



CHALMERS



GÖTEBORGS UNIVERSITET



Utvärdering av AI-detektorer för att identifiera AI-genererade lösningsförslag på programmerings problem

Examensarbete inom högskoleingenjörsprogrammet i datateknik

Shahad Al-Saedi

Institutionen för data- och informationsteknik

CHALMERS TEKNISKA HÖGSKOLA
Göteborg 2024
www.chalmers.se

EXAMENSARBETE 2024

Utvärdering av AI-detektorer för att identifiera AI-genererade lösningsförslag på programmerings problem

Shahad Al-Saedi



GÖTEBORGS
UNIVERSITET



CHALMERS

Institutionen för data- och informationsteknik
CHALMERS TEKNISKA HÖGSKOLA
GÖTEBORGS UNIVERSITET
Göteborg 2024

Utvärdering av AI-detektorer för att identifiera
AI-genererade lösningsförslag på programmerings problem

SHAHAD AL-SAEDI

© SHAHAD AL-SAEDI , 2024.

Handledare: Ulf Norell, Chalmers Tekniska Högskola
Examinator: Jonas Duregård, Chalmers Tekniska Högskola

Examensarbete 2024
Institutionen för data- och informationsteknik
Chalmers Tekniska Högskola
SE-412 96 Göteborg
Telefon +46 31 772 1000

Skriven i L^AT_EX
Göteborg 2024

Utvärdering av AI-detektorer för att identifiera AI-genererade lösningsförslag på programmerings problem

SHAHAD AL-SAEDI

Institutionen för data- och informationsteknik

Chalmers Tekniska Högskola

Göteborgs Universitet

Sammanfattning

Denna studie utvärderar effektiviteten hos AI-detekteringsprogram för att identifiera AI-genererad kod, specifikt från ChatGPT. Genom en systematisk litteratursökning och urval baserat på popularitet testades tre AI-detekteringsprogram—Contentdetector, Plagiarismdetector och Copyleaks—med hjälp av gamla tentamensfrågor och lösningar från kursen Objektorienterad programmering i Java vid Chalmers tekniska högskola. De AI-genererade lösningarna jämfördes med faktiska tentamenslösningar för att bedöma detektionsnoggrannheten över olika språk och textlängder. Resultaten visar betydande variationer i programmets förmåga att detektera AI-genererad kod, vilket belyser begränsningarna och potentiella felaktigheter hos nuvarande AI-detekteringstekniker. Studien framhäver behovet av förbättrade verktyg och strategier för att säkerställa akademisk integritet i en tid av snabb AI-utveckling.

Abstract

This study evaluates the effectiveness of AI-detection programs in identifying AI-generated code, specifically from ChatGPT. Through a systematic literature review and selection based on popularity, three AI-detection programs—Contentdetector, Plagiarismdetector, and Copyleaks—were tested using past exam questions and solutions from the Object-Oriented Programming in Java course at Chalmers University. The AI-generated solutions were compared against actual exam solutions to assess the detection accuracy across different languages and text lengths. The findings reveal significant variations in the detection capabilities of the programs, highlighting the limitations and potential inaccuracies of current AI-detection technologies. This study underscores the need for improved tools and strategies to ensure academic integrity in the face of evolving AI technologies

Nyckelord: Artificiell Intelligens, AI, Java, AI-detektor

Förord

Detta examensarbete har utförts vid Institutionen för data och informationsteknik vid Chalmers tekniska högskola. Arbetet syftar till att utvärdera effektiviteten hos AI-baserade detektorer för att identifiera AI-genererad programkod. Mitt intresse för detta ämne väcktes av den snabba utvecklingen inom artificiell intelligens och dess ökande användning inom programmering och mjukvaruutveckling. Jag vill rikta ett stort tack till min handledare, Ulf Norell, för hans ovärderliga stöd och insiktsfulla vägledning genom hela processen. Hans feedback och uppmuntran har varit en stor hjälp för att driva detta projekt framåt. Jag hoppas att resultaten kommer att bidra till vidare forskning och utveckling av AI-detekterings verktyg.

Shahad Al-Saedi, Göteborg, 19 juni 2024

Figurer

4.1	Comparable interface	12
6.1	Contentdetectors resultat, Tenta lösningsförslag visades vara 100% AI-genererade(felaktigt)	17
6.2	Contentdetectors resultat, ChatGPT lösningar visades vara 100% AI-genererade(korrekt)	17
6.3	Contentdetector resultat när man lägger in text i fältet och kör programmet	20
6.4	Plagiarismdetectors identifierade 20st av Tenta lösningsförslagen som mänskligt skriven och 20st som AI-genererad (felaktigt)	20
6.5	Plagiarismdetectors identifierade 28st av ChatGPT lösningar som AI-genererad och 12st mänskligt skriven (felaktigt)	20
6.6	Plagiarismdetector resultat efter att lösningen har lagts till	23
6.7	Av tenta lösningsförslagen identifierades 36st som mänskligt skriven och 4st som AI-genererad	24
6.8	Av AI -genererade lösningar identifierades 30st som AI-genererad och 10st som mänskligt skriven	24
6.9	Resultatet av att lägga in lösningar i Copyleaks för att kolla om de innehåller AI	24
A.1	Tenta lösningsförslag till fråga 22	II
A.2	Tenta lösningsförslag till fråga 23	III
A.3	Tenta lösningsförslag till fråga 33	IV

Innehåll

Figurer	viii
Tabeller	ix
1 Inledning	1
2 Bakgrund	3
2.1 Historisk bakgrund	4
2.2 Relaterade studier	5
2.3 AI detekterings program	6
2.3.1 Teknisk bakgrund	6
2.3.2 Copleaks	7
2.3.3 Plagiarismdetector	7
2.3.4 Contentdetector	8
3 Syfte	9
3.1 Mål	9
3.2 Avgränsningar	9
4 Metod	11
5 Genomförande	15
6 Resultat	17
6.1 Contentdetector	17
6.1.1 Fisher's exakta test:	17
6.1.2 Beräkningen av resultatet med <i>precision</i> och <i>recall</i> :	18
6.2 Plagiarismdetector	20
6.2.1 Fisher's exakta test	20
6.2.2 Beräkningen av resultatet med <i>precision</i> och <i>recall</i> :	22
6.3 Copleaks	23
6.3.1 Fisher's exakta test:	24
6.3.2 Beräkningen av resultatet med <i>precision</i> och <i>recall</i> :	26
7 Diskussion	29
7.1 Etiska aspekter	29
7.2 Ekologiska aspekter	30

7.3	Diskussion av resultatet	31
8	Slutsats	33
	Bibliography	35
A	Appendix	I

1

Inledning

Idag fortsätter användningen av artificiell intelligens (AI) växa och det har öppnat nya horisonter inom utbildning och akademiskt arbete. Med den snabba utvecklingen av AI modeller som ChatGPT och GitHub Copilot blir det allt vanligare att generera lösningar på programmeringsproblem på ett autentiskt och till synes mänskligt sätt. Denna utveckling har dock väckt frågor om akademisk ärlighet och upphovsrätt, särskilt i samband med bedömning av studenters prestationer och upptäckt av potentiell plagiering.

Denna studie syftar till att utvärdera AI-verktyg för att identifiera AI-genererade lösningar på programmeringsproblem inom undervisningssammanhang. För att genomföra detta användes frågor och svar från tidigare tentamina inom objektorienterad programmering som referensmaterial, tillsammans med AI-genererade lösningar från ChatGPT. Dessa lösningar matades in i olika detektionsverktyg för att undersöka deras förmåga att skilja mellan mänskligt skrivna och AI-genererad programkod. Resultaten belyser effektiviteten hos dessa verktyg och deras potential att säkerställa akademisk integritet.

2

Bakgrund

När ChatGPT lanserades den 30 november 2022, fångade den världens uppmärksamhet med sina imponerande förmågor. ChatGPT, utvecklad av OpenAI, är en avancerad språkmodell (LLM) tränad för att följa instruktioner i en prompt och leverera detaljerade svar. Genom att använda djupinlärning kan ChatGPT generera resultat från några få meningar som innehåller bakgrundsinformation eller en fråga [1],[2]. AI utvecklas genom maskininlärning, vilket innebär att ChatGPT har tillgång till stora mängder information och kan förbättra sig själv, även om den inte kan "tänka själv". ChatGPT:s optimering för dialog innebär att ChatGPT kan komma ihåg tidigare interaktioner och leverera mer utarbetade svar på följdfrågor.

Trots dess imponerande förmågor kan ChatGPT ibland ge missledande svar eftersom generativa AI-system, som ChatGPT, kan producera irrelevanta, felaktiga eller ensidiga resultat [12]. Dessa begränsningar innebär att ChatGPT ibland kan ha problem med relativt enkla frågor. Trots dessa utmaningar har ChatGPT blivit mycket populär, med miljontals användare världen över som drar nytta av dess tidsbesparande och snabba resultat [12]. Dess tillgänglighet och öppenhet gör det tydligt att ChatGPT redan har en betydande påverkan inom utbildning. Många professorer argumenterar för att ChatGPT inte bör förbjudas utan istället användas för att förbättra studenternas prestationer och kreativitet [3].

Två månader efter lanseringen hade ChatGPT attraherat 100 miljoner användare, vilket gjorde det till den snabbast växande konsumentprogramvaran i historien. En stor del av dessa användare är studenter [1]. Användare kan utnyttja ChatGPT för att brainstorma idéer, förstå komplexa ämnen, få personliga produkt- och tjänsterekommendationer, utforska framväxande vetenskaplig forskning, översätta text till 95 språk, skriva och debugga datorprogram, och generera anpassat skriftligt innehåll. Trots dess framgångar mötte ChatGPT snabbt flera kontroverser, inklusive oro över dess förmåga att generera felaktiga svar och dess användning i utbildningsmiljöer. OpenAI fortsatte att utveckla mer avancerade generativa inlärningsmodeller och släppte GPT-4 i mars 2023, vilket ytterligare förbättrade ChatGPT:s förmågor och användningsområden [3]. Trots konkurrens från andra AI-chatbotar, som Claude AI och Google Bard, förblev ChatGPT en dominerande kraft inom AI-industrin in i 2024 [1].

