

Seminární práce z předmětu Umělá inteligence I

Analýza změn ICQ statusů

Cíl práce

Cílem mé práce mělo být původně srovnání úspěšnosti predikce ICQ statusu (nikoli statutu [link 1]) založené na běžné statistické analýze a metody využívající neuronovou síť.

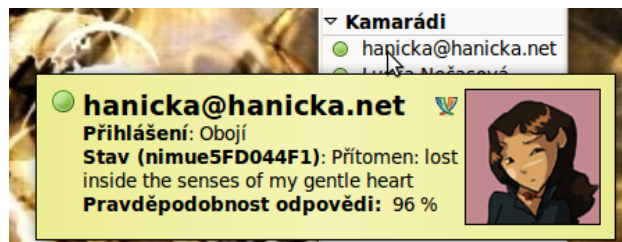
Vedlejším cílem pak bylo získat praktické zkušenosti s nějakou knihovnou pro neuronové sítě a použít takové prostředky, aby byla moje práce později prakticky využitelná.

Motivace a použité technologie

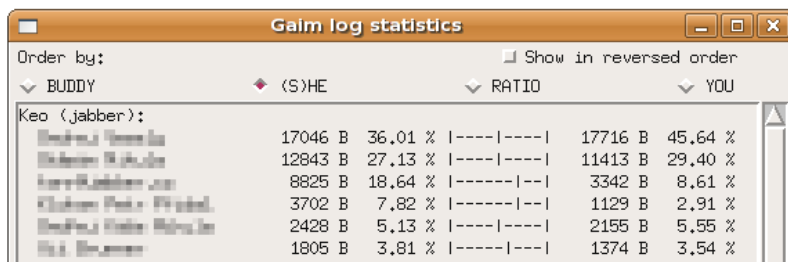
Pro použití chronologického záznamu změn ICQ statusů jsem se rozhodl proto, že popisuje lidské chování. To je obecně považováno za nedeterministické, ale jistá (individuální a nepřímá) závislost na čase v podobě ICQ statutu jistě existuje. Hledání této závislosti je pak vhodným úkolem pro neuronovou síť.

Vzhledem k tomu, že zatím neexistuje vhodné serverové řešení, které by záznam změn statusů umožňovalo (např. knihovna pro PHP/Perl), použil jsem běžný instant messenger (IM) Pidgin. Ten umožňuje aktivace tzv. systémového záznamu, který poměrně konzistentně zachycuje čas a změnu stavu každého přihlášeného uživatele v kontaktní listu. Důvodů pro použití programu Pidgin bylo víc:

- Už existují programy pro analýzu komunikace (ovšem ne změn statutů), např. *Gaim log statistics* [link 2], nebo plugin *Předpovídání dostupnosti kontaktu* [link 3] (ten však řeší jen pravděpodobnost odpovědi online uživatele – tedy jeho ochoty reagovat).
- Vedlejším cílem byla potenciální praktická použitelnost – zde se nabízí prostor vyvinout plugin. Ten by na požádání informoval uživatele s jakou pravděpodobností se testovaný kontakt během zadaného časového úseku pravděpodobně připojí.
- Podpora dalších patnácti protokolů a dostupnost pod OS GNU/Linux.



Ukázka funkce pluginu *Předpovídání dostupnosti kontaktu*



BUDDY	(S)HE	RATIO	YOU
Keo (jabber):			
17046 B	36,01 %	---- ----	17716 B 45,64 %
12843 B	27,13 %	---- ----	11413 B 29,40 %
8825 B	18,64 %	----- ---	3342 B 8,61 %
3702 B	7,82 %	----- ---	1129 B 2,91 %
2428 B	5,13 %	---- ----	2155 B 5,55 %
1805 B	3,81 %	----- ---	1374 B 3,54 %

Ukázka z programu *Gaim log statistics*

Volba Pidginu vedla pak vedla k rozhodnutí použít jazyk C, protože je velmi dobře dokumentovaným jazykem pro vývoj pluginů tohoto IM.

