Analysis of knowledge requirements for speech and text alignment problem

Bartosz Kalińczuk

September 25, 2013

Audio model based alignment with word granularity

Phoneme alignment

Limited knowledge alignment

Synthesizer

Conclusions

Required knowledge

- knowledge about alphabet and punctuation marks
- phoneme set
- limited knowledge of graphemes to phonemes conversions
- model parameters for each phoneme

Input

- prepared audio model → for any similar language
- audio recording
- accurate text to be aligned

Algorithm

- convert text to HMM
    - create dictionary with word phoneme represantation
    - create simple language grammar from text
    - convert above grammar to HMM using prepared phoneme models and dictionary
- apply search algorithm to find best scoring HMM state sequence
    - variation of Viterbi's algorithm
    - instead of all possible states it keeps only a priority queue with best performing sequences

English audio model

- "Doktor Piotr"
  After **38** seconds and after **62** words it incorrectly assigned 3 seconds to word “części” and it never recovered.
- “Boże Narodzenie”
  It has gone wrong after **257** seconds and **503** words on word “czarownicach”.

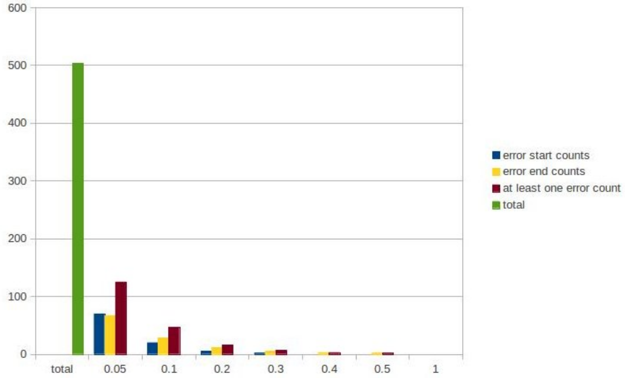
English audio model

The statistics for “Boże Narodzenie” however shows, that before it goes bad it actually aligns first **503** words quite nicely:

- Maximum difference (start or end): **0.559s**
- Maximum difference (start or end), if label was to short at one end: **0.559s**
- Average difference (start or end): **0.032s**

English audio model

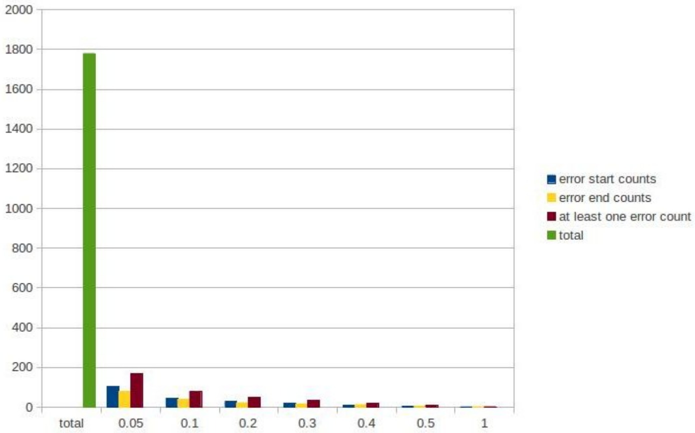
Error counts depending on time thresholds:



Russian audio model - “Boże Narodzenie” statistics

- Total number of words: **1779**
- Maximum difference (start or end): **2.451s**
- Maximum difference (start or end), if label was to short at one end: **0.543s**
- Average difference (start or end): **0.016s**

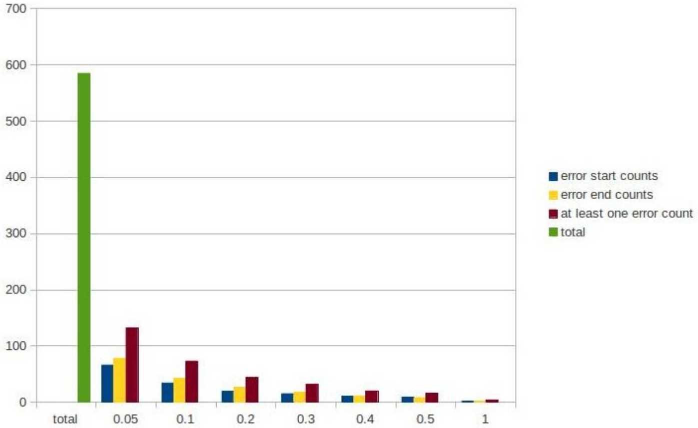
Russian audio model - “Boże Narodzenie” error counts