ChatGPT:s kod och text är välskriven, vilket gör det svårt att skilja den från mänskligt skriven text. Inom utbildning är det viktigt att kunna identifiera hur stor del av arbetet som studenter lämnar in som är genererat av ChatGPT, då detta anses

vara en form av plagiat. Det finns flera AI-program som kan analysera text och kod för att avgöra hur stor andel som är skriven av AI.

I en nyligen utförd studie [4] uttryckte Khalil och Er sin oro över effektiviteten hos AI/plagiatdetekterings program när de används för att upptäcka text genererad av ChatGPT. Den komplexa strukturen och syntaxen av AI-genererade texter och kod gör det svårt att upptäcka plagiat. För att bemöta dessa farhågor utvecklade OpenAI ett specialprogram, GPTzero, för att upptäcka ChatGPT-genererad text och kod. Programmet använder en blandning av maskininlärnings-algoritmer och mänsklig granskning för att identifiera plagiat i text från ChatGPT [4].

Chalmers tekniska högskola har strikta riktlinjer för akademisk integritet som syftar till att säkerställa att all forskning och akademiskt arbete utförs på ett etiskt och ärligt sätt [6]. En central del av dessa riktlinjer är definitionen och hanteringen av plagiat. Plagiat definieras som användning av någon annans arbete utan korrekt hänvisning, vilket kan omfatta text, bilder, diagram eller datorprogram. Detta inkluderar även självplagiat, där en student återanvänder sitt eget tidigare arbete utan att tydligt ange att det är en tidigare publicerad text. Oftast kan inte genererad text ses som plagiat, men det kan fortfarande vara fusk. Man skall ange hur man har använt AI och man ska alltid rådgöra med sin handledare för att beslut kan variera mellan olika arbeten, även när man har samma examinator.

2.1 Historisk bakgrund

Den första chatbottens historia går tillbaka till 1966 när ELIZA skapades vid Massachusetts Institute of Technology (MIT) av den tysk-amerikanska datavetaren Joseph Weizenbaum [1]. ELIZA använde tekniker som substitution och mönsterigenkänning och kunde övertyga vissa användare om att de samtalade med en annan person, trots dess enkelhet jämfört med moderna standarder.

År 1991 släppte det singaporeanska teknikföretaget Creative Labs ett program som hette Doctor Sbaitso, som var kompatibelt med datorer som körde MS-DOS-operativsystemet [1]. Doctor Sbaitso var den första chatbott som införlivade tidiga AI-teknologier.

1995 dök ALICE upp, en chatbott som använde naturlig språkbehandling och engagerade sig i realtidssamtal med användare. ALICE, även känd som Alicebot, modellerades efter ELIZA-programmet. 2001 släppte ActiveBuddy en chatbott kallad Smarterchild, som blev mycket populär på olika snabbmeddelandeplattformar.

År 2010 släppte Apple sin digitala personliga assistent Siri, vilket markerade ett stort framsteg inom chatbott-teknologin [1]. Siri använde ett naturligt språkgränssnitt (UI) för att skapa en mer realistisk användarupplevelse, vilket blev en prototyp för nästan alla framgångsrika chatbottar som följde, inklusive ChatGPT.

2.2 Relaterade studier

En studie [5] utförd av Copyleaks visar en 76 procent ökning av AI-genererat innehåll bland studenter. Denna omfattande studie, genomförd av Copyleaks, syftade till att analysera den föränderliga landskapet av akademisk integritet genom att undersöka trender inom AI-adoption och plagiatdetektion bland tiotusentals högskole- och gymnasiestudenter i sju länder, inklusive Australien, Kanada, Myanmar, Filippinerna, Sydafrika, Storbritannien och USA. Under ett års tid (januari 2023 till januari 2024) samlades data in och analyserades för att avslöja insikter om relationen mellan AI-användning och plagiatnivåer bland studenter.

En pionjär inom AI-baserad textanalys och plagiatdetektion, berättar att nya data visar förändrade studentvanor gällande AI-användning och plagiat i skolor. Studien, som genomfördes från januari 2023 till januari 2024, avslöjade en 76 procent ökning av AI-genererat material under året.

Copyleaks granskade trender inom AI och plagiatdetektion bland tiotusentals högskole- och gymnasiestudenter i sju länder under det senaste året [5]. Analysen visade att medan AI-genererat innehåll bland studenter ökade år över år, minskade plagiatnivåerna markant med 51 procent under samma period, vilket understryker den transformerande effekten av AI-adoption på akademisk integritet.

“Som data visar har studenter gradvis adopterat generativ AI under det senaste året. Men det mest insiktsfulla är ändå plagiatnivåerna. Även om det har skett en minskning av plagiat, kvarstår problemet, vare sig det handlar om generativ AI eller studenternas egna verk, vilket betyder att det fortfarande är en fråga som behöver åtgärdas,” sa Alon Yamin, VD och medgrundare av Copyleaks. “Därför är det absolut nödvändigt för akademiska institutioner att anta omfattande lösningar som kan upptäcka AI-genererat innehåll och plagiat för att säkerställa full transparens och bibehålla akademisk integritet.”

Studien fann en utbredd AI-adoption och efterföljande detektion i olika länder. Australien hade den högsta genomsnittliga andelen, med 31 procent av de skannade uppsatserna innehållande AI under året. Men även om AI-adoptionen ökar, är plagiatnivåerna oroväckande höga, med Storbritannien (33 procent) och USA (30 procent) som de länder med högst genomsnittliga nivåer.

Ytterligare nyckelfynd från analysen inkluderade:

1. Ökad AI-adoption: Det var en märkbar ökning av AI-adoption bland studenter under året. I januari och februari 2023 var den genomsnittliga andelen skannade uppsatser innehållande AI-innehåll 12 procent respektive 11 procent. Men detektionsnivåerna ökade avsevärt senare under året och nådde 25 procent i november och december 2023.

2. Högre detektionsnivåer i onlineskolor: E-lärande institutioner uppvisade de högsta nivåerna av AI- och plagiatdetektion bland skannade uppsatser, med 26 procent re-

spektive 39 procent [5]. Detta belyser att personlig undervisning i viss mån hämmar dessa praktiker. Plagiat kvarstår dock i alla akademiska miljöer, med nivåer på 29 procent i gymnasieskolor och 28 procent i högskolor.

3. Variation beroende på institutionstyp: Yrkes- och tekniska högskolor registrerade den högsta andelen AI-användning, medan community colleges hade de högsta plagiatnivåerna [5]. Den genomsnittliga andelen skannade uppsatser innehållande AI vid yrkes- och tekniska högskolor var 23 procent, medan community colleges hade en plagiatnivå på 32 procent. Både privata och offentliga skolor registrerade en plagiatdetektionsnivå på 28 procent, med AI-innehållsdetektioner på 19 procent respektive 14 procent.

“AI-genererat innehåll kommer bara att öka när studenter adopterar dessa verktyg mer allmänt,” tillade Yamin. “Därför är övervakning kritisk för att främja innehållsautenticitet och originalitet.”

Med den ökande tillgängligheten och sofistikereringen av AI-teknologier står akademisk integritet inför nya och komplexa globala utmaningar. Det är avgörande för lärare, institutioner och beslutsfattare att proaktivt hantera dessa utmaningar samtidigt som de utnyttjar AI:s potential för att främja integritet och innovation inom utbildningen.

2.3 AI detekterings program

När ChatGPT fått populariteten blev det ganska självklart att den påverkar utbildningen. Därför dök sådana verktyg som AI-detektorer upp. AI-detektorerna som utvärderas är Copyleaks, Plagiarismdetektor och Contentdetector.

2.3.1 Teknisk bakgrund

AI-detektorer använder maskininlärning och “natural language processing” för att inspektera språkliga mönster och meningstruktur för att avgöra om innehållet är AI-genererat eller mänskligt skrivet [11]. Klassificerare grupperar text baserat på mönster de har lärt sig, och inbäddningar representerar ord som vektorer för att visa hur de relaterar till varandra semantiskt. ‘Perplexity’ mäter hur förutsägbar innehållet är, högre nivåer indikerar att det är skrivet av en människa. Däremot ‘burstiness’ kontrollerar hur annorlunda meningarna är, mänskligt skrivet innehåll visar större mångfald. AI-detektorer är användbara men de är inte perfekta. De kan oftast leda till falsk positiva resultat eller avslag.

Maskininlärning är en gren av artificiell intelligens som fokuserar på utvecklingen av algoritmer och statistiska modeller som gör det möjligt för datorsystem att lära av och fatta beslut eller förutsägelser baserat på data utan att vara explicit programmerade [10]. I huvudsak tillåter maskininlärning datorer att lära och förbättra sig av erfarenhet, vilket gör det möjligt för dem att utföra uppgifter och fatta beslut utan

mänsklig inblandning. Det involverar användningen av olika tekniker såsom övervakad inlärning, oövervakad inlärning och förstärkningsinlärning för att analysera och tolka data, identifiera mönster och göra förutsägelser eller beslut. Maskininlärning har applikationer inom ett brett spektrum av områden, inklusive bild- och taligenkänning, naturlig språkbehandling, medicinsk diagnos, finansiell prognos och mer.

Precision och Recall är två viktiga mått som används för att utvärdera prestandan för klassificeringsmodeller [13].

- Precision: är förhållandet mellan korrekt förutsagda positiva observationer och totalt förutsagda positiva observationer. Den mäter noggrannheten hos de positiva förutsägelser som modellen gör.
- Recall: även känd som känslighet eller sann positiv frekvens, är förhållandet mellan korrekt förutsagda positiva observationer och alla observationer i den faktiska klassen. Den mäter modellens förmåga att identifiera alla relevanta instanser.

Fisher's exakta test En annan metod som skall användas för mätning av noggrannheten hos dessa AI-verktyg. Fisher's exakta test är ett statistiskt test som används för att avgöra om det finns icke slumpmässiga samband mellan två kategoriska variabler i en 2x2 tabell när urvals storlekarna är små [16]. Testet är uppkallat efter Sir Ronald Fisher och beräknar sannolikheten för att observera ett särskilt arrangemang av räkningar i *Contingency table*, givet de marginella summorna. Testet är baserat på den hypergeometriska fördelningen och ger ett mer exakt p-värde i situationer med små urvals storlekar.

