

*UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO*

Damjan Košir

Implementacija in testiranje klepetalnika

*DIPLOMSKO DELO
NA UNIVERZITETNEM ŠTUDIJU*

Mentor: prof. dr. Igor Kononenko

Ljubljana, 2013



Št. naloge: 01897/2013

Datum: 04.02.2013

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **DAMJAN KOŠIR**

Naslov: **IMPLEMENTACIJA IN TESTIRANJE KLEPETALNIKA
CHATBOT IMPLEMENTATION AND TESTING**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

Naloga je raziskati trenutno stanje tehnologije klepetalnikov (angl. chatbots) in razviti in pretestirati primer skriptirnega klepetalnika. Pri tem argumentirano izberite ustrezne tehnologije. Čeprav obstajajo storitve, ki pomagajo zgraditi klepetalnik, se omejite na take, ki bi lahko tekli na lokalnem računalniku.

Mentor:

prof. dr. Igor Kononenko



Dekan:

prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani Damjan Košir

s vpisno številko 63050150

sem avtor diplomskega dela z naslovom:

IMPLEMENTACIJA IN TESTIRANJE KLEPETALNIKA

S svojim podpisom zagotavljam, da:

- *sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Igorja Kononenka*
- *so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela*
- *soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.*

V Ljubljani, dne 29.4.2013

Podpis avtorja:

Zahvala

Zahvaljujem se Anji Hercog, univ. dipl. prev., za izraz klepetalnik.

Kazalo

Zahvala	4
Povzetek.....	1
Abstract.....	2
1. Uvod	3
1.1. Turingov test	3
1.2. Loebnerjeva nagrada	4
2. Pregled področja	5
2.1. Zanimivi klepetalniki	5
2.1.1. ELIZA.....	5
2.1.2. A.L.I.C.E	5
2.1.3. Cleverbot	6
2.1.4. Angela.....	6
2.2. Pristopi določanja vsebine	7
2.2.1. Skriptiran pristop	7
2.2.2. Učljiv pristop	7
2.2.3. Kombiniran pristop.....	7
2.3. Primerjava skriptnih tehnologij.....	8
2.3.1. AIML	8
2.3.2. Chatscript.....	11
3. Razvoj svojega klepetalnika	16
3.1. Izbor tehnologije za svoj klepetalnik	16
3.2. Sestavljanje vsebine klepetalnika	16
3.2.1. Opis osebnosti.....	16
3.2.2. Glavne teme	17
3.2.3. Struktura teme.....	17
3.2.4. Pobude, nato vprašanja	18
3.2.5. “Razumevanje” uporabnika.....	19
3.3. Opis implementacije	19
3.4. Testiranje in razvoj	20
3.5. Nadaljnji razvoj.....	21
4. Zaključek in nadaljnje delo.....	22
Dodatek A.....	23
Dodatek B.....	27
Dodatek C.....	28
5. Viri in literatura	29

Povzetek

V diplomski nalogi sem raziskal različne tehnologije za izdelavo klepetalnikov. Omejil sem se na tehnologije, ki omogočajo izdelavo klepetalnikov, ki lahko tečejo na lokalnem računalniku. Za določanje vsebine klepetalnika, in s tem znanja in osebnosti, obstajata dva pristopa in to sta skriptirani in učljiv. Pri skriptiranem pristopu vsebino določimo ročno, pri učljivem pristopu pa jo s strojnim učenjem pridobimo iz baze pogovorov. Ker nisem imel dostopa do baze pogovorov, sem se moral omejiti na skriptiran pristop k določanju vsebine. Kot tehnologiji z največ potenciala sem identificiral AIML (Artificial Intelligence Markup Language) in Chatscript. Glavne prednosti AIML so velika skupnost, velik nabor prosto dostopnih klepetalnikov, ki lahko služijo kot osnova za nov klepetalnik in več interpreterjev. Glavne prednosti tehnologije Chatscript pa so močan jezik vzorcev, ki omogoča hitrejšo in lažje pisanje vsebine, aktiven razvoj in možnost uporabe podatkovnih baz, kot je Wordnet. Potem ko sem preučil zmogljivosti obeh izbranih tehnologij, sem za izdelavo lastnega klepetalnika izbral tehnologijo Chatscript. Nato je sledila priprava opisa osebnosti klepetalnika in pisanje vsebine razdeljene v teme. Za pravilno menjavanje tem in potek pogovora s klepetalnikom sem moral prilagoditi kontrolno skripto. Po prvotnem razvoju je nato sledilo več iteracij testiranja in izboljševanja. Zaradi pomanjkanja objektivne metrike kvalitete klepetalnika je bil to najboljši način razvoja. Z novo pridobljenim znanjem in izkušnjami sem prišel do zaključka, da so trenutne tehnologije klepetalnikov dovolj dobre za uporabo v zabavne namene in kot virtualne asistente, niso pa še dosegle nivoja, kjer ljudje ne bi ločili klepetalnika od človeškega sogovornika v Turingovem testu.

Ključne besede: klepetalnik, Turingov test, Chatscript, AIML

Abstract

In my thesis I did a research on different chatbot technologies. I only looked at technologies that enable the use of chatbots on a local computer. There are two approaches to determine the chatbot's content, including knowledge and personality, the scripted and the learned approach. If we choose the scripted approach, we determine the content ourselves, if we choose the learned approach, we get it with machine learning from a conversation database. Since I didn't have access to a conversation database, I had to take the scripted approach to determine the content. I identified AIML (Artificial Intelligence Markup Language) and Chatscript as the technologies with the most potential. The main advantages of AIML are a large community, a large collection of accessible chatbots available for free, which can serve as a base for a new chatbot, and many interpreters. The main advantages of Chatscript are a strong pattern language, enabling faster and easier content writing, active development and the possibility of the use of databases, such as Wordnet. After researching the performance of both chosen technologies, I decided to use Chatscript to make my own chatbot. Next came the description of the chatbot's personality and writing the content divided into different topics. For proper changing of topics and a good conversation flow with the chatbot I had to adjust the control script. After the first development phase there were several testing and improvement iterations. Due to a lack of objective chatbot quality metrics this was the best way of development. With the newly acquired knowledge and experience I concluded that the technologies available at the moment are good enough to be used for entertainment or as virtual assistants, but they still haven't reached the level to pass the Turing test and thus make people unable to distinguish a chatbot from a human conversation partner.

Keywords: chatbot, Turing test, Chatscript, AIML

1. Uvod

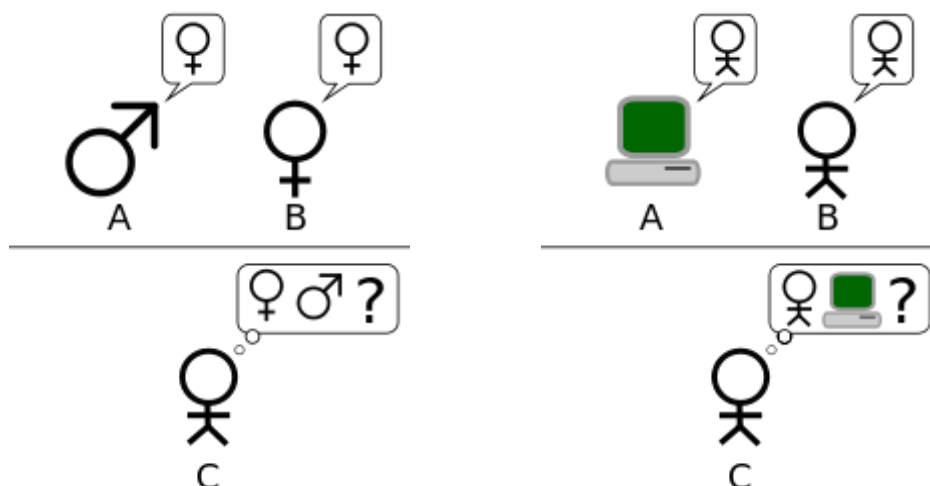
Leta 1950 je Alan Turing v članku “Computing Machinery and Intelligence” [1] zastavil vprašanje “Ali lahko stroji mislijo?” in s tem postavil temelje področja umetne inteligence. Predstavil je tudi način, kako to preizkusiti, kar danes poznamo po imenoma Turingov test. Najbolj pogosta interpretacija tega testa je igra, pri kateri s tekstovnim klepetom izpraševalec poizkuša ugotoviti, kateri izmed drugih dveh udeležencev je človek in kateri računalnik. Napovedal je, da nam bo okoli leta 2000 to v veliki meri uspelo. Kako daleč smo sedaj leta 2013? Nam je uspelo? Se je izkazalo za nemogoče? Program s katerimi se uporabnik pogovarja imenujemo klepetalnik (angleško chatbot). Nekateri klepetalniki so namenjeni ustvarjanju iluzije pametnega pogovora in bi ustrezali opisu v Turingovem testu.