Russian audio model - “Doktor Piotr” sample statistics

- Total number of words **585**
- Maximum difference (start or end): **1.354s**
- Maximum difference (start or end), if label was to short at one end: **0.534s**
- Average difference (start or end): **0.033s**

Russian audio model - “Doktor Piotr” sample error counts



Input

- prepared audio model → for any similar language
- audio recording
- accurate text to be aligned
- word alignment

Problem definition

For each word and assigned audio part find best phoneme sequence

- each phoneme is represented with a single state
- state is a frame scorer
- state sequence is a sequential assignment of states to audio frames
- best state sequence is the one, that has the smallest score
- sequence score is the sum of scores for each frame using assigned scorer

Algorithm outline

- in Sphinx a phoneme is modelled with a triple state HMM
- each state is a SenomeScorer, which calculates log likelihood of frame emission (by the state)
- DP algorithm calculates best state sequence
    - iterates over frames
    - partial solution is calculated for each state, where the best sequence ends with the state

Algorithm outline

- EM technique applied:
  - Expectation step
    Alignment of phonemes given previously trained distribution
  - Maximization step
    Calculation of normal distribution for each phoneme from assigned frames

Testing sample

Tests are performed using Corpora corpus, which contains audio recordings of digits, names and some unusual sentences for tens of different speakers, and all of these recordings are tagged with phonemes.

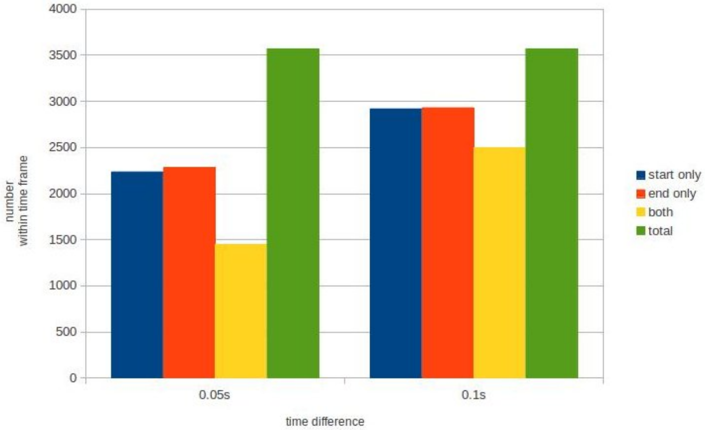
Merged recording contains a 7 minutes and 26 second of audio data, a total of **843** spoken words and **3611** phoneme labels.

Statistics of phoneme alignment with Russian audio model

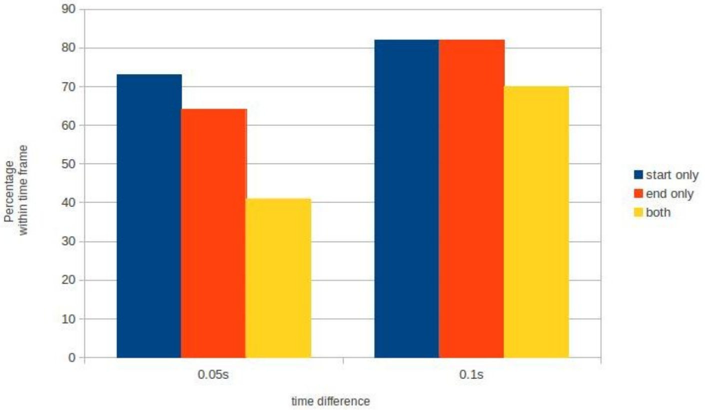
The test was A resulting match contained **3570** of pairs.

- Average start difference: **0.0571s**
- Average end difference: **0.0574s**
- Maximum time difference:**0.516s**

