Úvod do počítačového zpracování řeči

Luděk Bártek[1]

1

Obsah

Úvod

- Hlavní rozdíly oproti rozpoznávání slov:
  - nelze vytvořit analogii databáze vzorů
  - prozodické faktory
  - nutnost určovat hranice mezi slovy
  - výplňkové zvuky a chyby řeči
- Řešení - statistický přístup
  - použití jazykových modelů
  - HMM vrátí stejnou pravděpodobnost např. pro slova "máma" a "nána"
    1. máma je častější - vhodné použít máma

Jazykové modely

- Posloupnost slov (promluva) W = (w(1)w(2)...w(n)).
- Posloupnost akustických vektorů - O = O(o(1)o(2)...o(t)).
- Chceme nalézt W* (množinu všech promluv) maximalizující P(W—O).
- Dle Bayesova pravidla platí:P(W*—O) = max P(W—O) = max P(W)*P(O—W)/P(O)
- Pro nalezení maxima potřebujeme znát:
  - model řečníka P(O—W)
  - jazykový model P(W)
- Model řečníka se nahrazuje pravděpodobností generování W odpovídajícím Markovovým modelem.
- Trigramový model:
  - Platí: $P(w(n)|w(1)..w(n-1)) \cong P(w(n)|w(n-2)w(n-1))$

Rozpoznávání tématu - topic recognition

- Úspěšnost rozpoznávání plynulé řeči 50-99
  - úkolu
  - jazyku
  - mluvčím
  - ...
- Úspěšnost rozpoznávání může zvýšit:
  - znalost tématu promluvy
  - použití gramatiky pro rozpoznávání řeči.
- Mění se stavový prostor a pravděpodobnosti trigramů
  - např. mějme burzovní zprávy - bylo rozpoznáno slovo honey nebo money?
- Známé téma - může být přesnější jazykový model.

Gramatiky pro podporu rozpoznávání řeči

- Umožňují omezit množinu rozpoznávaných promluv:
  - výhoda - vyšší úspěšnost rozpoznávání
  - nevýhoda - nižší volnost vyjadřování
- Používají se bezkontextové gramatiky.
- V praxi často používané formáty gramatik:
  - JSGF (`http://www.w3.org/TR/jsgf/`) - původně definována v Java Speech API (`http://java.sun.com/products/java-media/speech/`)
  - SRGS (`http://www.w3.org/TR/speech-grammar/`) - součást standardů W3C Voice Browser Activity (`http://www.w3.org/Voice`)
    - Určeny pro tvorbu dialogových a hlasových rozhraní.

Ukázka gramatiky ve formátu JSGF

```
#JSGF
 <koren> = Chci jet <cim>.|
           Chci jet <cim> z <odkud> do <kam>.|
           Chci jet <cim> z <odkud> do <kam> v <kdy>.;
 <cim> = vlakem | autobusem;
<odkud> = <czMesto>;
<kam> = <czMesto>;
<kdy> = <czCas>;
```

Ukázka odpovídající gramatiky v XML formátu SRGS

```
<grammar root="koren" version="1.0" xml:lang="cs-CZ">
 <rule id="koren">
  <one-of>
   <item>Chci jet <ruleref uri="\#cim"/>.</item>
   <item>Chci jet <ruleref uri="\#cim"/>
    z <ruleref uri="url db názvů stanic"/>
    do <ruleref uri="url db názvů stanic"/>
   </item>
   ...
  </one-of>
 </rule>
```

Ukázka odpovídající gramatiky v XML formátu SRGS

Pokračování

```
 <rule id="cim">
  <one-of>
   <item tag="vlak">vlakem</item>
   <item tag="autobus">autobusem</item>
    ...
  </one-of>
 </rule>
</grammar>
```

Ukázka gramatiky v ABNF formátu SRGS

```
root=$koren;
language = cs-CZ;
...
$koren = Chci jet $cim. |
         Chci jet $cim z $<url db stanic>
          do $<url db stanic>|
         ...
$cim = autobusem {$out=autobus} | vlakem {$out=vlak}
```

Úvod

- Úkol:
  - Převod psaného textu na mluvenou řeč.
  - Co nejpřirozenější řeč - ideálně k nerozeznání od člověka:
    - správná intonace
    - správné umístění přízvuků
    - správná koartikulace
    - správný rytmus
    - ...