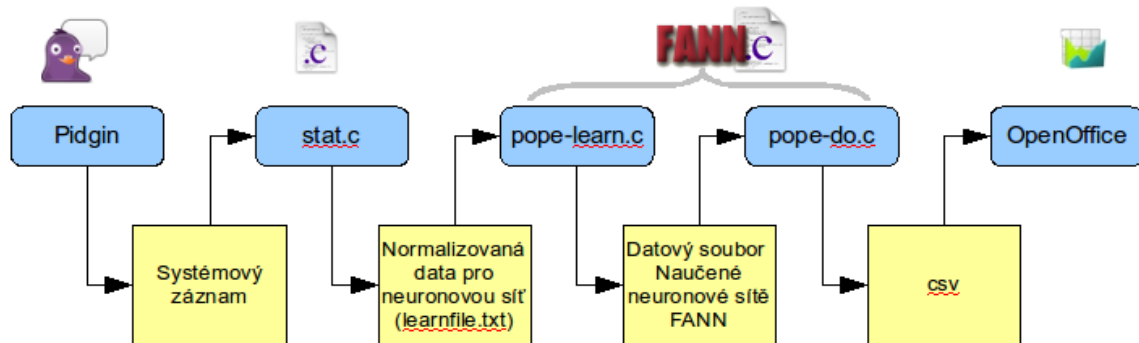
Knihovnu Fast Artificial Neural Network Library (FANN) [link 6] jsem si vybral především proto, že mě oslovily práce studentů z VUT v Brně [link 4,5], má přehledně

zpracovaný tutoriál a podle dokumentace i široké možnosti ladění. Později jsem však tuto důležitou technologii vyměnil za Neural Network toolbox Matlabu.

Pro grafovou prezentaci dat jsem využil tabulkový procesor Calc.

Vlastní práce

Chtěl jsem mít možnost rozdělit celkové zpracování dat do více fází, aby bylo možné zpracovávat přibývající údaje v záznamu postupně. Jednotlivé fáze jsou znázorněny v následujícím schématu.



Datový tok v původním návrhu systému na analýzu změn statutů

Zpracování systémového záznamu Pidginu

Vzhledem k tomu, že záznam změn statusů provádí Pidgin pokaždé, dojde-li ke změně, jsou záznamy v čase značně nerovnoměrné. Takto vypadá úsek z běžného systémového záznamu:

```
---- kiki (252507794) změnil stav z Přítomen na Přítomen @ 23.2.2009 15:23:59 ----
---- smoke.it (287581308) změnil stav z Přítomen na Přítomen @ 23.2.2009 15:23:59 ----
---- Andrew se stal nečinným @ 23.2.2009 15:24:09 ----
---- Tomkava (271761434) změnil stav z Pryč na Odpojen @ 23.2.2009 15:24:10 ----
---- Andrew (285864213) změnil stav z Přítomen na Přítomen @ 23.2.2009 15:24:13 ----
---- Andrew se stal činným @ 23.2.2009 15:24:24 ----
---- Andrew (285864213) změnil stav z Přítomen na Přítomen @ 23.2.2009 15:24:27 ----
---- ersin (262836180) změnil stav z Přítomen na Přítomen @ 23.2.2009 15:24:47 ----
---- +++ 240590309 se odhlásil @ 23.2.2009 15:25:51 ----
```

Záznamy jsou navíc v mnoha souborech, které jsou unikátně pojmenovány v závislosti na čase, např.: 2009-02-23.143359+0100CET.txt

Bylo tedy nutné napsat pomocný program, který by změnil formát dat tak, aby byly jednoduše zpracovatelné knihovnou FANN. Ten měl být o poznání jednodušší:

```
3219 1 5
1
3118 1 5
-1
1129 1 6
1
3555 1 6
```

Liché řádky jsou vstupy do neuronové sítě a následující (sudé) požadovaný výstup. V ukázce je první číslo lichého řádku hash příslušného UIN, následuje den v týdnu a hodina. Na sudých řádcích je status uživatele v danou hodinu: -1 pro odpojeného a 1 pro připojeného.

Ačkoliv je na tento úkol výborně hodí např. Perl, kvůli možné integraci knihovny FANN a kompatibility s Pidginem (pro potřeby případného pluginu) jsem se držel jazyka C.

