Vicuna

Selekcijski zadatak – Kvantitativni ML inženjer

Uvod

Vaš zadatak je implementirati i kontejnerizirati jednostavan API koji pruža funkcionalnost predviđanja buduće cijene financijskog instrumenta na temelju povijesnih podataka. Dostupan je skup podataka s satnim OHLCTV vrijednostima na linku:

Naziv	Opis
Date	Datum koji označava sat u kojem su izmjereni podaci.
Open	Početna cijena sata
High	Najviša cijena sata
Low	Najniža cijena sata
Close	Završna cijena sata
Trades	Ukupan broj transakcija u satu
Volume	Ukupan broj trade-anih jedinica instrumenta u satu

Cilj je izraditi prediktivni model koji na temelju povijesnih podataka predviđa sljedeću vrijednost cijene (ili povrata). Preporučujemo da razmotrite predviđanje povrata (return) umjesto same cijene, jer su povrati tipično stacionarniji i prikladniji za modele strojnog učenja. U nastavku se nalaze detaljniji opisi za sve tri cjeline, u kojima se nalaze upute te pitanja na koja je potrebno odgovoriti. Pritom imajte na umu da velik naglasak stavljamo na kreativnost i proaktivnost, tako da ne postoji "strogo točno" rješenje. Zadatak ste slobodni riješiti koristeći proizvoljne tehnologije i knjižnice, dokle god su tražene funkcionalnosti ostvarene. Vremenski rok za rješavanje iznosi 10 dana od trenutka kada dobijete zadatak. Rješenje koje nam pošaljete bi trebalo sadržavati sljedeće:

1. Izvorni kod (Jupyter notebook ili Python skripte): analiza i generiranje feature-a, treniranje i evaluacija modela, kod REST API-a i Docker konfiguracije.
2. Izvještaj (PDF, Markdown, notebook ili tekstualni dokument): opis svega što je napravljeno po svakoj fazi, odgovori na pitanja iz zadatka, vizualizacije, tablice i komentari.

Svrha zadatka je pokazati vještine u raznim domenama ML inženjerstva, ali i pokazati kreativnost i snalaženje u rješavanju problema. To nam pomaže u odabiru užeg kruga kandidata, stoga nema smisla prepisivati kôd iz postojećih rješenja. Rješenja koja su očito u potpunosti ili djelomično prepisana biti će odbačena. Ukoliko ne uspijete riješiti neke dijelove zadatka, pošaljite nam djelomično rješenje i opišite probleme koje ste imali, kako ste ih pokušali riješiti i zašto mislite da niste uspjeli. Važno je da se potrudite, a ne da savršeno riješite zadatak. Također, slobodno šaljite konstruktivna pitanja e-mailom – nećemo riješiti zadatak umjesto vas, ali vas možda možemo usmjeriti :)

1 Analiza i priprema podataka

Prije treniranja modela potrebno je provesti analizu podataka i kreirati značajke ("feature engineering"). Osnovni skup podataka sadrži samo "sirove" ili „raw" OHLCTV vrijednosti, a zadatak studenta je samostalno generirati izvedene feature-e.

Prijedlozi za moguće feature-e:

- **Lagged returns:** npr. return_t-1, return_t-5, return_t-10
- **Volatilnost:** standardna devijacija povrata kroz prozor od n dana
- **Moving averages:** SMA(5), SMA(20), EMA(10)
- **Volume indicators:** omjer trenutnog volumena i prosječnog volumena zadnjih 10 dana
- **Price ratios:** High/Low, Close/Open
- **Momentum:** razlika Close_t - Close_t-5
- **RSI, MACD, Bollinger bands** (ako žele uključiti tehničke indikatore)
- **Time-based features:** dan u tjednu, mjesec, sezonski efekti

Osim izrade značajki, potrebno je dobro razumjeti podatke. Neki od postupaka koje možete provesti nalaze se u nastavku (ne morate napraviti sve, i slobodno napravite nešto što nije predloženo):

1. **Čišćenje skupa podataka** (Što s nedostajućim (null) vrijednostima)
2. **Transformacija skupa podataka** (Kada učitate podatke, jesu li sve kolone ispravnog tipa?)
3. **Kreiranje značajki.**

4.Vizualizacija skupa podataka. (Kako su značajke distribuirane? Postoje li neke očite međuovisnosti između značajki?)

