



CHALMERS



# AI-baserat träningsstöd för löpning

En förstudie om genomförbarhet och strategisk implementering

Kandidatarbete vid Institutionen för fysik och astronomi

Christopher Boissier, Cornelia Ivanovic, Fabian Johannesson, Daniel Melin Romstad, Teodor Tambur, Ebba Thenander

---

Institutionen för fysik och astronomi

CHALMERS TEKNISKA HÖGSKOLA

Göteborg 2026

[www.chalmers.se](http://www.chalmers.se)



# AI-baserat träningsstöd för löpning

En förstudie om genomförbarhet och strategisk implementering

## *Författare*

Christopher Boissier, TKAUT

Cornelia Ivanovic, TKAUT

Fabian Johannesson, TKAUT

Daniel Melin Romstad, TKDAT

Teodor Tambur, TKAUT

Ebba Thenander, TKTFY



**CHALMERS**

AI-baserat träningsstöd för löpning

En förstudie om genomförbarhet och strategisk implementering

CHRISTOPHER BOISSIER, CORNELIA IVANOVIC, FABIAN JOHANNESSON, DANIEL  
MELIN ROMSTAD, TEODOR TAMBUR, EBBA THENANDER

© CHRISTOPHER BOISSIER, CORNELIA IVANOVIC, FABIAN JOHANNESSON, DANIEL  
MELIN ROMSTAD, TEODOR TAMBUR, EBBA THENANDER, 2026.

Handledare: Magnus Karlsteen, Institutionen för fysik och astronomi.

Uppdragsgivare: Janne Wanhainen, Svenska Friidrottsförbundet.

Examinator: Jan Swenson, Institutionen för fysik och astronomi.

Kandidatarbete 2026

Institutionen för fysik och astronomi

Chalmers Tekniska Högskola

SE-412 96 Göteborg

Telefon +46 31 772 1000

Omslagsbild: 'Running' hämtad 8 maj 2026 från Encyclopædia Britannica ImageQuest.

Skriven i L<sup>A</sup>T<sub>E</sub>X

Göteborg 2026

AI-baserat träningsstöd för löpning

En förstudie om genomförbarhet och strategisk implementering

CHRISTOPHER BOISSIER, CORNELIA IVANOVIC, FABIAN JOHANNESSON, DANIEL  
MELIN ROMSTAD, TEODOR TAMBUR, EBBA THENANDER

Institutionen för fysik och astronomi

Chalmers Tekniska Högskola

## Sammanfattning

Inom friidrottens grenar medel- och långdistanslöpning är personliga träningsupplägg en viktig faktor för idrottslig framgång, men skapandet och justeringen av dessa är en tidskrävande process för tränare. Tränares metodik varierar ofta beroende på erfarenhet och filosofi, vilket ställer höga krav på flexibilitet i digitala verktyg. Detta arbete utforskar hur avancerade AI-modeller kan användas för att automatisera delar av denna beslutslogik, i syfte att avlasta tränare administrativt och möjliggöra ett ökat fokus på teknisk analys och aktiv coaching. Genom en inledande behovsanalys har en träningsplattform utvecklats som fungerar som ett AI-baserat beslutsstöd. Systemet integrerar en regressionsmodell för prediktion av löptempo, baserad på individuell Garmin-data, med en modell baserad på hämtningsförhöjd generation som skapar strukturerade träningsprogram utifrån utbildningsmaterial och idrottarens profil. För att möjliggöra individanpassade justeringar har en stor språkmodell (LLM) finjusterats på specifik tränar-logik för att hantera utövares dagsform och status. Resultatet är en webbaserad prototyp som demonstrerar hur AI och maskininlärning kan komplettera mänsklig expertis inom medel- och långdistanslöpning.

## AI-based Running Support

A Feasibility Study of Generative Training Support for Running

### Abstract

In the field of middle- and long-distance running in athletics, personalized training plans are an important factor for athletic success, yet the creation and adjustment of these plans remain a time-consuming process for coaches. Coaching methodologies often vary based on individual experience and philosophy, placing high demands on the flexibility of digital tools. This thesis explores how advanced AI models can be utilized to automate parts of this decision-making logic, aiming to reduce the administrative workload for coaches and enable an increased focus on technical analysis and active coaching. Through a needs analysis, a training platform has been developed to serve as an AI-based decision support system. The system integrates a regression model for predicting running pace, based on individual Garmin data, with a model based on retrieval-augmented generation for generating structured training schedules from educational material and the athlete's profile. To enable individualized adjustments, a Large Language Model has been fine-tuned on specific coaching logic to handle athlete's current physical condition. The result is a web-based prototype demonstrating how AI and machine learning can complement human expertise within middle- and long-distance running.

## Författarnas tack

Detta kandidatarbete, *AI-baserat träningsstöd för löpning*, har utförts under våren 2026 vid Institutionen för Fysik på Chalmers tekniska högskola. Arbetet har genomförts av sex studenter i samarbete med Svenska Friidrottsförbundet.

Vi vill rikta ett särskilt tack till vår handledare Magnus Karlsteen för värdefull feedback, vägledning och uppmuntran under projektets gång. Vi tackar även vår examinator Jan Swenson för stöd i den akademiska processen.

Ett stort tack riktas till Svenska Friidrottsförbundet för förtroendet att utföra detta projekt, med ett särskilt omnämmande till Janne Wanhainen för hans engagemang och det material som tillhandahållits. Vi vill även tacka de tränare inom friidrotten som genom intervjuer och enkäter bidragit med sin expertis.

Slutligen vill vi tacka alla de som bidragit med träningsdata till projektet. Ett speciellt tack till Johan Lindh och hans löpgrupp för deras ovärderliga hjälp i flera av arbetets skeden.

Christopher Boissier, Cornelia Ivanovic, Fabian Johannesson, Daniel Melin Romstad, Teodor Tambur, Ebba Thenander, Göteborg, maj 2026

## Användning av AI-verktyg i projektet

I detta kandidatarbete har artificiell intelligens (AI) haft två roller: dels som ett centralt undersökningsobjekt i utvecklingen av projektets träningsplattform, och dels som ett stödverktyg i arbetsprocessen. Nedan beskrivs hur AI-verktyg har hanterats.

- **AI-verktyg och omfattning:** Stora språkmodeller, främst ChatGPT (OpenAI), Gemini (Google) och Claude (Anthropic), har utgjort en väsentlig del av utvecklingsarbetet. Detta inkluderar generering av kod för projektet samt implementering av plattformens webbaserade användargränssnitt. Verktygen har även använts för textbearbetning, språklig granskning av rapporten och sammanställning av intervjuer (där även Copilot använts i begränsad omfattning).
- **Ansvar och verifiering:** Projektgruppen har agerat arkitekter och granskare för allt innehåll. All kod som genererats av AI, inklusive hemsidans funktionalitet och den matematiska logiken i analysmodellerna, har testats, validerats och vid behov korrigerats manuellt. Projektgruppen tar fullt ansvar för det slutgiltiga resultatet i rapport, bilagor och tillhörande kodarkiv.

# Innehåll

<b>1</b>	<b>Inledning</b>	<b>1</b>
1.1	Syfte och mål . . . . .	2
1.2	Avgränsningar . . . . .	2
<b>2</b>	<b>Teori</b>	<b>3</b>
2.1	Träningslära . . . . .	3
2.2	Regressionsbaserad tempomodell . . . . .	4
2.2.1	Applikationer med löpdata . . . . .	4
2.2.2	Statistisk modellering . . . . .	4
2.3	AI-baserad träningsplanering . . . . .	5
2.3.1	Regelbaserade system . . . . .	5
2.3.2	Hämtningsförhöjd generation . . . . .	6
2.4	Webbapplikationer . . . . .	7
2.5	Språkmodeller och finjustering . . . . .	7
2.5.1	Transformer-arkitekturen och tokenisering . . . . .	7
2.5.2	Parameter-effektiv finjustering med LoRA . . . . .	8
<b>3</b>	<b>Metod</b>	<b>9</b>
3.1	Behovsanalys . . . . .	9
3.2	Regressionsbaserad tempomodell . . . . .	10
3.2.1	Datainsamling och urvalskriterier . . . . .	10
3.2.2	Databehandling och variabelselektion . . . . .	10
3.2.3	Statistisk modellering . . . . .	11
3.2.4	Tidsviktning och individualisering . . . . .	11
3.2.5	Modellutvärdering och jämförelse . . . . .	11
3.3	AI-baserad träningsplanering . . . . .	12
3.3.1	Hämtningsförhöjd generation . . . . .	12
3.3.2	Regelbaserade system . . . . .	13
3.4	Systemsammankoppling i webbplattform . . . . .	14
3.4.1	Utveckling av webbplattform . . . . .	15
3.4.2	Utvärdering av träningsplaneringssystemet . . . . .	16
3.5	Finjusterad språkmodell för utövaråterkoppling . . . . .	16
3.5.1	Tränarenkäter . . . . .	17
3.5.2	Analys och standardisering av enkätdata . . . . .	17
3.5.3	Val av språkmodell och optimeringstekniker . . . . .	18
3.5.4	Finjustering av språkmodellen . . . . .	18
3.5.5	Utvärdering av finjusterad modell . . . . .	19
<b>4</b>	<b>Resultat</b>	<b>20</b>
4.1	Behovsanalys . . . . .	20
4.2	Tempomodell . . . . .	20
4.3	Hämtningsförhöjd generering . . . . .	22
4.4	Träningsplaneringssystemet . . . . .	22
4.5	Finjusterad språkmodell för utövaråterkopplings . . . . .	24

<b>5</b>	<b>Diskussion</b>	<b>26</b>
5.1	Regressionsbaserad tempomodell . . . . .	26
5.1.1	Dataunderlagets begränsningar . . . . .	27
5.1.2	Modellens parametrar och fysiologisk relevans . . . . .	27
5.1.3	Tidsviktning och statistisk metod . . . . .	28
5.1.4	Källor till osäkerhet . . . . .	28
5.2	AI-baserad träningsplanering . . . . .	28
5.2.1	Regelbaserade system . . . . .	29
5.2.2	Hämtningsförhöjd generation . . . . .	29
5.3	Finjusterad språkmodell . . . . .	30
5.4	Samhälleliga och etiska aspekter . . . . .	30
<b>6</b>	<b>Slutsats</b>	<b>32</b>
	<b>Referenser</b>	<b>32</b>
<b>A</b>	<b>Intervjufrågor</b>	<b>II</b>
<b>B</b>	<b>Exempel på träningsplanering och pass</b>	<b>IV</b>
<b>C</b>	<b>Fördefinierad träningsmall</b>	<b>VII</b>
C.1	Veckomallar . . . . .	VII
C.2	Passbankens struktur . . . . .	VII
C.3	Exempel på detaljerade passbeskrivningar . . . . .	VIII

# 1

## Inledning

Löpning är ett område inom idrott där digitala träningsverktyg fått stort genomslag. Idag finns flera lösningar, såsom Garmin och Strava, där man kan registrera träningspass, samla in prestationsdata, följa sin utveckling och i vissa fall få individanpassade rekommendationer. Trots detta finns det ett behov inom svensk friidrott av mer individanpassade system som bättre stödjer tränarnas arbetssätt och coachning. Inom friidrottsgrenen löpning behöver träningen i högre grad anpassas efter prestationsnivå, träningsbakgrund och individuella mål. Kraven skiljer sig mellan aktiva inom exempelvis medel- och långdistans, både vad gäller intensitet, volym, återhämtning och träningsinnehåll [19].

Tränare inom friidrotten arbetar i stor utsträckning ideellt och har därför ofta begränsat med tid och resurser för uppgifter utöver den direkta träningsverksamheten. Därtill varierar erfarenheten och utbildningsbakgrunden inom träning och coachning, vilket kan påverka vilka förutsättningar som finns för planering, uppföljning och träningsmässig analys. Detta innebär ett behov av verktyg som inte bara samlar träningsrelaterad information, utan också kan underlätta planering, uppföljning och rådgivning. Det finns därför ett värde i lösningar som kan stödja tränarens resonemang, exempelvis genom återkoppling kring passens uppbyggnad, balansen mellan hårda och lugna dagar, progression över tid samt hur träningen förhåller sig till den aktivas situation och mål.

Mot denna bakgrund blir artificiell intelligens (AI) särskilt intressant. AI har under de senaste åren utvecklats snabbt och används i allt fler sammanhang där analys, rekommendationer och individanpassat stöd efterfrågas. Samtidigt finns fortfarande ett glapp mellan teknikens möjligheter och dess praktiska tillämpning inom idrott. Utmaningar som databegränsningar, resurskrav och svårigheter att införa avancerade modeller i verkliga verksamheter har bidragit till att utvecklingen inom området gått långsammare än den tekniska potentialen möjliggör [14]. Mycket av den befintliga användningen tycks dessutom vara inriktad på bredare tränings-sammanhang, där rekommendationerna ofta är generella och inte fullt ut anpassade till de krav som finns inom friidrottens löpgrenar. Detta skapar ett relevant utvecklingsområde i skärningspunkten mellan löpningsapplikationer, friidrottens specialiserade behov och möjligheten att använda AI som stöd i coachning och träningsplanering.

### 1.1 Syfte och mål

Syftet med projektet är att realisera en webbaserad prototyp som demonstrerar hur AI-baserat beslutsstöd kan användas för att generera och anpassa träningsupplägg i praktiken. För att uppnå detta mål är projektet uppdelat i följande delmål:

- Utveckla en regressionsmodell som, baserat på historisk träningsdata, predikterar lämpligt tempo för nya pass utifrån en angiven intensitetsfaktor.
- Utveckla en modell baserad på hämtningsförhöjd generation för att generera strukturerade träningsupplägg och integrera denna med regressionsmodellen för temposättning.
- Finjustera en stor språkmodell (LLM) i syfte att anpassa träningsrekommendationer efter en utövers dagsform, samt utvärdera denna i jämförelse med generella språkmodeller.
- Utvärdera de tekniska lösningarnas prestanda och träffsäkerhet i relation till mänsklig expertis genom kvalitativa tester.
- Utforma en webbplattform som visualiserar systemets output och illustrerar hur beslutsstödet kan presenteras för både tränare och utövare.

### 1.2 Avgränsningar

För att skapa en tillförlitlig modell och säkerställa projektets genomförbarhet har arbetet avgränsats inom flera områden. På grund av bristande tillgång till klinisk data samt avsaknad av medicinsk kompetens inom projektgruppen, exkluderas medicinska bedömningar och diagnoser helt från arbetet.

Arbetet avgränsas även till medel- och långdistanslöpning. Detta eftersom friidrott omfattar många grenar med skilda fysiologiska krav och olika träningslogik. En universell lösning för samtliga grenar bedömdes inte vara genomförbart i detta projektet på grund av begränsade resurser. Dessutom finns det i nuläget en brist på insamlad data i många andra discipliner. I ett tidigt skede undersöktes möjligheten att lägga till kompletterande styrketräning till löpning men efter förarbete och utvärdering av insamlad data ansågs även detta vara för komplext för att kunna ge ett gott resultat inom tidsramen.

Modellen avgränsas till att hantera instruktioner på svenska för att tillgodose svenska friidrottsförbundets verksamhet. Fortsättningsvis, begränsas träningen av maskininlärningsmodellen till existerande öppna språkmodeller som finjusteras med insamlad data istället för utveckling av en grundarkitektur på grund av tidsbegränsningar samt mängden av data.

Målgruppen för arbetet är idrottare på ungdoms- och seniornivå; modellen tar därmed inte hänsyn till de specifika krav som ställs på elitidrottare. Av praktiska skäl har datainsamlingen begränsats till vuxna individer som själva kan ge samtycke till delning av personlig data. Slutligen har projektet avgränsats tekniskt till att enbart hantera data från Garmin-användare.

# 2

## Teori

I avsnittet nedan beskrivs den relevanta teorin för projektet. Syftet är att ge en teoretisk grund för de metoder och analyser som används i arbetet. Först introduceras träningslära med inriktning på löpning. Därefter presenteras de teoretiska delar som är kopplade till arbetets regressionsbaserade tempomodell. Sedan beskrivs teorin bakom den AI-baserade träningsplaneringen samt webbplattformen. Avslutningsvis behandlas språkmodeller och finjustering.