Program *stat.c* (viz příloha 1) umožňuje volit časovou granularitu výstupů (v ukázce vypisuje

stav každého kontaktu každou hodinu) a přikládat jednotlivým statusům různé váhy (viz funkce *statusvalue* v kódu). Po jednoduché úpravě může program vypisovat jakékoli časové údaje (měsíc, rok, pořadí týdne v roce, minuta, ...).

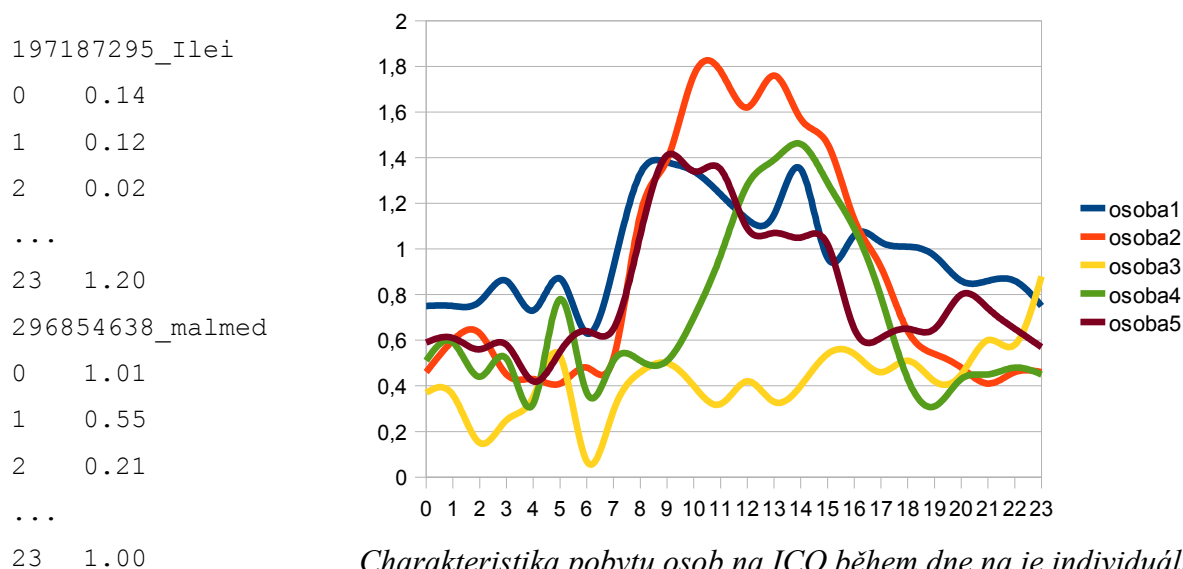
Pokus o použití FANN

Ačkoliv měla knihovna FANN připravená normalizovaná a zjednodušená data přesně podle pokynů v tutoriálu, její funkce žalostně selhaly. Žádná dostupná verze knihovny se nebyla schopna vypořádat s daty jinými, než popisuje tutoriál na webu.

Popis chyby je popsán v příloze 2 a od vývojářů dosud nevzešlo žádné vysvětlení, které by mi pomohlo problém vyřešit. Panovala však shoda, že problém není v logice knihovny, programu a že na daných datech by měla knihovna fungovat.

Běžná statistická analýza

Úpravou programu *stat.c* a použitím dalšího pomocného programu jsem dostal údaje do formátu, který agreguje údaje pro každé UIN a každý den v týdnu na prostý seznam míry výskytu příslušného člověka v příslušnou hodinu na ICQ.



Tento formát umožňuje snadnou tvorbu grafů v tabulkovém procesoru a srovnání míry výskytu jednotlivých lidí vůči sobě jak je vidět z grafu výše. Na ose x jsou hodiny, na ose y míra dostupnosti kontaktu (hodnota 2 značí 100% dostupnost, hodnota 0 žádnou dostupnost).

Použití Neural Network toolbox Matlabu

Toolbox *nntool* byl vybrán jako nouzové řešení, když selhala FANN. Jeho nevýhodou je pochopitelně mnohem menší flexibilita a horší možnosti automatizace práce s daty. Mezi jeho přednosti pak pěkná vizualizace názorné možnosti měnit parametry neuronové sítě.

Nntool používá standardní formát dat Matlabu. Pokěněd neobvyklé je zvolení jako vstupní řady pro neuronovou síť sloupce v matici. Následují ukázky trénovacích dat.

