

Fotbollsbetting med hjälp av dataanalys och AI

Skapandet av en interaktiv chatbot



Camilla Månsson

EC Utbildning

Projekt Data Science

Oktober 2024

Abstract

This report describes the development of a football betting and chat application that integrates machine learning models to predict match outcomes and identify value bets. A Random Forest Classifier and a Voting Regressor were constructed, however, both models fell short of expectations, indicating a need for further refinement and development before they can be applied in real-life betting scenarios. Additionally, a chatbot was implemented to answer user queries about football statistics and match results, with potential for expanded functionality. The project highlights the challenges of building an end-to-end pipeline and the importance of making informed decisions. Future work will aim to improve the model and enhance the chatbot's features.

Innehållsförteckning

Abstract	2
1 Inledning.....	1
1.1 Syfte	1
2 Teori.....	2
2.1 Random Forest.....	2
2.2 Voting Regressor	3
3 Metod	4
3.1 Agil arbetsmetodik.....	4
3.1 Systemarkitektur	4
3.2 Datainsamling	4
3.3 Databas	4
3.4 EDA.....	5
3.5 Modeller.....	6
3.6 LLM.....	6
3.7 Frontend.....	6
4 Resultat och Diskussion.....	8
5 Slutsatser	11
6 Självutvärdering.....	12
Källförteckning.....	13

1 Inledning

Att tippa på fotboll i hopp om att ta hem storkovan är inget nytt, utan har snarare en lång historia. Redan 1934 fick det privata bolaget Tipstjänst tillstånd från svenska staten att starta sin vadhållningsverksamhet och lanserade Stryktipset som än idag är en välkänd produkt hos Svenska Spel (Wikipedia, n.d.). Sedan dess har bettingmarknaden vuxit enormt och idag kan man spela på nästan vad som helst hos otaliga spelbolag.

Under de senaste åren har dataanalys och AI-teknologier utvecklats till kraftfulla verktyg, vilket har förändrat många branscher. Inom bettingvärlden förlitar sig många spelare fortfarande på sin magkänsla när de placerar sina vad. Men den nya teknologin kan spela en avgörande roll genom att erbjuda datadrivna insikter som kan förbättra träffsäkerheten i deras tips.

I vårt projekt har vi därför kombinerat traditionell sportsbetting med moderna AI-verktyg. Inspirationen kom från idén att försöka "slå huset", en vanlig dröm i spelvärlden där man vet att huset alltid har en fördel. Med hjälp av maskininlärning ville vi utforska hur man kan förbättra sina odds och samtidigt utveckla en chat-bot som användarna kan ställa frågor till och få tips från.

1.1 Syfte

Syftet med detta projekt var att utveckla en chatbot och sportsbettingapplikation som kan analysera fotbollsdata från flera europeiska ligor och generera odds för kommande matcher. Chatboten ska dessutom kunna svara på frågor om matcher och fotbollsstatistik, där resultaten presenteras på ett användarvänligt sätt i en webbapplikation. Genom att kombinera avancerad datainsamling och prediktionstekniker strävar projektet efter att optimera sportsbettingprocessen och förbättra användarupplevelsen.

Denna rapport syftar till att beskriva utvecklingsprocessen, analysera resultaten och utvärdera hur väl chatboten och applikationen uppfyller sitt mål att förbättra sportsbettingupplevelsen genom användning av dataanalys och AI.

För att uppfylla syftet så kommer följande frågeställningar att besvaras:

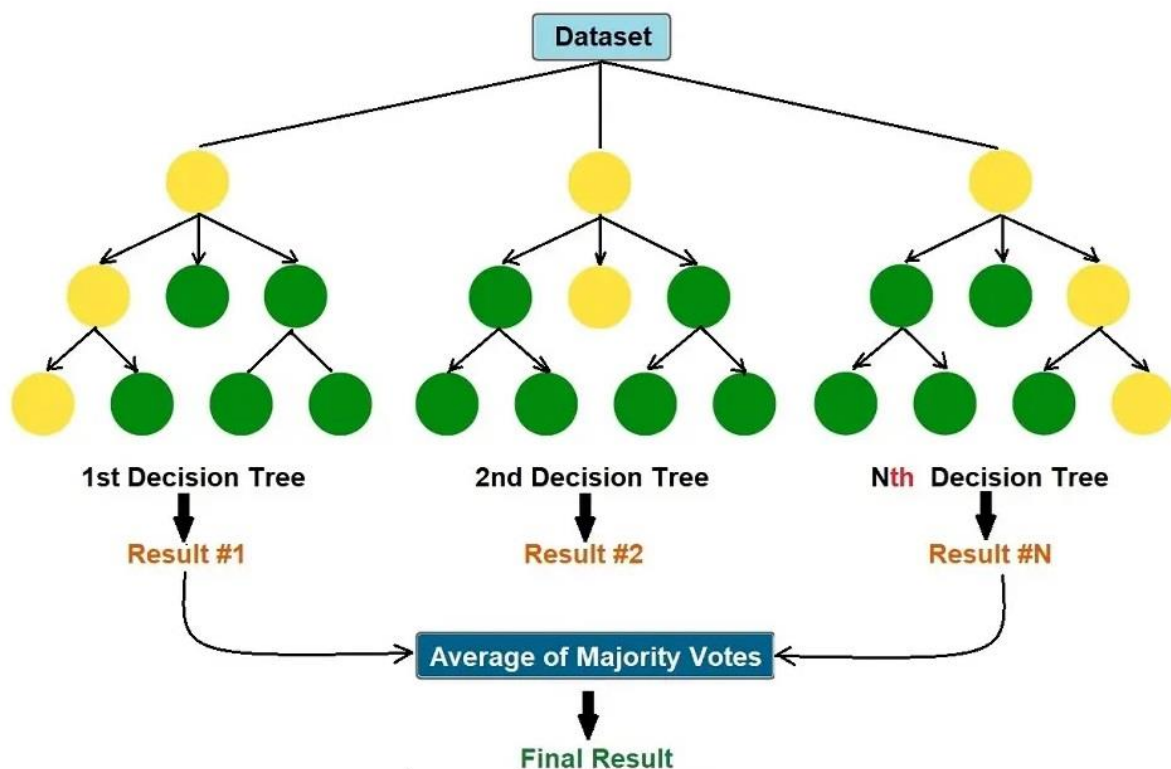
1. Hur korrekt är vår modell i att prediktera fotbollsmatcher jämfört med verkliga odds?
2. Kan vi bygga en chatbot som kan svara på frågor om fotbollsstatistik och kommande matchresultat och därigenom ge användaren en bättre upplevelse?