### 2.1 Träningslära

Träningslära är en tvärvetenskap som handlar om vilka faktorer som påverkar prestation och hur utveckling kan maximeras. Inom uthållighetsträning är detta särskilt viktigt, eftersom utveckling inte bara beror på träningsmängd utan också på hur träningen fördelas, belastningen stegras och återhämtningen planeras.

Inom löpträning görs ofta en periodisering för hela träningsåret, där olika faser har olika syften [30, s. 38–39]. Målet är att skapa progression, minska skaderisken och nå toppform vid rätt tidpunkt [19, s.28–29]. Grundträningen syftar till att bygga upp en aerob bas genom hög träningsvolym och låg intensitet. I uppbyggnadsfasen blir träningen mer specifik, med större inslag av exempelvis tröskelträning och intervaller. Tävlingsfasen fokuserar på prestation, där träningen anpassas mot tävlingsfart och formtoppning. Återhämtningsfasen innebär lättare träning och vila för att kroppen ska hinna återställa sig och minska risken för överbelastning.

För att prestationen ska förbättras behöver träningsbelastningen successivt öka. Belastningen påverkas bland annat av träningsmängd, frekvens, intensitet och kvalitet [19, s. 81]. Hur stor anpassningen blir beror på individens träningsnivå, och belastningen bör stegras gradvis för att undvika överträning och skador [30, s. 11–12].

Efter belastning minskar en individs funktionsförmåga, vilket leder till trötthet och nedbrytning av musklerna. Vid vila efter träningen återhämtar sig individen, och funktionsförmågan höjs mot den tidigare nivån, vilket kallas superkompensation. Därför är återhämtning en viktig del i träningsfasen och en fullständig återhämtning tar 24–48 timmar [19, s. 78–79].

En vanlig modell inom uthållighetsträning är polariserad träning, där ungefär 80 % av träningen är lågintensiv och 20 % högintensiv [30, s. 39–40]. Träningen bör också i hög grad vara anpassad till det utvecklingsområde som prioriteras, utan att bli så ensidig att risken för överträning ökar [30, s. 12–13]. Ytterligare anpassning efter individens ålder och mognad behöver också göras, eftersom olika fysiska förmågor är olika träningsbara i olika utvecklingsfaser [19, s. 77–78].

Det finns flera olika sätt att kvantifiera och mäta träning. Till exempel genom följande mått:

- **Intensitetsfaktor (IF)**: Intensitetsfaktor mäter hur intensivt ett träningspass är och definieras enligt följande:

$$\text{IF} = \frac{\text{NP}}{\text{FTP}} \approx \frac{\text{medelpuls}}{\text{tröskelpuls}}. \quad (2.1)$$

där NP är normaliserad kraft och FTP är funktionell tröskelkraft [6]. IF kan även generaliseras till en funktion av medelpuls och tröskelpuls där tröskelpuls är den högsta puls en person kan hålla i en timme.

- **Training Stress Score (TSS):** Kombinerar varaktighet och intensitet till ett värde för att skatta den totala belastningen. TSS beräknas enligt följande:

$$\text{TSS} = \frac{\text{sec} \cdot \text{NP} \cdot \text{IF}}{\text{FTP} \cdot 3600} \cdot 100 = \frac{\text{sec} \cdot \text{IF}^2}{3600} \cdot 100 \quad (2.2)$$

där sec är varaktighet på passet.

## 2.2 Regressionsbaserad tempomodell

Syftet med den regressionsbaserade tempomodellen är att prediktera lämpliga måltempo för framtida träningspass baserat på historisk data. Genom att statistiskt modellera sambandet mellan intensitet och prestation kan individuella rekommendationer genereras. Nedan följer en redogörelse för den teoretiska bakgrund och de statistiska metoder som ligger till grund för modellen.

### 2.2.1 Applikationer med löpdata

Att betrakta data från tidigare träningspass är en viktig del av att förstå och analysera träning och därmed även en viktig pusselbit för att skapa nya träningspass. Det finns många olika applikationer som samlar och sammanställer löpdata och även många sätt att tillgå samma data. Nedan förklaras några.

- **Strava:** En blandning av ett socialt medium och en träningsplattform, där man kan samla utförda träningspass och dela dem med andra. De har även en API som man kan använda för att extrahera sin träningsdata från passen.
- **Garmin Activity API:** En API som tillhör Garmin Developer Program. Den gör data insamlat med bland annat Garmins träningsklockor åtkomlig [8]. Enligt beskrivningen ger det tillgång till FIT-filer som innehåller träningspass med full aktivitetsdata. Utövarens data är tillgänglig via API:n så fort denne har gett tillåtelse och kräver inga extra steg eller input utöver det.
- **GarminDB:** Ett från Garmin fristående Pythonbibliotek som kan användas för att extrahera data som samlats in med Garmin-klockor. Datan är samma som Garmin själva läser in vilket innebär att den har samma detaljnivå [9].
- **Garmin Connect:** Garmins egna applikation där en individ kan se och extrahera sina aktiviteter och sin träningsdata i CSV-filer.

### 2.2.2 Statistisk modellering

Longitudinell data består av upprepade observationer från samma individ över tid, vilket innebär att observationerna inte är oberoende. Detta bryter mot centrala antaganden i klassiska regressionsmodeller och kan leda till underskattade standardfel och missvisande inferens om beroendet ignoreras. För att hantera korrelerade observationer finns det några typer av statistiska modeller. En vanlig är en linjär blandad modell (Linear Mixed Model, LMM) vilket

är en modell för longitudinell data där flera observationer görs på samma individ [28]. Den matematiska modellen för tre parametrar kan beskrivas enligt nedan

$$y_{ij} = \beta_0 + \beta_1 x_{1,ij} + \beta_2 x_{2,ij} + \beta_3 x_{3,ij} + \gamma_{0,j} + \epsilon_{ij}$$

där  $y_{ij}$  är responsvariabeln,  $x_{1,ij}, x_{2,ij}, x_{3,ij}$  är förklarande variabler,  $\beta_0$  modellens globala intercept,  $\beta_1, \beta_2, \beta_3$  fasta effekter,  $\gamma_{0,j}$  är det slumpmässiga interceptet för individ  $j$  och  $\epsilon_{ij}$  residualfel.

Ett alternativ för analys av longitudinell data är generaliserade estimeringskvationer (GEE), som används för att skatta populationsmedel samtidigt som beroendestrukturen mellan observationer inom samma individ tas hänsyn till genom en specificerad korrelationsstruktur. Den matematiska modellen för tre parametrar ser ut enligt nedan

$$y_{ij} = \beta_0 + \beta_1 x_{1,ij} + \beta_2 x_{2,ij} + \beta_3 x_{3,ij} + \epsilon_{ij} \quad (2.3)$$

med samma parameterförklaringar som LMM. Till skillnad från LMM inkluderas inga individ-specifika slumpstermer explicit i modellen, utan beroendet mellan observationer hanteras genom den valda korrelationsstrukturen.

För GEE påverkar valet av korrelationsstruktur modellens effektivitet. En vanligt förekommande korrelationsstruktur är den autoregressiva modellen av ordning ett, AR(1), där korrelationen mellan observationer avtar med tidsavståndet mellan dem, vilket gör den särskilt lämplig för tidsdata där närliggande observationer tenderar att vara mer lika [26]. Andra exempel på korrelationsstrukturer är utbytbar där man antar konstant relation mellan varje mätning och oberoende där det inte antas finnas någon korrelation mellan mätningar. Eftersom GEE baseras på kvasi-sannolikhet används anpassade modellvalskriterier, såsom Quasi-likelihood under the Independence Model Criterion (QIC), för att jämföra olika modellspecifikationer.

Residualer, definierade som skillnaden mellan observerade och predikterade värden, används för att analysera modellens avvikelser och identifiera systematiska mönster som modellen inte fångar. Vid modellering med flera förklarande variabler kan multikollinearitet uppstå, vilket kan leda till instabila parameterestimater och försvåra tolkningen av enskilda parametrar.

För att utvärdera modellernas parametrar analyserades skattade koefficienter, standardfel och p-värden. p-värden används för att undersöka om observerade samband mellan variabler och responsvariabeln statistiskt kan skiljas från noll. Om p-värdet är  $p \leq 0,01$ ;  $0,01 < p \leq 0,05$ ;  $0,05 < p \leq 0,1$  tyder det på att det finns övervägande, starkt respektive svagt bevis för att parametern kan beräknas av modellen [17].

## 2.3 AI-baserad träningsplanering

AI-baserad träningsplanering kan utformas på flera sätt. I detta avsnitt beskrivs de två angreppssätt som använts i projektet: regelbaserade system och hämtningsförhöjd generation.

### 2.3.1 Regelbaserade system

Regelbaserade system är en form av AI där beslut tas utifrån fördefinierade regler som kan kallas kontrollregler. Reglerna är ofta utformade som villkor, vilket betyder att systemet utför en viss åtgärd om ett visst tillstånd uppfylls [15]. Det gör att regelbaserade system lämpar sig väl för förutsägbara scenarier där tydliga regler kan användas.

Till skillnad från maskininlärning bygger regelbaserade system inte på att systemet själv identifierar mönster i data. I stället utformas reglerna manuellt av människor [13].

Eftersom beslutslogiken i ett regelbaserat system uttrycks genom regler blir systemet transparent [15]. Det kan också underlätta granskning av systemets beslut, eftersom reglerna är explicit formulerade. Det innebär dock att varje ny situation behöver granskas på nytt och att reglerna konstant behöver justeras manuellt. Regelbaserade system kan därför bli mindre lämpliga i komplexa miljöer där det är svårt att beskriva alla relevanta mönster och variabler i förväg [13].

### 2.3.2 Hämtningsförhöjd generation

Hämtningsförhöjd generation, även känt som Retrieval-Augmented Generation (RAG), fungerar genom att först hämta relevant information från ett avgränsat kunskapsunderlag och därefter använda denna information som stöd när en generativ språkmodell skapar ett svar. Syftet med denna metod är att förbättra noggrannheten och relevansen i modellens utdata genom att förankra genereringen i externa källor i stället för att enbart förlita sig på modellens förtränade kunskap [18].

Processen i ett RAG-system inleds med att användarens fråga omvandlas till vektorrepresentationer, det vill säga numeriska vektorer som representerar textens semantiska innehåll. Genom vektorbaserad sökning kan systemet sedan identifiera de textsegment i kunskapsunderlaget som är mest relevanta för frågan. Dessa hämtade informationsfragment används därefter som utökad kontext för den stora språkmodellen, som genererar ett svar baserat på både frågan och det tillagda underlaget.

I ett RAG-system delas dokumenten upp i mindre delar, så kallade textsegment. Syftet med detta är att göra stora dokument sökbara, så att endast de segment som är mest relevanta för en fråga behöver hämtas. Vid segmentering behöver strategi, segmentstorlek och eventuell överlappning mellan segmenten bestämmas. Vanliga strategier är till exempel segmentering med fast längd och segmentering som tar hänsyn till meningar eller stycken. Överlappning mellan segment används ofta för att minska risken att relevant information går förlorad vid segmentgränser [12].

För att matcha en sökfråga mot de textsegment som skapats kan olika informationshämtningsmetoder användas. Exempel på sådana metoder är TF-IDF (termfrekvens-invers dokumentfrekvens) och semantiska vektorrepresentationer. TF-IDF bygger främst på lexikal likhet, vilket innebär att metoden väger in vilka ord och ordkombinationer som förekommer i både sökfrågan och textsegmentet. Metoden fungerar därför väl när relevanta träffar uttrycks med liknande ordval, men kan vara mer begränsad när textavsnitt har liknande innebörd utan att använda samma formuleringar [23].

Semantiska vektorrepresentationer utgör i stället ett alternativ där innehållsmässig likhet kan fångas även när exakt samma ord inte används, vilket gör metoden väl lämpad för att hitta textsegment som är begreppsmässigt relevanta för frågan [23]. Likheten mellan en sökfråga och ett textsegment kan beräknas med ett mått som kallas cosinuslikhet, vilket används för att rangordna hur väl olika textsegment motsvarar sökfrågan. Ett högre värde innebär att sökfrågan och textsegmentet bedöms vara mer innehållsmässigt lika. De textsegment som hämtas sammanställs därefter till ett kontextunderlag som skickas vidare till språkmodellen som stöd för genereringen.

## 2.4 Webbapplikationer

En webbapplikation kan beskrivas som ett webbaserat system där användaren inte bara tar del av information, utan även påverkar hur systemet fungerar genom sina handlingar. I motsats till en traditionell statisk webbsida, där innehållet främst visas utan större påverkan från användaren, kan en webbapplikation ta emot data, behandla den och anpassa resultatet efter olika val och behov [25].

Funktionen i en webbapplikation bygger på kommunikationen mellan klient och server. När användaren utför en handling i webbläsaren skickas en begäran till servern, som bearbetar informationen och skickar tillbaka ett svar. Denna struktur gör det möjligt att erbjuda mer avancerade funktioner, såsom autentisering, formulärhantering, navigering mellan olika delar av systemet samt presentation av innehåll som varierar mellan användare [25].

## 2.5 Språkmodeller och finjustering

Språkmodeller kan delas in i proprietära och öppna modeller. Proprietära modeller såsom GPT och Gemini erbjuder ofta hög prestanda, men begränsar tillgängligheten för finjustering och kontroll. Fortsättningsvis kräver dessa även användning via API vilket är kostsamt, medför integritetsbegränsningar samt begränsar reproducerbarhet. Öppna modeller erbjuder dock lokal körning vilket möjliggör att modellen kan tränas för specifika problem och data kan lagras lokalt. Detta möjliggör större kontroll över datahantering och lokal lagring av träningsdata, då datan stannar lokalt hos värden [10]. Vidare skiljs basmodeller, tränade på textprediktion, och instruct-modeller, som optimerats för att följa specifika instruktioner och föra dialog.

Ett centralt problem vid träning av modeller är överanpassning, där modellen memorerar träningsdatan istället för att lära sig generella mönster. Genom att analysera förlustkurvornas utveckling går det att identifiera om modellen börjar övertränas, vilket är viktigt för att säkerställa att modellen kan generalisera till ny, tidigare osedd data.

Vid träning av stora språkmodeller används gradientackumulering, vilket innebär att gradienter summeras över flera träningssteg innan modellens vikter uppdateras. Metoden används för att minska minnesanvändningen och möjliggöra träning av större modeller på begränsad hårdvara.

Fortsättningsvis delas datan in i tränings-, validerings- och testdata enligt en procentuell fördelning. Träningsdatan används för att anpassa modellens parametrar, valideringsdatan används för att övervaka modellens prestanda under träningen och identifiera överanpassning, medan testdatan används för den slutliga utvärderingen av modellen på tidigare osedd data.

### 2.5.1 Transformer-arkitekturen och tokenisering

Moderna språkmodeller såsom Llama 3.1 8B bygger på Transformer-arkitekturen. Denna arkitektur använder en mekanism kallad självuppmärksamhet, vilket tillåter modellen att väga betydelsen av olika ord i en mening, oberoende av deras avstånd till varandra. Detta är avgörande för att modellen ska kunna förstå kontexten i komplex data där modellen måste ta i beaktning tidigare information [33].

I Transformer-arkitekturen bearbetas text genom tokenisering, där ord delas upp i mindre enheter kallade tokens. Detta påverkar hur mycket kontext modellen kan behandla samtidigt.