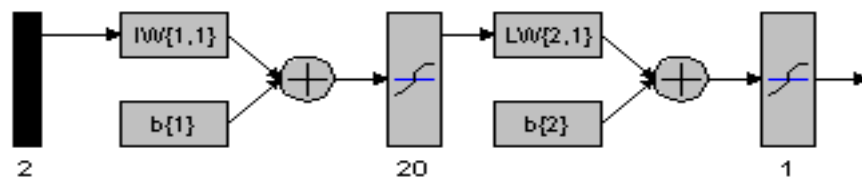
Část vstupní matice zachycující jedno páteční odpoledne (číslo 4 v první řádce značí pátek, spodní řádek jednotlivé hodiny):

```
[ ... 4 4 4 4 ... ;
  ... 13 14 15 16 ... ]
```

Část výstupní matice popisující průběh statusu v danou dobu (*targets*).

[... 0 1 1 0 ...]

Při návrhu sítě jsem nemohl opírat o žádné praktické zkušenosti. Zvolil jsem klasickou síť s dopředným šířením signálu a se zpětným šířením chyby (feed-forward backprop) a implicitní nastavení training function, adaptation learning function a performance function.



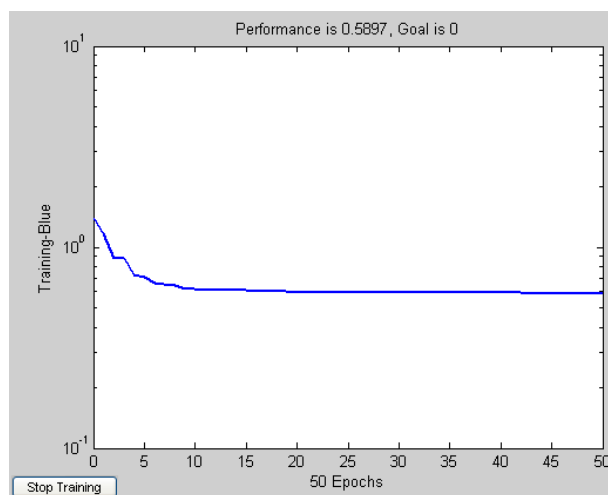
Architektura použité neuronové sítě

Problém je relativně jednoduchý – proto stačí třívrstvá síť (počítám vstupní i výstupní vrstvu) ale s dvaceti neurony ve skryté vrstvě – charakteristika křivky je vzhledem k počtu hodin náročnější na „paměť sítě“. Nechtěl jsem však překročit rozlišený počet hodin, aby si síť zachovala nějakou míru generalizace a neučila se data „natvrdo“.

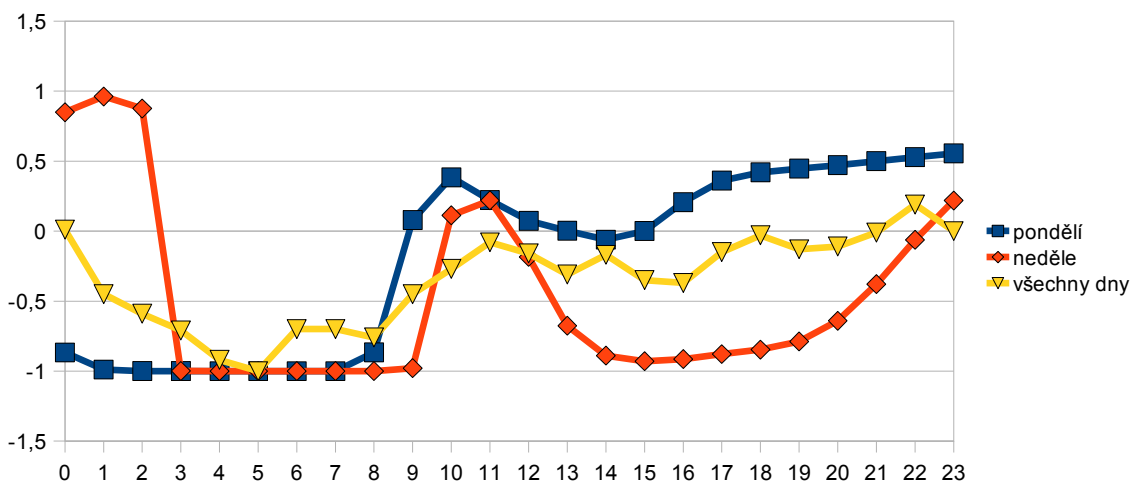
Při analýze neuronovou sítí jsem se zaměřil na jediné UIN s cílem znázornit výskyt uživatele všechna pondělí a neděle v průběhu dne. Do grafu jsem také přidal charakteristiku získanou běžnou statistickou analýzou pro všechny dny v týdnu.