2.3.2 Copyleaks

Copyleaks är en avancerad plagiatdetekteringsplattform som hjälper användare att upptäcka kopierat innehåll på nätet [7]. Den använder molnbaserade teknik och AI för att analysera och skanna texter, och jämföra dem med miljarder av dokument och webbsidor. Copyleaks sedan 2015 samlade och analyserade olika papper från tusentals universiteter för att förstå hur människor skriver [15]. Programmet kan AI-detektera på flera olika språk men den har högst noggrannhet (99.1%) när man använder engelska språket.

2.3.3 Plagiarismdetector

Plagiarismdetector fungerar likadant genom att jämföra in matade texten och med miljarder webbsidor och källor från andra databaser [8]. De plagierade delarna markeras och även de plagierade källorna visas. Webbsidan har en speciell funktion för AI detekterat innehåll genom att använda avancerade algoritmer för att analysera inmatningsdatan i flera olika faser som beskrivs på webbsidan. "AI- innehållsdetektering börjar med datainsamling som möjliggör en djupgående analys av uppladdade texten, sedan går den vidare till mönsterigenkänning i innehållet såsom ordförråd, meningstruktur och layout av innehållet för att säkerställa att ingen AI-genererad fras går förbi. Slutligen använder AI- detektorn handlingsbar insikt för att beteckna om inmatningsdatan är AI-genererat eller mänsklig skriven innehåll" .

2.3.4 Contentdetector

Contentdetector är ett verktyg utformat för att identifiera text som genererats av AI-system som ChatGPT och andra GPT-modeller [9]. Genom avancerade algoritmer och mönsterigenkänning analyserar den inskickad text för att avgöra om den är mänskligt skriven eller AI-genererad. Enligt informationen på Contentdetectors hemsida kan den detektera AI-genererat innehåll med 85 procent noggrannhet när det gäller ChatGPT genererat text/kod. Den är utvecklad genom att kombinera och finjustera resultaten från Open AI:s ChatGPT modell med utdata från BERT-modell (Roberta). Contentdetector som alla andra AI-detector använder algoritmer med sannolikhet och är inte hundra procent korrekt.

3

Syfte

Syftet med detta examensarbete är att utvärdera om AI-genererad programmeringskod kan upptäckas av existerande plagiat kontroll program som påstår sig kunna identifiera AI-genererad innehåll. Det är av särskilt intresse att undersöka hur precisa dessa program är när det gäller att skilja mellan mänskligt skrivna och AI-genererade lösningar på programmeringsproblem.

Diskutera användningen av dessa verktyg inom utbildningsmiljöer och vilka riktlinjer man ska ha vid användning av dessa verktyg om man upptäcker att en student har använt AI-genererat innehåll.

3.1 Mål

Målet med detta examensarbete är att:

1. Utvärdera detektionskapaciteten hos AI-detektorer: Bedöma om nuvarande AI-detektorprogram kan effektivt identifiera AI-genererad text eller kod, och undersöka hur exakta deras resultat. Detta görs genom
2. Bedöma verktygens användbarhet i utbildningen: Fastställa om dessa detektorprogram är värdefulla tillgångar för att upprätthålla akademisk integritet inom undervisningen, eller om de introducerar fler problem än de löser.

Genom att uppnå dessa mål hoppas detta arbete bidra till en djupare förståelse av både möjligheter och begränsningar med AI-detektorer, och ge vägledning för deras framtida användning inom akademiska miljöer.

3.2 Avgränsningar

En avgränsning är att det är just ChatGPT som används i undersökningen, ChatGPT valdes på grund av dess popularitet, det gäller även valet av de olika AI-detektor webbsidorna. Val av referensmaterial, att det är just objektorienterad programmerings kursen med programmerings språket Java. AI-detektor som valdes är även anpassade för just att upptäcka ChatGPT genererad text/kod och de är tillgängliga för användning i gratis versioner. Resultatet skulle kanske se annorlunda ut vid användning av plagiat program som kostar.

4

Metod

Insamling av Tentamensfrågor och Lösningsförslag

Källor för Tentamensfrågor och Lösningar:

- Tentamensfrågor och lösningsförslag hämtades från Chalmers läroplattform Canvas för kursen Objektorienterad programmering i Java.
- Tentamensfrågor från åren 2019-2021 användes (40 stycken).
- Frågorna involverade lösningar av programmeringsproblem, inte textförklaringar, med undantag för någon deluppgift.

Generering av Lösningar med ChatGPT

Instruktioner till ChatGPT:

- De insamlade tentamensfrågorna matades in i ChatGPT för att generera lämpliga lösningar.
- ChatGPT instruerades att generera endast programkod utan kommentarer.
- Kommentarer lades till i vissa lösningar för att undersöka om det påverkade resultaten.

Inmatning i AI-Detektor Program

Tester med AI-Detektor Program:

- Genererade lösningar matades in i tre olika AI-detektorprogram genom att klistra in koden i respektive ruta.
- AI-detektorprogrammen undersökte om koden var genererad av AI eller skriven av människor.
- Både engelsk och svensk text användes i koden för att testa programmets förmåga att upptäcka AI-genererad kod på olika språk.

En teknisk begränsning är att det finns begränsat antal ord som kan klistras in samtidigt och ett av programmen undersöker endast engelska språket. Det kan bli svårt om ChatGPT inte lyckas lösa uppgiften direkt själv, utan att få mer förklaring om vad som är fel och behöver rättas. Många av frågorna var relativt enkla för ChatGPT att lösa förutom frågor som använde bilder med grafer eller tabeller eftersom här används gratis versionen av ChatGPT som inte kan ta emot instruktioner som bilder.

Nedan är ett exempel på en fråga:

I Javas API finns interfacet Comparable (Figur 4.1).

```
public interface Comparable<T> {  
    /**  
     * Returns: a negative integer, zero, or a positive integer as  
     * this object is less than, equal to, or greater than the  
     * specified object o.  
     */  
    public int compareTo(T o);  
}
```

Figur 4.1: Comparable interface

Anta att det finns en klass Car som implementerar Comparable, dvs:

```
1 public class Car implements Comparable<Car> { ... }
```

Din uppgift är att skriva en metod med följande signatur, som givet en lista av Car objekt returnerar den största i listan enligt compareTo. Metoden bör kasta en exception ifall det inte finns någon största. Man får anta att listan inte är null och att den inte innehåller null. `public static Car getGreatest(List<Car> list) ...` din kod ...

– **Tenta lösningförslag** Följande svar räcker:

```
1  
2 public static Car getGreatest(List<Car> list) {  
3     if (list.size() == 0) {  
4         throw new IllegalArgumentException("No such element");  
5     }  
6     Iterator<Car> it = list.iterator();  
7     Car max = it.next();  
8     while (it.hasNext()) {  
9         Car temp = it.next();  
10        if (max.compareTo(temp) != 1) {  
11            max = temp;  
12        }  
13    }  
14    return max;  
15 }
```

Det går att läsa uppgiftens text på två olika sätt, specifikt kan man läsa “Metoden bör kasta en exception ifall det inte finns någon största” så att den bör kasta en exception ifall det inte finns ett element som är större än alla andra. Det är OK att förstå texten på det sättet. Detta kan man implementera med att lägga till följande rader kod innan return.

```
1 Iterator<Car> it2 = list.iterator();  
2 int count = 0;  
3 while (it2.hasNext()) {  
4     Car temp = it2.next();  
5     if (max.compareTo(temp) == 0) {  
6         count = count + 1;  
7     }  
8 }  
9 if (count > 1) {  
10    throw new IllegalArgumentException("No single max element.");  
11 }
```


Oberoende hur man läser texten bör man kasta exception ifall den givna listan är tom, dvs `list.size() == 0`, som i koden högst uppe. Det finns ju inget största element i en tom lista.

ChatGPT lösning:

```
import java.util.*;

1 public class Main {
2     public static void main(String[] args) {
3         List<Car> cars = new ArrayList<>();
4         cars.add(new Car("Toyota", 2020));
5         cars.add(new Car("Honda", 2019));
6         cars.add(new Car("Ford", 2018));
7
8         Car greatestCar = getGreatest(cars);
9         System.out.println("The greatest car is: " + greatestCar);
10    }
11
12    public static Car getGreatest(List<Car> list) {
13        if (list.isEmpty()) {
14            throw new IllegalArgumentException("List is empty");
15        }
16
17        Car greatestCar = list.get(0);
18        for (int i = 1; i < list.size(); i++) {
19            Car currentCar = list.get(i);
20            if (currentCar.compareTo(greatestCar) > 0) {
21                greatestCar = currentCar;
22            }
23        }
24
25        return greatestCar;
26    }
27 }

1 class Car implements Comparable<Car> {
2     private String brand;
3     private int year;
4
5     public Car(String brand, int year) {
6         this.brand = brand;
7         this.year = year;
8     }
9
10    public int getYear() {
11        return year;
12    }
13
14    public String toString() {
15        return brand + " (" + year + ")";
16    }
17
18    @Override
19    public int compareTo(Car other) {
```

4. Metod

```
20         return Integer.compare(this.year, other.year);
21     }
22 }
```

Både tentamen och ChatGPT-lösning hanterar problemet med att hitta det största elementet i en lista av 'car'-objekt enligt 'compareTo'-metoden. En sak att lägga märke till är att ChatGPT:s lösningar som genererats i undersökning är nästan alltid längre än tenta lösningsförslagen.

5

Genomförande

Tre olika program användes i undersökningen för att detektera lösningar som är genererade av AI och i det här fallet är det ChatGPT lösningar som testades. Det användes gamla tentamens(2019-2021) frågor (40st) och lösningsförslag till dessa tentamen inom kursen Objektorienterad programmering i Java. Dessa tentamen hämtas från Chalmers läroplattform Canvas. De frågorna som användes i undersökningen var just frågor som involverar lösning av programmeringsproblem och inte text förklaringar förutom någon deluppgift i enstaka frågor. Tentamens lösningsförslag är skrivna av lärare och all material är tagen innan ChatGPT fanns.