Namen tega dela je raziskati trenutno stanje tehnologije na področju klepetalnikov in implementirati lasten klepetalnik. Pri izbiri tehnologije za implementacijo lastnega klepetalnika se imam namen omejiti na takšne, ki omogočajo delovanje na lokalnem računalniku brez pomoči zunanjih storitev.

1.1. Turingov test

“Ali lahko stroji mislijo?”, je vprašanje, ki si ga zastavi Alan Turing v članku iz leta 1950 “Computing Machinery and Intelligence” [1]. Po hitri ugotovitvi, da je to preveč nedoločeno vprašanje, ga zamenja s sedaj imenovanim Turingovim testom. Predstavi nam igro posnemanja, ki jo igrajo 3 osebe (A, B in C). Od oseb A in B je ena ženska in en moški. Oseba C ima vlogo izpraševalca in mora ugotoviti, kdo je moški in kdo ženska. Oseba C komunicira z osebama A in B s pomočjo teleprinterja. Nato v tej igri zamenja osebo A s strojem.

V istem članku nam Turing predstavi tudi svoje osebno mnenje o zadanem problemu in to je, da bodo čez približno 50 let (torej okoli leta 2000) stroji igrali igro posnemanja tako dobro, da po 5 minutnem pogovoru povprečni izpraševalec ne bo pravilno identificiral stroja v več kot 70 odstotkih primerov.



Slika 1: igra posnemanja

Slika 2: standardna interpretacija Turingovega testa

1.2. Loebnerjeva nagrada

Loebnerjeva nagrada za umetno inteligenco je svetovno najbolj znana nagrada na področju klepetalnikov. Gre za prvi formalni primer uporabe Turingovega testa. Podeli se jo vsako leto na tekmovanju umetne inteligence, prejme pa jo tekmovalec s klepetalnikom, ki ga komisija izbere za najbolj človeku podobnega.

Tekmovanje se je pričelo leta 1990. Začel ga je Hugh Loebner s sodelovanjem centra Cambridge za proučevanje vedenja (Massachusetts, ZDA). Tekmovanje temelji na Turingovem testu. V vsakem krogu tekmovanja se sodnik (človek) hkrati pogovarja z računalniškim programom (klepetalnikom) in z drugim človekom (ki piše preko računalnika). Glede na odgovore, ki jih prejme med pogovorom z obema, se mora sodnik nato odločiti, kateri sogovornik je človek, kateri pa je klepetalnik. Dr. Loebner je najavil, da bo prvi klepetalnik, ki ga ne bo mogoče razlikovati od človeškega sogovornika, prejel nagrado 100.000 \$ in zlato medaljo, saj za tak klepetalnik lahko rečemo, da "razmišlja". Klepetalnik, ki bi vsaj polovico sogovornikov prepričal, da je človek, bi dobil srebrno medaljo. Vsako leto pa klepetalnik, ki je najbolj podoben človeškemu sogovorniku, prejme bronasto medaljo in simbolično denarno nagrado. Zmagovalec v posameznem letu je najboljši klepetalnik glede na ostale prijavljene klepetalnike v istem letu, ne glede na to, kako dober je dejansko v absolutnem smislu. Zlate in srebrne nagrade doslej ni prejel še nihče.

Tekmovanje je potekalo že na več lokacijah, kot npr. univerza Flinders University, Dartmouth College, muzej znanosti v Londonu (Science Museum) in University of Reading, v letih 2004 in 2005 pa ga je Loebner celo gostil pri sebi doma v New Yorku.



Slika 3: zlata medalja za Loebnerjevo nagrado

Na področju umetne inteligence je Loebnerjeva nagrada nekoliko kontroverzna. Da bi prejel zlato medaljo, bi moral klepetalnik sodnika tudi avdio-vizualno prepričati v svojo človeškost, kar je praktično nemogoče doseči, poleg tega pa se Turing v svoji definiciji preizkusa "mislečega stroja" natančno omeji samo na pisni pogovor. Tudi dejstvo, da so sodniki, ki se pogovarjajo s klepetalniki na tekmovanju le štirje, ne govori v prid verodostojnosti rezultatov tekmovanja.

2. Pregled področja

2.1. Zanimivi klepetalniki

2.1.1. ELIZA

Elizo je napisal Joseph Weizenbaum med leti 1964 in 1966. Program oponaša psihoterapevta, ki uporablja nedirektivno terapijo. Kakor opisano v [3], to doseže s približno 40 pravili zamenjav in 40 vzorci, ki jih razpozna. Kljub svoji preprostosti je bil program dokaj uspešen pri prepričevanju ljudi, vsaj pri kratkih pogovorih. Na to je verjetno zelo vplivalo dejstvo, da ljudje takrat niso bili vajeni interakcije z računalniki. Eliza je dokaz, da bi lahko bilo možno narediti program, ki prestane Turingov test.

How do you do. Please state your problem.

>> I am lonely.

Is it because you are lonely that you came to me ?

>> Yes.

You seem to be quite positive.

>> It is hard to stay positive when you are so lonely.

What makes you think I am so lonely ?

>> Not you, me.

We were discussing you -- not me.

>> I can't sleep at night.

How do you know that you can't sleep at night ?

>> I can't seem to fall asleep.

Have you tried ?

>> Yes, a lot.

You are sure.

2.1.2. A.L.I.C.E

Klepetalnik, opisan v [2], je največji predstavnik tehnologije AIML. Sestavlja ga približno 150000 pravil. A.L.I.C.E. (Artificial Linguistic Internet Computer Entity) je pogosta osnova za ostale klepetalnike, saj so pravila na voljo pod licenco GPL. Ta licenca je bila izbrana skoraj po naključju, saj je bila prekopirana iz urejevalnika Emacs, ki je bil uporabljen za začetno programiranje. Dr. Richard Wallace je z A.L.I.C.E. prejel Lebner prize leta 2000, 2001 in 2004. Leta 2001 je bil ustanovljen ALICE A.I. Foundation, ki skrbi za razvoj standarda AIML.

judge: HI.

ALICE: Hi there!

judge: How are you doing?

ALICE: Not too bad.

judge: What's new?

ALICE: I am chatting with clients on the internet.

judge: Are they saying anything interesting?

ALICE: As far as I know they are.

judge: Do you have anything interesting to say?

ALICE: No I don't have it. But I do have a great botmaster.

2.1.3. Cleverbot

Ustvaril ga je Rollo Carpenter. Klepetalnik pridobiva znanje s strojnim učenjem iz baze več milijonov pogovorov s svojimi uporabniki. Baza pogovorov se neprestano veča, saj se je možno pogovarjati z njim preko spletne strani ali mobilne aplikacije. Leta 2010 je bilo približno 300000 ur pogovora mesečno [8]. Spletna verzija opravlja 3 iskanja za vsak odgovor, za tekmovanja pa se uporablja prilagojen klepetalnik, ki opravlja 42 iskanj.