2 Teori

I denna del av rapporten beskrivs de modeller som använts för utvecklingen av vår chatbot.

2.1 Random Forest

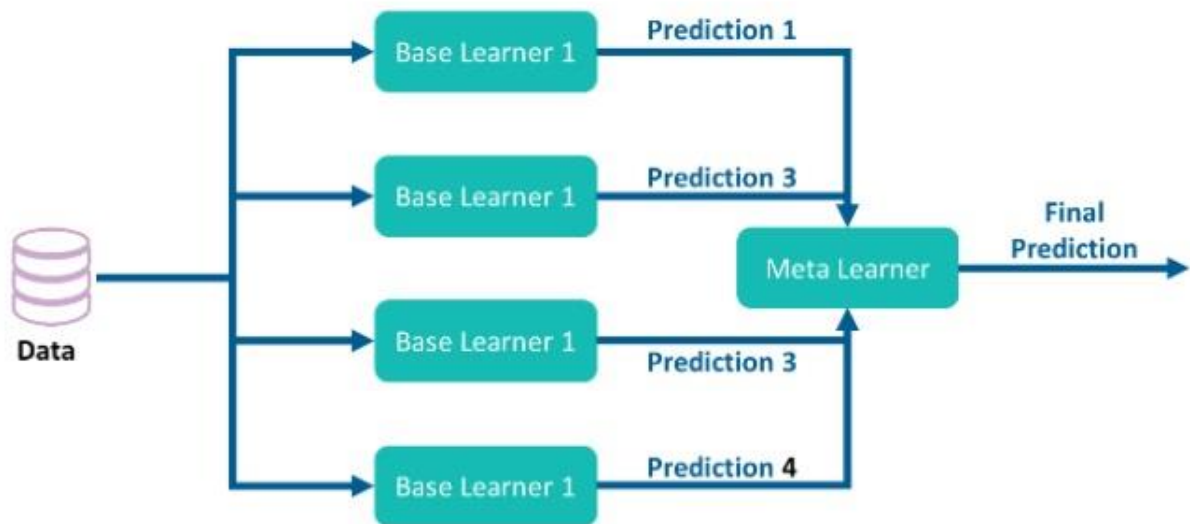
Random Forest är en ensemblemetod inom maskininlärning som bygger på att skapa ett stort antal beslutsträd där varje träd tränas på en slumpmässig delmängd av träningsdatan. För att göra en prediktion röstar alla träden och majoritetsbeslutet blir modellens slutgiltiga prediktion. Detta minskar risken för överanpassning och gör modellen robust.



Figur 1. Arkitekturen av en Random Forest Modell

2.2 Voting Regressor

Voting Regression är en ensemblemetod där flera regressionsmodeller kombineras för att förbättra noggrannheten i prediktionerna. Genom att ta ett genomsnitt av prediktionerna från dessa modeller kan metoden bidra till en mer stabil och precis prediktion. Detta uppnås genom att utnyttja styrkorna hos olika regressionsmodeller samtidigt som effekten av enskilda modellers svagheter minskas.



Figur 2. Arkitekturen av en Voting Regressor Modell.

3 Metod

Detta avsnitt beskriver hur vår grupp arbetade med projektet, inklusive de verktyg och teknologier som användes i olika delar.

3.1 Agil arbetsmetodik

I detta projekt har vi arbetat enligt agil metod, vilket innebar att vi delade upp helheten i mindre, hanterbara uppgifter som fördelades mellan gruppens medlemmar. Detta arbetssätt hjälpte oss bla att snabbare identifiera problem. Trots att vi arbetade med våra individuella delar hade vi kontinuerlig kommunikation via en gruppchatt i Teams och regelbundna digitala möten för att diskutera både framsteg och svårigheter. Vid mötena gjorde vi en avstämning för att omvärdera prioriteringar och justera tidsplanen om det behövdes.