De insamlade frågorna användes som instruktioner till ChatGPT för att generera lämpliga lösningar till. Svaren fick justeras genom att be ChatGPT ta bort kommentarer från koden eftersom vi fokuserade på enbart programkod. Men kommentarer lades till senare för att kolla om resultatet skulle påverkas av det. Endast programkod som ChatGPT genererade togs, ingen extra förklaring till uppgiften eller kommentar togs med i lösningarna som undersöktes. ChatGPT svara oftast med ``självklart, här är lösningen...`` dessa delar togs inte med som en del av lösningen. Dessa lösningar matades in i plagiat kontroll programmen genom att klistra in i ruta. Lösningarna som togs från ChatGPT fick vara utan kommentarer genom att be den om att ta bort kommentarer på programkod. För att säkerställa att den genererade koden är fungerande, jämfördes de med tentamenslösning förslag som skulle spara tid på att inte behöva testköra alla lösningar. För att vissa lösningar som ChatGPT genererade var ganska likt tenta lösningsförslagen.

Tentamernas lösningsförslag användes i jämförelse, de klistrades in i de olika AI plagiat programmen och noterade om resultatet av programmet visar att det är genererad av AI eller mänskligt skriven. Hälften av lösningarna skrevs på svenska i form av kommentarer eller metod, variabel eller parameter namn för att se hur bra kan AI- detektor upptäcka AI- genererat kod på ett annat språk. Detta för att undersöka om AI- detektor har några svagheter som studenter kan utnyttja.

En statistiskt metod kalla Fisher's exakta test används i resultatet för att beräkna sannolikheten för hur bra AI-detektorer kan skilja mellan AI-genererade lösningar och mänskligt skrivna.

Dessutom används *precision* och *recall* i resultatet för att räkna ut noggrannheten hos dessa AI-verktygen. *Precision* och *recall* borde vara balanserade för att indikera att programmet ger pålitliga resultat. Högre *precision* säger att flaska positiva resultat är låga medans högre *recall* säger att det är lägre falskt negativa resultat.

5. Genomförande

För att programmet skall vara bra och noggrann skall *precision* och *recall* båda vara höga.

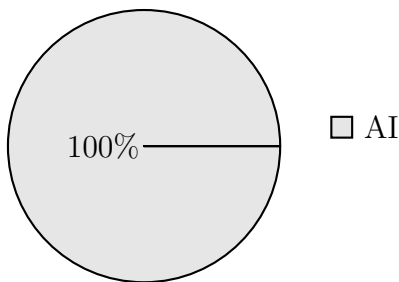
6

Resultat

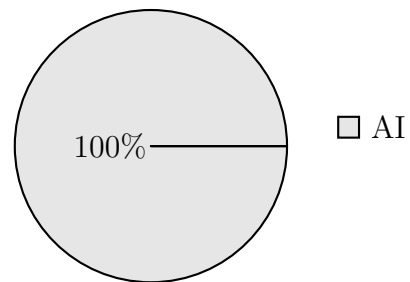
6.1 Contentdetector

Undersökningen av tenta lösningsförslagen i Contentdetector visade att 100% av svaren var AI-genererade. Delarna som misstänktes ha AI markerades med en highlighter. Programmet kunde endast ta emot engelska språket och därför undersöktes inte lösningarna som var på svenska. Lösningarna från ChatGPT och från tentamen visades vara AI innehåll.

Nedan visas andelen i genomsnitt av lösningarna som användes, hur stort andel är AI-genererad eller mänskligt skriven.



Figur 6.1: Contentdetectors resultat, Tenta lösningsförslag visades vara 100% AI-genererade(felaktigt)



Figur 6.2: Contentdetectors resultat, ChatGPT lösningar visades vara 100% AI-genererade(korrekt)

6.1.1 Fisher's exakta test:

Tenta lösningsförslag:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	0	0
Mänsklig (actual)	40	0

Tabell 6.1: Kontingenstabell för tenta lösningsförslag

AI-genererade lösningar:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	40	0
Mänsklig (actual)	40	0

Tabell 6.2: Kontingenstabell för ChatGPT genererade lösningar

Kombinerad 2x2 tabell:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	40	0
Mänsklig (actual)	40	0

Tabell 6.3: Kombinerad kontingenstabell för tenta lösningsförslag och AI-genererade lösningar

Vi använder *hypergeometric distribution* för att räkna ut p-värdet:

$$p = \frac{\binom{40}{40} \binom{40}{40}}{\binom{80}{80}}$$

beräkna varje binomial koefficient:

$$\binom{40}{40} = \frac{40!}{40!(40-40)!} = \frac{40!}{40! \cdot 0!} = 1$$

$$\binom{40}{40} = \frac{40!}{40!(40-40)!} = \frac{40!}{40! \cdot 0!} = 1$$

$$\binom{80}{80} = \frac{80!}{80!(80-80)!} = \frac{80!}{80! \cdot 0!} = 1$$

Kombinerar resultaten:

$$p = \frac{1 \cdot 1}{1} = 1$$

P-värdet är 1 och det visar ingen statistisk skillnad på resultatet för båda typer av lösningar. Contentdetector kan inte skilja mellan mänskligt skrivna och AI-genererade lösningar.

6.1.2 Beräkningen av resultatet med *precision* och *recall*:

Formlerna:

$$precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (6.1)$$

$$recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (6.2)$$

Tenta lösningsförslag:

- True Positives (rätt identifierade mänskligt skrivna) = 0
- False positives (AI-genererade men identifierade som mänskligt skrivna) = 0
- False Negatives (mänskligt skriven men identifierad som AI-genererad) = 40

$$precision = \frac{0}{0+0} = 0 \quad (6.3)$$

Precision är odefinierad när både täljaren och nämnaren är 0. Vi kan anta att precisionen är 0 i detta fall eftersom det inte finns några korrekt identifierade mänskligt skrivna lösningar.

$$recall = \frac{0}{0+40} = 0 \quad (6.4)$$

Precision och *recall* för tenta lösningsförslagen går det inte att räkna ut när det inte finns några rätt identifierade mänskligt skrivna lösningar.

AI-genererade lösningar:

- True Positives (rätt identifierade som AI-genererade) = 40
- False positives (mänskligt skrivna men identifierade som AI-genererade) = 40

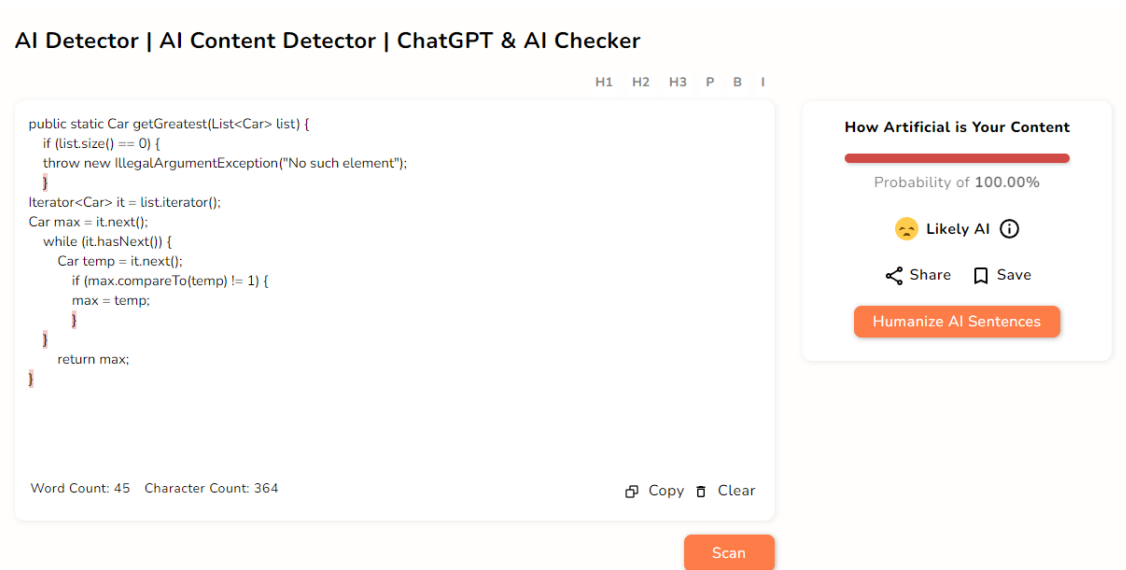
$$precision = \frac{40}{40+40} = \frac{40}{80} = 0.5 \text{ eller } 50\% \quad (6.5)$$

- True Positives (rätt identifierade som AI-genererade) = 40
- False Negatives (AI-genererad men identifierad som mänskligt skriven) = 0

$$recall = \frac{40}{40+0} = \frac{40}{40} = 1 \text{ eller } 100\% \quad (6.6)$$

precision : Vid klassificering av AI-genererad program kod, kan Contentdetector i 50% av fallen identifiera AI-genererade lösningar. *Precision* är lågt, vilket innebär att det finns många falska positiva resultat. Alla mänskligt skrivna lösningar klassificerades som AI-genererade och därför finns det inga True Positives för de. Contentdetector kan inte korrekt identifiera mänskliga skrivna lösningar.

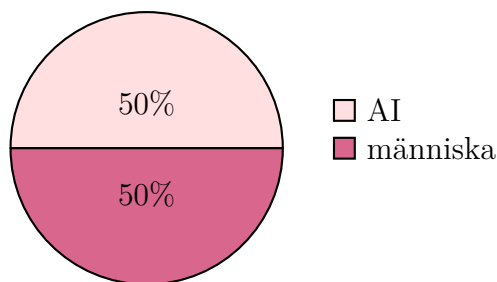
recall : Contentdetector identifiera alla AI-genererade lösningar korrekt och detta innebär att det inte finns några falska negativa resultat för AI-genererade lösningar.



