelu LLM i jak weryfikował jego sugestie.