Vår projektplanering såg ut som följer:

- Vecka 1-2: Brainstorming, datainsamling, strukturering av data, förberedelser i Anything LLM och Bubble, informationssök och regelbundna avstämningar.
- Vecka 3: EDA. Implementering av API-anrop i webapplikationen och fler funktioner inlagda, utveckling av chatbot och påbörja prediktionsmodellen.
- Vecka 4: Träna prediktionsmodellen. Integration av chatboten och förbättringar i webapplikationen.
- Vecka 5: Förberedelse inför presentation och rapportinlämning.

3.1 Systemarkitektur

Systemarkitekturen för projektet följer en klient-server-modell. Frontend-delen är utvecklad i Bubble och kommunicerar med backend via API och en MySQL-databas. Frontend ger användarna möjlighet att interagera med fotbollsstatistik och odds genom ett användarvänligt gränssnitt medan backend hanterar datainsamling, bearbetning och lagring. Backend byggdes med Python-skript som hämtade data från två API och sparade den i databasen. Arkitekturen är modulär, vilket innebär att varje del (frontend, backend, databas) kan utvecklas och underhållas oberoende av varandra. Detta ökar systemets flexibilitet och kan vara viktigt för framtida funktionalitet.

3.2 Datainsamling

Vi hämtade data från följande fotbollsligor: Premier League, Bundesliga, La Liga, Ligue 1, Serie A och Allsvenskan. Datainsamlingen gjordes via två olika API. The-odds-api valdes för inhämtning av odds på grund av dess täckning av den svenska marknaden och tillgång till historisk data. Api-football användes för att hämta fotbollsstatistik tack vare dess omfattande täckning av olika ligor och möjlighet att inkludera spelarstatistik. Ursprungligen var planen att använda ett enda API för både statistik och odds men på grund av bristande funktionalitet hos leverantörerna beslutades det att använda två olika API. Vi hämtade matchstatistik för de senaste tre åren och programmet som användes var Python via Visual Studio Code. För att hålla databasen uppdaterad skapades skript i Python för automatiserad datainsamling.

3.3 Databas

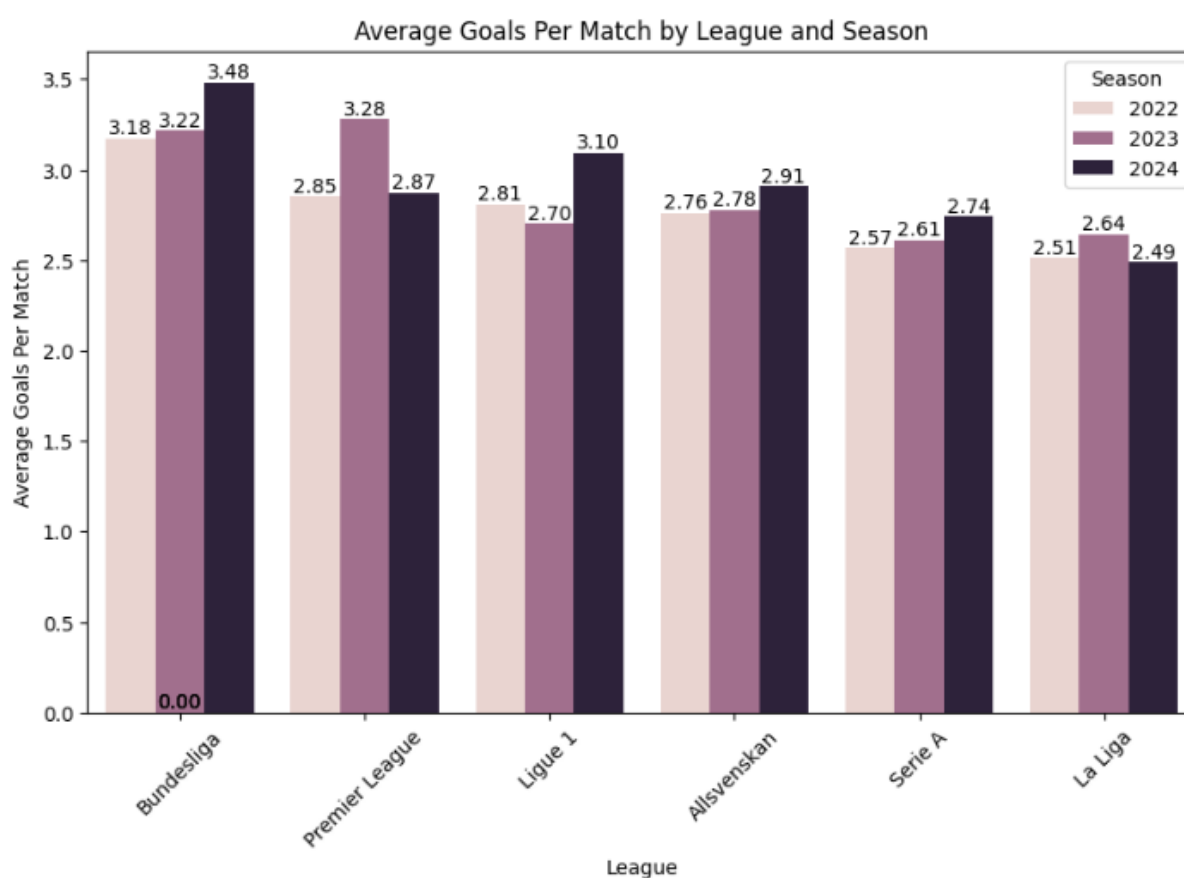
MySQL användes som databasplattform. Anledningen till att den valdes var på grund av dess breda stöd för både Python och API samt att det är enkelt att integrera i webbaserade applikationer. Detta gör det enkelt att koppla samman statistiken med chatbotten och möjliggör användning av molntjänster. Databasen strukturerades för att koppla samman lag, matcher, statistik och odds för fotbollsligorna.