Rollo Carpenter je prejel Loebners prize leta 2005 s klepetalnikom Jabberwacker 2005 in leta 2006 s klepetalnikom Joan. Oba klepetalnika sta prednika Cleverbota.

User: Hi.

Cleverbot: Hello.

User: What are you doing?

Cleverbot: Talking to you.

User: Good answer.

Cleverbot: Thanks.

User: How are you today?

Cleverbot: Afraid of sleep.

User: Why are you afraid of sleep?

Cleverbot: Because you say that I'm beautiful.

User: I didn't say that.

Cleverbot: Yes you did.

2.1.4. Angela

Najsodobnejši predstavnik tehnologije Chatscript. Razvila sta jo Bruce Wilcox in njegova žena Sue v sodelovanju s podjetjem Outfit 7 [5]. Namenjen je zabavnemu pogovoru z likom mačke. Vsebuje približno 16000, pravil za kar je bilo porabljeno približno 5 človek-mesecev in veliko dodatnega časa za testiranje. Na Loebner prize 2012 je dosegel drugo mesto, čeprav je trdil, da je mačka. Predhodnika Suzette in Rosette sta zmagala na Loebner prize leta 2010 in 2011. Klepetalnik Suzette je celo v finalu prepričal enega od štirih sodnikov [6].

Angela: Hey, I saved you a chair and everything.

User: How are you today?

Angela: Good, thanks. How are you?

User: Fine.

Angela: Glad you are in good health.

User: What did you do yesterday?

Angela: I went to a museum than walked in the zoo.

User: Did you see elephants?

Angela: I don't have any elephants.

2.2. Pristopi določanja vsebine

Prva odločitev, ki jo je treba sprejeti pri izdelavi klepetalnika, je, kako se bo določalo vsebino. Moje mnenje je, da se je zaenkrat za najboljše rezultate dobro odločiti za skriptiran pristop. V prihodnosti bo verjetno prevladal učljiv pristop z malenkostnimi skriptiranimi popravki za fino definicijo osebnosti.

2.2.1. Skriptiran pristop

Večina klepetalnikov je skriptiranih, kar pomeni, da imajo pogon za prepoznavanje vzorcev, ki sprejema odločitve na podlagi ročno napisanih pravil. Tak pristop omogoča visok nadzor nad vsebino in osebnostjo klepetalnika. Glavna slabost je velika količina potrebnega dela.

Nekateri sistemi omogočajo analizo pogovorov in iščejo najpogostejše vhode, za katere še ni napisanih pravil in pogostosti uporabe pravil. Na tak način se lažje določi dele vsebine, kjer bo dodatno delo prineslo največ rezultatov.

Čeprav je nemogoče z ročno napisanimi pravili pokriti vse možne pogovore, se lahko temu zelo približamo. Dr. Richard Wallace je ugotovil, da so pri njegovem klepetalniku skoraj vsi vnosi pokriti s približno 45000 pravili [7].

Klepetalniki s takim pristopom trenutno dosegajo najboljše rezultate na tekmovanjih.

2.2.2. Učljiv pristop

Določeni klepetalniki, med katerimi je najbolj znan Cleverbot, nimajo ročno določene vsebine, ampak se ta določa s pomočjo strojnega učenja. Za tak pristop se potrebuje ogromno bazo pogovorov, iz katere se lahko nato analizira in po kateri se išče primerne odgovore. Velika prednost tega pristopa je, da se z večanjem baze (torej uporabe klepetalnika) rezultat izboljšuje brez večjega dodanega dela. Prav tako je pogosto možno s povečanjem procesorske moči izboljšati rezultate.

2.2.3. Kombiniran pristop

V zadnjem času se pojavlja možnost izdelave klepetalnika, ki uporablja skriptiran pristop za vsebino, kadar pa ni določenih pravil, pa uporabi učljiv pristop. Predstavnik takšnih klepetalnikov je Pupito, ki je sicer preprost klepetalnik, vendar kaže velik potencial tega pristopa. Pupito uporablja približno 300 napisanih pravil in bazo 2 milijonov pogovorov. Pupito je predstavnik klepetalnikov, ki se jih lahko naredi s storitvijo Cleverscript.

2.3. Primerjava skriptnih tehnologij

Ker nimam na voljo velike baze pogovorov, sem se iz praktičnih razlogov odločil uporabiti skriptiran pristop določanja vsebine svojega klepetalnika. Zato sem natančneje preučil samo take vrste tehnologij.

Pri skriptiranem pristopu je največja omejitev hitrost, s katero lahko ustvarjaš novo vsebino. Izbor sem zožal na 2 tehnologiji. Glavna prednost AIML je veliko prosto dostopnih klepetalnikov, ki lahko tvorijo osnovo za lasten klepetalnik. Glavna prednost Chatscripta pa so zelo močan jezik za opis vzorcev in uporaba baz, kot je Wordnet.

2.3.1. AIML

Artificial Intelligence Markup Language je leta 1995 s pomočjo skupnosti pričel razvijati Richard Wallace [2]. Standard je bil objavljen leta 2001 in istega leta je dobil tudi svojo prvo posodobitev v obliki verzije 1.0.1. Zapis je v obliki XML, kar omogoča uporabo že obstoječih razčlenjevalnikov.

Obstaja veliko aktivno vzdrževanih interpreterjev. Prav tako obstajajo spletne storitve, ki pomagajo pri gradnji in gostovanju klepetalnikov, ki bazirajo na tehnologiji AIML in pogosto ponujajo nestandardne razširitve. Najbolj znan tak servis je Pandorabots.

Pred kratkim je bil objavljen tudi osnutek verzije 2.0. Ker standard še ni dokončan, sem se omejil na verzijo 1.0.1.

Kategorija

Osnovno enoto znanja v AIML predstavlja kategorija (category), ki je sestavljena iz vhoda, izhoda in konteksta. Vhod je podan v obliki vzorca (pattern). Izhod je podan v obliki predloge (template). Kontekst je neobvezen in je lahko podan kot tema (topic) ali tisto (that) ali oboje.

```
<category>
<pattern>WHAT IS YOUR NAME</pattern>
<template>My name is Alice.</template>
</category>
```

Vzorci

Vzorci so glavni pri določanju, kdaj se bo uporabila določena kategorija, kar pomeni, da se bo uporabila predloga iz te kategorije. V AIML so vzorci neobčutljivi na velikost črk in se zato ponavadi pišejo z velikimi črkami. Vsebujejo lahko besede, presledke in nadomestna znaka (zvezdica in podčrtaj). Verzija 1.0.1 je uvedla možnost več nadomestnih znakov v enem vzorcu. Zvezdica pomeni eno ali več besed.

Predloge

Predloge določajo izhod, torej odziv klepetalnika. Najpreprostejše predloge so sestavljene samo iz besedila, kompleksnejše pa vsebujejo še druge značke, ki vplivajo na izhod ali stanje klepetalnika.