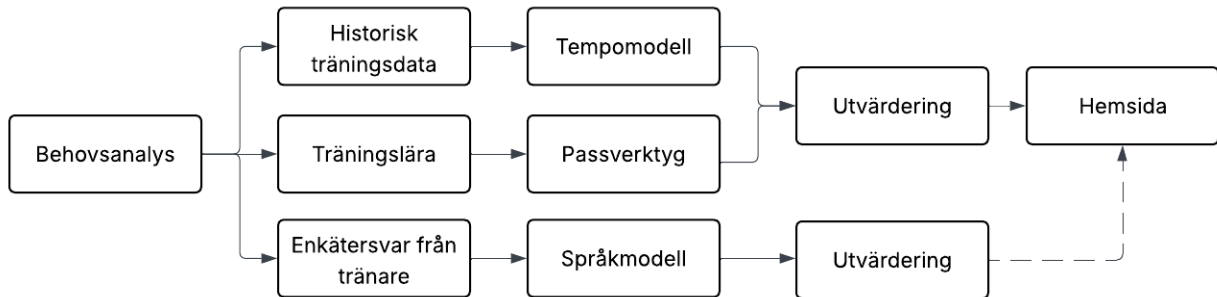
### 2.5.2 Parameter-effektiv finjustering med LoRA

För att specialisera en modell krävs övervakad finjustering (SFT), där modellen tränas på uppgiftsspecifik data. Då fullständig träning av miljarder parametrar kräver enorma beräkningsresurser används ofta Low-Rank Adaptation (LoRA). Teorin bakom LoRA är att de viktuppdateringar som sker under träningen har en låg inneboende dimension. Detta innebär att majoriteten av modellens ursprungliga vikter fryses och förblir oförändrade, medan träningen fokuseras på ett fåtal lågdimensionella matriser. Detta reducerar kraven på grafikminne drastiskt utan betydande prestandaförlust [11]. Genom att kombinera detta med 4-bitars kvantisering kan modellens storlek minskas ytterligare, vilket sänker minnesanvändning med cirka 70 % med bibehållen prestanda [3].

# 3

## Metod

Detta avsnitt behandlar projektets metod, uppdelad i tre huvudfaser. Först presenteras en regressionsmodell för prediktion av individanpassande tempon baserat på historisk löpdata. Därefter beskrivs systemet för träningsplanering, vilket kombinerar RAG med regelbaserad logik för att generera strukturerade pass och scheman. Slutligen avhandlas finjusteringen av en stor språkmodell som fungerar som en interaktiv tränar-persona för att tolka utövarens dagsform och justera träningsrekommendationerna. Nedan följer en översiktsfigur på projektets uppbyggnad.



**Figur 3.1:** Översiktsbild över de olika metoddelarna och hur de ska kopplas ihop. Det streckade är tänkt men inte genomfört.

### 3.1 Behovsanalys

För att identifiera kravspecifikationen och användarnyttan av en AI-baserad träningsplattform genomfördes en kvalitativ behovsanalys. Undersökningen fungerade som beslutsunderlag för systemets funktionalitet, prioriterade utvecklingsområden samt utformningen av webbplattformen.

Som ett första steg genomfördes en förstudie i form av kvalitativa och semistrukturerade intervjuer med aktiva friidrottstränare. Syftet var att skapa en djupare förståelse för tränarens roll i praktiken, hur beslut fattas i träningsvardagen samt vilka faktorer som vägs in vid individanpassning av träning. Totalt intervjuades sex friidrottstränare samt en aktiv friidrottare. Intervjuerna genomfördes digitalt och utgick från en gemensam intervjuguide, se Appendix A, men anpassades efter respondentens roll samt tidigare svar för att möjliggöra fördjupning och nyanserade resonemang. Särskilt fokus lades på hur tränare arbetar med planering, uppföljning och skadeprevention, samt vilka delar av träningsprocessen som potentiellt skulle kunna stödjas av en AI-baserad lösning.

Frågorna var strukturerade kring flera centrala områden, inklusive träningsupplägg, återhämtning, datainsamling, uppföljning samt användning av teknik och träningsplattformar. Exempel på frågeställningar rörde hur tränare anpassar träning efter dagsform, hur data från exempelvis träningsklockor används i beslutsfattande, samt vilka moment i tränarrollen som upplevs som

mest tidskrävande och komplexa.

## 3.2 Regressionsbaserad tempomodell

Nedan beskrivs utvecklingen av den statistiska modell som används för att generera tempoförslag. Arbetet omfattar datainsamling, förbehandling, modellering samt utvärdering.

### 3.2.1 Datainsamling och urvalskriterier

För att ta fram relevanta tempoförslag till olika träningspass var första steg att identifiera en lämplig datakälla för historisk träningsdata. För att identifiera en lämplig datakälla definierades två primära urvalskriterier:

- **Teknisk extraherbarhet:** Applikationen ska erbjuda ett dokumenterat gränssnitt eller exportmöjligheter som tillåter programmatisk åtkomst till rådata.
- **Juridiskt tillstånd:** Applikationens användarvillkor ska tillåta att datan behandlas och analyseras inom ramen för ett akademiskt projekt.

Utifrån dessa kriterier genomfördes en förundersökning där dokumentation och licensavtal för Strava API, Garmin Activity API, GarminDB och Garmin Connect granskades. Denna process syftade till att säkerställa att studiens datainsamling skedde på ett etiskt och juridiskt korrekt sätt.

Till följd av denna förundersökning valdes Garmin Connect som datakälla. För att underlätta för användare att ladda ner sin data skapades en stegvis guide på hur datan skulle hämtas från Garmin Connects hemsida och i guiden inkluderades bland annat att användaren först behöver ladda in aktiviteter för att de skulle inkluderas i exporten.

Efter att datakällan valts skapades en testgrupp bestående av frivilliga från en löpargrupp i Kungälv. Detta var en grupp vars tränare tidigare hade medverkat i intervjuer för behovsanalys och gruppen låg i stort i linje med projektets avgränsningar. För att skapa en mer robust modell kompletterades även datan med data från sex frivilliga studenter som använde samma datakälla för att följa sin träning. Totalt inkluderades tio löpare, vilket genererade en datamängd om 1 779 löppass.

### 3.2.2 Databehandling och variabelselektion

Datasäkerhet och personlig integritet har varit styrande för val av arbetsmetod. Medan externa AI-verktyg har använts som stöd vid utveckling och optimering av analyskoden, har all faktisk bearbetning av rådata och statistisk modellering genomförts uteslutande i en lokal miljö. Inga personuppgifter eller träningsdata har överförts till externa språkmodeller eller molnbaserade tjänster.

I denna lokala miljö bearbetades rådata i CSV-format med Python-biblioteket `pandas`. Förbehandlingen innefattade exkludering av irrelevanta aktivitetstyper samt omvandling av medeltempo från min/km till s/km. För att hantera extremvärden beräknades ett medeltempo per individ, varpå observationer som avvek mer än två standardavvikelser från individens medelvärde exkluderades. Datat kompletterades även med träningsmått som TSS och IF beräknade enligt Ekvation (2.1) och (2.2), där tröskelpulsen beräknades som 0,95 av den uppmätta maxpulsen för en individ.

Initiala förklarande variabler valdes utifrån fysiologisk relevans:

- **Belastningsmått:** Intensitetsfaktor (IF) och Training Stress Score (TSS).
- **Rörelsedata:** Distans och höjdvariation (stigning/nedför per kilometer).
- **Miljöfaktorer:** Temperatur och luftfuktighet.

### 3.2.3 Statistisk modellering

För att modellera den longitudinella datan utvärderades initialt både LMM och GEE. De preliminära jämförelserna genomfördes under den tidiga modellutvecklingen och indikerade att GEE gav robust hantering av korrelation mellan observationer från samma individ genom explicit specifikation av en korrelationsstruktur. Utifrån detta vidareutvecklades GEE-modellen och användes som grund för den slutgiltiga tempomodellen.

Efter jämförelse av olika korrelationsstrukturer med hjälp av QIC valdes korrelationsstrukturen AR(1). Antagandet innebär att träningspass nära varandra i tiden antas vara starkare korrelerade än pass med längre tidsintervall.

Variabelselektionen genomfördes iterativt. Multikollinearitet analyserades genom parameterestimaten stabilitet och standardfel. Variabler med instabila skattningar exkluderades successivt. Exempelvis exkluderades TSS till förmån för IF på grund av stark korrelation mellan variablerna. Vissa variabler, såsom nedförslutning, behölls trots begränsad fysiologisk intuition då de förbättrade modellens prediktionsförmåga.

### 3.2.4 Tidsviktning och individualisering

För att fånga förändringar i löparnas fysiologiska status och dagsform tillämpades en exponentiell tidsviktning, där nyligen genomförda träningspass gavs högre vikt än äldre observationer. Vikten  $w_i$  för observation  $i$  definierades som  $w_i = e^{-\lambda d_i}$ , där  $d_i$  representerar antalet dagar sedan passet genomfördes och  $\lambda$  är en sönderfallsparameter. I studien valdes  $\lambda = 0,005$  heuristiskt för att gradvis minska äldre observationers påverkan utan att helt exkludera dem ur datamängden.

Viktningen applicerades både vid modellestimering och vid beräkning av individernas residualbaserade korrigeringsstermer. GEE-modellen kompletterades med en individspecifik biaskorrigering, definierad som det tidsviktade medelvärdet av individens historiska residualer.

Den slutgiltiga prediktionen genererades därefter i två steg: först beräknades en populationsbaserad prediktion med GEE-modellen, varpå den individspecifika biaskorrigeringen adderades för att slutgiltigt justera tempoförslaget.

### 3.2.5 Modellutvärdering och jämförelse

Modellens prestanda och anpassningsförmåga utvärderades genom att beräkna det genomsnittliga absolutfelet, uttryckt i sekunder per kilometer (s/km), för de fem senaste passen för varje löpare. För att efterlikna ett realistiskt användarscenario låg fokus vid utvärderingen på dessa specifika pass, vilka analyserades i förhållande till löparens totala träningshistorik.

Slutligen genomfördes en jämförelse mellan den tidsviktade GEE-modellen och en oviktad LMM över samma datamängd för att utvärdera effekten av tidsviktningen samt valet av statistisk metod.

Den kompletta koden återfinns i `garmin_global_model_tidvikt.ipynb` i GitHub-arkivet [2].

### 3.3 AI-baserad träningsplanering

Den AI-baserade träningsplaneringen utvecklades som en kombination av en RAG-baserad lösning och regelbaserad logik, för att generera individuellt anpassade planer utifrån strukturerad användardata och ett avgränsat kunskapsunderlag inom träningslära och löpträning.

#### 3.3.1 Hämtningsförhöjd generation

Kunskapsunderlaget bestod av tre PDF-baserade utbildningsmaterial som extraherades till textfiler: *Fröidrottens allmänna träningslära* [19] för generella tränings- och belastningsprinciper, *Löptränare* [30] för löpspecifik progression, intensitetsfördelning och återhämtning, samt *Träning i uppbyggnadsstadiet: Fröidrott* [24] för åldersanpassad planering och medel- och långdistansspecifika principer.

Dokumentet delades upp (segmenterades) i överlappande textsegment om 800 tecken med 150 teckens överlappning. Segmentgränserna justerades vid menings- eller radbrytningar och segment kortare än 50 tecken exkluderades. Varje segment omvandlades till vektorrepresentationer med modellen `paraphrase-multilingual-MiniLM-L12-v2`. Segmentstorleken 800 tecken valdes för att passa inom vektorrepresentationsmodellens kontextfönster, och överlappningen på 150 tecken bedöms som rimlig för att minska risken att information vid segmentgränser går förlorad. Segment under 50 tecken exkluderades eftersom de sällan bär meningsfullt innehåll.

Vid generering byggde systemet sju sökfrågor till kunskapsbasen utifrån fyra fördefinierade teman: periodisering, nyckelpass, återhämtning och teknik. Varje tema hade flera sökfrågemallar med varierande ordval, för att täcka in olika aspekter av träningsupplägget och fånga upp relevant innehåll även när det uttrycks med andra begrepp än temats namn. Mallarna anpassades efter idrottaren genom att disciplin och träningsfas fylldes i. För en medeldistanslöpare i uppbyggnadsfas blev exempelvis en av sökfrågorna under temat periodisering ”medeldistans uppbyggnad periodisering månadsplanering”.

Varje sökfråga jämfördes med textsegmenten genom cosinuslikhet och de mest relevanta plockades ut. Både medel- och långdistans använde sig av samtliga dokument i kunskapsunderlaget, men användaren kunde även manuellt välja vilka dokument som skulle ingå i sökningen.

Urvalet av textsegment skedde i två steg. Först hämtades, för varje sökfråga, de fyra textsegment som hade högst cosinuslikhet med frågan. Fyra valdes för att ge tillräckligt många kandidater att filtrera bland i nästa steg, men inte så många att svagt relevanta segment togs med. Med sju sökfrågor gav detta upp till 28 kandidater innan dubletter togs bort. Därefter filtrerades kandidaterna så att högst två segment per tema och högst tre segment per dokument fick ingå. Eftersom det fanns fyra teman innebar detta att högst åtta segment kunde väljas totalt. Två per tema valdes för att alla fyra teman skulle få plats. Tre per dokument valdes för att alla tre dokument skulle kunna bidra utan att något enskilt dokument dominerade. På så sätt blev alla fyra teman representerade utan att något enskilt tema eller dokument dominerade.

De utvalda segmenten sammanställdes till ett kontextunderlag som skickades till språkmodellen Claude Sonnet 4.5 via Anthropic API, tillsammans med icke-identifierande information om idrottaren såsom löpinriktning, träningsfas, träningsvana, träningsmängd, mål, begränsningar och i vissa fall tävlingsresultat. Direkt identifierande uppgifter som namn och kontaktuppgifter exkluderades. Utifrån detta genererade modellen förslag på träningspass och månadsplanering.

För att välja metod för informationshämtning genomfördes en jämförelse mellan TF-IDF och vektorbaserad sökning med samma textsegment och kunskapsunderlag som beskrivits ovan. Vektorrepresentationsmetoden använde modellen `paraphrase-multilingual-MiniLM-L12-v2`. TF-IDF-modellen jämförde sökfrågan med textsegmenten utifrån vilka ord och ordkombinationer de hade gemensamt, där ovanliga ord väger tyngre än vanliga. Mycket sällsynta och mycket vanliga ord filtrerades bort eftersom de inte bidrog till att särskilja segmenten från varandra. De exakta parametervärdena finns i filen `compare_retrieval_detailed.py` i projektets GitHub-arkiv [2]. För båda metoderna användes cosinuslikhet som likhetsmått, och värden under 0,05 exkluderades från TF-IDF-resultaten eftersom de bedömdes innehålla för få gemensamma termer med sökfrågan för att vara relevanta.

Jämförelsen genomfördes med disciplinen medeldistans och träningsfasen uppbyggnad, med följande fyra sökfrågor:

- medeldistans uppbyggnad träningsplanering veckoschema
- medeldistans tröskelpass uthållighet nyckelpass
- återhämtning vilodag hard easy polariserad löpning
- uppbyggnad intensitetszoner 80 20 tröskelträning

För varje sökfråga hämtades de åtta högst rankade textsegmenten från respektive metod, varefter identiska segment avduplicerades inom varje resultatlista. Resultaten analyserades utifrån likhetsvärden, antal gemensamma träffar, skillnader i hämtade segment samt en kvalitativ bedömning av segmentens användbarhet som underlag för träningsplanering.

### 3.3.2 Regelbaserade system

Som komplement till RAG-systemet användes kontrollregler implementerade i Python, dels för att säkerställa en tillåten veckostruktur före genereringen, dels för att efter genereringen kontrollera och vid behov justera planen. Reglerna utgick från träningsprinciperna i avsnitt 2.1 och sammanfattas i tabell 3.1, med exempel på regler, den fullständiga regeluppsättningen finns i filen `rag_knowledge.py` i projektets GitHub-arkiv [2]. Om planen inte uppfyllde reglerna, eller om RAG-lösningen inte kunde användas, aktiverades en regelbaserad reservlösning där passförslag genererades utifrån fördefinierade träningsmallar och villkorsstyrd logik, se Appendix C samt filerna `ai_schedule.py` och `exercise_bank.py` i projektets GitHub-arkiv [2].

Claude Sonnet 4.5 valdes som språkmodell för RAG-systemet på grund av modellens starka förmåga till textgenerering och resonemang, samt dess väldokumenterade API och relativt enkla integrationsprocess. Detta möjliggjorde snabb implementation och stabil generering av träningsrekommendationer inom projektets tidsram [1].