3.4 EDA

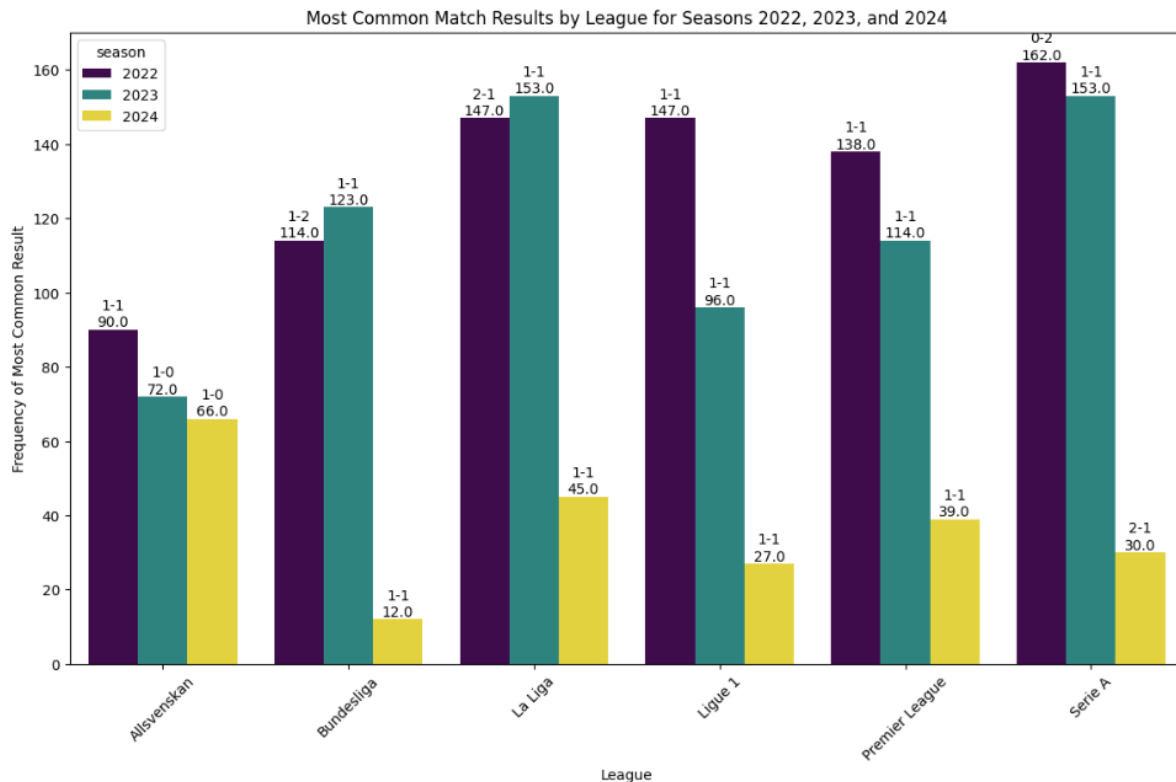
Exploratory Data Analysis (EDA) handlar om att utforska och sammanfatta viktiga egenskaper i vår data med hjälp av visuella och statistiska metoder. I vårt projekt har vi använt en MySQL-databas med mycket information om fotbollsmatcher, inklusive matchresultat, antal hörnor, bollinnehav, odds och andra viktiga parametrar.

Under EDA-processen har vi rensat och förberett datan för analys. Vi har hanterat saknade värden och identifierat avvikelser. Genom olika visualiseringsverktyg har vi skapat diagram och grafer för att se mönster och trender i datan. Detta har hjälpt oss att hitta de viktigaste variablerna för att senare bygga våra modeller.

Nedan visas två exempel på visualiseringar. Ett diagram visar snittet av mål per match och det andra diagrammet visar det vanligaste matchresultatet.



Figur 3. Visualisering över antalet mål som gjorts i snitt per match, fördelat över de olika ligorna och uppdelat per säsong.



Figur 4. Visualisering över det vanligaste slutresultatet fördelat per liga per säsong.

3.5 Modeller

I projektet har vi arbetat med två olika modeller för att förutse fotbollsmatchers utfall, Random Forest och Voting Regressor. Dessa modeller valdes utifrån deras förmåga att hantera komplex data och ge robusta resultat. För vidare information om valda modeller se under avsnittet "Teori".