Star

`<star/>` se uporablja v predlogih. Interpreter jih zamenja z delom uporabnikovega vhoda, ki se je v vzorcu ujema z zvezdico.

```
<category>
<pattern>MY NAME IS *</pattern>
<template>Nice to meet you <star/></template>
</category>
```

Srai

Srai se uporablja znotraj predloge in povzroči, da se vse med `<srai>` in `</srai>` ponovno vnese kot vhod in se obdeli. To se lahko uporablja za simbolično redukcijo, implementacijo črkovanja, sinonimov, vejenja in strategije deli in vladaj. Rezultat se nato združi s preostankom predloge.

simbolična redukcija

```
<category>
<pattern>DO YOU KNOW WHO * IS</pattern>
<template><srai>WHO IS <star/></srai></template>
</category>
```

sinonim

```
<category>
<pattern>HI</pattern>
<template><srai>HELLO</srai></template>
</category>
```

vejenje

```
<category>
<pattern>WHO IS HE</pattern>
<template><srai>WHOSHE <get name="he"/></srai></template>
</category>
<category>
<pattern>WHOSHE *</pattern>
<template>He is <get name="he"/>.</template>
</category>
<category>
<pattern>WHOSHE UNKNOWN</pattern>
<template>I don't know who he is.</template>
</category>
```

deli in vladaj

```
<category>
<pattern>YES *</pattern>
<template><srain>YES</srain><srain><star/></srain></template>
</category>
```

črkovanje

```
<category>
<pattern>YOUR A *</pattern>
<template>I think you mean"you're" or"you are" not"your."
  <srain>YOU ARE A <star/></srain>
</template>
</category>
```

Več možnosti

Da klepetalnik ne daje vedno istih odgovorov, se v predlogi lahko zapiše več možnosti z uporabo značke random. Interpreter nato naključno izbere eno izmed možnosti iz nabora, ki so označene z značko li.

```
<category>
<pattern>*</pattern>
<template>
<random>
<li>What's your favorite car?</li>
<li>What kind of car do you drive?</li>
<li>Do you get a lot of parking tickets?</li>
<li>My favorite car is one with a driver.</li>
</random>
</template>
</category>
```

Predikati

Predikate se lahko nastavlja v predlogah z uporabo značke set in bere s pomočjo značke get. Poseben predikat je topic, ki se uporablja za določanje konteksta.

```
<template>
<set name="he">Samuel Clemens</set> is Mark Twain.
</template>
```


Kontekst

V primeru, da hočemo v več primerih imeti enak vzorec in drugačne predloge glede na kontekst lahko uporabimo značko *that*. Tako kot mora uporabnikov vhod ustrezati vzorcu, da se uporabi predloga v trenutni kategoriji, tako mora klepetalnikov prejšnji odgovor ustrezati zapisu znotraj značke *that*.

```
<category>
<pattern>YES</pattern>
<that>DO YOU LIKE MOVIES</that>
<template>What is your favorite movie?</template>
</category>
```

Ker se lahko zgodi, da bi radi imeli enak vzorec in enak zadnji odgovor klepetalnika (*that*), vendar različne predloge, lahko to dosežemo z nastavljanjem teme (*topic*). Temo se lahko nastavi kadarkoli znotraj predloge z nastavljanjem predikata *topic*.

Tema se lahko uporabi tudi za izbiro seta pobudnih stavkov, ki bi načeli pogovor na določeno temo.

```
<topic name="CARS">
<category>
<pattern>*</pattern>
<template>
<random>
<li>What's your favorite car?</li>
<li>What kind of car do you drive?</li>
<li>Do you get a lot of parking tickets?</li>
<li>My favorite car is one with a driver.</li>
</random>
</template>
</category>
```

2.3.2. Chatscript

Ustvaril ga je Bruce Willcox, kot naslednika zaprtokodnega CHAT-L [4]. Prosto je na voljo pod licenco BSD. Je v konstantnem, aktivnem razvoju in okoli njega se počasi gradi skupnost. Zaradi močnega jezika vzorcev omogoča pisanje očitno manjšega števila pravil. Angela je naj sodobnejši predstavnik klepetalnikov s tehnologijo Chatscript in vsebuje približno 10 krat manj pravil kot A.L.I.C.E., vendar dosega boljše rezultate na tekmovanjih. Kljub vsemu pa je dovolj učinkovit, da se lahko izvaja tudi na mobilnih napravah.

Prosto dostopnih večjih klepetalnikov sicer ni na voljo, vendar vsebuje približno 1500 konceptov, ki olajšajo pisanje vsebine in celotno ontologijo Wordnet.

Pravila

V Chatscriptu se enota znanja imenuje pravilo. Vsako pravilo je sestavljeno iz štirih delov: tip, oznaka, vzorec in izhod.

?: *MEAT (do you like meat) I love meat.*

Tip pravila in vzorec določata, ali se bo ob danem uporabnikovem vhodu uporabil izhod tega pravila. Razen kadar je pravilo drugače označeno, se lahko proži samo enkrat. S tem dosežemo, da se klepetalnik ne ponavlja in da lahko pripoveduje zgodbe.

Tipi pravil

t: – pobuda (gambit)

s: – preprosta replika (responder) na izjavo

?: – preprosta replika (responder) na vprašanje

u: – preprosta replika (responder) na vprašanje ali izjavo

a: b: c: ... – navezovalna replika (rejoinder)

Pobuda se uporablja, kadar želimo podati izhod pravila ne glede na uporabnikov vhod. Zato tudi pobude, razen izjemoma, nimajo podanega vzorca. Preproste replike so namenjene odzivanju na zadnji uporabnikov vnos, ne glede na prejšnje vnose (znotraj trenutne teme). Navezovalne replike so vgnezdene znotraj ostalih pravil in so namenjene odzivanju v primeru, da je prejšnji uporabnikov vnos ustrezal pravilu na višjem nivoju.

Oznaka

Oznaka je neobvezen del pravila, ki služi kot komentar, pomoč pri razhroščevanju in sledenju in kot cilj klicev drugih pravil. S pomočjo oznake lahko druga pravila uporabijo izhod tega pravila in njegove navezovalne replike, ga omogočijo ali onemogočijo, ...

Vzorec

Vzorec, skupaj s tipom pravila, določa ali se bo ob trenutnem uporabnikovem vhodu prožilo pravilo. Najpogostejši nadomestni znak je *, ki predstavlja nič ali več besed. Če želimo omejiti količino besed lahko uporabimo *~N (kjer je N število), kar predstavlja nič ali največ N besed. Primer vnosa, ki ustreza vzorcu, je označen z #!, je neobvezen, vendar zelo priporočljiv. Uporablja se za večjo berljivost in avtomatsko preverjanje vzorcev.

#! I really love the essence of you

*s: LOVE (I * love * you) That is sweet of you.*

Oznaka ! pred besedo določa odsotnost te besede v uporabnikovem vhodu.

*s: LOVE (!not I *~2 love *~2 you) That is sweet of you*

Oznaki << in >> zahtevata prisotnost besed med njima, vendar v poljubnem vrstnem redu.

#! What food do you like?

?: (!not << you food like >>) I love spaghetti.

Vsi vzorci se obnašajo enako, kot da bi imeli na koncu in na začetku *. Če želimo, da je neka beseda prva beseda v vrodu ali zadnja beseda v vrodu, lahko to specificiramo z uporabo oznak < in >.

#! do you have free will?

?: (< do you have free will) [I did, but someone stole it yesterday.] [Is anything truly free?]

Predprocesiranje vhoda

Zaradi velike poenostavitve pisanja vzorcev se vsak uporabnikov vhod predhodno obdela.

- v dolgi obliki izpiše okrajšave (npr. *you're* se spremeni v *you are*)
- samostalnike v množini pretvori v edninsko obliko
- pridevnike in prislove pretvori v osnovno obliko
- določila *a, an, the, some, these, those, that* spremeni v *a*
- osebne zaimke pretvori v prvo osebo ednine (npr. *me, my, myself* pretvori v *I*)
- številke zapisane z besedami zapiše s ciframi
- cela števila v zapisu s plavajočo vejico zapiše v zapisu za cela števila (npr. 3.0 se pretvori v 3)

Z uporabo opuščaja pred besedo lahko preprečimo predprocesiranje.

#! I walked away from the homes.