**Tabell 3.1:** Sammanfattning av centrala kontrollregler i det regelbaserade systemet.

Regelkategori	Syfte	Exempel på regel	Användning
Pass och passroller	Avgränsa planen till löpträning.	Pass ska vara löpningsbaserade och ha roller som kvalitet, volym, återhämtning eller teknik.	Används när systemet instruerar språkmodellen om vilken typ av pass den får föreslå.
Ålder	Anpassa belastning efter ålder.	Barn får inga hårda pass och max 3 pass/vecka. Ungdomar får max 1 hårt pass/vecka.	När veckans grundstruktur skapas och när den färdiga planen kontrolleras.
Intensitet och fas	Begränsa hård belastning.	Max antal hårda pass styrs av träningsfas, t.ex. 1 i grundträning, 2 i uppbyggnad/tävling och 0 i återhämtning.	Vid bestämning av hur hård veckan får vara och när intensiteten kontrolleras efter generering.
Veckostruktur	Skapa variation, prioritering och återhämtning.	Veckan ska ha minst två passroller, normalt ett nyckelpass och sen ska ett hårt pass följas av lätt pass eller vila.	När systemet bygger veckans upplägg och när passens ordning granskas.
Passkvalitet	Säkra genomförbara hårda pass.	Hårda pass ska innehålla uppvärmning, huvuddel och nedvarvning.	Används när de genererade passbeskrivningarna kontrolleras.
Månadsplanering	Kontrollera periodisering.	Vecka 4 behandlas som återhämtningsvecka med 50–70 % av normalvolym och utan hårda pass.	När systemet skapar och kontrollerar en plan över fyra veckor.

### 3.4 Systemsammankoppling i webbplattform

Tempomodellen, beskriven i sektion 3.2, integrerades med den AI-baserade träningsplaneringen, beskriven i sektion 3.3. Efter att ett pass genererats, antingen via RAG eller via den regelbaserade reservlösningen, beräknade tempomodellen en rekommenderad tempointervall som presenterades tillsammans med passbeskrivningen. När data från Garmin saknades hämtades tempoangivelserna antingen direkt från den RAG-genererade passbeskrivningen eller, i annat fall, genom regelbaserade uppskattningar utifrån användarprofil, prestationsnivå, passtyp och underlag.

Den ursprungliga tempomodellen, `garmin_global_model_tidvikt.ipynb` [2], byggde på Garmin-data där medelpuls, total stigning och totalt nedför var direkt mätta. Eftersom AI-genererade pass saknar sådana mätvärden behövde modellen anpassas. Koefficienterna och de personliga tempooffsets som tagits fram i regressionsanalysen exporterades som fasta värden till en Python-modul, `tempo_model.py` [2], och tre kompletterande anpassningar infördes. IF räknades fram genom att programmet läste passets beskrivning och letade efter särskilda ord och distanser. Hittade det exempelvis ordet "tröskel" eller distansen "800 m" så valdes ett högt IF-värde, medan ord som "lugn" eller "jogg" gav ett lägre värde. Variablerna stigning/km och nedför/km antogs vara samma och det skapades fyra fördefinierade lutningsprofiler kopplade till underlagstyp (bana: 0 m/km, plan väg 5 m/km, kuperat 15 m/km och väldigt kuperat 30

m/km) som användaren själv valde. Tempot justerades dessutom efter eventuella repetitions-distanser, eftersom kortare repetitioner generellt löps i högre tempo än motsvarande IF i ett kontinuerligt distanspass.

Modellen kunde tillämpas i två lägen. För de tio löpare vars data från Garmin användes för att träna modellen fanns personliga biaskorrigeringar redan inbyggda. För övriga användare implementerades en kalibreringsfunktion där de själva kunde ladda upp historisk träningsdata från Garmin i CSV-format, varefter plattformen jämförde modellens prognoser med deras faktiska loggade tempon och beräknade en personlig offset i sekunder per kilometer. Denna offset adderades till modellens predikterade tempo vid framtida passgenereringar, så att även användare utanför den ursprungliga träningsuppsättningen kunde få individanpassade tempoindikationer.

För att demonstrera träningsplaneringssystemet utvecklades en webbplattform, som beskrivs nedan.

### 3.4.1 Utveckling av webbplattform

Webbplattformen utvecklades i Python med `Flask` och utformades som en lokalt körd, serverbaserad webbapplikation. `Flask` är ett webbramverk skrivet i Python, det är skapat för att göra webbapplikationer snabbt och enkelt [21]. Applikationen innehåller funktioner för inloggning, navigering, registrering av träningspass, planering av träning samt presentation av sammanställningar för användaren. Gränssnittet utvecklades med HTML, som är standarden för att strukturera innehåll på webbsidor med hjälp av taggar [31], och CSS, som används för att styra hur innehållet presenteras visuellt, exempelvis färger, typsnitt och layout [16]. Vad som visades på sidan anpassades automatiskt utifrån vilken användare som var inloggad och vilken data som fanns tillgänglig.

Kodstrukturen delades upp mellan flera filer med olika ansvarsområden. En huvudfil samordnade applikationens sidor och centrala funktioner, medan övriga filer hanterade bland annat autentisering, datamodeller, träningsloggar, planerade pass, testresultat, skador, tävlingsresultat och AI-relaterade funktioner. I prototypen lagrades användardata lokalt i en `SQLite`-databas, ett filbaserat databassystem som inte kräver en separat databasserver [27], där information om användare, aktiva, träningsloggar och planerade pass sparades.

I webbplattformen lästes tävlingsresultat in och matchades mot användarprofiler utifrån namn, födelseår och klubb. Detta för att både visa tävlingshistorik i gränssnittet och även som underlag vid passgenerering.

Tävlingsresultat hämtades genom webbskrapning från <https://www.friidrottsstatistik.se/>. Webbskrapning innebär att information hämtas automatiskt från webbsidor genom att läsa av deras HTML-struktur och extrahera relevanta dataelement, så att informationen kan struktureras och användas vidare [22]. Detta genomfördes i Python med biblioteket `requests` i ett separat skript. Relevant information extraherades ur HTML-innehållet med reguljära uttryck och sparades lokalt i JSON-format.

Programmeringsarbetet genomfördes med hjälp av Claude Cowork och Codex. Verktygen användes för att ta fram kod, utveckla funktioner, strukturera delar av applikationen och ge stöd vid felsökning.

### 3.4.2 Utvärdering av träningsplaneringssystemet

Utvärderingen av det slutgiltiga verktyget för träningsplanering med implementerad tempomodell gjordes i två steg.

Första steget var en kvalitativ utövaranalys. För att bedöma de genererade passens rimlighet testades några pass inom varje kategori av en användare med registrerad träningsdata. Ett genererat månadsschema skickades även ut till tre personer som bidragit med data till projektet men dessa tre gjorde en bedömning utan att genomföra passen. Samtliga fyra svarade på en enkät med frågor om tempo och passupplägg.

I enkäten frågades det inledningsvis efter första reaktion på planeringen. Sedan följde frågorna nedan med skalan 1–5 på alla förutom Fråga 7. Till var och en av dessa frågor fanns även möjlighet till en kommentar i fritext. Avslutningsvis frågades även om det fanns någon period där belastning blev orimligt hög/låg och vad personen hade ändrat om hen fick ändra en sak med planen samt övriga tankar och kommentarer.

**Fråga 1:** Hur rimliga upplever du att de föreslagna tempotiderna för de högintensiva passen/tröskelpassen är?

**Fråga 2:** Hur rimliga upplever du att de föreslagna tempotiderna för distanspassen är?

**Fråga 3:** Hur rimliga upplever du att tempotiderna är för återhämtningspassen?

**Fråga 4:** Hur rimliga upplever du tempoanpassningarna för olika typer av kupering?

**Fråga 5:** Hur väl matchar den genererade planen din nuvarande träningsnivå och historik?

**Fråga 6:** Hur väl stämmer intensitetsfördelningen (balansen mellan lugna pass och högintensiva pass) överens med din träningsfilosofi?

**Fråga 7:** Upplevs uppvärmning och nedvarvning som tillräcklig i förhållande till huvudpasset?

**Fråga 8:** Hur bedömer du progressionskurvan (ökningen av träningsmängd) över tid?

**Fråga 9:** Känns veckostrukturen (t.ex. placering av vilodagar och långpass) logisk och genomförbar i din vardag?

**Fråga 10:** Hur stor är sannolikheten att du skulle följa denna plan om du hade ett lopp som mål?

Då det kan vara skillnad på hur en individ tränar och vad som är optimal träning var det andra steget en kvalitativ tränaranalys. Detta gjordes genom en intervju med en tränare till några av löparna som bidragit med data för att analysera samma genererade pass som skickats till hens löpare. Intervjufrågorna som ställdes kan ses i `tranarutvardering_mall.txt` i GitHub-arkivet [2].

## 3.5 Finjusterad språkmodell för utövaråterkoppling

För att skapa en personlig tränar-persona krävdes en process som sträckte sig från datainsamling till teknisk optimering. Arbetet innefattade utformning av enkäter för att fånga tränarlo-

gik, val av en lämplig basmodell, samt finjustering av denna i ett högpresterande datorkluster. Nedan beskrivs de steg och metoder som användes för att utveckla och utvärdera modellens förmåga att ge kontextuella träningsrekommendationer.

### 3.5.1 Tränarenkäter

Utformningen av enkäterna syftade till att skapa ett datamaterial som kunde användas för finjustering av AI-modellen. För att modellen skulle kunna ge relevanta och kontextanpassade rekommendationer behövde den tränas på situationer som liknar de avvägningar en tränare gör i praktiken. Enkäterna konstruerades utifrån de inledande intervjuerna för att fånga tränarnas beslutslogik i olika typer av träningsituationer, och totalt skapades 133 scenarier fördelade över fem enkäter, som blev besvarade av 15 tränare. Dessa finns tillgängliga i projektets GitHub-arkiv [2].

Varje scenario utformades med en återkommande struktur bestående av profil, aktuell status och dagens planerade pass. Denna struktur valdes för att skapa ett tydligt samband mellan indata och beslut, vilket var viktigt för AI-träningen. Genom att variera profiler, träningsmål och typer av problematik kunde datan täcka flera olika typer av användare och situationer. Scenarierna inkluderade även olika tidpunkter i beslutsprocessen, såsom situationer före, under och efter ett pass, för att modellen skulle tränas på flera typer av bedömningar.

Svarsalternativen utformades som konkreta handlingsval, exempelvis att genomföra passet, modifiera det, byta till alternativ träning eller vila. Därutöver gavs möjlighet att ange ett eget svar med motivering. En viktig del i utformningen var också att inkludera gränsfall och situationer utan ett självklart rätt svar, eftersom målet var att modellen skulle lära sig att hantera nyanserade beslutssituationer snarare än att följa enkla, fasta regler.

### 3.5.2 Analys och standardisering av enkätdata

Enkätsvaren utgjorde grunden för den data som användes vid finjusteringen av AI-modellen. De varierade dock i längd, struktur och detaljnivå. En stor andel av svaren bestod av korta, standardiserade svarsalternativ, medan andra svar utgjordes av mer omfattande och utförligt formulerade rekommendationer från tränare. Denna inkonsekvens innebar utmaningar vid finjustering av AI-modellen, då modellen riskerade att lära sig otydliga och motsägelsefulla mönster.

För att uppnå en mer sammanhängande datastruktur utformades ett antal fördefinierade mallar för hur svaren från tränarna skulle struktureras. Mallarna bestod av en kort beskrivning av användarens situation, en rekommendation kopplad till det planerade passet baserat på tränarens ursprungliga svar, en kort motivering samt kompletterande information för särskilda fall. Den komplimenterade informationen inkluderade ytterligare detaljer eller nyanser i tränarnas svar som inte direkt passade in i mallens struktur. Med mallarna som utgångspunkt genomfördes en manuell bearbetning av samtliga svar från enkäterna. Svaren omformulerades för att tydligare kopplas till den specifika situationen för varje scenario samtidigt som den underliggande rekommendationen från tränarna bevarades. Utöver det ändrades otydliga eller generella formuleringar till kontextspecifika resonemang baserat på det planerade träningspasset samt användarens status.

### 3.5.3 Val av språkmodell och optimeringstekniker

Som basmodell för projektet valdes Meta Llama 3.1 8B Instruct. Valet motiverades av modellens prestanda i förhållande till dess storlek, samt dess tillgänglighet som öppen källkod [10]. Detta möjliggjorde lokal finjustering och inferens på Chalmers Vera-kluster utan beroende av externa API-tjänster. Modellstorleken valdes genom en avvägning mellan språkförståelse och beräkningsresurser. Modellen behövde vara tillräckligt kapabel för att hantera svenska språket, samtidigt som den behövde kunna finjusteras inom projektets tidsram och tillgängliga GPU-resurser. Instruct-versionen av modellen valdes framför basversionen då den redan tränats för att följa instruktioner och föra dialog, vilket gav en stabilare utgångspunkt för utvecklingen av en konversationsbaserad tränarassistent.

För att möjliggöra finjustering av modellen inom tillgängliga beräkningsresurser användes flera optimeringstekniker. Modellens minnesanvändning reducerades genom kvantisering, samtidigt som LoRA användes för parameter-effektiv finjustering där endast ett begränsat antal parametrar uppdaterades under träningen.

Vidare användes gradientackumulering för att möjliggöra en effektiv batchstorlek på 16 trots begränsningar i GPU-minnet. Träningen optimerades även genom användning av 16-bitars flyttal och en 8-bitars AdamW-optimerare, vilket reducerade minneskraven och möjliggjorde effektiv träning på Nvidia A40 GPU med 48 GB VRAM på Chalmers Vera-kluster.

Den fullständiga implementationen och kod finns under mappen LLM, tillgänglig i projektets GitHub-arkiv [2].

### 3.5.4 Finjustering av språkmodellen

Finjusteringen genomfördes som SFT, där modellen tränades på exempel bestående av användarprompter och tillhörande tränarsvar. Träningen utfördes med parameter-effektiv finjustering genom LoRA. Träningen konfigurerades även med så kallat completion-only loss, där förlustfunktionen endast beräknades på modellens genererade svar och inte på promptstrukturen. Fördelningen för datan var en 80/15/5 fördelning, där 80 % av datan användes för träning, 15 % för validering och 5 % för test.

Implementeringen genomfördes i Python med hjälp av `HuggingFace` och `Unsloth`. För att styra modellens beteende definierades en systeminstruktion som instruerade modellen att agera som en erfaren och försiktig löptränare med fokus på återhämtning och långsiktig utveckling. Modellen instruerades även att undvika att generera symptom eller information som inte uttryckligen nämnts av användaren.

Träningen genomfördes på Chalmers Vera-kluster med Nvidia A40 GPU (48 GB VRAM) via resurshanteraren Slurm. Antalet epoker begränsades till tre för att minska risken för överanpassning givet den relativt begränsade datamängden. För att övervaka träningsprocessen användes `Weights & Biases (WandB)`, där tränings- och valideringsförlust jämfördes för att identifiera eventuella tendenser till överanpassning.

Rådatan konverterades till ett strukturerat JSONL-format där varje träningsinstans bestod av en systeminstruktion, en användarprompt och ett tillhörande tränarsvar. Genom denna struktur tränades modellen att generera dialogbaserade tränarrekommandationer.

Den maximala sekvenslängden begränsades till 1024 tokens för att balansera kontextlängd mot minnesanvändning och träningstid. Centrala hyperparametrar som användes i finjusteringen

presenteras i Tabell 3.2.

**Tabell 3.2:** Hyperparametrar från finjusteringen av språkmodellen.

Parameter	Värde
Learning Rate	$2 \times 10^{-5}$
Epoker	3
Effektiv batchstorlek	16 ( $1 \times 16$ )
LoRA Rank ( $r$ )	16
LoRA Alpha	32
LoRA Dropout	0,05
Maximal sekvenslängd	1024 tokens
Optimerare	AdamW (8-bit)

### 3.5.5 Utvärdering av finjusterad modell

För att utvärdera modellens prestation genomfördes en jämförande analys mellan den finjusterade modellen och basmodellen. Datan delades upp i tränings-, validerings- och testdata enligt en 80/15/5-fördelning, där 80 % av datan användes för träning, 15 % för validering och 5 % för testning. Testdatan (5 %) motsvarade 20 scenarier som låg till grund för utvärderingen.