Druhy syntézy řeči

- Druhy syntézy řeči
  - ve frekvenční oblasti
  - v časové oblasti
  - korpusová
  - problémově orientovaná syntéza:
    - hlášení nádražního rozhlasu
    - automatizované linky telefonické podpory

Fáze syntézy řeči

1. Fonetický přepis.
2. Syntéza fonetické transkripce
3. Případný postprocessing:
   - intonace
   - správné časování - modifikace délky fonémů, ...
   - větné přízvuky
   - ...

Fonetický přepis

- Slouží k přesnému, jednoznačnému zápisu mluvené řeči.
- Využívá fonetickou abecedu:
  - mezinárodní fonetická abeceda - IPA (součást standardu UNICODE)
  - 7bitový přepis IPA pomocí ASCII - SAMPAma:S se dobr'e / ma:S se dobRe
- Nelze si pamatovat fonetický přepis každé promluvy - nutno zabezpečit automatický přepis:
  - fonologická pravidla
- Při transkripci češtiny se některé české znaky nevyužívají:
  - ch - x
  - w - v
  - y/ý - i/í
  - q - kv
- Koartikulace

Pravidla fonetického přepisu češtiny

- ch → x
- ů → ú
- w → v
- q → kv
- y → i
- ý → í
- ě → je /po b,p,f,v
- dě, tě, ně, mě
  - dě → ďe
  - tě → ťe
  - ně → ňe
  - mě → mňě

Pravidla fonetického přepisu češtiny

- di, ti, ni
  - di $\longrightarrow$ ďi
  - ti $\longrightarrow$ ťi
  - ni $\longrightarrow$ ňi
- X:
  - x $\longrightarrow$ ks — začátek slova před samohláskou, mezi samohláskami nebo před neznělou souhláskou a nebo na konci slova, s výjimkou ex¡samohláska¿ $\longrightarrow$ egz
  - x $\longrightarrow$ gz — před znělou souhláskou

Změny na při spojování souhlásek

- Dochází k nim při spojování souhlásek.
- Způsobeny přenastavováním mluvidel.
- 2 druhy:
  - spodoba znělosti - změna znělosti párových souhlásek
    - ZPS → ZPS
    - NPS → NPS
    - dub → dup
    - zpěv → spjef
    - sběr → zbjer
    - když → gdiš
  - spodoba artikulační - při spojení dvou souhlásek s různou artikulací
    - banka, tango
    - tramvaj, nymfa
    - punťa, pindík
    - odpovědně, sto dní, vodní
    - ts → c, tš → č
    - ds → c, dš → č

On-line přístupné ukázky syntézy řeči

- AT&T Labs Natural Voices© Text-To-Speech (`http://www2.research.att.com/~ttsweb/tts/demo.php`)
- Free demo to create avatars using TTS by SitePal (`http://www.oddcast.com/home/demos/tts/tts_example.php`)
- Cepstral Text-to-Speech (`http://cepstral.com/demos/`)
- Festival Online Demo (`http://www.cstr.ed.ac.uk/projects/festival/onlinedemo.html`)
- Speechtech s.r.o. (`http://www.speechtech.cz/cs/produkty/demo.html#Iva210`)

Syntéza ve frekvenční oblasti

- Emulace funkce hlasového ústrojí pomocí FM syntezátoru.
- Nutno uchovávat:
  - frekvenční charakteristika použitého hlasu
  - parametry buzení.
- Využívá:
  - systém frekvenčních generátorů - simulují hlasivky
  - filtry a zesilovače - simulace rezonance v dutinách
  - Tyto komponenty ovládány parametry modelu.
- Nejběžněji použité způsoby kódování zdroje:
  - Řečová syntéza formantového typu - uchovávají se parametry průběhu jednotlivých formantů a buzení.
  - LPC řečová syntéza - uchovávají se F0, příznak znělosti, amplituda budícího signálu G a koeficienty LPC,