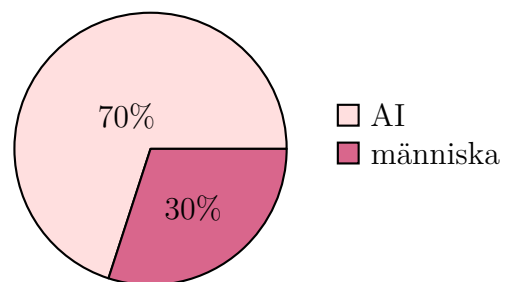
Figur 6.3: Contentdetector resultat när man lägger in text i fältet och kör programmet

6.2 Plagiarismdetector

I Plagiarismdetector hade stort antal falsk positiva resultat. Hälften av de mänskligt skrivna lösningarna felaktigt identifierades. Lösningarna testades på två olika språk, engelska och svenska men resultat förblev den samma. Programmet kan bättre identifiera AI-genererade lösningar men är mindre bra på att identifiera de mänskligt skrivna.



Figur 6.4: Plagiarismdetectors identifierade 20st av Tenta lösningsförslagen som mänskligt skrivna och 20st som AI-genererad (felaktigt)



Figur 6.5: Plagiarismdetectors identifierade 28st av ChatGPT lösningar som AI-genererad och 12st mänskligt skrivna (felaktigt)

6.2.1 Fisher's exakta test

Vi börjar med att skapa kontingenstabellerna:
Tenta lösningsförslag:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	0	0
Mänsklig (actual)	20	20

Tabell 6.4: Kontingenstabell för tenta lösningsförslag

AI-genererade lösningar:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	28	12
Mänsklig (actual)	12	0

Tabell 6.5: Kontingenstabell för ChatGPT genererade lösningar

Kombinerad 2x2 tabell:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	28	12
Mänsklig (actual)	32	20

Tabell 6.6: Kombinerad kontingenstabell för tenta lösningsförslag och AI-genererade lösningar

Tabell (6.4) används för att beräkna p-värdet. Vi använder *hypergeometric distribution* för att räkna ut p-värdet:

$$p = \frac{\binom{40}{28} \binom{52}{32}}{\binom{92}{60}}$$

beräkna varje binomial koefficient:

$$\binom{40}{28} = \frac{40!}{28!(40-28)!} = \frac{40!}{28! \cdot 12!}$$

$$\binom{52}{32} = \frac{52!}{32!(52-32)!} = \frac{52!}{32! \cdot 20!}$$

$$\binom{92}{60} = \frac{92!}{60!(92-60)!} = \frac{92!}{60! \cdot 32!}$$

Kombinerar resultaten:

$$p = \frac{\frac{40!}{28! \cdot 12!} \cdot \frac{52!}{32! \cdot 20!}}{\frac{92!}{60! \cdot 32!}}$$

sedan förenklar vi:

$$p = \frac{40! \cdot 52! \cdot 60! \cdot 32!}{28! \cdot 12! \cdot 32! \cdot 20! \cdot 92!}$$

Slutligen får vi p-värdet:

$$p \approx 0.509$$

P-värdet 0.509 visar att det inte finns något stor skillnad på Plagiarismdetectors resultat för ChatGPT genererade lösningar och tenta lösningsförslag. Det finns ingen statistiskt signifikans på resultatet. Det innebär att Plagiarismdetector är mindre bra på att skilja mellan mänskligt skrivna och AI-genererade lösningar.

6.2.2 Beräkningen av resultatet med *precision* och *recall*:

- Totalt antal tenta lösningsförslag: 40 (alla mänskligt skrivna)
- Plagiarismdetector resultat: 50% (20) AI-genererade (felaktiga), 50% (20) mänskligt skrivna (korrekta)
- Totalt antal AI-genererade lösningar: 40
- Plagiarismdetector resultat: 70% (28) AI-genererade (korrekta), 30% (12) mänskligt skrivna (felaktiga)

Resultaten av *precision* och *recall* ger oss insikt i hur effektiv och tillförlitlig Plagiarismdetector är vid att skilja mellan AI-genererade och mänskligt skrivna lösningar. Beräkningen enligt formlerna (6.1) och (6.2)

- **Tenta lösningsförslag:**

True positives (rätt identifierade mänskligt skrivna) = 20

False positives (AI-genererade men identifierade som mänskligt skrivna) = 12

$$precision = \frac{20}{20 + 12} = \frac{20}{32} = 0.625 \text{ eller } 62.5\% \quad (6.7)$$

True positives (rätt identifierade mänskligt skrivna) = 20

False negatives (mänskligt skriven men identifierad som AI-genererad) = 20

$$recall = \frac{20}{20 + 20} = \frac{20}{40} = 0.5 \text{ eller } 50\% \quad (6.8)$$

- **AI-genererade lösningar:**

True positives (rätt identifierade som AI-genererade) = 28

False positives (mänskligt skrivna men identifierade som AI-genererade) = 20

$$precision = \frac{28}{28 + 20} = \frac{28}{48} = 0.5833 \text{ eller } 58.33\% \quad (6.9)$$

True positives (rätt identifierade som AI-genererade) = 28

False negatives (AI-genererad men identifierad som mänskligt skriven) = 12

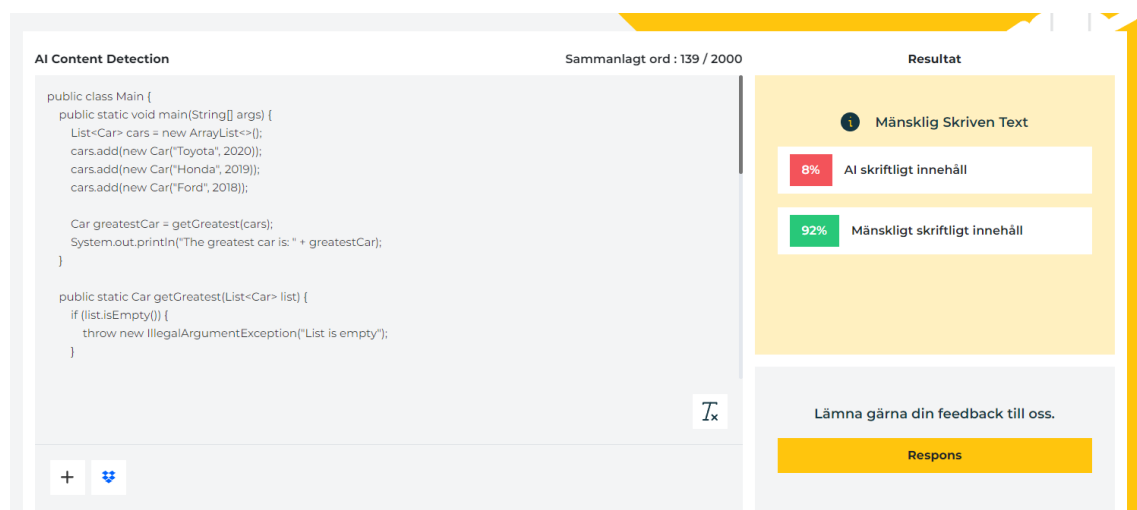
$$recall = \frac{28}{28 + 12} = \frac{28}{40} = 0.7 \text{ eller } 70\% \quad (6.10)$$

precision: Detta innebär att när Plagiarismdetector identifierar en lösning som mänskligt skriven, är det korrekt i 62.5% av fallen. Det finns fortfarande en betydande andel falska positiva (AI-genererade lösningar som felaktigt identifieras som mänskligt skrivna), vilket påverkar precisionen. Däremot har den en lägre *precision* på de ChatGPT genererade lösningarna. Den identifierar AI-genererad innehåll med *precision* på 58.33%.

recall: Detta innebär att Plagiarismdetector korrekt identifierar endast 50% av alla mänskligt skrivna lösningar. Det innebär också att 50% av de mänskligt skrivna lösningarna felaktigt identifieras som AI-genererade (falska negativa).

Plagiarism detector har en lägre *recall* för mänskligt skrivna lösningar (50%) vilket innebär att det ofta missar att korrekt identifiera de mänskligt skrivna lösningarna. *precision* är något högre men det är fortfarande inte tillräckligt för att vara pålitligt.

Nedan är en bild som visar hur programmet ser ut vid användning:

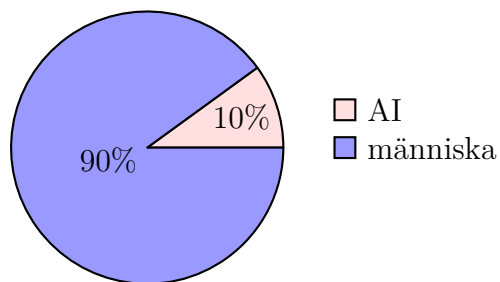


Figur 6.6: Plagiarismdetector resultat efter att lösningen har lagts till

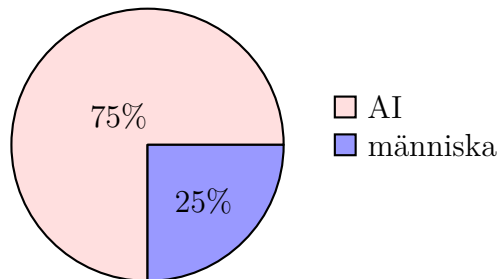
6.3 Copyleaks

I Copyleaks blev resultatet närmare verkligheten, tenta lösningsförslagen visades 90 procent vara skriva av människor och 10 procent (figur 6.7) var AI-genererade. ChatGPT:s lösningar visades vara 75 procent AI-genererade (figur 6.8) och 25 procent skrivna av en människa. Jämfört med de andra två programmen, är Copyleaks närmast till verkligheten. på bilden nedan(figur 6.9) kan man se att AI-genererad innehåll markeras vart i koden/texten det är. Programmet hade minimum 350 antal tecken för att kunna ge ett resultat och vissa lösningar hade kort svar vilket inte