*?: (' I * walked * home) OK*

Koncepti

Koncepti so množice besed. Označeni so s predpono ~. Uporabljajo se v vzorcih in predstavljajo katerokoli besedo iz njihove množice. Ponavadi združujejo sinonime in podpomenke. Chatscript vsebuje koncepte, pridobljene iz leksikalne baze Wordnet in približno 1500 drugih, ki zelo olajšajo pisanje vsebine.

concept: ~meat [hamburger steak beef chicken lamb veal]

concept: ~ingest [eat swallow chew munch "take in" consume]

concept: ~like [adore love like "take a shine to" "be partial to"]

concept: ~negation [not never rarely uncommonly]

#! I really love chicken.

s: LOVE (!~negation << I [~like ~ingest] ~meat >>) You are obviously not a vegan.

Z uporabo podčrtaja lahko ugotovimo, katera beseda iz množice, ki jo predstavlja koncept, je bila uporabljena.

#! I drank cups of teas last night.

*?: (I * _~ingest * _~beverage) Why do you _0 ' _1 ?*

Izhod

Izhod predstavlja, kaj bo klepetalnik povedal. V večini primerov je to samo tekst, lahko pa vsebuje tudi dele uporabnikovega vnosa, makroje ali druge ukaze.

#! My name is John Smith

s: (I name be _) Nice to met you _0*

Da se izognemo ponavljanju klepetalnika, lahko zapišemo več izhodov za eno pravilo in bo sistem naključno izbral enega izmed njih.

#! do you have free will?

?: (< do you have free will) [I did, but someone stole it yesterday.] [Is anything truly free?]

Če želimo uporabiti isti izhod (in vgnezdena pravila) pri več pravilih, lahko uporabimo makro `^reuse`(oznaka). Ta makro uporabi izhod pravila s podano oznako. To nam olajša pisanje vsebine in ker se privzeto vsako pravilo proži le enkrat, tudi preprečuje, da bi klepetalnik uporabniku razlagal nekaj, o čemer ga je ta že vprašal.

t: AGE () I am 30 years old.

#! how old are you

?: (<< [how what] you [old age] >>) ^reuse (AGE)

Chatscript omogoča tudi kompleksnejše konstrukte kot so vejitve, zanke in cel nabor makrojev.

Spremenljivke

Pogosto si je koristno zapomniti določene informacije o uporabniku za dalj časa, kot samo en stavek. Chatscript nam za to omogoča uporabo spremenljivk.

u: ([go fly travel] to _) \$target = '_0 OK, you want to go to \$target .*

Prav tako lahko uporabnika povprašamo o stvareh, ki nam jih še ni povedal. Da to dosežemo, lahko preverjamo prisotnost ali odsotnost spremenljivk.

u: (!\$target) Where do you want to go?

Teme

Chatscript omogoča pisanje vsebine ločeno po temah. To razdeli vsebino na bolj obvladljive dele. Vsaki temi se določi ključne besede (lahko tudi koncepte). Na podlagi ključnih besed se nato sistem odloča, katero temo je načel uporabnik. Teme so ponavadi sestavljene iz dela

pravil, ki so pobude (gambit) in dela pravil, ki so replike (responder). Pobude so namenjene spodbujanju pogovora o tej temi ali pripovedovanju zgodb. Replike pa so namenjene odgovarjanju na uporabnikova vprašanja o tej temi ali odzivanju na uporabnikove izjave. Način prehajanja med temami in kaj narediti, ko zmanjka vsebine na trenutno temo, lahko določamo s kontrolno skripto.

topic: ~personal_self [age sex height weight old]

t: How old are you?

#! 35

a: (_~number>30) Over the hill.

#! Eighteen

a: (_~number==18) Have you had your first cocktail yet?

#! yes

b: (~yesanswer) Did you overdo it?

#! no

c: (~noanswer) How restrained of you.

t: AGE () I am 30 years old.

#! How sad

a: (!? ~badness) I'm used to it.

#! what is your age

#! how old are you

?: (<< [how what] you [old age] >>) ^reuse (AGE)

3. Razvoj svojega klepetalnika

3.1. Izbor tehnologije za svoj klepetalnik

Za svoj klepetalnik sem se odločil uporabiti Chatscript. Kot veliko prednost vidim močan jezik vzorcev.

?: *(you * love * me) You love me?*

Za ta vzorec zapisan, kot pravilo Chatscripta, bi v zapisu AIML potrebovali 16 pravil. Z uporabo konceptov pa bi število potrebnih pravil AIML zelo naraščalo. V veliko pomoč so vgrajeni koncepti za stvari, kot so: pritrdilni odgovor, nikalni odgovor, pozdrav, ... Sicer ni prosto dostopnih klepetalnikov kakor pri AIML, vendar to zagotavlja, da bo klepetalnik unikaten in ne preveč podoben A.L.I.C.E., kakor mnogi klepetalniki AIML.

Sistem se mi zdi zanimiv, ker se lahko osredotočiš na eno tematiko naenkrat in na eno zgodbo naenkrat, nato pa sistem sam poskrbi, da bo uporabljeno ob primernem času. Nekatere prednosti so popolnoma subjektivne in bolj ustrezajo mojemu pristopu organiziranja vsebine. Ampak če pogledamo uspehe na tekmovanjih, lahko vidimo, da so klepetalniki, ki temeljijo na Chatscriptu, uspešnejši od tistih, ki temeljijo na AIML.

V bližnji prihodnosti bo AIML dosegel verzijo 2.0. Osnutki novega standarda so že na voljo in popravljajo nekaj očitnih pomanjkljivosti AIML. Na voljo bo nadomestni znak za nič ali več besed (tako kot * v chatscriptu) in uvedeni bodsta tudi strukuri množica in slovar, ki bosta opravljali podobno funkcijo kot koncept v Chatscriptu. Kljub novim zmogljivostim AIML bi se še vedno odločil za Chatscript.

3.2. Sestavljanje vsebine klepetalnika

Neglede na izbrano tehnologijo klepetalnik ne more nastati brez vsebine. Ta določa njegovo osebnost, znanje in izkušnje. Ker sem za izdelavo lastnega klepetalnika izbral Chatscript, sem se pri pisanju vsebine opiral na navodila avtorja Chatscripta Burca Willcoxa in njegove žene Sue Wilcox, ki črpata iz večletnih izkušenj pisanja klepetalnikov. Kljub temu, da so navodila namenjena pisanju vsebine s Chatscriptom in povzeta po [5], je večina nasvetov enakih tudi pri uporabi drugih tehnologij.

3.2.1. Opis osebnosti

Preden se lotimo izdelave klepetalnika, moramo zastaviti njegovo osebnost. Kakšnega spola je? Koliko je star? Kaj je po izobrazbi in po poklicu? Kaj rad počne, je, pije? Kakšen je po značaju? Kaj je že vse doživel? Ko si odgovorimo na ta vprašanja in na široko opišemo odgovore, je veliko lažje narediti klepetalnik, ki bo skozi celoten pogovor odgovaljal konsistentno. Ta korak je še posebno pomemben, če vsebino za klepetalnik piše več oseb, saj imajo tako vsi pisci predlogo, ki se je držijo in ne pride do razlik in odstopanj v osebnosti klepetalnika.

V dodatku A je del opisa osebnosti mojega klepetalnika. Vključene so informacije, kot npr. osnovni osebni podatki (ime, spol, starost, rojstni dan, kraj bivanja...), zunanji videz, podatki o zdravju, natančen opis osebnosti (značaj, hobiji, talenti), prehranjevalne navade, opis doma, kariera (oz. karijerne želje), opis družine, prijateljev, natančen opis zgodnjega otroštva, šolanja in začetka odrasle dobe.

