

Sistem za preporuku sledeće narudžbine hrane

Članovi tima

1. Balsa Bulatovic SV37-2021
2. Vladimir Popov SV29-2021

Motivacija

Savremeni korisnici aplikacija za dostavu hrane (Glovo, Wolt) suočavaju se sa problemom prevelikog izbora i gubitka vremena na pretragu. Često biraju ista jela ili restorane, bez uzimanja u obzir trenutnog konteksta (vreme dana, vremenske prilike, dostupne promocije). Osim toga mnogi korisnici su osteljivi na određene namirnice (gluten, orašasti plodovi, mleko,...) ili se jednostavno hrane prema određenom režimu ishrane (vegetarijanski, veganski, halal, itd.). Predloženi sistem koristi pravila i znanje o navikama korisnika da bi preporučio sledeću narudžbinu. Time se korisniku olakšava proces odlučivanja i povećava zadovoljstvo, dok platforme ostvaruju veću zadržanost korisnika i promociju restorana, a samim tim i veći profit.

Pregled problema

Problem preporuke sledeće narudžbine hrane podrazumeva uzimanje u obzir više faktora: istorija narudžbina, vreme dana, vremenske prilike, dostupni popusti, budžet korisnika, restrikcije u ishrani, i inicijalne preferencije korisnika. Na tržištu postoje rešenja za preporuke uglavnom u okviru aplikacija za dostavu poput Wolt-a i Glovo-a, ali većina se oslanja na klasične algoritme filtriranja i ne pruža transparentnost zaključivanja. Skup preporuka takvih aplikacija je ograničen i uglavnom se svodi isključivo na istoriju poručivanja ili na to koji restoran plati više da bi bio prisutniji u preporukama. Naš sistem uvodi pravila bazirana na znanju, forward-chaining sa više nivoa, accumulate funkcije, CEP i backward chaining, kako bi se što transparentije, fleksibilnije i objektivnije donela preporuka sledeće narudžbine.

Metodologija rada

Uloge

1. Korisnik – želi da dobije preporuku za sledeću narudžbinu hrane.
2. Sistem (Drools engine) – na osnovu pravila, događaja i znanja o kontekstu predlaže narudžbine.

Ulaz u sistem:

Korisnik i kontekst definišu sledeće ulazne podatke:

- Istorija narudžbina (jela, restorani, cena, vreme)
- Restrikcije (alergije, režimi ishrane, itd.)
- Preferencije (tip kuhinje, sastojci, tolerancija na ljuto, npr. ljubitelj italijanske kuhinje i ne preferira majonez)
- Grupa korisnika (korisnik može kreirati grupnu narudžbinu)
- Budžet i tipične potrošnje
- Vreme dana i dan u nedelji
- Vremenske prilike (toplo/hladno, kiša, sneg)
- Promocije i popusti restorana
- "Surge" cene i dostupnost dostave

Izlaz iz sistema:

- Lista preporučenih jela/restorana sa obrazloženjem
- Notifikacije o najpovoljnijim ponudama (popusti, vremenski specifične akcije)
- Izveštaji o navikama korisnika (koliko puta naručuje određenu kuhinju, prosečna potrošnja)

Baza znanja

Forward-chaining

1. Ako je temperatura $< 5^{\circ}\text{C}$ i korisnik često naručuje supe → kreiraj Craving(SOUP).
2. Ako postoji craving i promocija kuhinje → formiraj CandidateCuisine.
3. Ako postoji candidate cuisine i jelo ispunjava budžet i restrikcije → filtriraj listu.
4. Na osnovu restrikcija korisnika formiraj listu jela koja se mogu preporučiti korisniku
5. Ako je jelo pripada preferiranoj kuhinji korisnika , formiraj pozitivan ScoreModifier
6. Ukoliko je doba dana jutro, formiraj pozitivan ScoreModifier doručaka
7. Ukoliko jelo ima veći nivo ljutine od korisnikove preferirane tolerancije, formiraj negativan ScoreModifier tog jela
8. (Ukoliko je porudžbina grupna)
 - a. Za svakog korisnika u grupnoj porudžbini izabрати jelo sa najvećim skorom – formirati TopDishForUser.
 - b. Ukoliko više korisnika deli TopDishForUser, preporučiti to jelo
 - c. Ukoliko prethodno nije aktivirano, naći najpopularniju vrstu kuhinje u toj grupi – TopCuisineForGroup
 - d. Na osnovu TopCuisineForGroup, pronaći jelo iz te kuhinje sa najvećim zbirom individualnih skorova

9. Ako su jela filtrirana → kreiraj Recommendation.

Accumulate primer

- Ako prosečna cena poslednjih 10 narudžbina ≤ 1200 RSD → filtriraj jela skuplja od $1.5\times$ proseka.
- Ako broj narudžbina iz nekog restorana > 5 u poslednjih mesec dana → povećaj skor tog restorana.
- Ako korisnik u proseku naručuje bar 2 puta nedeljno → preporuči promotivne pakete (combo ponude).
- Ukoliko je korisnik u prošlosti naručivao isto jelo (više od jednom), povećaj rangiranje jela
- Ukoliko korisnik pravi grupnu porudžbinu i ne može se jednoglasno odrediti preporučeno jelo - predost pri preporuci imaju jela koja pripadaju najomiljenijem tipu kuhinje među grupom
- Akumulacija svih ScoreModifier-a kako bi se izračunao konačan skor među preporučenim jelima

CEP primeri

1. Ako je vreme 18h–23h i postoji promocija za italijansku hranu u poslednjih 30 minuta i korisnik je u poslednjih 7 dana naručio italijansku kuhinju → preporuči pizzu/pastu.
2. Ako postoji SurgeEvent(multiplier > 2.0) u poslednjih 10 minuta i WeatherEvent(rain=True) → isključi restorane bez besplatne dostave.
3. Ako se pojavi sekvenca PromotionEvent → SurgeEvent → PromotionEvent u roku od 60 minuta za isti restoran → pojačaj skor tog restorana.
4. Ukoliko korisnik ostavi negativan feedback na npr. italijanska jela tri puta onemogućiti preporučivanje italijanskih jela narednih 7 dana

Template primeri

1. Doba dana:

- Ako je vreme između 7h i 11h → preporuči doručak.
- Ako je vreme između 11h i 15h → preporuči ručak.
- Ako je vreme između 18h i 23h → preporuči večeru.

Backward Chaining

1. Upit objašnjenja: “Zašto je sushi preporučen?” → sistem vraća lanac dokaza (petak večer → craving riba → promocija japanske hrane → istorija sushi narudžbi).
2. Upit sa uslovom: “Koje preporuke imaju ocenu restorana > 4.5 ?” → backward chaining filtrira preporuke.

3. Agregirani upit: “Koliko puta su poslednjih mesec dana preporuke bile vege jela?” → backward chaining vraća broj i procenat.

Klasni dijagram sistema

Entiteti:

- Group
- UserProfile
- Order
- Context
- WeatherEvent / PromotionEvent / SurgeEvent / UserFeedbackEvent
- Dish
- Restaurant
- Recommendation