Utvärderingen baserades på en jämförelse mellan den finjusterade modellen och en otränad basmodell. För varje scenario genererade båda modellerna ett svar baserat på samma input, vilket sedan jämfördes med tränarnas rekommendationer och bedömdes utifrån fördefinierade kriterier. Bedömningen genomfördes manuellt och baserades på hur väl modellernas svar överensstämde med tränarnas rekommendationer samt hur väl resonemangen bakom rekommendationerna var kopplade till indatan. Samtliga genererade svar från både basmodellen och den finjusterade modellen, tillsammans med tillhörande tränarrekommandationer, har även sammanställts i ett separat dokument `evaluation_outputs.txt` som finns tillgängligt i projektets GitHub-arkiv [2].

För att möjliggöra en strukturerad och jämförande utvärdering definierades fem bedömningskriterier: beslut, majoritet, logik, inputkoppling och språk. Varje utdata bedömdes utifrån en poängskala från 0 till 2, där 0 motsvarar låg kvalitet och 2 hög kvalitet. Summan av kriterierna gav ett totalpoäng per scenario, med ett maximalt värde på 10.

- **Beslut:** I vilken utsträckning modellen ger tydliga och konkreta rekommendationer.
- **Majoritet:** Hur väl rekommendationen överensstämmer med den vanligaste tränarrekommandationen.
- **Logik:** Hur rimligt och sammanhängande resonemanget är i relation till scenariot.
- **Inputkoppling:** Hur väl svaret använder relevant information från indatan.
- **Språk:** Hur tydligt, konsekvent och lättförståeligt svaret är formulerat.

För varje scenario beräknades ett totalpoäng baserat på summan av poängen från de fem kriterierna. Totalpoängen användes sedan för en analys där medelvärden, skillnader i totalpoäng samt antal scenarion där respektive modell presterade bättre jämfördes. Utöver den statistiska sammanställningen genomfördes även en övergripande analys av modellernas svar, där återkommande mönster och avvikelser identifierades och antecknades.

# 4

## Resultat

I detta avsnitt redovisas resultatet av projektet, i form av tre delar. Den första är resultat från intervjuer med tränare, den andra delen handlar om den AI-baserade träningsplaneringen och tempomodellen. Den tredje redovisar resultatet från den finjusterade språkmodellen.

### 4.1 Behovsanalys

Intervjuerna visade att tränarna ser individanpassning som en central men tidskrävande del av tränarrollen. Detta gäller särskilt i större grupper där aktiva har olika träningsbakgrund, mål, skadehistorik och dagsform. I praktiken sker många anpassningar utifrån tränarens erfarenhet, dialog med den aktive och observationer under träning, snarare än genom ett samlat digitalt beslutsunderlag. Resultaten i detta avsnitt baseras på de genomförda intervjuerna, vars sammanställda underlag finns tillgängligt i projektets GitHub-arkiv under [Intervjuer](#) [2].

Ett tydligt tema var behovet av bättre överblick över dagsform och återhämtning. Data som hjärtfrekvensvariabilitet (HRV), vilopuls, sömn, upplevd ansträngning och skadekänningar används i dag i begränsad utsträckning, men uppfattades som mycket värdefull om den kan sammanställas enkelt och kopplas till konkreta rekommendationer. För löpning lyftes särskilt automatisk pulszonsanalys, volymkontroll, skadevarningar och dagsformsbaserad justering av intervaller som efterfrågade funktioner.

Intervjuerna visade också att användningen av kroppsburen teknik, exempelvis smartklockor och aktivitetsarmband, skiljer sig mellan grengrupper. Inom medel- och långdistans uppfattades puls-, tempo- och volymdata som relevant för uppföljning av belastning. Inom sprint och hopp är pulldata mindre central, eftersom träningen i högre grad bygger på explosivitet, kraftutveckling och teknik.

Tränarna efterfrågade framför allt ett enkelt system som minskar administration och gör det lättare att följa träning över tid. Ett snabbt loggningsflöde för exempelvis Borgskalan, känsla, sömn och smärta bedömdes vara avgörande för att aktiva ska använda systemet konsekvent. Tränarna betonade också behovet av att kunna se trender över tid, både för prestation och återhämtning, snarare än att enbart analysera enskilda pass.

Slutligen framkom att AI bör fungera som ett beslutsstöd snarare än ersätta tränaren. Tränarna ville kunna behålla kontrollen över träningsplaneringen, men få hjälp med att uppmärksamma risker och föreslå justeringar. En särskilt efterfrågad funktion var att tränaren ska kunna lägga in ett grundpass som systemet sedan individanpassar automatiskt utifrån varje aktivs historik, dagsform och belastning.

### 4.2 Tempomodell

Datan som användes till tempomodellen var inhämtad från Garmin Connect. CSV-filen som extraherades från Garmin Connect innehöll genomförda aktiviteter som rader, med kolumner för

datavärden. Dessa var bland annat snittpuls, maxpuls, distans, stigning, tidsåtgång, maxtempo, snittempo.

**Strava:** Strava undersöktes utefter aspekterna teknisk extraherbarhet och juridiskt tillstånd, där det föll på juridiskt tillstånd i och med att under punkt 2.14.4 i Stravas användningsavtal står det specifikt att man ej får träna modeller relaterade till artificiell intelligens med hjälp av deras API [29].

**Garmin Activity API:** Projektgruppen ansökte om tillåtelse att få använda Garmin Activity API men blev nekade på grund av avsaknad av användarpolicy.

**GarminDB:** Data laddas ner från Garmin i JSON-format för att sedan konverteras till Pythonbiblioteket `pandas` dataframe. Antingen gör projektgruppen det åt utövaren, eller så gör utövaren det själv. För att projektgruppen skall kunna göra det kräver det tillgång till användares egna Garminkonton, vilket inte är tillåtet under Garmins användarpolicy [7]. Detta bryter mot den juridiska aspekten. Det fanns alternativ som inte gjorde det, det ena var att utövaren själv använde GarminDB för att ladda ner sin data, men detta krävde teknisk kunskap. Det andra var att utveckla en applikation som gör detta åt utövaren. Detta bedömdes vara ett för stort projekt för att hamna inom arbetets tidsramar.

Valet av manuell export från Garmin Connect innebar att modellen begränsades till de parametrar som fanns tillgängliga i standardrapporten. Utifrån dessa förutsättningar valdes de fysiologiskt mest relevanta variablerna för att prediktera löptempo. Den resulterande modellen formulerades som:

$$\text{Tempo} \sim \text{Stigning/km} + \text{Nedför/km} + \text{IF}.$$

koefficienterna för modellen redovisas i Tabell 4.1.

**Tabell 4.1:** Parametervärdena för tempomodellen samt dess statistiska signifikans.

Variabel	Koefficient	Standardfel	p-värde
Intercept	551,57	38,76	0,00
Stigning/km	1,03	0,62	0,01
Nedför/km	2,68	1,27	0,03
IF	-301,64	65,93	0,00

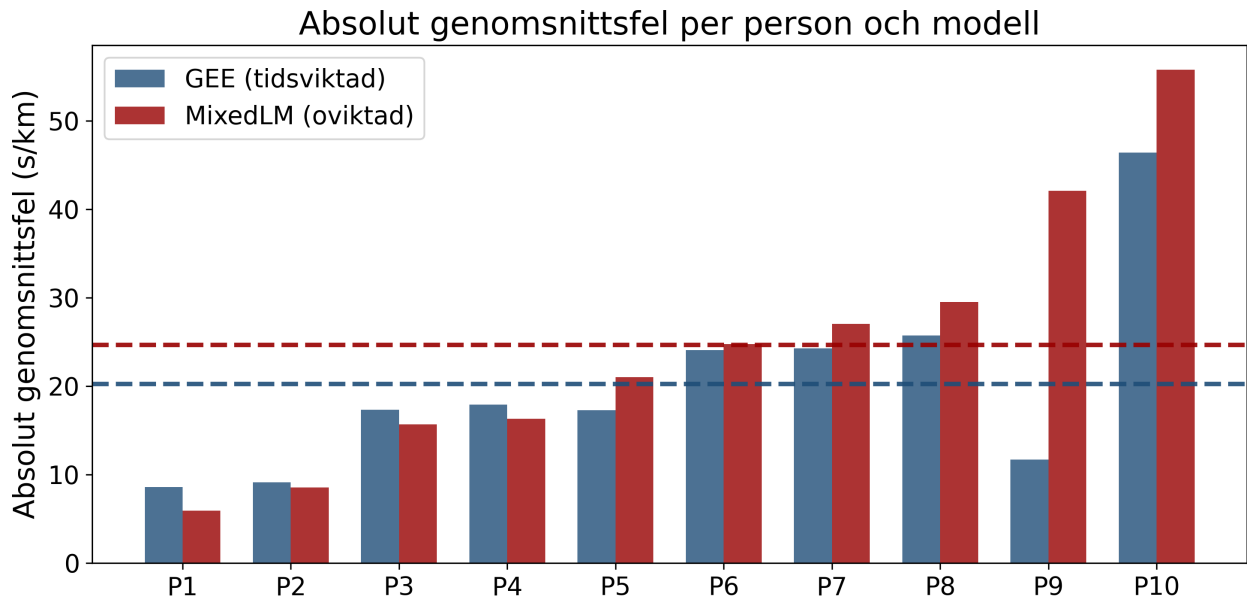
IF uppvisar en negativ koefficient (-301,6) och är statistiskt signifikant ( $p < 0,001$ ). Även Nedför/km har en positiv och signifikant koefficient (2,68,  $p = 0,034$ ). Stigning/km har en positiv koefficient (1,03), men är inte statistiskt signifikant på 5 %-nivån ( $p = 0,099$ ).

Den korrelationsstruktur som användes var autoregressiv då den gav lägst QIC-värde. Resultaten för beräkning av QIC-värdena för olika korrelationsstrukturerna presenteras i Tabell 4.2.

**Tabell 4.2:** QIC-värden för olika korrelationsstrukturer för tempomodellen.

Korrelationsstruktur	QIC-värde
Oberoende	2131,72
Utbytbar	2400,33
Autoregressiv	2083,33

Validering av absolut snittfel per kilometer för de senaste fem passen för varje löpare med och utan tidviktning gav resultat enligt Figur 4.1. Då den tidsviktade GEE-modellen gav ett snittfel på 20,23 s mot LMMs 24,66 s på de fem senaste passen valdes GEE-modellen.



**Figur 4.1:** Absolut snittfel i tempo för de olika löparna på de fem senaste passen i sekunder/kilometer.

### 4.3 Hämtningsförhöjd generering

Jämförelsen mellan TF-IDF och vektorrepresentationer visade att metoderna i stor utsträckning hämtade olika textsegment. Av totalt 62 valda segment var endast två gemensamma mellan metoderna. Resterande segment fördelades jämnt, med 30 unika för TF-IDF och 30 unika för vektorrepresentationer. Sedan gav också vektorrepresentationerna genomgående högre cosinuslikhet än TF-IDF. För sökrågan ”medeldistans uppbyggnad träningsplanering veckoschema” låg resultaten från den vektorbaserade metoden mellan 0,7666 och 0,7123, medan resultaten från TF-IDF låg mellan 0,1124 och 0,0577. I den kvalitativa genomgången framkom att TF-IDF oftare hämtade allmänna och teoretiska textsegment, som beskrivningar av utvecklingsstadier eller klassificeringar av uthållighetsformer, medan vektorrepresentationer oftare hämtade konkret innehåll i form av veckoplaner, exempel på tröskel- och intervallpass samt praktiska rekommendationer. Det detaljerade utfallet från jämförelsen redovisas i filen `comparison_results_detailed.txt` i projektets GitHub-arkiv [2].

### 4.4 Träningsplaneringssystemet

Webbplattformen resulterade i att fungera som en prototyp för att demonstrera den utvecklade AI-baserade träningsplaneringen med tempomodell, riktad både mot idrottare och tränare. Den omfattar separata vyer för de två användargrupperna, vilket gör det möjligt för idrottare att registrera och följa sin träning samtidigt som tränare kan administrera och överblicka flera aktivas träningsupplägg. Via hemsidan kan användare registrera profiler, logga genomförda träningspass samt generera träningsplaneringar på månadsnivå, där antal träningspass per vecka

kan specificeras. Systemet utformades för att kunna samla träningsrelaterad information på ett ställe och samtidigt erbjuda stöd för individanpassad planering.

I Appendix B visas ett exempel på en månadsplanering från webbplattformen samt detaljer kring passen under den första veckan. Exemplet avser löpare P4 från den enkätbaserade utövarstudien.

Utvärderingen av systemet kan delas upp i två dimensioner: bedömningar gjorda dels av en erfaren tränare, dels av löpare, där varje grupp utvärderade både träningsuppläggets logik och tempomodellens rekommendationer. De genererade träningsuppläggen som användes under utvärderingarna kan ses i `Genererade_träningsupplägg`-mappen i projektets GitHub-arkiv [2].

Löparnas utvärdering samlades in via enkäter. Svaren till frågorna om GEE-modellens tempoprediktionsförmåga redovisas i Tabell 4.3.

**Tabell 4.3:** Sammanfattning av utvärdering gjord av fyra personer gällande tempomodellens precision. P1, P3 och P4 har bedömt ett månadsschema utan att genomföra passen medan P7 testat ett pass i varje passkategori.

Fråga och tema	P3	P1	P4	P7	Kommentar
1, Tröskeltempo	5	4	2	3	P1 och P3 upplevde tempot som för högt, medan P4 noterade att föreslaget tempo var lägre än nuvarande träningsfart.
2, Distanstempo	3	4	2	3	P1 upplevde tempot som högt. P4 noterade avsaknad av pass över 21 km. P3 noterade liten temposkillnad mellan distanspass och återhämtningspass.
3, Återhämtningstempo	3	3	1	2	P4 och P7 beskrev tempot som onaturligt långsamt. P1 efterfrågade längre passlängd för denna kategori.
4, Kupering	2	4	4	–	P3 noterade för högt tempo i övre intervallet. P4 angav att terrängalternativen var svåra att tolka/bedöma.

Enkätsvaren angående träningsupplägg och logik presenteras i Tabell 4.4.

**Tabell 4.4:** Sammanfattning av utvärdering gjord av fyra personer gällande träningsuppläggets logik och struktur. P1, P3 och P4 har bedömt ett månadsschema utan att genomföra passen medan P7 testat ett pass i varje passkategori.

Fråga och tema	P3	P1	P4	P7	Kommentar
5, Historik	2	3	4	5	P3 påpekar att långpassen är för korta och att den totala distansen för en vecka är i underkant på hans nuvarande.
6, Balans	4	2	4	5	P1 skriver att hen generellt springer långsammare både på tröskelpassen och långpassen så fartskillnaderna är ungefär samma.
7, Upp/nedvarvning	Ja	Nej	Ja	Nej	P7 och P3 ansåg upp- och nedvarvning vara långa, medan P1 efterfrågade längre uppvärmning inför specifika högintensiva pass.
8, Progression	3	3	2	–	P4 uttrycker att tävlingsdistansen saknas. P3 uttrycker en saknad av en veckovis översikt men anser att det är bra att träningsvolymen minskar fjärde veckan för återhämtning.
9, Logik	5	1	3	4	P1 och P4 angav specifikt att långpassen bör läggas på helger för genomförbarhet.
10, Användbarhet	2	3	3	4	P1, P3 och P4 uppgav att de redan följer befintliga träningsprogrammen vilket påverkade sannolikheten att byta till den genererade planen.