Error counts using Russian audio model

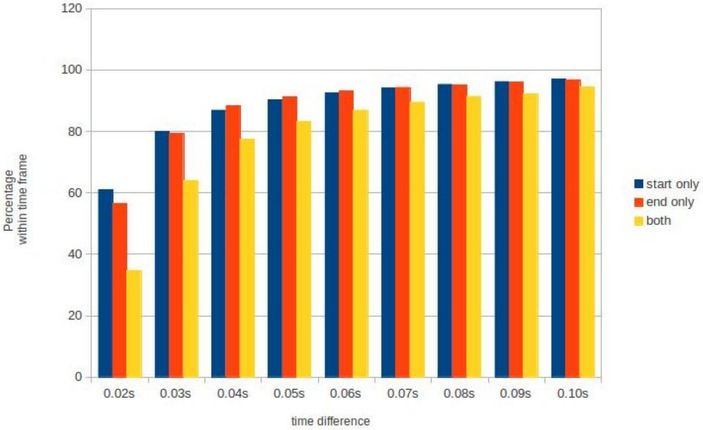


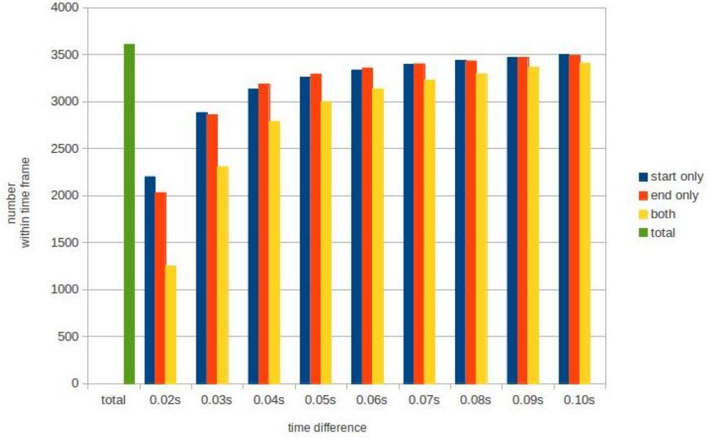
Error counts (percentage) using Russian audio model



Matched phonemes contained **3611** pairs.

- Average start difference: **0.02402s**
- Average end difference: **0.02439s**
- Maximum time difference:**0.5131s**


120
100
80
60
40
20
0
Percentage within time frame
0.02s
0.03s
0.04s
0.05s
0.06s
0.07s
0.08s
0.09s
0.10s
time difference
🟦 start only
🟥 end only
🟨 both


4000
3500
3000
2500
2000
1500
1000
500
0
number
within time frame
total
0.02s
0.03s
0.04s
0.05s
0.06s
0.07s
0.08s
0.09s
0.10s
time difference
start only
end only
both
total

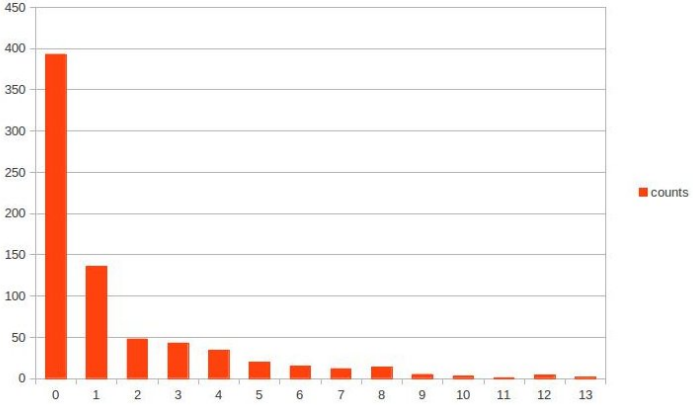
Required knowledge

- knowledge about alphabet and punctuation marks
- knowledge about speech frequency range and subjective perception
- phoneme set [1]
- limited knowledge of graphemes to phonemes conversions [1]

[1] not really needed

- detect pauses/speech parts
- split text to parts by punctuation marks
- calculate expected time of each text part
- match speech and text parts
    - sum of time differences should be minimized
    - parts are matched sequentially
    - DP algorithm solves this problem

- time differences:
  - there were **370** chunks (50,7%) which time frame were within a **0.5s** difference,
  - average time difference was **0.77s**
  - standard deviation was **0.84s**
  - maximum time difference was **11.21s**
- word differences:
  - **393** (53.8%) chunks had **0** difference in words
  - **136** (18.9%) chunks where different by **1** word (missing or additional)
  - **48** (6.6%) different by **2** words
  - **56** (7.7%) with a difference over **5** words


450
400
350
300
250
200
150
100
50
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
🟥 counts