Ačkoliv by bylo zajímavé srovnat metody a hlavně jejich úspěšnost, v čase určeném této práci pro takové porovnání již není prostor. I tak z grafu plyne, že obě metody jsou použitelné a získané charakteristiky mají své snadno vysvětlitelné opodstatnění.

Graf mj. ukazuje, že uživatel zůstává v neděli dlouho do noci (cca do druhé hodiny) vzhůru a nedělní odpoledne tráví ve srovnání s jinými dny převážně offline. Typická je špička kolem oběda, která je pro všechny dny společná.



Klesající chyba v průběhu učení (50 epoch)



Hodinové charakteristiky vybraného kontaktu v různých dnech (míra dostupnosti v rozmezí -1 až 1)

Potenciální možnosti analýzy pomocí neuronové sítě

Pokud by mi to rozsah práce umožňoval zaměřil bych se ve srovnání obou metod na analýzu chybovosti predikce. Jsem přesvědčen, že by neuronová síť byla při dostatečně rozsáhlém vzorku dat úspěšnější, protože:

- dokáže dynamiku vývoje statusu (např. pokud byl uživatel online dlouho do noci, pravděpodobně nebude online brzo ráno).
- je schopná reflektovat nepředpokládané pravidelnosti (předmět informatika na střední škole jednou za 14 dní)
- lépe se vyrovná s nedostatkem dat a v teoreticky může přihlídnout k jinému uživateli s podobnou charakteristikou (viz dále)

Samotnou kapitolou by pak bylo rozdělení uživatelů (clustering) podle jejich podobných charakteristik, které se v dynamice změn statusů projevují (např.: spolužáci jsou společně online ve škole, kamarádi jsou společně offline na výletě). Zvýšenou generalizací by pak třeba bylo možné rozlišit např. Vysokoškoláky od středoškoláků, pracující od studujících apod.

Další rozměr by systému přinesla evidence záznamů proběhlých konverzací případně agregace s daty jiných zdrojů (Skype, e-mail, Twitter, sociální sítě, SMS, ...)

Závěr a praktické zkušenosti

Cíl práce mi negativně ovlivnil problém knihovny FANN v jehož světle se pak neprojevila užitečnost a síla jazyka C. Řešení problému s knihovnou v kombinaci s použitím nízkourovňového jazyka na zpracování textu vedlo v velké časové ztrátě a tudíž nezbyl prostor na podrobnější analýzu výsledků.

Práce s rozsáhlým (1,6MB) vzorkem dat se ukázala být v OpenOffice poměrně zdlouhavá – narazil jsem na omezení 65 635 řádků v tabulce a tvorba tak velkých kontingenčních tabulek není efektivní. Při další práci bych data ukládal do nějaké relační databáze (např. Firebird) odkud se dají data čerpat jednoduchými selecty.

Pokud bych chtěl provádět nad daty nějakou predikci, byl dle mého názoru vzorek pěti týdnů dat moc malý.

Literatura

1. Jazykový rozbor slova statut
<http://www.proofreading.cz/status-nebo-statut>
2. program Gaim log statistics
<http://www.keo.cz/?p=23>
3. Pluginy Pidginu
<http://developer.pidgin.im/wiki/ThirdPartyPlugins>
4. bakalářská práce Jana Beránka (detekce hran v obraze)
www.fit.vutbr.cz/study/DP/rpfile.php?id=3045
5. bakalářská práce Jakuba Žáka (rozpoznání obličejů)
www.fit.vutbr.cz/study/DP/rpfile.php?id=5203
6. tutoriál FANN
<http://leenissen.dk/fann/html/files2/gettingstarted-txt.html>
7. CONVEY Peter, High Roger: Mezi chaosem a řádem