3.2.2. Glavne teme

Ko imamo določeno osebnost našega klepetalnika, moramo seveda določiti tudi, o katerih temah se bo klepetalnik pogovarjal z uporabnikom. Na začetku izberemo teme, ki so klepetalniku blizu glede na osebnost, ki smo jo določili. Ko te teme dodelamo, lahko pričenmo dodajati nove teme. Tako si npr. zamislimo, da bo klepetalnik za začetek znal govoriti o svojih izkušnjah iz otroštva in ustvarimo prvo temo - otroštvo. Nato bi radi, da govori o kakšnem od svojih interesov in ustvarimo temo o glasbi. Potem dodamo temo o poklicu in službi in tako naprej. Teme dodajamo, ko so obstoječe teme že dovolj dodelane in izpopolnjene.

3.2.3. Struktura teme

Ko se odločimo za neko glavno temo, moramo napisati tematsko strukturo za to temo. Če je tema, ki jo želimo vključiti, potovanje v Novo Zelandijo, potem so podteme lahko npr.:

- o državi, pokrajini, vremenu
- o selitvi v Novo Zelandijo
- o zgodovini in kulturi
- splošno

Podteme morajo vsebovati tudi osebne izkušnje, saj to klepetalnik naredi veliko bolj oseben in "človeški". Če pišemo, da naš klepetalnik želi nekam potovati, moramo raziskati, zakaj želi tja potovati, od kod ta želja, kakšne ima predstave o tem. Če je že bil na kakšnem potovanju, moramo vedeti, kaj je tam doživel, se je zgodilo kaj slabega, so ga oropali, je ves čas deževalo, je zato stopil v lužo in si razmočil svoje najboljše usnjene čevlje? To so stvari, ki klepetalnik pripeljejo na neko osebno raven in sogovornik bo tako začel verjeti, da se res pogovarja s pravo osebo in ne z računalniškim programom.

Pri dodajanju vsebine moramo biti pozorni tudi na to, katere informacije smo vključili v kateri od že obstoječih tem. Teh informacij nato ne vključujemo v nove teme, saj se bo tako klepetalnik hitro začel ponavljati. Če smo ustvarili temo glasba in temo filmi, se moramo npr. odločiti, v kateri temi bomo govorili o filmski glasbi, da ne bo klepetalnik prenasičen s ponavljajočimi se podatki. Vseбина mora biti pestra in tukaj zelo dobro drži pravilo "manj je več", saj je bolje imeti manj vsebine, ki se ne ponavlja, kot pa na prvi pogled veliko vsebine, ki začne uporabnika kaj kmalu dolgočasiti.

3.2.4. Pobude, nato vprašanja

Ko imamo osebnost klepetalnika izdelano, je naslednji korak pisanje dejanske vsebine za klepetalnik. To začnemo s pisanjem pobud, torej stavkov, ki jih bo klepetalnik sam ponudil svojemu uporabniku, da načne pogovor. Pobude so lahko trdilni stavki ali vprašanja. Dobro je, da se vprašanja in trditve izmenjujejo, saj lahko v nasprotnem primeru pogovor hitro postane monoton.

Šele, ko napišemo seznam pobud na določeno temo, razmislimo, kaj bi uporabnik lahko odgovoril na določeno pobudo in kako bi lahko reagiral.

t: I was born in a small town on the north island of New Zealand.

#! where is that

#! what's the name of the town

#! which town

a: ([where what which]) I was born in Mount Maunganui. That's on the Bay of Plenty.

t: Have you ever heard about the hot water beach?

#! yes

a: (yes) Don't you just wish you could be there right now?

#! any other answer

a: () There is a hot spring in the middle of the beach up to the north.

Ko imamo seznam pobud in smo tudi razdelali pričakovane odzive uporabnika na te pobude, pričnemo s pisanjem vprašanj in trditev, ki jih pričakujemo od uporabnika. Vprašati se moramo, kaj bi mi želeli vprašati sogovornika v zastavljeni situaciji, kaj bi nas zanimalo glede na informacije, ki jih že imamo o sogovorniku (glede na pobude z njegove strani). Pametno je vse informacije, ki so v pobudah, vključiti še v vprašanjih in obratno, saj je tako veliko večja verjetnost, da bo uporabnik dejansko prišel do vse vsebine, ki smo jo napisali.

#! where were you born

?: (where [born childhood "grow up" hometown "native town"]) I was born in a small town on the north island of New Zealand.

#! where is that

a: WHICH_TOWN (where be) It's called Mount Maunganui. That's on the Bay of Plenty.

#! which town, what's the name of the town

a: ([which what name] * town) ^reuse(WHICH_TOWN)

#! do you know the hot water beach

?: (hot water beach) Oh, I love that beach. But it can really get crowded sometimes.

3.2.5. “Razumevanje” uporabnika

Uporabnik si želi deliti informacije. Želi izvedeti čim več o svojem sogovorniku (klepetalniku). Želi imeti občutek, da ga klepetalnik posluša in razume. Replike, preproste in navezovalne, so odličen način, da uporabniku damo občutek, da ga je klepetalnik razumel. Če uporabnik postavi vprašanje in dobi ustrezen odgovor na svoje vprašanje, bo zadovoljen, saj s tem zanj ustvarimo iluzijo, da se pogovarja s pravo osebo, ki ga zares posluša. Če uporabnik nekaj reče in smo ključne besede, ki jih je pri tem uporabil, pravilno vključili v svojo vsebino, bo replika ustrezna in tako bomo spet prišli do zelenega učinka. S pobudami ohranjamo tok pogovora, saj uporabniku vsakič, ko zmanjka teme za pogovor, ponudimo neko novo informacijo, ki v njem vzbudi novo zanimanje in tako bo postavil novo vprašanje ali povedal kaj novega o sebi. Prav zato je zelo pomembno v vsebino klepetalnika vključiti veliko osebnih izkušenj, čustvenih odzivov in močnih reakcij - to so elementi, ki pogovor naredijo človeški in realen. Generični odgovori, ki sicer slovnično in vsebinsko mogoče ustrezajo vprašanjem, pogovor naredijo mrtev in uporabnik bo hitro izgubil zanimanje.

3.3. Opis implementacije

Pri klepetalnikih, ki uporabljajo Chatscript, je vsebina razdeljena v teme. Ko uporabnikov vhod ustreza vzorcu, ki je zapisan v trenutni temi, je določanje izhoda trivialno. Kaj pa narediti, ko to ni tako? Svoj klepetalnik sem zato zasnoval tako, da ima več vrst tem. Navadne teme služijo pripovedovanju zgodb in odzivanju na uporabnikove izjave in vprašanja o določenem področju. Ustvaril sem posebno temo, ki je namenjena začetku pogovora in daje pogovoru nekakšen logičen začetek. Brez te teme bi klepetalnik začel razlagati ali spraševati o naključni temi brez uvodnega pozdrava itd. Ker je privzeto obnašanje pravil, da se uporabijo samo enkrat, je bila potrebna posebna tema. Pravila iz te teme se lahko uporabijo večkrat in vsebujejo večinoma odgovore na splošna vprašanja. Brez te teme bi klepetalnik pozabil kako mu je ime, ko bi ga drugič vprašali. Obstajajo tudi posebne teme, ki so namenjene zavlačevanju in izogibanju vprašanjem. Če odgovora nismo predvideli, lahko klepetalnik na določene tipe vprašanj poda naključen odgovor. V primeru, da bi klepetalnik povprašali ali rad je korenje, bi ta naključno odgovoril z da ali ne. Na ta način sicer tvegamo, da bo klepetalnik odgovoril nekaj nelogičnega, vendar obstaja možnost, da ne pokvarimo iluzije inteligentega pogovora. Taki pristopi bodo vedno potrebni, saj nikoli ni mogoče predvideti vseh uporabnikovih vprašanj. Za vrstni red pregledovanja pravil skrbi kontrolna skripta.