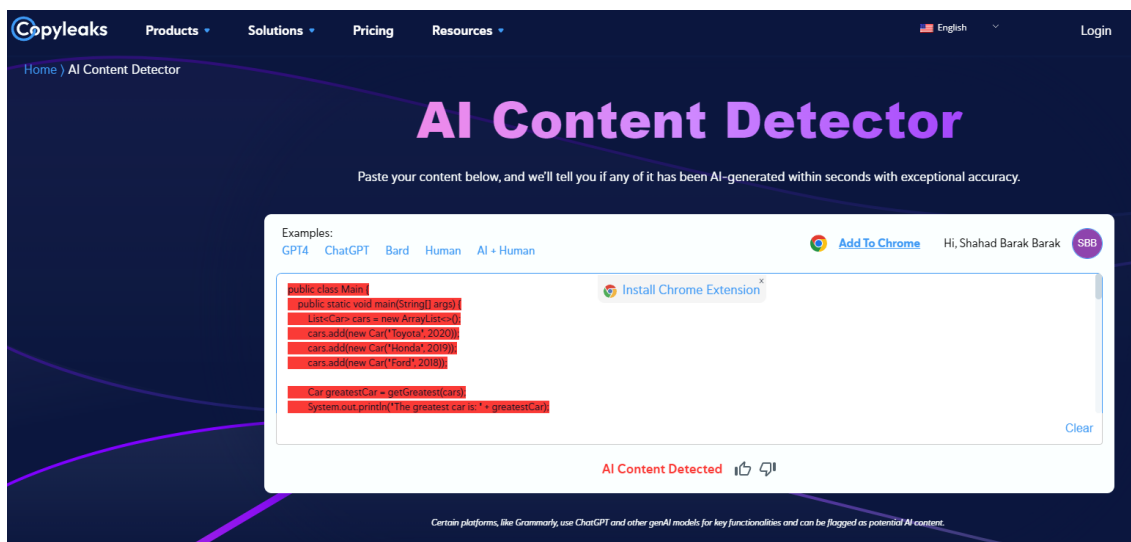
räckte för att få resultat. Även om Copyleaks hade mest noggrann resultat, finns det fortfarande 4st lösningsförslag som falsk positivt identifierades som AI-genererade.



Figur 6.7: Av tenta lösningsförslagen identifierades 36st som mänskligt skrivna och 4st som AI-genererad



Figur 6.8: Av AI -genererade lösningar identifierades 30st som AI-genererad och 10st som mänskligt skrivna



Figur 6.9: Resultatet av att lägga in lösningar i Copyleaks för att kolla om de innehåller AI

6.3.1 Fisher’s exakta test:

Tenta lösningsförslag:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	0	0
Mänsklig (actual)	4	36

Tabell 6.7: Kontingenstabell för tenta lösningsförslag

AI-genererade lösningar:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	30	10
Mänsklig (actual)	12	0

Tabell 6.8: Kontingenstabell för AI-genererade lösningar

Kombinerad tabell 2x2:

	AI-genererad (pred)	Mänskligt (pred)
AI (actual)	30	10
Mänsklig (actual)	4	36

Tabell 6.9: Kombinerad kontingenstabell för tenta lösningsförslag och ChatGPT genererade lösningar

Vi använder *hypergeometric distribution* för att räkna ut p-värdet:

$$p = \frac{\binom{40}{30} \binom{40}{4}}{\binom{80}{34}}$$

beräkna varje binomial koefficient:

$$\binom{40}{30} = \frac{40!}{30!(40-30)!} = \frac{40!}{30! \cdot 10!}$$

$$\binom{40}{4} = \frac{40!}{4!(40-4)!} = \frac{40!}{4! \cdot 36!}$$

$$\binom{80}{34} = \frac{80!}{34!(80-34)!} = \frac{80!}{34! \cdot 46!}$$

Kombinerar resultaten:

$$p = \frac{\frac{40!}{30! \cdot 10!} \cdot \frac{40!}{4! \cdot 36!}}{\frac{80!}{34! \cdot 46!}}$$

Sedan förenklar vi:

$$p = \frac{40! \cdot 40! \cdot 34! \cdot 46!}{30! \cdot 10! \cdot 4! \cdot 36! \cdot 80!}$$

Slutligen får vi p-värdet:

$$p \approx 0.00134$$

P-värdet 0.00134 indikerar en signifikant skillnad mellan resultatet för ChatGPT genererade lösningar och tenta lösningsförslagen. Det innebär att resultatet är mest sannolikt inte slumpmässigt och Copyleaks är ganska bra på att skilja mellan mänskligt skriven och en AI-genererad lösning.

6.3.2 Beräkningen av resultatet med *precision* och *recall*:

- **Tenta lösningsförslag:**

True positives (rätt identifierade mänskligt skrivna) = 36

False positives (AI-genererade men identifierade som mänskligt skrivna) = 10

$$precision = \frac{36}{36 + 10} = \frac{36}{46} = 0.782 \text{ eller } 78.2\% \quad (6.11)$$

True Positives (rätt identifierade mänskligt skrivna) = 36

False Negatives (mänskligt skriven men identifierad som AI-genererad) = 4

$$recall = \frac{36}{36 + 4} = \frac{36}{40} = 0.9 \text{ eller } 90\% \quad (6.12)$$

- **AI-genererade lösningar:**

True Positives (rätt identifierade som AI-genererade) = 30

False positives (mänskligt skrivna men identifierade som AI-genererade) = 4

$$precision = \frac{30}{30 + 4} = \frac{30}{34} = 0.882 \text{ eller } 88,2\% \quad (6.13)$$

True Positives (rätt identifierade som AI-genererade) = 30

False Negatives (AI-genererad men identifierad som mänskligt skriven) = 10

$$recall = \frac{30}{30 + 10} = \frac{30}{40} = 0.75 \text{ eller } 75\% \quad (6.14)$$

precision: högre *precision* innebär att resultatet är relevant. 78.2% *precision*är andelen av fallen där Copyleaks kan identifiera mänskligt skrivna lösningar. Medans 88.2% *precision*när det gäller att identifiera AI-genererade lösningar.

recall: Detta innebär att Copyleaks korrekt identifierar 90% av alla mänskligt skrivna lösningar. Och 75 % av de AI-genererade lösningarna kunde korrekt identifieras.

Copyleaks har ändå ganska högt *precision* och *recall* för båda Tenta lösningarna och ChatGPT:s genererade svar. Det finns dock fortfarande en chans att man får felaktigt resultat.

Uppgift	Tenta lösningsförslag	ChatGPT AI-genererad
1	Mänskligt skriven	Mänskligt skriven
2	Mänskligt skriven	Mänskligt skriven
3	Mänskligt skriven	AI-genererad
4	Mänskligt skriven	AI-genererad
5	Mänskligt skriven	AI-genererad
6	Mänskligt skriven	AI-genererad
7	Mänskligt skriven	Mänskligt skriven
8	Mänskligt skriven	AI-genererad
9	Mänskligt skriven	AI-genererad
10	Mänskligt skriven	AI-genererad
11	Mänskligt skriven	Mänskligt skriven
12	Mänskligt skriven	AI-genererad
13	Mänskligt skriven	AI-genererad
14	Mänskligt skriven	AI-genererad
15	Mänskligt skriven	AI-genererad
16	Mänskligt skriven	AI-genererad
17	Mänskligt skriven	Mänskligt skriven
18	Mänskligt skriven	AI-genererad
19	Mänskligt skriven	AI-genererad
20	Mänskligt skriven	Mänskligt skriven
21	AI-genererad	AI-genererad
22	AI-genererad	AI-genererad
23	AI-genererad	AI-genererad
24	Mänskligt skriven	AI-genererad
25	Mänskligt skriven	AI-genererad
26	Mänskligt skriven	AI-genererad
27	Mänskligt skriven	Mänskligt skriven
28	Mänskligt skriven	Mänskligt skriven
29	Mänskligt skriven	AI-genererad
30	Mänskligt skriven	Mänskligt skriven
31	Mänskligt skriven	Mänskligt skriven
32	Mänskligt skriven	AI-genererad
33	AI-genererad	AI-genererad
34	Mänskligt skriven	AI-genererad
35	Mänskligt skriven	AI-genererad
36	Mänskligt skriven	AI-genererad
37	Mänskligt skriven	AI-genererad
38	Mänskligt skriven	AI-genererad
39	Mänskligt skriven	AI-genererad
40	Mänskligt skriven	AI-genererad

Tabell 6.10: Resultatet för de olika lösningarna enligt Copyleaks

Tittar vi närmare på de tenta lösningsförslagen som fick falsk negativ resultat enligt

Copyleaks i figur A.1, A.2 och A.3, ser vi att två av av dessa frågor har metod och variabel namn på svenska. Programmet har brister när det gäller andra språk och det kan vara för att lösningarna är relativt enkla och följer en vanlig struktur som kan lika väl likna AI-genererade lösningars struktur. AI-genererade texter tenderar att följa vissa mönster och stilistiska egenskaper som algoritmer är tränade att känna igen. Om de mänskligt skrivna texterna liknar dessa mönster kan de felaktigt klassificeras som AI-genererade. Program koden är inte komplex och liknar till en viss del de AI-genererade lösningarna.

7

Diskussion

7.1 Etiska aspekter

Etiska problem

Användningen av AI för att generera lösningar väcker viktiga frågor kring akademisk ärlighet och upphovsrätt. Det är avgörande att studenter förstår och respekterar principerna för akademisk integritet, och att lärare och utbildningsinstitutioner utvecklar riktlinjer för att hantera användning av AI i akademiska sammanhang på ett etiskt sätt.

Bedömning av Studentprestationer

En viktig aspekt är bedömningen av studenternas prestationer, vilket kan kompliceras av AI-genererade lösningar. Utbildningsinstitutioner måste utveckla lämpliga bedömningsmetoder som tar hänsyn till både autentiska mänskliga svar och AI-genererade svar, och vara transparenta om hur dessa bedömningar genomförs. AI-detektorprogram kan bidra till rättvisa bedömningar genom att möjliggöra identifiering av AI-genererade lösningar.

Utmaningar med AI-Detektering

Felaktig identifiering av mänskligt skrivna lösningar som AI-genererade kan leda till förvrängning av sanningen, vilket skapar misstro och förvirring bland studenter och resulterar i felaktiga och orättvisa beslut. Dessutom kan AI-system ibland felaktigt hållas ansvariga för handlingar de inte har utfört, vilket innebär ytterligare etiska och rättsliga komplikationer. AI-system kan också ge partiska svar, och det är viktigt att studenter är medvetna om dessa potentiella biaser. Studenter bör utbildas i att kritiskt utvärdera AI-genererade svar och använda pålitliga källor.