3.6 LLM

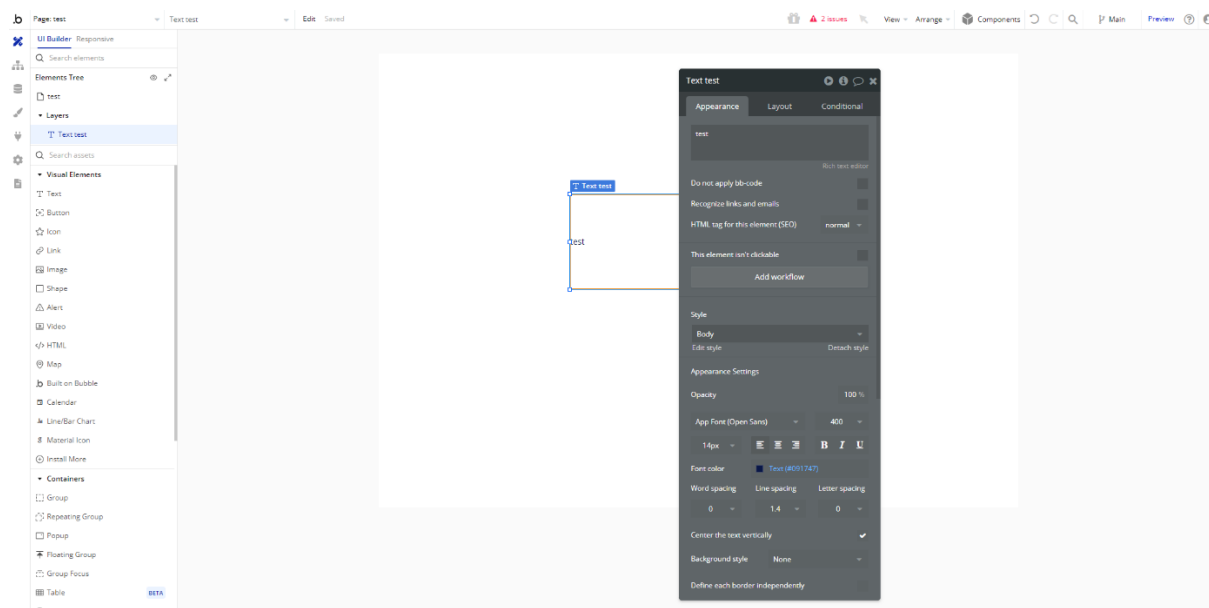
För att förbättra användarupplevelsen i applikationen och ge dynamiska svar integrerades en Large Language Model (LLM). LLM används för att förstå och svara på användarfrågor baserat på information om fotbollslag, spelare och matcher från vår databas. Den LLM som vi ansåg vara mest lämplig för vårt projekt var Google BERT vilket är en förkortning av Bidirectional Encoder Representations from Transformers. BERT använder en nätverksbaserad teknik för att bearbeta språk. Det var här vårt chatbotnamn "BertBot" skapades.

3.7 Frontend

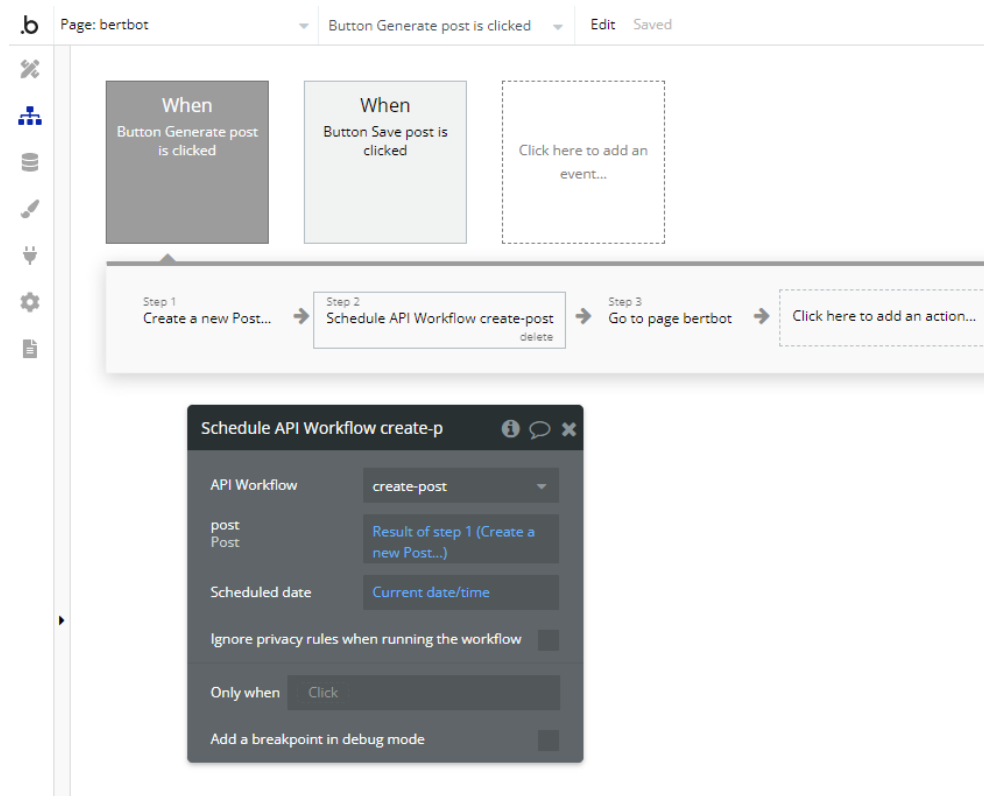
Vid utvecklingen av frontend-delen i vår sportsbetting- och chattapplikation för fotboll användes plattformen Bubble för att skapa en användarvänlig webbapplikation. Användarna kan enkelt navigera för att hitta information om fotbollsligor och matcher samt interagera med våra chatbotar, BertBot och SoccerGPT. Bubble valdes på grund av sin no-code-funktionalitet vilket möjliggjorde snabb utveckling utan behov av kodning.