Syntéza ve frekvenční oblasti

- Výhody
  - menší paměťové nároky - uchovávají se pouze parametry modelu.
- Nevýhody:
  - oproti syntéze v časové oblasti může být výsledek méně přirozený - "robotické" hlasy
  - Softwarová - výpočetně relativně náročné - lze implementovat přímo na úrovni HW
    - skládání jednotlivých frekvencí, které tvoří příslušné fonémy
    - řešení koartikulace
    - ...
  - Neexistuje dostatečně přesný matematický model

Využití syntézy ve frekvenční oblasti

- Využití dříve:
  - malé paměťové nároky
  - domácí počítače (Amiga, Atari, ...)
  - syntéza realizována většinou hardwarově
- Dnes:
  - Syntéza na zařízeních s nedostatkem paměti.
  - Syntéza realizovaná hardwarově pomocí zákaznických obvodů.
- Doplnění syntézy v časové oblasti o prozodické jevy:
  - Větná intonace
  - ...
  - Realizováno programově pomocí modifikace F_0 a formantů.

Schéma syntetizéru formantového typu

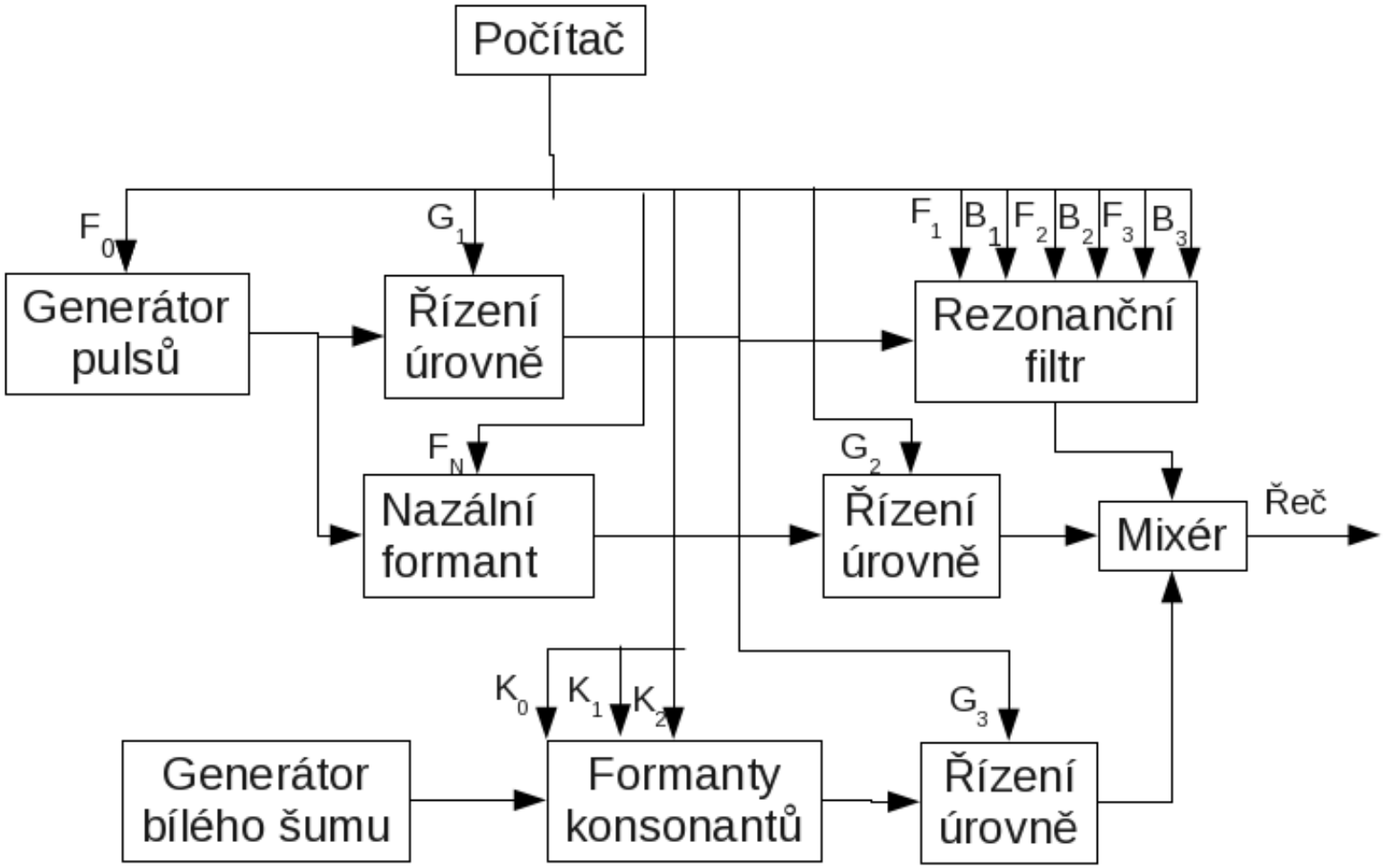
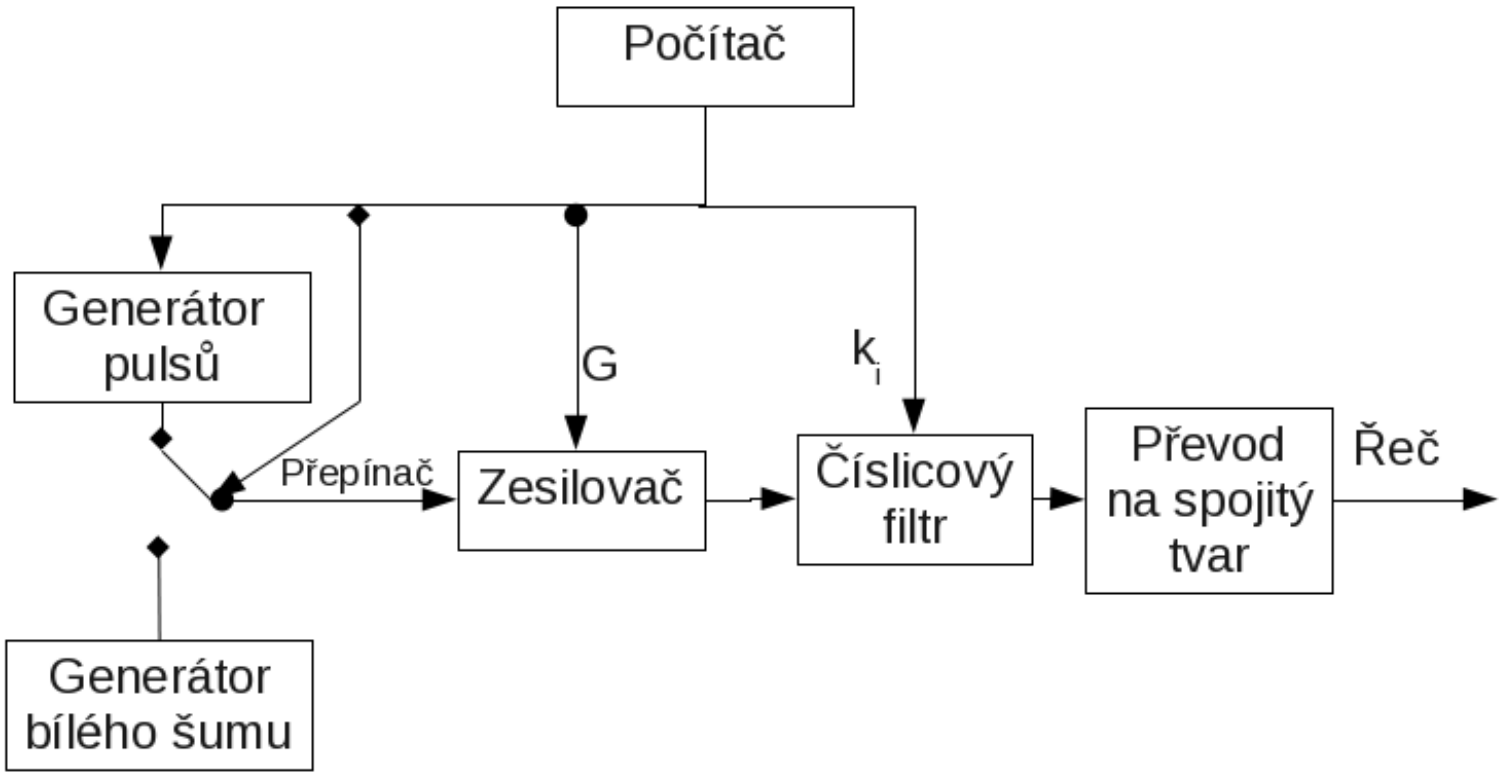