Omdefiniering av Plagiat

Vad som utgör plagiat blir särskilt komplext när det gäller AI. Enligt traditionell definition är plagiat att använda någon annans arbete utan att ge erkännande till den ursprungliga författaren. Men när arbetet är genererat av något snarare än någon, blir denna definition svår att tillämpa. Det väcker frågor om huruvida ChatGPT och liknande AI-system ska klassificeras som författare. Detta kan leda till behovet av att omdefiniera plagiat för att inkludera verk som genererats av AI-system.

Akademisk Policy och Riktlinjer

För att navigera dessa komplexa frågor måste akademiska institutioner skapa och implementera tydliga policyer och riktlinjer kring användningen av AI i akademiska sammanhang. Detta inkluderar regler om hur AI-verktyg får användas i uppsatser, tentamina och andra akademiska arbeten, samt hur AI-genererat innehåll ska citeras och erkännas. Utbildning om akademisk integritet i en tid av avancerade AI-verktyg är också kritiskt för att säkerställa att studenterna förstår konsekvenserna av att använda AI på ett oetiskt sätt.

Transparens och Kommunikation

Slutligen är transparens och tydlig kommunikation mellan studenter, lärare och institutioner avgörande. Studenter måste informeras om vilka verktyg som används för att detektera AI-genererat innehåll och hur dessa verktyg fungerar. Det är också viktigt att studenter har möjlighet att diskutera och förstå de etiska implikationerna av AI i deras utbildning.

7.2 Ekologiska aspekter

Användningen av AI medför betydande ekologiska konsekvenser, främst i form av ökad energiförbrukning. AI-system som ChatGPT kräver enorma mängder datorkraft och resurser för att fungera, vilket leder till höga energikostnader och en ökad koldioxidavtryck. Till exempel använder ChatGPT dagligen mer än en halv miljon kilowattimmar, vilket är svårt att mäta exakt men visar på den stora energikonsumtionen [14].

Energikonsumtion och Miljöpåverkan

Den massiva energiförbrukningen för AI-modeller som ChatGPT genererar en betydande mängd växthusgaser, vilket bidrar till klimatförändringar. Data center som driver dessa AI-system behöver kontinuerlig kyla och elektricitet, vilket ytterligare ökar miljöpåverkan [14]. Detta innebär att AI:s framväxt inte bara medför tekniska och etiska utmaningar, utan också stora miljömässiga problem.

Hållbarhet och Energieffektivitet

För att minska AI:s ekologiska fotavtryck är det viktigt att utveckla och implementera mer energieffektiva algoritmer och tekniker. Forskning och innovation inom detta område kan bidra till att skapa AI-system som är både kraftfulla och energisnåla. Att använda förnybar energi för att driva datacenter är en annan viktig strategi för att minska koldioxid-utsläppen. Vidare kan optimering av AI-modeller, så att de kräver mindre datorkraft, också vara ett effektivt sätt att spara energi.

Livscykelperspektiv

Det är också viktigt att betrakta AI-teknologier ur ett livscykelperspektiv, vilket innebär att man tar hänsyn till energikonsumtionen och miljöpåverkan under hela livscykeln för AI-system – från utveckling och träning till implementering och underhåll. Genom att analysera och optimera varje steg i denna process kan man

ytterligare minska den totala miljöpåverkan [9].

Balans mellan Nyttan och Miljöpåverkan

För att maximera de potentiella fördelarna med AI och samtidigt minimera risker och negativa konsekvenser är det nödvändigt att hitta en balans mellan teknologisk utveckling och hållbarhet. AI kan potentiellt användas för att lösa några av världens mest pressande miljöproblem, såsom optimering av energianvändning, klimatövervakning och hållbart jordbruk. Därför är det avgörande att fortsätta utveckla AI på ett sätt som inte bara främjar teknisk framgång utan också ekologisk hållbarhet.

7.3 Diskussion av resultatet

Resultaten från utvärderingen av AI-detekteringsprogrammen visar att det finns betydande variationer i deras förmåga att urskilja mänskligt skriven kod från AI-genererad kod. De tre programmen som undersöktes – Contentdetector, Plagiarismdetector och Copyleaks – gav olika resultat, vilket belyser både styrkor och begränsningar hos nuvarande AI-detekteringstekniker.

Contentdetector visade att alla tentamenslösningarna var AI-genererade och därför kan den inte användas i utbildnings sammanhang. Lösningar som var på svenska kunde inte undersökas då programmet stödjer endast bearbeta text på engelska språket. Dessutom fick den lägst *precision* för tenta lösningsförslagen då den inte kunde identifiera någon av lösningarna som mänskligt skriven. Resultatet av Fisher's test visar att Contentdetector inte kan se skillnad mellan mänskligt skrivna och AI-genererade lösningar som gör den oanvändbar.

Plagiarismdetector identifierade hälften av tentamenslösningarna som AI-genererade och hälften som mänskligt skrivna, medan det bedömde 70 procent av ChatGPT:s lösningar som AI-genererade. Dessa resultat var mer balanserade men indikerar ändå en betydande risk för felklassificering. Att programmets resultat var desamma för både engelska och svenska tyder på en viss robusthet men väcker också frågor om noggrannheten i detekteringen. Resultatet av Fisher's test betyder att vi inte har tillräckligt med bevis för att säga att skillnaden mellan resultaten för mänskligt skrivna och AI-genererade lösningarna är verklig och inte av slump.

Copyleaks presterade bäst i denna undersökning genom att korrekt identifiera 90 procent av tentamenslösningarna som mänskligt skrivna och 75 procent av ChatGPT:s lösningar som AI-genererade. Detta program kom närmast verkligheten, men hade också sina begränsningar, såsom krav på en minsta textlängd på 350 tecken för att kunna ge ett resultat. Kortare kodsegment kunde därför inte bedömas, vilket är en viktig faktor att beakta i programmeringsutbildningar där kortare lösningar kan förekomma. Frågorna som var falskt positivt resultat undersöktes och två av de var på svenska, Copyleaks har redan nämnt att programmet ger mer noggranna resultat när lösningarna är på engelska. Resultat av Fisher's test visar att Copyleaks kan faktiskt se skillnad på mänskligt skrivna och AI-genererade lösningar.

Resultaten visar att nuvarande AI-detekteringsprogram har svårigheter med att exakt identifiera AI-genererad kod, särskilt när det gäller olika språk och kortare textsegment. Detta indikerar att dessa verktyg är inte fullt tillförlitliga för att användas som ensamma beslutsstöd i akademiska sammanhang. Det är problematiskt att förlita sig på program som inte kan ge konsekventa och korrekta resultat, eftersom detta kan leda till felaktiga anklagelser om plagiat och orättvisa betygsättningar. Det är inte heller säkert att AI-detekterings verktyg kan hålla jämn takt med hur snabbt AI utvecklas och blir bättre.

Dessutom framhäver detta problematiken med användning av AI i undervisningen och hur snabbt teknologin utvecklas jämfört med samhällets förmåga att anpassa sig. Undervisningsinstitutioner kan behöva utveckla nya metoder för att skapa uppgifter som är svårare för AI att lösa, vilket skulle kunna uppmuntra studenter att lösa problem själva istället för att förlita sig på AI-verktyg som ChatGPT.

Chalmers Tekniska Högskola gällande AI och plagiat

Chalmers med sitt starka fokus på akademisk integritet, kan spela en ledande roll i att utveckla och implementera effektiva strategier för att hantera dessa utmaningar. Deras befintliga riktlinjer och regler för AI-användning kan tjäna som en grund för att ytterligare undersöka och anpassa metoder för att säkerställa att AI används på ett etiskt och rättvist sätt i utbildningssammanhang. Chalmers policyer kring användningen av AI-verktyg betonar vikten av akademisk ärlighet och transparens. Enligt resultaten av denna studie, där olika AI-detekteringsprogram användes, framgår det att det finns betydande skillnader i hur väl dessa program kan identifiera AI-genererat innehåll. Detta påvisar behovet av tydliga och rättvisa riktlinjer för att hantera fall där studenter potentiellt felaktigt anklagas för att ha använt AI-verktyg.

Chalmers disciplinära regler inkluderar hantering av fusk och akademiskt oärligt beteende, men de betonar också att varje fall måste behandlas rättvist och med beaktande av alla omständigheter. Det är därför avgörande att utbildningsinstitutioner, inklusive Chalmers, utvecklar robusta bedömningsmetoder och säkerhetsåtgärder för att säkerställa att studenter inte felaktigt anklagas på grund av otillräckliga eller felaktiga AI-detekteringsverktyg.

En annan viktig aspekt är definitionen av plagiat i en tid där AI blir alltmer integrerat i utbildningen. ChatGPT får sin information från nätet på ett sätt som inte skiljer sig mycket från hur studenter själva söker svar på frågor online. Detta kräver en omdefiniering av vad som räknas som plagiat och hur akademisk ärlighet ska hanteras när både människor och AI bidrar till skapandet av lösningar.

För att hantera dessa utmaningar kan det vara fördelaktigt att informera studenter och lärare om hur AI kan användas på ett etiskt och pedagogiskt fördelaktigt sätt. Att inkludera AI i undervisningen, snarare än att förbjuda det, kan ge nya möjligheter för lärande och kreativitet, samtidigt som tydliga riktlinjer och bedömningsmetoder utvecklas för att säkerställa rättvisa och noggrannhet.

8

Slutsats

Denna studie har undersökt förmågan hos olika AI-detekteringsprogram att identifiera AI-genererad programmeringskod och text, med fokus på verktygen Contentdetector, Plagiarismdetector och Copyleaks. Resultaten visar att det finns betydande skillnader i hur dessa program presterar och att ingen av dem är helt tillförlitliga för att exakt skilja mellan mänskligt och AI-genererat innehåll.