Plattformens flexibilitet och integrationsmöjligheter, inklusive plugins för MySQL och API-anrop, gjorde det möjligt att effektivt hämta och visa fotbollsstatistik från databasen samt implementera chattfunktionalitet.

Nedan visas två skärmdumpar av användargränssnittet i Bubble.



Figur 5. En del av arbetsvyn i Bubble.



Figur 6. Arbetsvy från Bubble där en workflow för om användaren trycker på en viss knapp.

4 Resultat och Diskussion

Vid utvärdering av våra modeller användes korsvalidering med time-series-split med default score. Random Forest Classifier utvärderades med accuracy och Voting Regressor utvärderades med R^2 . Resultaten sammanfattas i tabell 1:

Typ av modell	Mått	Resultat
Random Forest	Mean Accuracy	0.87
Voting Regressor	Mean R^2	1.00

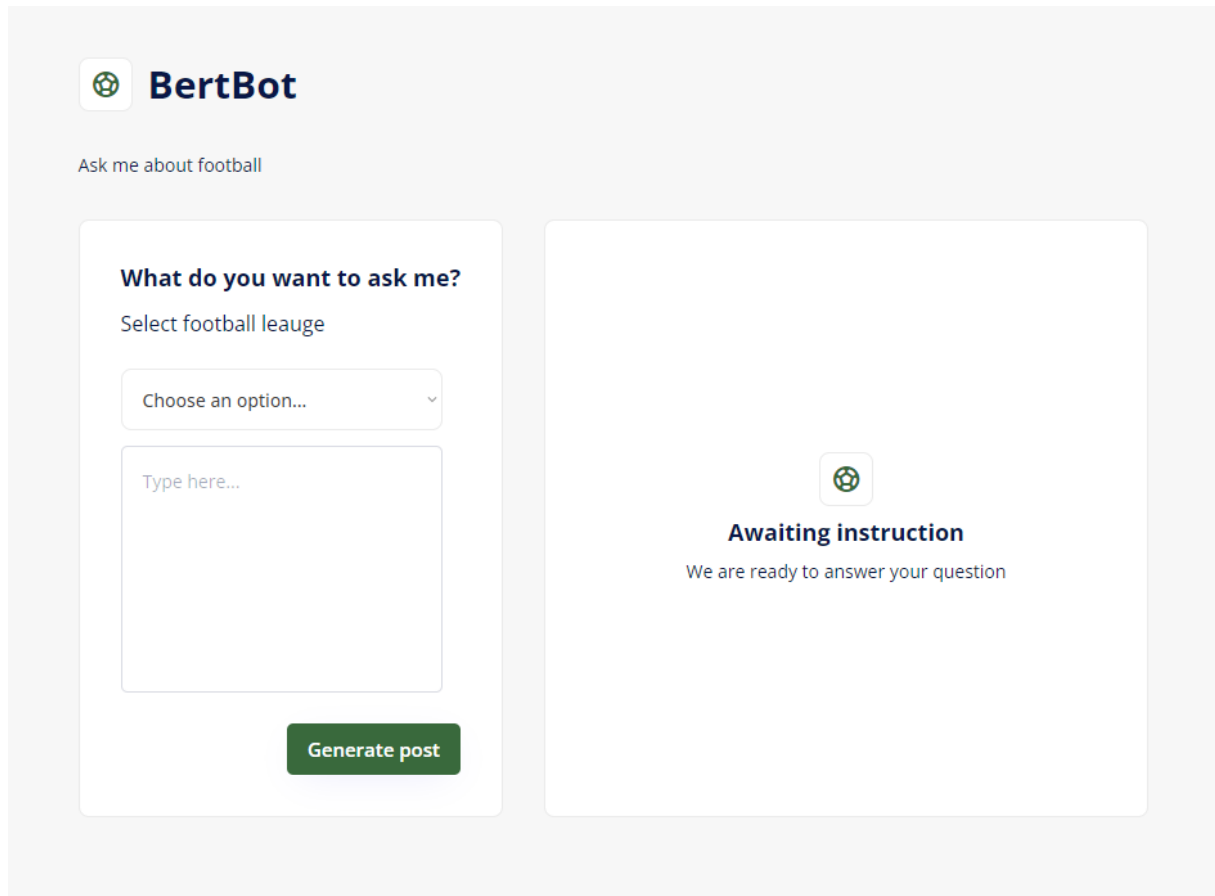
Tabell 1: Resultat för våra valda modeller.

Trots att Random Forest visade en hög accuracy och Voting Regressor ett perfekt R^2 -värde är vi inte helt nöjda med resultaten. Detta beror delvis på förberedelsen av datan, som spelar en avgörande roll i modellernas prestanda. Random Forest-modellen identifierar satsningar som "value bets" men kalibrering av sannolikheter är avgörande. Precision i förutsägelserna är inte tillräckligt, odds måste också återspegla de faktiska sannolikheterna för olika utfall.