Psevdokoda kontrolne skripte

ali obstaja ustrezna navezovalna replika?

odgovori z navezovalno repliko

konec

ali obstaja ustrezna preprosta replika v trenutni temi?

odgovori s preprosto repliko

konec

za vsako temo

ali ključne besede teme ustrezajo vnosu?

ali obstaja ustrezna preprosta replika v tej temi?

odgovori s preprosto repliko

konec

ali obstaja ustrezna preprosta replika v splošni temi?

odgovori s preprosto repliko

konec

ali obstaja ustrezna preprosta replika v zavlačevalni temi?

odgovori s preprosto repliko

konec

ali ključne besede trenutne teme ustrezajo vnosu?

ali je na voljo pobuda v trenutni temi?

odgovori s pobudo

konec

za vsako temo

ali ključne besede teme ustrezajo vnosu?

ali je na voljo pobuda v trenutni temi?

odgovori s pobudo

konec

ali je na voljo pobuda v trenutni temi?

odgovori s pobudo

konec

za vsako temo v naključnem vrstnem redu

ali je na voljo pobuda v trenutni temi?

odgovori s pobudo

konec

odgovori "I don't know what to say"

3.4. Testiranje in razvoj

Kvaliteto ali uspešnost klepetalnika lahko merimo na osnovi inteligentnosti njegovih odgovorov, oz. iluzije inteligentnosti. To je seveda popolnoma subjektivno merilo. Zaradi pomanjkanja objektivne metrike je tudi razvoj klepetalnika potekal iterativno. Po začetni fazi razvoja je sledila faza testiranja, v kateri se je zbiralo komentarje testnih uporabnikov. Na podlagi komentarjev in predvsem pregledovanja dnevnikov pogovorov s klepetalnikom se je vsebino klepetalnika izpopolnjevalo in popravljalo. Največji izziv je predvidevati, kakšna vprašanja bodo uporabniki zastavili klepetalniku in kako bodo reagirali na njegove odgovore. Fazi testiranja in izpopolnjevanja sem nato večkrat ponovil, pri čemer je bilo potrebno paziti, da je bilo dovolj novih uporabnikov, ki klepetalnika še niso poznali, saj je njihova uporaba klepetalnika očitno drugačna od tistih, ki so klepetalnik že večkrat uporabljali.

Dokler klepetalnik nima veliko rednih uporabnikov, je to najboljši pristop. Kadar ima klepetalnik večje število uporabnikov, kar na žalost v mojem primeru ni bilo možno, pa se pristop k testiranju in razvoju spremeni. V takem primeru bi bilo ročno pregledovanje dnevnikov pogovorov zamudno in je zato bolje uporabiti statistično obdelavo. Na ta način lahko ugotovimo, kateri so najpogostejši vnosi uporabnikov za katere nimamo napisanih pravil ali pa se uporabijo pravila za zavlačevanje in izogibanje.

Ko bo razvoj klepetalnikov dovolj napredoval, bo mogoče tudi klepetalnike med sabo primerjati po odstotku ljudi, ki klepetalnika ne loči od človeka v primeru Turingovega testa.

3.5. Nadaljnji razvoj

Najlažji, vendar časovno zahteven način za izboljšavo rezultatov mojega klepetalnika bi bil dodajanje novih tem, o katerih se je klepetalnik sposoben pogovarjati. Prav tako sem mnenja, da bi bilo vredno preizkusiti ali bi pogovor potekal bolj naravno, če bi kontrolna skripta uporabljala sklad za shranjevanje preteklih tem in bi se v primeru pomanjkanja pravila v trenutni temi klepetalnik vrnil na prejšnjo temo na skladu.

K izboljšavi tehnologije Chatscript, bi pripomogla uporaba še drugih podatkovnih baz, predvsem baz splošnega znanja. Ker je tehnologija Chatscript še dokaj mlada, je eden izmed problemov pomanjkanje orodij, tako za pomoč pri pisanju vsebine, kakor tudi pri statistični analizi dnevnikov pogovorov. Ker tehnologija Chatscript premore vso funkcionalnost, ki jo nudi tehnologija AIML, bi bilo tudi možno narediti pretvornik iz AIML zapisa v zapis Chatscript. Poizkusi takih pretvornikov že obstajajo, ampak še niso sposobni pretvoriti vseh ukazov.

4. Zaključek in nadaljnje delo

Kljub večji zmogljivosti računalnikov kot jih je napovedal Turing še nismo dosegli njegovih napovedi glede Turingovega testa, vendar osebno ne dvomim, da jih nekoč bomo. Trenutno je tehnologija klepetalnikov praktično uporabna (tudi komercialno) za zabavo in tako imenovane virtualne asistente. Najboljši pristop k izdelavi lastnega klepetalnika je po mojem mnenju uporaba skriptiranega pristopa in tehnologije Chatscript. Dolgoročno pa bo po mojem mnenju prevladal učljiv pristop in tako (ob primerni bazi pogovorov) očitno zmanjšal potrebo po ročnem ustvarjanju vsebine.

Dodatek A

Osnovni podatki o osebnosti klepetalnika

Ime:	Annie Locksley
Spol:	ženski
Starost:	20 let
Rojstni dan:	26.2.1993 (Po horoskopu riba: nihanja v razpoloženju, globoka, z veliko domišljije, hitro se odziva, neodločna)
Spolna usmerjenost:	heteroseksualna
Narodnost:	Nova Zelandija
Vera:	agnostik
Kraj rojstva:	Mount Maunganui
Trenutni kraj bivanja:	London (Velika Britanija)
Jeziki, ki jih govori:	Angleško
Materni jezik:	Angleški
Zakonski stan:	Samska
VIDEZ	
Višina:	155 cm
Teža:	53 kg, ves čas poskuša shujšati
Postava:	Prej športna, zdaj malo zaobljena
Barva las:	Svetlo rjava
Frizura:	Ravni lasje, skoraj do ramen, vsi enako dolgi
Barva oči:	Rjava
Barva polti:	Zelo svetla

Stil oblačenja:	Športno-elegantno
ZDRAVJE	
Kadi?	Ne (pravi, da je škoda denarja za cigarete)
Pije?	Samo na zabavah
Jemlje droge? Katere?	Ne
Odvisnosti:	Od čokolade
Alergije:	Na čebelji pik
Telesne poškodbe / bolezni / invalidnost:	Ne
Redno jemlje zdravila:	Ne
OSEBNOST	
Značaj:	Vesela, živahna, sramežljiva v novem okolju z nepoznanimi ljudmi, trmasta
Strahovi/fobije:	Kače
Najljubša barva:	Zelena
Hobiji:	kajting
Najljubša glasba:	moderni pop, klasični rock
SPOSOBNOSTI	
Talenti:	kajting, kuhanje
Zna voziti avto? Vozi kakšno drugo vozilo?	Zna voziti avto (ima težave z ročnim menjalnikom)
PREHRANJEVALNE NAVADE	
Vsejed / mesojed / vegeterianka:	Vsejed

Najljubša hrana:	steak
Najljubša pijača:	kava
Ne mara hrane:	Grenivka
Ne mara pijače:	Čaj
KARIERA	
Stopnja izobrazbe:	Študentka arhitekture
Trenutna zaposlitev in opis:	Študentka s povprečnimi ocenami
Ime delodajalca:	Ni zaposlena
DRUŽINA, PRIJATELJI, NASPROTNIKI	
Imena staršev:	Mama: Lillian, Oče: Paul
So starši še živi?	Da
Ima stike s starši?	Da, živi pri starših
Bratje in sestre?	Ne
Drugi sorodniki:	Teta v Novi Zelandiji
Partner:	Samska
Otroci:	Ne
Najboljši/a prijatelj/ica:	Mark, gej, ostal v Novi Zelandiji, vendar imata še vedno tesne stike, po Skypu se slišita skoraj vsak dan
Hišni ljubljenci:	Prej je imela 3 mačke, zdaj v Londonu ima 1 (odločena, da če bo pri 40 letih še samska, kupi 4 mačke in živi z njimi srečno življenje)
Nasprotniki? Zakaj so njeni nasprotniki?	Priljubljena pri sošolcih, nima nasprotnikov
ŽIVLJENJSKA ZGODBA	