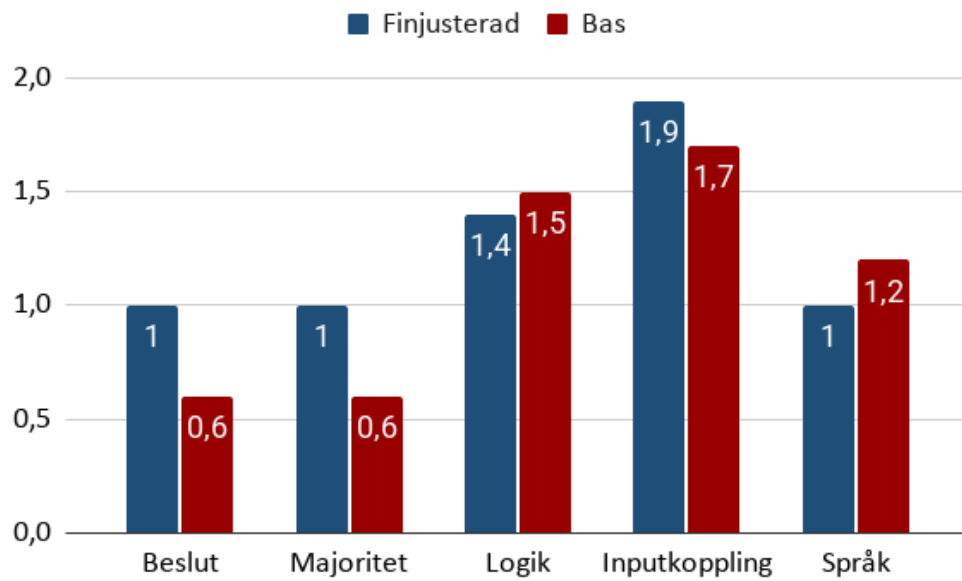
Tränaranalysen gjordes i en intervju och den indikerade att den AI-genererade planeringen överlag framstod som träningsmässigt rimlig och i flera avseenden relativt nära vad tränaren själv hade kunnat planera. Flera av passen uppfattades ligga i linje med löparnas nivåer och passens syften, och tränaren lyfte särskilt fram att planeringen innehöll genomtänkt uppvärmning och nedvarvning. Samtidigt menade tränaren att vissa justeringar hade kunnat stärka upplägget ytterligare. Den viktigaste förändringen hen själv hade gjort var att lägga till ett högintensivt intervallpass varannan vecka, exempelvis kortare intervaller med längre vila. Tränaren hade även lagt in ett långpass på cirka två timmar varje helg.

När det gäller tempomodellens rekommendationer framkom i tränarens bedömning vissa brister. För vissa pass uppfattades de rekommenderade temporna som något för höga, särskilt i huvuddelarna av tröskelpassen och i de tempoförslag som gavs för bana som underlag. I ett exempel efter skada ansågs det rekommenderade tempot vara rimligt utifrån passets syfte, men för just den aktuella löparen menade tränaren att det blev för långsamt, eftersom hen vet att löparen vanligtvis springer snabbare än rekommenderat. Hela tränaranalysen kan ses i filen `tranarutvardering_med_svar.txt` i projektets GitHub-arkiv [2].

## 4.5 Finjusterad språkmodell för utövaråterkopplings

Som en del av arbetet utvärderades språkmodellen för utövaråterkoppling som genererar träningsrekommendationer baserat på användarens dagsform och aktuella tillstånd. Resultatet av denna utvärdering presenteras i Figur 4.2, där de genomsnittliga poängen per kriterium för respektive modell visas. De individuella poängen för respektive scenario och modell är sammanställda i en Excel-fil `evaluation_scores.xlsx` som finns tillgänglig i projektets GitHub-arkiv

[2].



**Figur 4.2:** Genomsnittlig poäng per kriterium för den finjusterade modellen och basmodellen.

Baserat på de totala poängen per scenario uppnådde den finjusterade modellen ett högre genomsnittligt resultat på 6,2 av 10 jämfört med basmodellens resultat på 5,5. Vidare presterade den finjusterade modellen bättre i 10 av 20 fall medan basmodellen presterade bättre i 7 fall. Modellerna presterade likvärdigt i de återstående 3 fallen.

I Figur 4.2 framgår det att den finjusterade modellen uppnådde högre genomsnittliga poäng gentemot basmodellen för kriterierna beslut, majoritet och inputkoppling. Skillnaderna är som störst för beslut och majoritet där den finjusterade modellen presterar bättre. Däremot uppnådde basmodellen högre genomsnittliga poäng för logik och språk.

# 5

## Diskussion

I behovsanalysen belystes en skillnad mellan olika discipliners databehov. Medan det inom löpning finns mycket lättillgänglig data via pulsklockor, kräver grenar som sprint och hopp andra mätetal, exempelvis kraftutvecklingskurvor. Detta föll utanför projektets ramar men det finns potential för framtida utveckling om sätt att registrera mer relevant data implementeras.

För att adressera de identifierade behoven inom löpning togs en webbplattform fram som fungerat som en sammanlänkande prototyp snarare än en färdig produkt. Detta innebär att den fullständiga användarupplevelsen, inklusive den tänkta chatten baserad på en finjusterad språkmodell, återstår att utvärdera. Plattformen har dock tillämpat principer om enkelhet, vilket tränare starkt efterfrågat. Systemet har därför designats med en låg instegströskel där få parametrar är obligatoriska. Detta medför dock en avvägning; även om tröskeln för att börja använda systemet är låg, krävs historisk träningsdata för att tempomodellens funktionalitet ska kunna nyttjas fullt ut. Här finns en betydande potential i framtida integrationer med bärbar teknik, där passiv datainsamling av exempelvis sömnkvalitet och vilopuls skulle kunna möjliggöra en högre grad av dynamisk individanpassning utan att öka användarens administrativa börda.

Något som också efterfrågades av tränare var stöd i grundplaneringen och i nuläget automatiseras denna process helt i plattformen. Detta möjliggör enklare testning och gör hemsidan mer användbar för en bredare målgrupp. Det finns även en ”tränar-vy” som syftar till att återge kontrollen till tränaren för finjustering, men tidsramen förhindrade en iterativ utvecklingsprocess tillsammans med tränare. Detta är dock något som går att vidareutveckla i framtiden.

En betydande begränsning är studiens smala testunderlag, som främst består av studenter och motionärer i medelåldern. Avsaknaden av data från exempelvis ungdomar innebär att resultaten från tempomodellen inte nödvändigtvis kan generaliseras till denna grupp, vars fysiologiska förutsättningar skiljer sig markant. Dock har regler och scenarion för ungdomar varit med under utveckling så det hade fortfarande kunnat ge relevanta resultat men det är inte något som kan bekräftas av denna studie.

Arbetets styrka ligger i hur de tekniska modulerna kompletterar varandra: RAG-modulen bidrar med domänkunskap, reglerna med säkerhet och tempomodellen med personlig anpassning. Ytterligare hade den finjusterade språkmodellen kunnat innebära ett mer komplett stöd för träningsplanering, uppföljning och rådgivning om det hade varit sammankopplat med resterande delar. Systemet är dock, som visats, känsligt för brister i enskilda komponenter. För att ge en djupare förståelse för dessa beroenden följer nedan en detaljerad analys av varje moduls specifika styrkor och svagheter.

### 5.1 Regressionsbaserad tempomodell

Under denna rubrik diskuteras tempomodellen uppdelat i dataunderlagets begränsningar och påverkan, samt den statistiska modellen.

### 5.1.1 Dataunderlagets begränsningar

I utvärdering framkom det att tempoangivelserna ibland upplevdes som för höga (främst i högintensiva pass) och ibland upplevdes som för låga (främst i lågintensiva pass). En möjlig förklaring är att modellen ännu inte är tillräckligt träffsäker men det kan också vara en konsekvens av en för liten testgrupp. Just nu genomfördes testerna enbart på fyra personer och därför påverkar personliga preferenser resultatet mycket. Någon kan till exempel uppleva att ett pass är för långsamt även om det egentligen hade varit ett bättre tempo för den personen att hålla. Med det begränsade underlag som finns går det helt enkelt inte att säga hur väl modellen stämmer.

Modellen tränades även på ett begränsat dataunderlag, både i antal personer, variation och detalj. Datakällan som användes, Garmin Connect, gav tillgång till mätdata som maxtempo men på grund av tekniska problem med utrustningen är detta oftast missvisande. Alltså kunde endast aggregerad passdata såsom medelpuls och medelhastighet användas och variationer inom passen kunde inte analyseras. Detta påverkade främst högintensiva pass då de ofta består av intervaller eller åtminstone en uppvärmning och nedvarvning med stor intensitet och temposkillnad mot huvudpasset. Om en mer detaljerad datakälla hade använts hade tempomodellen sannolikt presterat bättre på intervallpass.

Datamängden dominerades dessutom av män i medelåldern eller i 20-årsåldern, vilket sannolikt begränsade modellens förmåga att generalisera till andra typer av löpare. Resultatet pekar därför på att modellen sannolikt skulle kunna förbättras genom träning på data från fler individer med större variation i exempelvis kön, ålder, prestationsnivå och träningsbakgrund.

CSV-formatet som Garmin Connect använder för dataexport passar väl för ändamålet. Det finns mängder av bra sätt att hantera CSV-filer på i Python så det är lätt att använda men ett problem utöver begränsade detaljen är att det krävde att utövare manuellt extraherade data och skickade det till projektgruppen. Detta behövde göras varje gång en ny aktivitet utförts, det fanns inget smidigare sätt att uppdatera med nya aktiviteter i prototypen. Det fungerade för denna förstudie men är inte skalbart för framtida utveckling med större användarbas.

En annan möjlig datakälla är Garmin Activity API. Detta alternativ valdes inte, som nämnt i resultatet, på grund av att projektgruppen blev nekad användning. Annars hade detta alternativ med största sannolikhet valts och vid en vidareutveckling av projektet, hade det varit en prioritet att få tillåtelse att använda det. Det kommer färdigutvecklat med funktioner för att ta ut data och det ger tillgång till full aktivitetsdata vilket gör det till ett bättre alternativ än att extrahera CSV-filer från Garmin Connect. En annan fördel är att det innehåller logik för att hantera godkännande av nya användare. Detta framstår därför som smidigare att använda samtidigt som den ger den önskade kvalitén på data.

Totalt användes 1779 löppass från sammanlagt tio löpare för att träna tempomodellen. Dessa löppass var dock inte jämnt fördelade över alla löpare. Det största datasetet för en löpare var 684 löppass och det minsta var 16. Det betyder att vissa individer hade en betydligt större påverkan på modellutformningen. Utvärderingen i Tabell 4.1 genomfördes dock på de fem senaste passen per person vilket innebar att ingen löpare vägde tyngre än någon annan.

### 5.1.2 Modellens parametrar och fysiologisk relevans

De parametrar som inkluderades i den slutgiltiga modellen var IF, stigning/km och nedför/km där IF påverkade modellen mest med en koefficient på -301,64. Detta är även fysiologiskt grundat då ökad intensitet sänker tempot. En koefficient som saknar en rimlig fysiologisk tolkning

är nedför/km som enligt modellen skulle öka tempot med 2,68 s/km. Denna variabel kan därför ha fungerat som en proxy för andra effekter som inte explicit modellerades.

Väderfaktorer var också något som undersöktes men som inte inkluderades i den slutgiltiga modellen. Detta då de inte gav en signifikant skillnad men framförallt eftersom positionsdata saknades i Garmin Connect-datan och att införa en vädervariabel under ett antagande att alla löppass för alla personer sker på samma plats är ett för starkt antagande. Med mer detaljerad data är detta dock en möjlighet att inkludera i framtida modeller.

### 5.1.3 Tidsviktning och statistisk metod

Resultaten indikerade att tidsviktning främst förbättrade prediktionerna för löpare vars träningsnivå eller träningsmönster förändrats över tid. För individer med stabil och långvarig träningshistorik kunde äldre data däremot fortfarande vara relevanta, vilket i vissa fall gjorde att tidsviktning försämrade prediktionsförmågan. Om man till exempel betraktar P9 så förbättrade en tidsviktning snittfelet från 42,1 till 11,7 s/km.

### 5.1.4 Källor till osäkerhet

Viktigt att notera är att det förekommer dataläckage i valideringen. Modellen har tränats på alla pass och därav även de fem senaste. Resultatet bör alltså betraktas med försiktighet då modellen förutsäger pass den redan sett.

Antagandet att tröskelpulsen motsvarade 95 % av individens högsta uppmätta puls innebär en osäkerhetskälla i modellen. Avvägning gjordes med resonemanget att personer oftast inte når upp till sin allra högsta puls i sin träning. Det här har dock en stor påverkan på resultatet då vissa löpare som är bättre på att ta ut sig mer i sina pass kommer få lite för höga tempon medan andra som inte gör det istället får lägre. Detta kan delvis förklara varför vissa ansåg tempona för låga och andra tempona för höga.

## 5.2 AI-baserad träningsplanering

Tränaranalysen ger en överlag positiv bild av den AI-baserade träningsplaneringen, där planerna uppfattades som träningsmässigt rimliga och i flera fall låg nära vad tränaren själv hade kunnat planera. Tolkningen ska samtidigt göras med försiktighet, eftersom analysen bygger på en enda tränares bedömning. Att tränaren särskilt lyfte fram uppvärmning och nedvarvning som genomtänkta delar tyder ändå på att regelverket lyckades fånga sådant som värdesätts i praktisk träningsplanering. Samtidigt visar de justeringar tränaren själv hade gjort, exempelvis att lägga till ett högintensivt intervallpass varannan vecka och ett långpass på cirka två timmar varje helg, att lösningen inte fullt ut nådde fram till det som en erfaren tränare hade planerat.

Tydligast blev detta i fallet med den skadade löparen, där den lugnare temporekommendationen var rimlig utifrån passets syfte, men där tränaren utifrån sin personkänedom ändå hade valt att sätta tempot något högre. Detta då löparen tenderar att ta ut sig för mycket och det hade varit troligare att hen följde tempot om hen fick en högre temporekomendation. Detta är en typ av individanpassning som ett AI-system har svårt att fånga, eftersom det förutsätter en tränare som känner utövaren tillräckligt väl för att kunna förutse hur denne kommer att tolka och förhålla sig till planen i praktiken. På så sätt pekar arbetet på att manuell planering fortfarande har ett tydligt värde, och att AI snarare bör ses som ett stöd i tränarens arbete än som en ersättning för det.

Utövaranalysen ger i stort en liknande bild, men med tydlig variation mellan deltagarna. Återigen så bör denna utvärdering betraktas som indikationer snarare än något som tillåter generella slutsatser, då den grundar sig på fyra deltagare. Att en plan som på pappret är välbalanserad ändå kan upplevas olika beroende på mottagaren stödjer tränaranalysens iakttagelse om att kännedom om individen är avgörande. Samtidigt framträder det att flera deltagare påpekade brister kopplade just till långpassen. P1 och P4 lyfte att långpassen bör läggas på helger för att vara genomförbara, medan P3 påpekade att långpassen var för korta och att den totala veckodistansen låg i underkant, vilket även tränaren tog upp som en justering hen själv hade gjort. Tillsammans pekar detta på att planeringen av just långpass, både vad gäller placering och omfattning, är en svaghet i lösningen.

### 5.2.1 Regelbaserade system

Systemet för den regelbaserade reservlösningen utgör i nuläget en prototyp som främst demonstrerar hur man hade kunnat garantera ett säkert träningsupplägg även när RAG inte kan användas. För att kunna bedöma om kvaliteten är tillräcklig, och för att vidareutveckla systemet, krävs både utvärdering och fortsatt arbete. Ett nästa steg skulle kunna vara en jämförande studie där tränare bedömer både RAG- och reservlösningens genererade planer blint, samt en analys av hur ofta reservlösningen aktiveras i praktiken. Ytterligare hade man med fördel behövt noggrant kvalitetssäkra de underliggande reglerna och passmallarna i samarbete med tränare, samt att låta reservlösningen anpassas ännu mer efter individ genom fler villkor som gör upplägget mer personligt än i dagsläget.

### 5.2.2 Hämtningsförhöjd generation

TF-IDF och vektorrepresentationer hämtade nästan helt olika textsegment, trots att sökfrågorna innehöll termer som är väl representerade i kunskapsbasen. Detta tyder på att skillnaden inte beror på att TF-IDF saknar matchande ord, utan på hur de två metoderna värderar innehållet. För den aktuella tillämpningen framstår vektorrepresentationer därmed som mer lämpligt, eftersom konkret innehåll som exempel på pass och veckoupplägg ger språkmodellen ett mer direkt underlag att utgå ifrån när den ska generera specifika träningsförslag.