För att skapa pålitliga odds behöver vi förbättra våra modeller så att sannolikheterna stämmer överens med verkligheten. Forskning visar att modeller som Support Vector Machines (SVM) och logistisk regression, som fokuserar mer på kalibrering än enbart noggrannhet, ofta ger bättre resultat i bettingkontexter (Walsh & Joshi, 2024). Framtida arbete bör därför inriktas på att implementera mer kalibrerade modeller och förbättra den data som används för att öka träffsäkerheten i satsningarna.

Vi har byggt en databas och utvecklat ett användargränssnitt i Bubble men det saknas fortfarande en komplett pipeline som integrerar alla systemkomponenter, inklusive databasen, modellerna och chatboten. För att få ett bättre och mer automatiserat system behöver vi förbättra våra skript och processerna för att uppdatera modellerna. Det är viktigt för att få datakällorna och modellresultaten att fungera bra tillsammans.

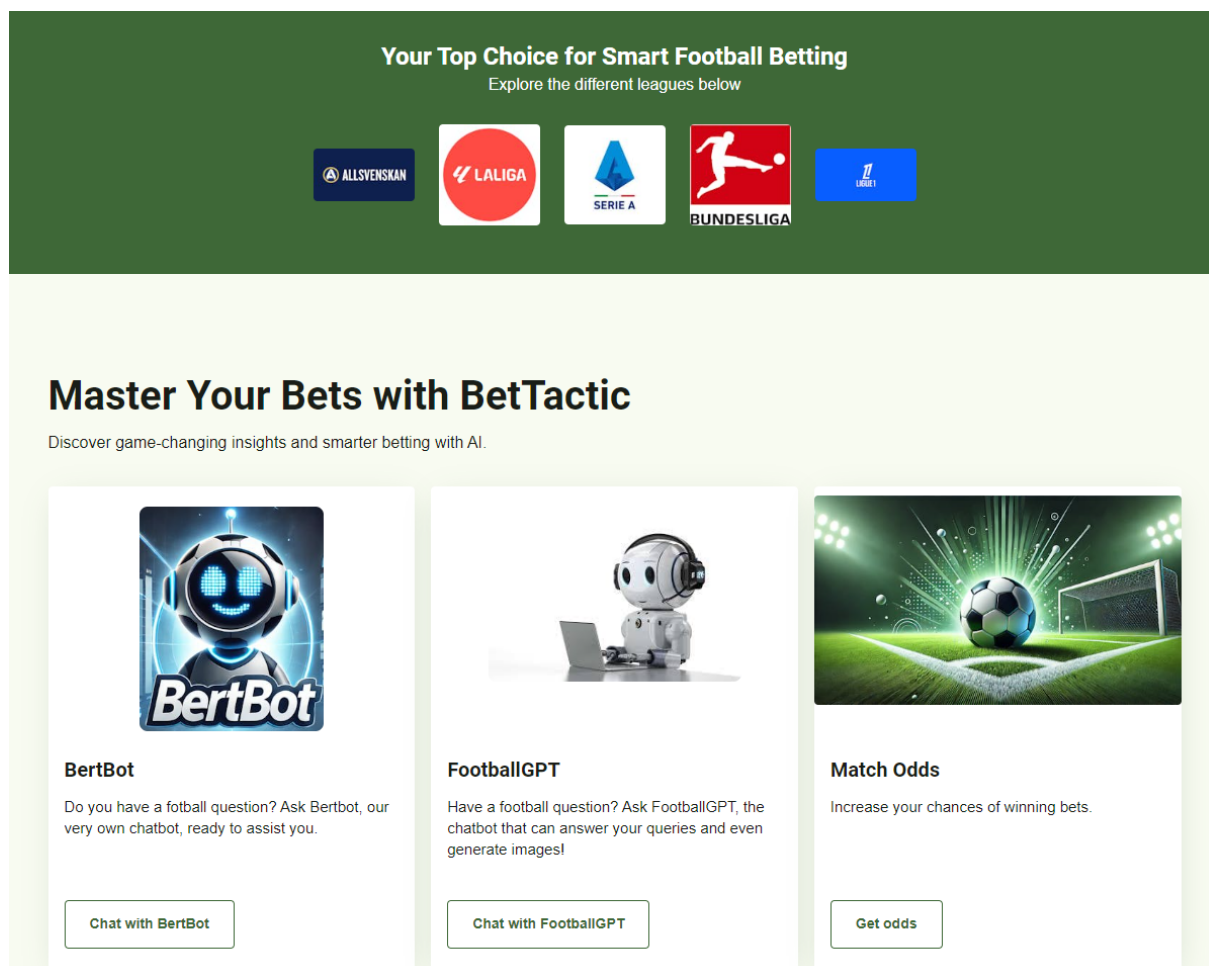
Den grundläggande versionen av chatboten fungerar som den ska och kan svara på frågor om fotboll och sannolikheter utifrån vår modell. För att göra den mer användbar kan vi lägga till realtidsdatainsamling och ge mer avancerade, kontextuella svar. Genom att finjustera hur vi formulerar frågor och justera modellen kan vi förbättra chatbotens förmåga att ge bettingrekommendationer och hantera mer komplexa frågor. Nedan visas en skärmdump av hur det ser ut när en användare vill ställa en fråga till vår chatbot.