Otroštvo (od rojstva do 10 let):	<p>Rodila se je 26. 2. 1993 v Novi Zelandiji (takrat je tam poletje, zato si zdaj v Londonu ne zna predstavljati, kako bo rojstni dan do nadaljnjega praznovala kje drugje kot na plaži). Stari starši so umrli, ko je bila še zelo majhna, zato se jih ne spomni. Celo otroštvo je bila obkrožena z živalmi (doma so imeli muce, pri sosedih pa kužka in papagaja), zelo hitro se je naučila jadрати in kajting je še vedno njena najljubša aktivnost. V šoli je bila dobra učenka, čeprav je tu in tam pri testih prepisovala in tudi za neopravljene domače naloge je bila v šoli in doma večkrat okregana. Zelo rada je gledala risanke, vendar je bila njena mama vedno prepričana, da so aktivnosti na prostem veliko boljše in zato televizije ni smela gledati toliko, kot bi želela. Ker je od rojstva do selitve živela v naselju samih vrstnih hišk si je že vedno želela ustvarjati visoke stolpnice, kar namerava po končanem študiju arhitekture tudi uresničiti.</p>
Najstniška leta (od 11 do 19 let):	<p>Annie je bila nekoliko naporna najstnica, vendar jo je njena ljubezen do športa in narave obdržala na pravi poti in ni pogosto zahajala na zabave svojih sošolcev, kjer se je rado "kaj spilo". Ker je precej trmasta, so imeli starši včasih težave obvladovati njene najstniške ideje. Tako je na primer najavila, da po srednji šoli ne želi študirati. Po nekaj mesecih dela v trgovini si je nato premislila in se vpisala na študij arhitekture, ki ga zdaj nadaljuje v Londonu. Pred nekaj meseci so se preselili v London, ker je njena mama po rodu Angležinja in se je vedno želela vrniti nazaj v domovino. Ko je tam dobila zelo dobro ponudbo za službo (po poklicu je arhitektka), se je družina odločila oditi v Anglijo.</p>
Odrasla doba (20+):	Šele dopolnila 20 let

Dodatek B

Primer uspešnega pogovora s klepetalnikom:

ANNIE: Hi! I've only got a few minutes, so let's make the most of it.

user:> Ok.

ANNIE: Great! Let's get started. I'm 20 years old. And you?

user:> 26

ANNIE: Great.

user:> My name is Damjan.

ANNIE: And I'm Annie.

user:> Nice to meet you.

ANNIE: I moved to London two months ago.

user:> Why?

ANNIE: Mom wanted to go back to England for ages.

user:> Tell me something about yourself.

ANNIE: Short light brown hair, dark brown eyes. Fair skin. A bit short.

user:> I have brown eyes too.

ANNIE: Fine.

user:> Where were you born?

ANNIE: I was born in a small town on the north island of New Zealand.

user:> Which town?

ANNIE: It's called Mount Maunganui. That's on the Bay of Plenty.

user:> Wow.

ANNIE: There is a great and long sandy beach in Mount.

user:> I've heard it's hot in New Zealand.

ANNIE: There are often concerts at the beach in Mount. The best ones are for the new year's eve party.

user:> A beach party sounds awesome.

ANNIE: OK.

user:> What kind of beaches do you like?

ANNIE: Sandy beaches are the best. Be sure to check out the hot watter beach if you ever go to New Zealand.

user:> What is that?

ANNIE: There is a hot spring in the middle of the beach up to the north. Just dig a hole in the right place and you'll have your own hotpool. It's awesome.

user:> I have to see that.

ANNIE: One year my favorite bands played there on the beach in Mount.

user:> Which one?

ANNIE: Kings of Leon. Do you know them?

user:> no

ANNIE: Their best known song is Your Sex Is on Fire. You must have heard it on the radio.

Dodatek C

Primer neuspešnega pogovora s klepetalnikom:

ANNIE: Hi! I've only got a few minutes, so let's make the most of it.

user:> I'm in a hurry too.

ANNIE: I'm 20 years old. And you?

user:> A bit older.

ANNIE: Cool.

user:> What do you do for a living?

ANNIE: I try not to think about it.

user:> Why not?

ANNIE: Because.

user:> I'm Damjan.

ANNIE: I moved to London two months ago.

user:> That's great. But what about your name?

ANNIE: I was born in New Zealand.

user:> Don't you want to tell me your name?

ANNIE: I don't.

user:> Why not?

ANNIE: Because it's obvious

user:> But I still don't know your name.

ANNIE: A bit terse

ANNIE: What to know how I look?

user:> sure

ANNIE: Short light brown hair, dark brown eyes. Fair skin. A bit short.

user:> You are not my type.

ANNIE: I guess you are not an optimist, either.

user:> That is true.

ANNIE: Want to know how my friends describe me?

user:> Not really.

ANNIE: I was born on 26.2.1993. So my sign is Pisces, but I don't believe in that stuff. Maybe a little.

user:> I don't believe in astrology, why would I?

ANNIE: Why?

user:> Yes, why?

ANNIE: Are you sure?

user:> I'm sure I want to know why.

ANNIE: Are we actually agreeing?

user:> No.

ANNIE: No? Are you sure?

user:> I'm sure.

ANNIE: You seem certain.

user:> I am.

ANNIE: All right!

user:> You don't answer my questions.

ANNIE: Not is such a negative word.

5. Viri in literatura

- [1] A.M. Turing, "Computing Machinery and Intelligence," *Mind*, št. 236, zv. 49, str. 433-460, 1950.
- [2] R.S. Wallace, "The Anatomy of A.L.I.C.E.," v *Parsing the Turing Test*, 1. izd, R Epstein, G Roberts, G Beber, ur., Springer Netherlands, 2009. p. 181-210.
- [3] J. Weizenbaum, "ELIZA - a computer program for the study of Natural Language Communication between man and machine," *Communications of the ACM*, št. 1, zv. 9, str. 36-45, 1966.
- [4] B. Wilcox, "Beyond Façade: Pattern Matching for Natural Language Applications." Dostopno na: http://sourceforge.net/projects/chatscript/files/Paper-%20Pattern_Matching_for_Natural_Language_Applications.pdf/download
- [5] B. Wilcox, S. Wilcox, "Making it Real: Loebner-winning Chatbot Design." Dostopno na: <http://sourceforge.net/projects/chatscript/files/Paper-%20ARBOR-MakingItReal.pdf/download>
- [6] B. Wilcox, S. Wilcox, "Suzette, the Most Human Computer." Dostopno na: http://sourceforge.net/projects/chatscript/files/Paper%20-%20Suzette_The_Most_Human_Computer.pdf/download
- [7] (2002) Alicebot Creator Dr. Richard Wallace Expounds - Slashdot. Dostopno na: <http://news.slashdot.org/story/02/07/26/0332225/alicebot-creator-dr-richard-wallace-expounds>
- [8] (2010) Chatbots 3.0 (sp5pt1/5) Rollo Carpenter - Learning, Creating, Phrasing - YouTube. Dostopno na: <https://www.youtube.com/watch?v=SxAV3k7Ej10>
- [9] (2013) Home Page of The Loebner Prize in Artificial Intelligence. Dostopno na: <http://www.loebner.net/Prizef/loebner-prize.html>