5. Odabir podskupa podataka. (Kako pravilo odabrati podskup podataka koji su u formatu vremenskih serija?)

6. Priprema skupa podataka za sljedeći korak. (Je li oblik u kojem se vaš skup podataka trenutno nalazi dovoljan i prikladan za sve potrebe koje ćete imati pri odabiru, treniranju i evaluaciji modela?)

Za vrijeme analize podataka i pripreme podataka za sljedeći korak, imajte na umu da se radi o podacima vremenskih serija, koji trebaju biti nešto drugačije tretirani od kros-sekcijskih podataka ili panel podataka. U financijskim podacima najbitnije je ne dopustiti izlijevanje (leakage) informacije iz testnog ili validacijskog podskupa – u trening skup – jer će to stvoriti „forward-looking bias“, što značajno utječe na rezultate. Razmišljajte da u danom trenutku „t“, uvijek morate imati samo informacije dostupne u tom trenutku ili prije – a ni slučajno informaciju koja dolazi nakon trenutka t (iz budućnosti).

Pitanja:

1. Što ste odlučili napraviti sa null vrijednostima i zašto?
2. Koje feature-e ste kreirali i zašto? Jeste li testirali njihovu informativnost?
3. Ako ste izrađivali vizualizaciju, opišite zašto ste to radili i kako vam je to doprinijelo razumijevanju podataka.
4. Koju metodu unakrsne validacije ste koristili? Kako ste pripremili skup podataka za sljedeći korak?

2 Modeliranje i evaluacija

Nakon što ste zadovoljni sa skupom podataka iz prethodne cjeline, vrijeme je pozabaviti se izradom i pripremom modela. U ovoj cjelini očekujemo da istražite modele koji su prikladni za rješavanje problema predviđanja cijena. Cilj je predvidjeti buduću vrijednost cijene ili povrata na temelju izgrađenih značajki. Model može biti bilo koje vrste (linearni modeli, stabla, neuronske mreže,…). Za razliku od prethodne cjeline, ovdje nećemo dati prijedloge mogućih postupaka jer su postupci za ovu cjelinu uglavnom “standardizirani” i pojavljuju se

u svakom AI/ML projektu. Ono u čemu se vaše rješenje može razlikovati je način na koji ste realizirali te postupke. Ponavljamo da ste potpuno slobodni odabrati bilo koji model iz bilo koje knjižnice, dokle god možete odraditi sve što je bitno za ovaj korak. Također, imajte na umu da ne trebate savršeno predvidjeti cijenu i mi nećemo sortirati kandidate po metrikama dobivenim iz rezultata. Radi se o istraživačkoj poziciji gdje će se više uzimati u obzir pristupi, razumijevanje i kritičko razmišljanje. Konačno, potrebno je na neki način trenirani model spremiti, kako biste ga mogli koristiti u API-u kojeg ćete izgraditi u sljedećem koraku.

Pitanja:

1. Jeste li koristili regresijski ili klasifikacijski model?
2. Koju ste ciljnu varijablu koristili?
3. Koje ste modele isprobali i koji se pokazao najboljim?
4. Kako ste procjenjivali performanse?
5. Kako biste znali da vaš model nije samo overfit na povijest?
6. Kako biste primijenili model u realnom vremenu (on-line predikcija)?

3 Izgradnja i kontejnerizacija API-a

Nakon što ste dobili trenirani model, potrebno je izraditi REST API koji omogućuje predikciju nove vrijednosti. API treba imati dvije pristupne točke:

- GET /info – vraća opis modela i upute za korištenje
- POST /predict – prima podatke o novim „raw“ cijenama (OHLCTV) i vraća predviđenu vrijednost.

Preporučujemo upotrebu FastAPI frameworka te Docker-kontejnerizaciju.

Pitanja:

1. U kojem formatu vaš API prima podatke?
2. Što se događa ako korisnik pošalje neispravne ili nepotpune podatke?

3. Kako ste integrirali model unutar kontejnera?
4. Koje su prednosti Docker-izacije?

Napomene i preporuke

Model ne mora savršeno predviđati cijenu. Više nas zanima vaš proces razmišljanja: kako pripremate podatke, birate feature-e, testirate modele i objašnjavate rezultate. Kreativnost i analitički pristup nose veću težinu od "točnog broja".

Rok i predaja

Rok za rješavanje zadatka je 10 dana od trenutka primitka podataka. Rješenje pošaljite zajedno s izvještajem i kodom u jednom arhivu ili putem repozitorija (GitHub, GitLab...).

Vicuna d.o.o.