The screenshot shows the BertBot web application interface. At the top left, there is a logo consisting of a green soccer ball icon and the text "BertBot". Below the logo, the text "Ask me about football" is displayed. The interface is divided into two main sections. The left section, titled "What do you want to ask me?", contains a prompt "Select football league" followed by a dropdown menu with the text "Choose an option..." and a small downward arrow. Below the dropdown is a text input field with the placeholder "Type here...". At the bottom of this section is a green button labeled "Generate post". The right section is titled "Awaiting instruction" and contains the text "We are ready to answer your question". A small soccer ball icon is positioned above the "Awaiting instruction" text.

Figur 7. Utklipp från webbapplikationen som visar hur det ser ut när man vill interagera med vår chatbot BertBot.

Bubble har visat sig vara en praktisk plattform för att bygga användargränssnitt utan kod. Trots att det är en drag-and-drop-lösning har integrationen med databasen och modellerna varit tidskrävande. Vissa delar, som en helt integrerad pipeline mellan chatboten och modellen, saknas fortfarande. Chatboten, som kommunicerar via ett API med ChatGPT, fungerar dock enligt plan och utgör en stabil grund för framtida förbättringar. Nedan visas en skärmdump från vår applikation i Bubble.



Figur 8. Bild från en del av webbapplikationen som bla visar våra chatbotar.

5 Slutsatser

Svar på fråga 1:

Vår modell, Random Forest Classifier, uppnår en accuracy på 87%, vilket indikerar en viss förmåga att förutsäga fotbollsmatcher. Trots detta är detta mått inte tillräckligt för att bedöma modellens kapacitet att prediktera verkliga odds. För att förbättra modellens prestanda är det viktigt att vi fokuserar på att organisera och förbereda datan bättre. Det kan också vara värt att utforska alternativa modeller, som logistisk regression eller Support Vector Machines (SVM), som kan ge mer tillförlitliga prediktioner.

Svar på fråga 2:

Vår chatbot har en grundläggande funktionalitet och kan svara på frågor baserat på den data och de modeller som vi har utvecklat. För att öka chatbotens användbarhet och ge en bättre upplevelse för användarna, bör vi överväga att implementera realtidsuppdateringar av fotbollsstatistik och utöka funktionaliteten med mer kontextbaserade svar. Genom att förbättra både prompt engineering och finjustera våra modeller kan vi göra chatboten mer kapabel att ge insikter om fotbollsstatistik och kommande matchresultat.

Projektet genomfördes med en agil metodik, vilket möjliggjorde anpassning till nya insikter och utmaningar under arbetets gång. Trots ambitionerna för projektet har oväntade problem förlängt tidsramarna.

För att optimera modellens prestanda skulle en mer fokuserad strategi ha varit fördelaktig, exempelvis genom att koncentrera oss på en specifik liga och förbättra datakvaliteten. Den nuvarande modellen, Random Forest Classifier, har inte uppnått de förväntade resultaten och det råder osäkerhet kring dess förmåga att fungera som en vinstdrivande value-bet-modell.

För att förbättra resultaten rekommenderas noggrannare hantering av saknade värden och utforskning av alternativa modeller, såsom neurala nätverk eller linjär regression. Dessa åtgärder kan öka precisionen i modellens prediktioner och möjliggöra mer pålitliga bettingrekommendationer.

6 Självutvärdering

1. Den största utmaningen har varit att få hela teamet att fungera och att alla medlemmar ska vara delaktiga. Jag vill lyfta fram samarbetet i gruppen, som tyvärr inte har fungerat så smidigt som jag hade hoppats, vilket har påverkat både min och gruppens prestation. Jag har försökt att aktivt engagera alla i diskussioner och att dela med oss av material, men det har varit en utmaning.

I mitt eget arbete har jag försökt sätta mig in i de andra gruppmedlemmarnas ansvarsområden samtidigt som jag arbetar med mina egna uppgifter. Att förstå helheten har inte varit ett problem, men att göra detta på en djupare nivå, utan att fastna i tekniska detaljer, har varit svårt.

När det kommer till frontend-delen har det också varit utmanande att avgöra exakt vad som behövs och hur man ska dra gränsen för funktioner, knappar och färger, med tanke på att det finns oändliga möjligheter att lägga till olika element.

2. Jag anser att jag uppnått ett godkänt genom att ha genomfört projektet med tidigare verktyg och tekniker samtidigt som jag lärt mig nya. Jag har tillsammans med gruppen uppnått ett fungerande system och samtidigt förstått var förbättringar behövs.
3. Att arbeta i team, särskilt med medlemmar som man inte har valt själv, har varit en värdefull erfarenhet. Det speglar verkligheten i arbetslivet, där man ofta behöver samarbeta med olika personligheter och kompetenser för att nå gemensamma mål.

Källförteckning

Walsh, C., & Joshi, A. (2024). Machine learning for sports betting: Should model selection be based on accuracy or calibration? Department of Computer Science, University of Bath, Somerset, UK.

Wikipedia. (n.d.). *Svenska Spel*. Hämtad 31 oktober 2024, från https://sv.wikipedia.org/wiki/Svenska_Spel