Schéma LPC syntetizéru



Syntéza v časové oblasti

- Princip
  - spojování navzorkovaných řečových segmentů uložených v databázi.
- Využívají se různé typy základních segmentů:
  - větší
    - lépe se modelují některé další charakteristiky jako intonace, přízvuky, ...
    - větší nároky na paměť - větší množství segmentů (potenciálně až 2n, kde n je délka segmentu)
    - příklady – slova, části vět, ...
  - menší
    - menší paměťové nároky - menší množství segmentů
    - horší možnost modelování větné intonace, přízvuků, ... (viz oblasti spektrální stacionarity řeči).

Používané řečové segmenty

- Alofóny
  - poziční varianty fonémů - obsahuje i části okolních fonémů
  - počet n^3 (n - počet fonémů)
- Difóny
  - začínají uprostřed jednoho fonému a končí uprostřed následujícího
  - počet n^2
  - často využívané pro syntézu i rozpoznávání:
    - MBrola (`http://tcts.fpms.ac.be/synthesis/mbrola.html`)

Používané řečové segmenty
Pokračování

- Trifóny
  - začínají uprostřed levého sousedního fonému a končí uprostřed pravého sousedního
  - počet n^3
  - často využívané pro rozpoznávání a syntézu
- Slabičné segmenty.
- Segmenty proměnné délky získané z korpusu.
- Rámce

Slabiky

- Slabika
  - Slabikovat se učí už děti v první třídě.
  - Nejmenší jednotka organizační jednotka řeči.
  - Nelze odvodit strukturu slabik - nejednoznačnost dělení některých slov na slabiky
    - funk-ční vs funkč-ní.
  - Počet slabik - uvádí se cca 10000.
  - Struktura slabiky
    - preatura (onset)
    - nukleus (vokalické jádro) - bývá to samohláska, příp. dvojhláska, sonora - např. k**r**k, frikativa - např. p**s**t, nazála - např. sed**m**
    - koda - nemusí se vyskytovat
    - nukleus + koda jsou považovány za základ slabiky
    - svahy – preatura a koda; jedná se většinou o jednu nebo více souhlásek.

Slabičné segmenty

- Definovány uměle
- Řešení nejednoznačnosti hranice slabiky.
- Frekventované slabičné typy:
  - V (samohláska/dvojhláska) - ú - kol
  - KV (souhláska - samohláska) - vo - da
  - KVK - jed-not-ka
  - KK - tr-sy
  - KKV - dna
  - KKVK - dmout
- Tvoří vice než 95 % slabik
- Umožňují automatickou segmentaci textu.
- Používají se např. v syntetizéru Demosthénes (doc. Kopeček LAF (LSD) FI)