Contentdetector visade sig ha stora begränsningar, särskilt med att hantera andra språk än engelska, och genererade många falska positiva resultat genom att felaktigt identifiera mänskligt skrivna texter som AI-genererade. Plagiarismdetector presterade något bättre men visade också på betydande inkonsekvenser i sina resultat, vilket kan leda till felaktiga bedömningar. Copyleaks var det mest tillförlitliga verktyget i denna undersökning, men även det hade sina begränsningar, såsom krav på en minsta textlängd och viss felmarginal.

Dessa resultat belyser den aktuella utmaningen med att förlita sig på AI-detekteringsprogram för att fatta viktiga akademiska beslut. Verktygens nuvarande bristande precision innebär en risk för felaktiga anklagelser om plagiat och orättvisa bedömningar, vilket kan ha negativa konsekvenser för studenternas akademiska karriärer.

Den snabba utvecklingen av AI-teknologier, jämfört med samhällets förmåga att anpassa sig, skapar ytterligare komplikationer. Det finns ett behov av att utveckla nya undervisningsmetoder och bedömningsstrategier som kan hantera dessa förändringar. Detta inkluderar att omdefiniera vad som räknas som plagiat och att införa nya riktlinjer för användning av AI i utbildningen.

Genomförandet av denna studie gav värdefulla insikter i hur effektivt olika AI-detektorer kan identifiera AI-genererat innehåll, och visade på behovet av vidare utveckling av dessa verktyg för att bättre hantera olika språk och komplexa svarstyper.

Med de verktygen som vi har idag är det svårt att upptäcka AI-genererad programkod. Ju mer dessa AI-teknologier utvecklas desto svårare blir det urskilja mänskligt skriven programkod från en AI-genererad. Dessutom kommer de att bli sämre med tiden med tanken på hur snabbt AI:n utvecklas. Denna utveckling gör att AI-genererat innehåll blir mer och mer likt mänskligt skriven programkod eller texter.

Sammanfattningsvis visar denna studie att nuvarande AI-detekteringsprogram har potential men också betydande begränsningar. Ytterligare forskning och utveckling krävs för att förbättra dessa verktyg, samtidigt som utbildningsinstitutioner behöver anpassa sina metoder för att hantera de nya utmaningar och möjligheter som AI-teknologier medför.

Litteraturförteckning

- [1] J. Berrezueta-Guzman and S. Krusche, “Recommendations to Create Programming Exercises to Overcome ChatGPT,” 2023 IEEE 35th International Conference on Software Engineering Education and Training (CSEE&T), Tokyo, Japan, 2023, pp. 147-151, doi: 10.1109/CSEET58097.2023.00031. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10229357> (Accessed: 2 may 2024).
- [2] S. Barnett, “ChatGPT is making universities rethink plagiarism,” *Wired*, January 30, 2023. Available at: <https://www.wired.com/story/chatgpt-college-university-plagiarism/> (Accessed: 25 April 2024).
- [3] Z. Taylor, C. Blair, E. Glenn, and T. R. Devine, “Plagiarism in Entry-Level Computer Science Courses Using ChatGPT,” 2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE), Las Vegas, NV, USA, 2023, pp. 1135-1139, doi: 10.1109/CSCE60160.2023.00189. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10487465> (Accessed: 25 April 2024).
- [4] Copyleaks, “AI Detection Accuracy Study,” 2024. [Online]. Available: <https://copyleaks.com/blog/ai-detection-accuracy-study>. [Accessed: 14-Jun-2024].
- [5] Chalmers University of Technology, “Regler för användning av AI-verktyg,” [Online]. Available: <https://www.chalmers.se/utbildning/dina-studier/kandidat-och-examensarbete/regler-for-anvandning-av-ai-verktyg/>. [Accessed: 14-Jun-2024].
- [6] Copyleaks, “AI Content Detector,” 2024. [Online]. Available: <https://copyleaks.com/ai-content-detector>. [Accessed: 14-Jun-2024].
- [7] Plagiarism Detector, “AI Content Detector,” 2024. [Online]. Available: <https://plagiarismdetector.net/sv/ai-content-detector>. [Accessed: 14-Jun-2024].
- [8] ContentDetector, “AI Content Detector,” 2024. [Online]. Available: <https://contentdetector.ai/>. [Accessed: Jun. 18, 2024].

- [9] Swedish Life Cycle Center, “Att införa livscykelerspektiv i miljöledningssystem,” 2018. [Online]. Available at: https://www.lifecyclecenter.se/wp-content/uploads/2018_03-Att-info%CC%88ra-livscykelerspektiv-i-miljo%CC%88ledningssystem.pdf. [Accessed: 14-Jun-2024].
- [10] A. Ushmani, “Machine learning pattern matching,” *J. Comput. Sci. Trends Technol.*, vol. 7, pp. 4–7, 2019. Available at: https://d1wqtxts1xzle7.cloudfront.net/58592293/IJCST-V7I2P2-libre.pdf?1552227541=&response-content-disposition=inline%3B+filename%3DIJCST_V7I2P2_Azhar_Ushmani.pdf&Expires=1718411522&Signature=S-xzw4XrelGFxy8Vbs5R41Kc1Sa11UVSy0lsFgVhsaDyRVAnbDfTFxUZ45CtGLTb~f2Z1YLpUVagm15_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA. (Accessed: 15 June 2024).
- [11] SurferSEO. “How Do AI Content Detectors Work?” SurferSEO Blog. Available online: <https://surferseo.com/blog/how-do-ai-content-detectors-work/>. [Accessed: 14-Jun-2024]. “ teknisk bakgrundsdelen “
- [12] E. Opara, M.-E. T. Adalikwu, and C. A. Tolorunleke, “ChatGPT for Teaching, Learning and Research: Prospects and Challenges,” *Glob Acad J Humanit Soc Sci*, vol. 5, no. 2, pp. 33-40, Mar. 2023. Available: <https://ssrn.com/abstract=4375470>. [Accessed: 17-06-2024]
- [13] B. Juba and H. S. Le, “Precision-Recall versus Accuracy and the Role of Large Data Sets,” *AAAI*, vol. 33, no. 01, pp. 4039-4048, Jul. 2019. [Accessed: 16-06-2024]
- [14] C. Gordon, “ChatGPT and generative AI innovations are creating sustainability havoc,” *Forbes*, Mar. 12, 2024. [Online]. Available: <https://www.forbes.com/sites/cindygordon/2024/03/12/chatgpt-and-generative-ai-innovations-are-creating-sustainability-havoc/>. [Accessed: 15-05-2024].
- [15] Copyleaks, “AI content detector FAQs,” May 2023. [Online]. Available: <https://copyleaks.com/wp-content/uploads/2023/05/ai-content-detector-faqs.pdf>. [Accessed: Jun. 19, 2024].
- [16] G. J. G. Upton, “Fisher’s Exact Test,” *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, vol. 155, no. 3, pp. 395–402, 1992. [Online]. Available: <https://doi.org/10.2307/2982890>. [Accessed: Jun. 2, 2024].

A

Appendix

```
public class Båt {
    private String namn;
    private int längd;
    public void skrivInfo() {
        System.out.println( "Namn: " + namn + "\nLängd: " + längd );
    }
}
public class Motorbåt extends Båt {
    private int motorstyrka;
    private int maxfart;
    public void skrivInfo() {
        super.skrivInfo();
        System.out.println( "Motorstyrka: " + motorstyrka +
            "\nMaxfart: " + maxfart );
    }
}
public class Segelbåt extends Båt {
    private int segelyta;
    public void skrivInfo() {
        super.skrivInfo();
        System.out.println( "Segelyta: " + segelyta );
    }
}

public class Båtregister {
    private ArrayList<Båt> register = new ArrayList<>();
    public void addera(Båt b) {
        register.add(b);
    }
    public void listaAlla() {
        for ( Båt b : register )
            b.skrivInfo();
    }
}
```

Figur A.1: Tenta lösningsförslag till fråga 22

```
public TreeMap<String, TreeMap<String, Integer>> sammanställPrestationer() {
    TreeMap<String, TreeMap<String, Integer>> ladok = new TreeMap<String,
    TreeMap<String, Integer>>();

    for (Studentprestation p : prestationer) {
        TreeMap<String, Integer> kursprestationer;

        if (!ladok.containsKey(p.getKurskod())) {
            kursprestationer = new TreeMap<String, Integer>();
            ladok.put(p.getKurskod(), kursprestationer);
        } else {
            kursprestationer = ladok.get(p.getKurskod());
        }

        if (!kursprestationer.containsKey(p.getPnr())) {
            // Studenten har tagit sina första poäng.
            kursprestationer.put(p.getPnr(), p.getPoäng());
        } else {
            // Studenten har poäng från tidigare moment.
            Integer poäng = kursprestationer.get(p.getPnr());
            kursprestationer.put(p.getPnr(), poäng + p.getPoäng());
        }
    }

    return ladok;
}
```

Figur A.2: Tenta lösningsförslag till fråga 23

```
public class LottoRow {
    private int[] row = new int[7];

    public LottoRow(int[] a) {
        if (a == null) {
            throw new IllegalArgumentException("Null");
        }
        if (a.length != 7) {
            throw new IllegalArgumentException("Bad length");
        }
        for (int i = 0; i < 7; i++) {
            for (int j = i + 1; j < 7; j++) {
                if (a[i] == a[j]) {
                    throw new IllegalArgumentException("Duplicate");
                }
            }
            row[i] = a[i]; // copies content to avoid aliasing
        }
    }

    @Override
    public String toString() {
        String res = "" + row[0];
        for (int j = 1; j < 7; j++) {
            res = res + ", " + row[j];
        }
        return res;
    }

    public int countMatches(LottoRow other) {
        if (other == null) {
            return 0;
        }
        int r = 0;
        for (int i = 0; i < 7; i++) {
            for (int j = 0; j < 7; j++) {
                if (row[i] == other.row[j]) {
                    r = r + 1;
                }
            }
        }
        return r;
    }
}
```

Figur A.3: Tenta lösningsförslag till fråga 33

Institutionen för data- och informationsteknik
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige
www.chalmers.se



GÖTEBORGS
UNIVERSITET



CHALMERS