Jämförelsen bygger dock på ett avgränsat material och endast fyra sökfrågor, vilket gör att resultatet inte kan generaliseras till RAG-system i stort. Sökfrågorna konstruerades manuellt för att täcka centrala teman i träningsplanering, men en annan uppsättning frågor hade kunnat ge en delvis annan bild av metodernas styrkor. Den kvalitativa bedömningen av segmentens användbarhet utfördes dessutom subjektivt i klassificeringen av vad som räknas som konkret respektive teoretiskt innehåll. Likhetsvärdena bör också tolkas med försiktighet eftersom de två metoderna producerar värden på olika nivåer. TF-IDF räknar enbart faktiska ordmatchningar, och eftersom ett textsegment bara innehåller en liten del av alla ord i kunskapsbasen blir värdena ofta låga även när innehållet är relevant. Med vektorrepresentationer vägs hela textens innebörd samman och ger därför genomgående högre värden. Det går alltså inte att jämföra likhetsvärdena rakt av mellan metoderna, utan jämförelsen handlar snarare om vilka segment respektive metod väljer ut än om vilken metod som ger högst siffervärde. Sammantaget pekar jämförelsen ändå på att hämtningssteg är en central del att fokusera på vid fortsatt utveckling av liknande lösningar, eftersom valet av metod direkt påverkar vilken typ av information språkmodellen får att arbeta med och därmed också karaktären på de genererade planerna.

### 5.3 Finjusterad språkmodell

Resultatet indikerar att den finjusterade modellen presterar bättre än basmodellen, men att förbättringen är relativt begränsad och inte jämnt fördelad över alla kriterier. De kriterier som visar en positiv effekt är beslut, majoritet och inputkoppling vilket tyder på att finjusteringens förbättring främst berör modellens förmåga att ge tydliga och situationsanpassade rekommendationer.

En möjlig förklaring till de observerade förbättringarna är att struktureringen av träningsdatan haft den avsedda effekten. Resultaten tyder på att modellen, genom träning på konsekvent utformade svar med tydlig koppling mellan indatan och rekommendationen, i större utsträckning fattar konkreta och situationsanpassade beslut baserade på relevant information. Detta framgår även i den jämförande analysen, där basmodellen oftare gav generella resonemang medan den finjusterade modellen i fler fall gav direkta förslag. Resultatet ligger i linje med syftet med finjusteringen och indikerar att den valda databehandlingen bidragit till de observerade förbättringarna.

Analysen av modellens svar visar även återkommande mönster på brister. Modellen har en tydlig tendens att vara för försiktig i sina rekommendationer, där den ofta föreslår att minska belastningen, även i fall där tränarnas rekommendationer är att genomföra passet som planerat. En trolig förklaring till detta är att en stor del av träningsdatan bestod av scenarier som var utformade på ett sätt där minskad belastning ofta framstod som en rimlig rekommendation. Att modellerna instruerades att agera som försiktiga tränare kan också ha påverkat resultaten. Detta ansågs dock vara en viktig del av implementationen för att i största möjliga utsträckning undvika att modellen genererade träningsrekommendationer som skulle kunna innebära en skaderisk. Detta kan ha lett till att modellen lärt sig en förenklad beslutslogik där negativa signaler i indatan ofta resulterar i reducerad belastning. Samtidigt kan den begränsade mängden träningsdata ha bidragit till att modellen inte fullt ut lärt sig att hantera mer nyanserade situationer, vilket både kan förklara de observerade bristerna och varför förbättringarna jämfört med basmodellen är relativt små.

En viktig aspekt att beakta är att resultaten bör tolkas med viss försiktighet. Utvärderingen baseras på ett begränsat antal scenarier (20 fall), och den finjusterade modellen tränades på en relativt liten datamängd (298 datapunkter), vilket innebär att resultaten inte nödvändigtvis går att generalisera. Vidare innehåller bedömningen av modellerna ett visst mått av subjektivitet, även om det har försökt minimeras genom användning av fördefinierade kriterier och en poängskala. Dessa två faktorer tyder på att resultatet bör ses som indikativa snarare än definitiva. För framtida arbete skulle en större och mer varierad datamängd kunna bidra till mer tillförlitliga resultat. Detta skulle även kunna leda till en tydligare skillnad i prestation mellan den finjusterade modellen och basmodellen. En utvärdering baserad på flera oberoende bedömare skulle dessutom kunna stärka bedömningens trovärdighet.

### 5.4 Samhälleliga och etiska aspekter

Aspekterna som behöver evalueras i relation till detta projekt är i huvudsak typen av data, användningen av datan, samt användningen av AI och vem som bär ansvaret när AI:n ger skadliga råd.

Att datan som tempomodellen är tränad på huvudsakligen består av män i 20- och medelåldern kan leda till att den presterar sämre och ger dåliga råd till personer som inte passar in i den

gruppen. Detta är något som bör tas i åtanke och utövaren dubbelkollar med fördel huruvida tempot är rimligt.

En annan etisk aspekt rör datahantering. I nuvarande prototyp skickas kontextunderlaget till språkmodellen Claude Sonnet 4.5 via Anthropic API för att generera träningsupplägg. Det betyder att utövarens träningsdata skickas till Anthropic's servrar, som ligger utanför EU. Även om datan inte innehåller namn eller personnummer finns det ändå en risk att en person identifieras utifrån mönster i datan, till exempel genom tävlingsresultat. Delar av datan klassas som hälsouppgifter och omfattas då av särskilt skydd enligt Art. 9 GDPR [5]. Att överföra data utanför EU väcker också frågor kring GDPR eftersom reglerna för sådana tredjelandsöverföringar är strikta [4].

För detta arbete anses nuvarande datahantering acceptabel, då utövaren har gett sitt samtycke, men om systemet skulle utvecklas till en färdig produkt skulle datahanteringen behöva göras om. Möjliga lösningar är att köra modellen på servrar inom EU, hosta den lokalt eller använda en mindre modell som kan köras direkt på användarens egen enhet.

Användning av AI i projektet har flera samhällliga och etiska aspekter som bör övervägas. Den första är miljöaspekten, där AI är energikrävande och därmed påverkar miljön negativt. Detta syns i att AI står för 2,5–3,7 % av de globala växthusgasutsläppen [20]. Även om AI står för en stor andel av globala utsläppen betyder det dock inte att projektets användning av AI har en så stor negativ påverkan att det inte borde användas. Speciellt eftersom projektet skulle bidra positivt till samhället genom att ge fler tillgång till personanpassade träningsupplägg som annars ofta ligger bakom en betalvägg. Dessutom är användningen av AI per person enligt flera rapporter fortfarande försumbar i termer av vattenanvändning och energiförbrukning. Enligt en rapport Google publicerat [32] är en prompt till Geminis energianvändning ekvivalent med att titta på TV i under nio sekunder. Detta anses tillräckligt lite för att vara försumbart.

Den andra aspekten handlar om ansvar, mer specifikt vem som bär ansvaret för om AI:n ger dåliga eller rentav skadliga råd och huruvida AI överhuvudtaget borde ha rollen som rådgivare där den kan resultera i skada. Här anser gruppen att ansvaret ligger hos utgivaren, vilket i detta fall är gruppen själva. Därför rekommenderas passen att granskas i samråd med en tränare, som kan avgöra huruvida förslaget är rimligt eller ej. Eftersom träning utan rådgivning kan innebära risker, bedömer gruppen att AI-genererade råd inte förvärrar situationen, förutsatt att utövaren samtidigt lyssnar på sin egen kropp. Tvärtom kan det förhoppningsvis utgöra en förbättring.

# 6

## Slutsats

Projektet demonstrerar att det är tekniskt genomförbart att realisera ett AI-baserat beslutsstöd för träningsplanering genom att integrera en regressionsmodell, RAG och finjusterad språkmodell. I förhållande till de uppställda delmålen har en regressionsbaserad tempomodell framgångsrikt utvecklats för att prediktera individanpassade tempoangivelser, och en RAG-baserad modell har visat sig kapabel att generera strukturerade träningsupplägg. Dessa har integrerats och sammankopplats i en webbplattform, för smidig interaktion. Den finjusterade språkmodellen uppvisade en förbättrad förmåga att ge anpassade rekommendationer jämfört med basmodellen, även om effekten var begränsad av ett litet träningsunderlag och en tendens till överdriven försiktighet i träningen.

Genom att lösningen utgår från användarens profil och träningshistorik skapas förutsättningar för en högre grad av individanpassade rekommendationer än vad generell träningsplaner erbjuder. Studien visar dock att AI i nuläget bör betraktas som ett komplement till, snarare än ersättning för mänsklig expertis. Särskilt tydligt blir det vid skadeproblematik och personkännedom, där en mänsklig tränares omdöme förblir centralt.

Sammantaget utgör de framtagna systemen en fungerande prototyp som belyser potentialen i att automatisera delar i träningsplanering. För att nå en fullständigt tillförlitlig produkt krävs dock mer omfattande datainsamling från diversifierade användargrupper, tätare integration mellan systemets moduler samt fortsatta användartester för att optimera samspelet mellan tränare, utövare och AI.

# Referenser

- [1] Anthropic. “Build with Claude - Overview”. 2026. URL: <https://platform.claude.com/docs/en/build-with-claude/overview> (hämtad 2026-05-08).
- [2] Christopher Boissier m. fl. “AI-baserat träningsstöd för löpning”. GitHub-arkiv. 2026. URL: <https://github.com/corneliaivanovic/Kandidat.git> (hämtad 2026-05-12).
- [3] Tim Dettmers m. fl. “QLoRA: Efficient Finetuning of Quantized LLMs”. 2023. DOI: 10.48550/arXiv.2305.14314.
- [4] European Parliament and Council of the European Union. “Regulation (EU) 2016/679 (General Data Protection Regulation), Article 44 – General principle for transfers”. 2016. URL: <https://gdpr-info.eu/art-44-gdpr/> (hämtad 2026-05-08).
- [5] European Parliament and Council of the European Union. “Regulation (EU) 2016/679 (General Data Protection Regulation), Article 9 – Processing of special categories of personal data”. 2016. URL: <https://gdpr-info.eu/art-9-gdpr/> (hämtad 2026-05-08).
- [6] Joe Friel. “Estimating Training Stress Score (TSS)”. TrainingPeaks. 2026. URL: <https://www.trainingpeaks.com/learn/articles/estimating-training-stress-score-tss/> (hämtad 2026-04-30).
- [7] Garmin. “Terms of Use”. 2026. URL: <https://www.garmin.it.com/terms-of-use/> (hämtad 2026-03-31).
- [8] Garmin Developer. “Activity API”. 2026. URL: <https://developer.garmin.com/gc-developer-program/activity-api/> (hämtad 2026-04-14).
- [9] Tom Goetz. “GarminDB”. GitHub-arkiv. 2026. URL: <https://github.com/tcgoetz/GarminDB> (hämtad 2026-04-14).
- [10] Aaron Grattafiori m. fl. “The Llama 3 Herd of Models”. 2024. DOI: 10.48550/arXiv.2407.21783.
- [11] Edward J. Hu m. fl. “LoRA: Low-Rank Adaptation of Large Language Models”. 2021. DOI: 10.48550/arXiv.2106.09685.
- [12] IBM. “Chunking strategies for RAG with LangChain and watsonx.ai”. URL: <https://www.ibm.com/think/tutorials/chunking-strategies-for-rag-with-langchain-watsonx-ai> (hämtad 2026-04-22).
- [13] IBM. “What is Machine Learning?” 2026. URL: <https://www.ibm.com/think/topics/machine-learning> (hämtad 2026-04-21).
- [14] Wenxi Li m. fl. “A review of artificial intelligence for sports: Technologies and applications”. I: *Intelligent Sports and Health* 1.3 (2025), s. 113–126. ISSN: 3050-5445. DOI: 10.1016/j.ish.2025.05.001.
- [15] Han Liu, Alexander Gegov och Mihaela Cocea. “Rule-based systems: a granular computing perspective”. I: *Granular Computing* 1 (2016), s. 259–274. DOI: 10.1007/s41066-016-0021-6.
- [16] Davood Mazinianian och Nikolaos Tsantalis. “CSSDev: Refactoring duplication in cascading style sheets”. I: *Proceedings of the 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. 2017, s. 63–66. DOI: 10.1109/ICSE-C.2017.7.
- [17] Wei Meng m. fl. “Application of p-Value Approach in Inversion of Initial Geo-Stress Field of Rock Masses”. I: *Zhongguo Tiedao Kexue/China Railway Science* 42.1 (2021), s. 71–79. DOI: 10.3969/j.issn.1001-4632.2021.01.09. URL: <https://www.scopus.com/inwar>

- d/record.uri?eid=2-s2.0-85102408691&doi=10.3969%2fj.issn.1001-4632.2021.01.09&partnerID=40&md5=31a38516339cc7ca028c0e696ccfb79b.
- [18] Microsoft Azure. “Vad är hämtningsförhöjd generation (RAG)?” Microsoft. 2026. URL: <https://azure.microsoft.com/sv-se/resources/cloud-computing-dictionary/what-is-retrieval-augmented-generation-rag> (hämtad 2026-04-22).
- [19] Dejan Mirkovic och Peter Wikström. “Friidrottens allmänna träningslära”. Opublicerat internt dokument mottaget via personlig kommunikation. Relevanta utdrag finns tillgängliga via projektets GitHub-arkiv [2]. 2017.
- [20] Haizhi Mu m. fl. “Artificial Intelligence and its Carbon Footprint”. I: *2025 IEEE Systems and Information Engineering Design Symposium, SIEDS 2025*. 2025, s. 31–35. DOI: 10.1109/SIEDS65500.2025.11021149. URL: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-105008421183&doi=10.1109%2fSIEDS65500.2025.11021149&partnerID=40&md5=08111152749b8359bac2f84235e25991>.
- [21] M. R. Mufid m. fl. “Design an MVC Model using Python for Flask Framework Development”. I: *2019 International Electronics Symposium (IES)*. Surabaya, Indonesia, 2019, s. 214–219. DOI: 10.1109/ELECSYM.2019.8901656.
- [22] Network of the National Library of Medicine. “Web Scraping”. URL: <https://www.nlm.nih.gov/resources/data/data-glossary/web-scraping> (hämtad 2026-04-22).
- [23] PyImageSearch. “TF-IDF vs. Embeddings: From Keywords to Semantic Search”. 2026. URL: <https://pyimagesearch.com/2026/02/09/tf-idf-vs-embeddings-from-keywords-to-semantic-search/> (hämtad 2026-04-22).
- [24] Anders Rydén. “Träning i uppbyggnadsstadiet – Friidrott – Medel- och långdistans”. PDF-utdrag ur kursmaterialet *Träning i uppbyggnadsstadiet: friidrott 14–17 år*. Relevanta utdrag finns tillgängliga via projektets GitHub-arkiv [2]. 2019.
- [25] Leon Shklar och Rich Rosen. *Web Application Architecture: Principles, Protocols and Practices*. 2. utg. John Wiley & Sons, 2009. ISBN: 9780470518601. URL: <https://download.e-bookshelf.de/download/0000/5960/33/L-G-0000596033-0002363371.pdf> (hämtad 2026-04-19).
- [26] Justine Shults m. fl. “A comparison of several approaches for choosing between working correlation structures in generalized estimating equation analysis of longitudinal binary data”. I: *Statistics in Medicine* 28.18 (2009), s. 2338–2355. DOI: 10.1002/sim.3622.
- [27] SQLite Consortium. “Appropriate Uses For SQLite”. URL: <https://sqlite.org/whentouse.html> (hämtad 2026-05-08).
- [28] statsmodels developers. “Linear Mixed Effects Models”. 2026. URL: [https://www.statsmodels.org/stable/mixed\\_linear.html](https://www.statsmodels.org/stable/mixed_linear.html) (hämtad 2026-05-07).
- [29] Strava. “API Agreement”. 2026. URL: <https://www.strava.com/legal/api> (hämtad 2026-04-14).
- [30] Svenska Friidrottsförbundet. “Löptränare – utdrag ur utkast till självstudier”. Opublicerat internt dokument mottaget via personlig kommunikation. Relevanta utdrag finns tillgängliga via projektets GitHub-arkiv [2]. 2025.
- [31] W3Schools. “What is HTML?” 2026. URL: [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp) (hämtad 2026-04-21).
- [32] Amin Vahdat och Jeff Dean. “How much energy does Google’s AI use? We did the math”. 2025. URL: <https://cloud.google.com/blog/products/infrastructure/measuring-the-environmental-impact-of-ai-inference> (hämtad 2026-05-08).
- [33] Ashish Vaswani m. fl. “Attention Is All You Need”. 2023. DOI: 10.48550/arXiv.1706.03762.

# A

## Intervjufrågor

- Är det okej att vi spelar in samtalet?

### Bakgrund

- Vilka grenar tränar du?
- Vilka åldrar tränar du?
- Tränar du enbart i grupp eller även med enskild coaching?
- Har du gått någon tränarutbildning? Vad har de handlat om?
- Hur mycket tid per vecka lägger du som coach per aktiv? (Både under träning och utanför träning)

### Träningsupplägg & planering

- Hur lägger du upp din träning generellt?
- Hur varierar träningen för olika grenar?
- Hur periodiserar du över året?
- Hur anpassar du träningen inför tävlingar?
- Hur anpassar du träningen efter dagsform?
- Hur ofta justerar du träningsupplägg baserat på dagsform?

### Styrka & kompletterande träning

- Hur värderar du styrketräning för löpare?
- Vad är viktigast inom styrketräning för löpare?
- Använder du färdiga styrkeprogram? Vad fungerar bra och vad fungerar mindre bra?
- Har du några tankar eller idéer kring styrkeprogrammet?

### Kost & återhämtning

- Är kost något som ingår i träningsupplägget?

- Hur pratar du om återhämtning med dina aktiva?
- Hur ser du på sömnens betydelse för prestation?

## Data & träningsklockor

- Vad loggar du idag i träningen?
- Är det något du inte loggar idag som du tror hade kunnat vara bra att logga?
- Hur värderar du metrics som Sleep Score, VO2 max, Body Battery, HRV och liknande från träningsklockor?
- Vilka mätvärden är mest värdefulla för dig som tränare?
- Hur använder du data för att justera träningen?

## Uppföljning & anpassning

- Hur följer du upp hur ett pass har känts för de aktiva?
- Hur anpassar du träningen när någon visar tecken på slitenhet?
- Vad tar mest tid i din planering och uppföljning?
- Vad hade underlättat din planering och uppföljning mest?

## AI & träningsplattform

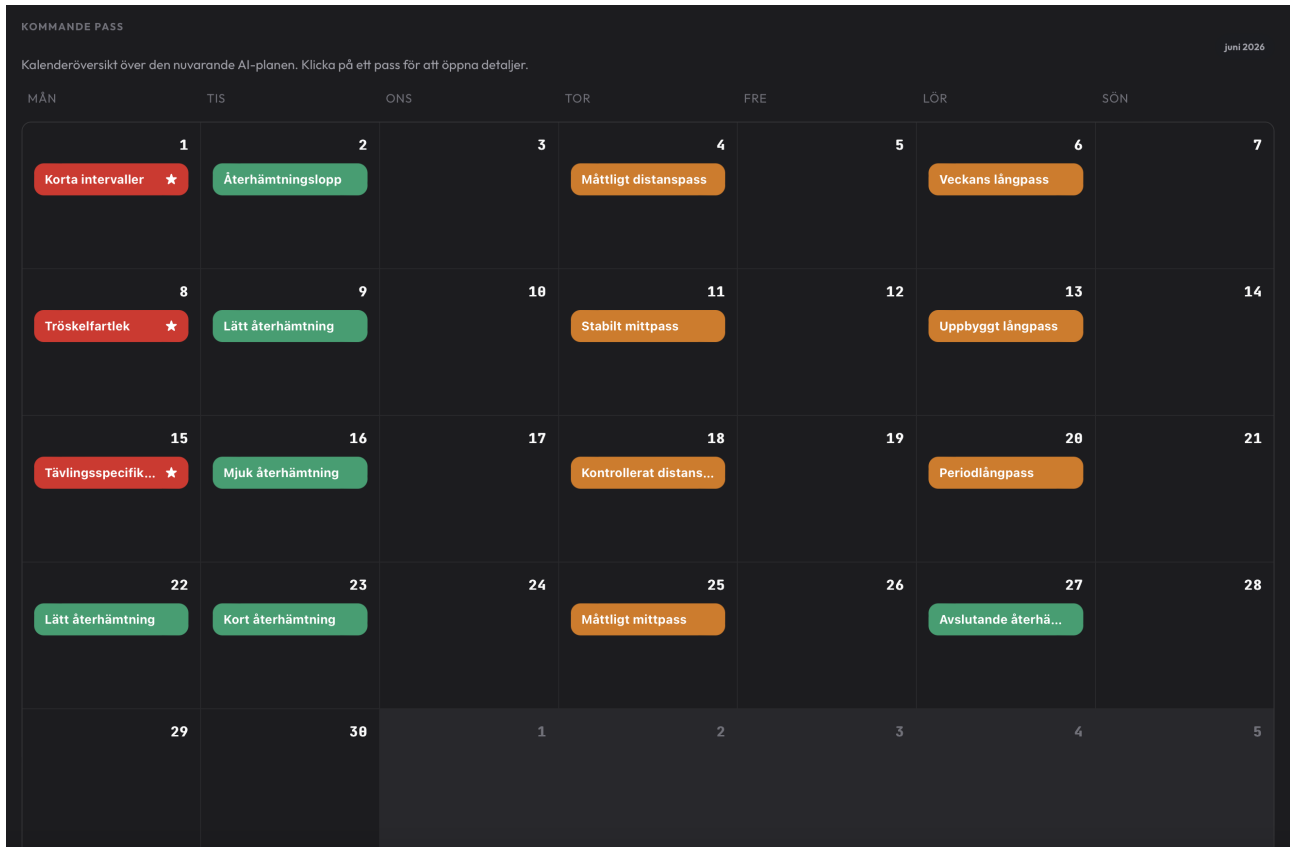
- Hur hade du värderat träningsförslag från en AI?
- Vilka funktioner och hjälpmedel hade du velat se i en träningsplattform?
- Hur skulle en optimal lösning för individanpassning i grupp se ut enligt dig?

## Avslutande frågor

- Får vi använda dina svar i vår rapport och citera dig?
- Skulle du kunna tänka dig att testa vår applikation?
- Hade du varit öppen för att genomföra en enkät på ca 25 flervalsfrågor?

# B

## Exempel på träningsplanering och pass



Figur B.1: Månadsplanering för löpare P4.

★ Nyckelpass
INTENSITET: HÖG
Fas: grundträning
×

## Korta intervaller

Måndag 01 juni · 75 min · uthållighet

### Passdetaljer

Syfte: Utveckla aerob kapacitet och tröskelhastighet genom korta intervaller. Veckans viktigaste kvalitetspass.

Rekommenderat underlag för passet: Bana. Tempovarianterna visar hur samma pass kan justeras om det genomförs på andra underlag.

Uppvärmning: 20 min lugn jogg i 6:30-7:00/km. Sedan 5 min dynamisk rörlighet (knälyft, häckspark, sidosteg, 2x20 meter vardera). Avsluta med 3 stegringslopp à 80 meter där du gradvis ökar farten från lugnt till snabbt men avslappnat, promenadvila 90 sek mellan varje.

Huvudpass: 10x300 meter i 5:05 min/km-5:25 min/km/km (cirka 1:33-1:36 per rep) med 90 sekunders aktiv vila (lätt jogg/promenad). Sikta på jämn känsla genom hela serien, inte för hårt i början. Känslan ska vara ansträngd men kontrollerbar.

Nedvarvning: 15 min mycket lugn jogg i 7:00-7:30/km, avsluta med 5 min statisk stretch för vader, lår och höftböjare.

Förklaring: Stegring/uppbyggnad/progressivt = börja kontrollerat och öka gradvis till snabbt men avslappnat, utan att sprinta fullt. Dynamisk rörlighet = rörliga övningar före passet, till exempel benpendlingar, höftöppnare, utfallssteg och lätta skips.

ALTERNATIVA TEMPON FÖR ANDRA UNDERLAG

**Plan väg:** 5:25 min/km-5:45 min/km

**Kuperat:** 6:01 min/km-6:21 min/km

**Väldigt kuperat:** 6:56 min/km-7:16 min/km

Figur B.2: Exempel på högintensivt pass för löpare P4.

INTENSITET: LÅG
Fas: grundträning
×

## Återhämningslopp

Tisdag 02 juni · 35 min · återhämtning

### Passdetaljer

Syfte: Aktiv återhämtning efter gårdagens hårda intervallpass. Främja blodcirkulation och mjuka upp muskulaturen.

Rekommenderat underlag för passet: Plan väg. Tempovarianterna visar hur samma pass kan justeras om det genomförs på andra underlag.

Uppvärmning: 5 min promenad/mycket lätt jogg för att komma igång.

Huvudpass: 25 min kontinuerlig lugn jogg i 6:45 min/km-7:15 min/km/km. Känslan ska vara lätt och konversationstempo, inga pulshöjningar. Fokus på avslappnad löpteknik med mjukt fotisätt.

Nedvarvning: 5 min promenad och lätt rörlighetssvängningar för benen.

ALTERNATIVA TEMPON FÖR ANDRA UNDERLAG

**Bana:** 6:25 min/km-6:55 min/km

**Kuperat:** 7:20 min/km-7:50 min/km

**Väldigt kuperat:** 8:16 min/km-8:46 min/km

Figur B.3: Exempel på lågintensivt pass för löpare P4.

**INTENSITET: MEDEL** Fas: grundträning

### Måttligt distanspass

Torsdag 04 juni · 55 min · uthållighet

**Passdetaljer**

Syfte: Bygga aerob bas med måttlig volym i kontrollerat tempo. Stödjer återhämtning inför helgens volympass.

Rekommenderat underlag för passet: Kuperat. Tempovarianterna visar hur samma pass kan justeras om det genomförs på andra underlag.

Uppvärmning: 10 min lätt jogg i 6:45-7:00/km för att mjuka upp.

Huvudpass: 40 min kontinuerligt löpande i 6:12 min/km-6:32 min/km/km. Tempot ska kännas kontrollerat och bekvämt, inte utmanande. Du ska kunna prata i korta meningar. I kuperad terräng håll jämn ansträngning snarare än jämnt tempo.

Nedvarvning: 5 min lätt jogg/promenad i 7:00-7:30/km och enkel stretch för vaderna.

ALTERNATIVA TEMPON FÖR ANDRA UNDERLAG

**Bana:** 5:17 min/km-5:37 min/km

**Plan väg:** 5:37 min/km-5:57 min/km

**Väldigt kuperat:** 7:08 min/km-7:28 min/km

Figur B.4: Exempel 1 på ett medelintensivt pass för löpare P4.

**INTENSITET: MEDEL** Fas: grundträning

### Veckans långpass

Lördag 06 juni · 65 min · uthållighet

**Passdetaljer**

Syfte: Bygga uthållighetsgrund genom veckans längsta pass. Grundsten i distansstråningen.

Rekommenderat underlag för passet: Kuperat. Tempovarianterna visar hur samma pass kan justeras om det genomförs på andra underlag.

Uppvärmning: 10 min mycket lugn jogg i 6:45-7:00/km för att värma upp försiktigt.

Huvudpass: 50 min kontinuerligt löpande i 6:20 min/km-6:40 min/km/km. Håll lugnt och stabilt tempo hela vägen, särskilt första halvan. I kuperad terräng kör på känsla och låt tempot variera naturligt med terrängen. Fokus på att bygga volym, inte prestera.

Nedvarvning: 5 min promenad/mycket lätt jogg och 5 min mjuk stretch för hela benen.

ALTERNATIVA TEMPON FÖR ANDRA UNDERLAG

**Bana:** 5:24 min/km-5:44 min/km

**Plan väg:** 5:45 min/km-6:05 min/km

**Väldigt kuperat:** 7:16 min/km-7:36 min/km

Figur B.5: Exempel 2 på ett medelintensivt pass för löpare P4.

# C

## Fördefinierad träningsmall

Den regelbaserade reservlösningen bygger på två komponenter: en veckomall som fördelar svårighetsgrader över veckans dagar, och en passbank organiserad efter gren, träningsfas och svårighetsgrad. Den fullständiga implementationen finns i `ai_schedule.py` i GitHub-arkivet [2].

### C.1 Veckomallar

Beroende på antal träningsdagar väljs en av fyra veckomallar. Varje dag tilldelas en svårighetsgrad som styr vilket pass som plockas från passbanken.

**Tabell C.1:** Veckomallar per antal träningsdagar.

Dagar/vecka	Mån	Tis	Ons	Tor	Fre	Lör	Sön
3 dagar	hårt	vila	medel	vila	hårt	vila	vila
4 dagar	hårt	lätt	medel	vila	hårt	vila	vila
5 dagar	hårt	lätt	medel	vila	hårt	lätt	vila
6 dagar	hårt	lätt	medel	lätt	hårt	lätt	vila

### C.2 Passbankens struktur

Passbanken är organiserad som en matris med 2 grenar (medel, distans), 4 träningsfaser (grundträning, uppbyggnad, tävling, återhämtning) och 3 svårighetsgrader (hårt, medel, lätt). Varje cell innehåller 1–2 fördefinierade pass.

**Tabell C.2:** Översikt av passbankens innehåll.

Gren	Fas	Hårt	Medel	Lätt
Medel	Grundträning	Tröskelpass; Intervaller 400m	Fartlek; Löpteknik och stegfrekvens	Lugn distanslöpning; Återhämtningsjogg
	Uppbyggnad	VO2max-intervaller; Tempointervaller 600m	Progressiv löpning; Tempojogg med stegringar	Lugn löpning; Aktiv vila
	Tävling	Tävlingsspecifikt	Aktivering	Lätt löpning; Vila + förberedelse
	Återhämtning	–	Lätt fartlek	Lugn jogg; Stretch & rörlighet
Distans	Grundträning	Tröskelpass långt; Intervaller 1000m	Medelstark löpning; Backlöpning och kuperad terräng	Lång löpning; Återhämtningsjogg
	Uppbyggnad	VO2max 1200m; Tempolöpning	Fartlek i terräng	Lugn distanslöpning; Aktiv vila
	Tävling	Tävlingsspecifikt	Aktivering	Lätt löpning; Vila
	Återhämtning	–	Lätt löpning	Mycket lugn jogg; Stretch & rörlighet

## C.3 Exempel på detaljerade passbeskrivningar

Varje mall innehåller utöver namn, typ, duration och intensitet även en strukturerad beskrivning med uppvärmning, huvudpass och nedvarvning. Nedan visas sex representativa exempel ur passbanken.

### Intervaller 400 m

Uppvärmning: 15 min jogg + dynamisk stretch + 4x100 m stigande.  
Huvudpass: 8x400 m (vila 90 s mellan).  
Nedjogg: 15 min + stretch.

### VO<sub>2</sub>max-intervaller

Uppvärmning: 15 min jogg + dynamisk stretch.  
Huvudpass: 5x1000 m på 90–95 % (vila 3 min).  
Nedjogg: 15 min + stretch.

### Lugn distanslöpning

45 min lugn löpning i zon 2. Jämn puls, avslappnat.

### Tröskelpass långt

Uppvärmning: 15 min jogg + dynamisk stretch.  
Huvudpass: 2x15 min i tröskeltempo (vila 3 min).  
Nedjogg: 15 min + stretch.

### Backlöpning och kuperad terräng

Uppvärmning: 10 min jogg.  
Huvudpass: 6x200 m uppförsbacke i medelhård fart (vila: jogg ner). Fokus: framåtlutning, knälyft och drivkraft från höft.  
Nedjogg: 10 min + stretch.

### Fartlek i terräng

Uppvärmning: 10 min jogg.  
Huvudpass: 35 min fartlek i kuperad terräng (1 min hårt i backar, lugnt emellan).  
Nedjogg: 10 min + stretch.



**CHALMERS**