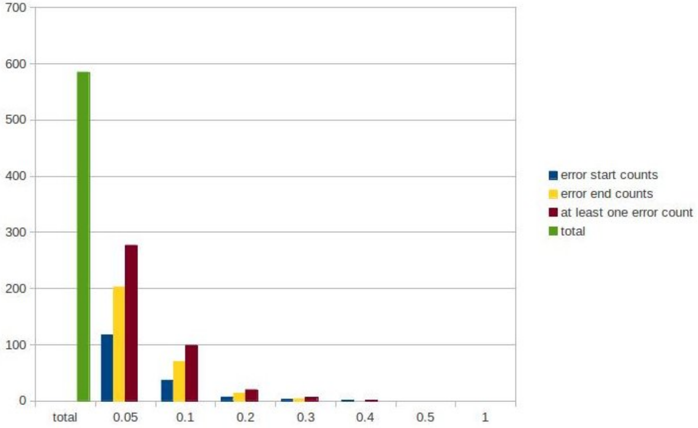
- around 54% of the chunks were correctly identified and around another 25% were nearly correct (less than 3 words difference)
- EM technique applied in similar fashion like in phoneme alignment part
- to improve runtime, only parts below certain threshold might be chosen

- merge between DP algorithm from phoneme alignment and word recognition using HMM from Sphinx library
- DP algorithm uses single state scorers for each phoneme
- no transition likelihood was used
- priority queue keeps only best performing sequence for constant number of possible ending states
- output phoneme assignment is converted to word alignment

Statistics - sample of “Doktor Piotr”

- Total number of words **585**
- Maximum difference (start or end): **0.422s**
- Maximum difference (start or end), if label was to short at one end: **0.371s**
- Average difference (start or end): **0.044s**

Error counts - sample of “Doktor Piotr”

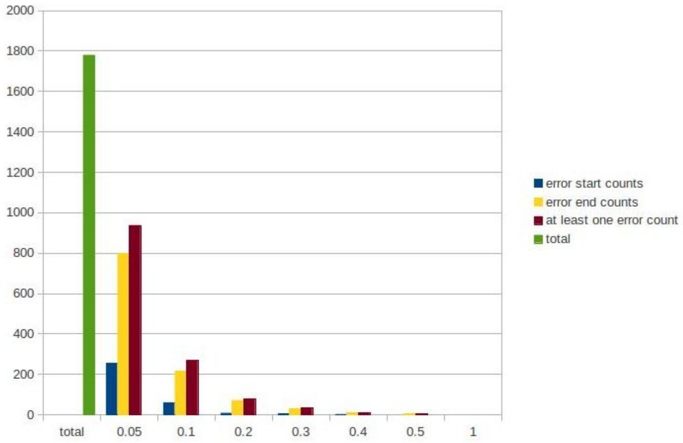


Dropping

Statistics - "Boże Narodzenie"

- Total number of words **1779**
- Maximum difference (start or end): **0.606s**
- Maximum difference (start or end), if label was to short at one end: **0.605s**
- Average difference (start or end): **0.046s**

Error counts - "Boże Narodzenie"



Dropping

Input

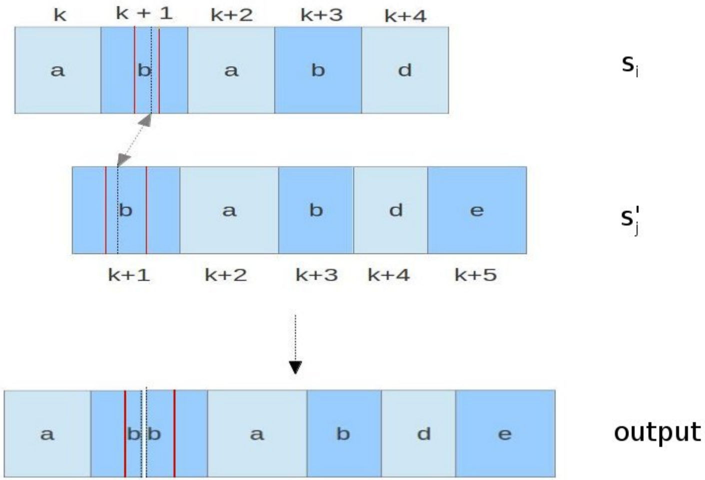
- audio recording
- phoneme alignment
- word alignment
- text to be synthesized

Algorithm outline

- for each word pick suitable candidates or synthesize one
- merge word candidates to create audio, as smoothly as possible

Word synthesis

- choose phoneme candidates (at least two at once)
- merge candidates so the total difference is minimized
- difference between two parts is a frame distance in best merging point
- best merging point candidates are in the middle of a phoneme


k
k + 1
k+2
k+3
k+4
a
b
a
b
d
s_i
b
a
b
d
e
s'_j
k+1
k+2
k+3
k+4
k+5
a
b b
a
b
d
e
output

Sample synthesized texts

Sample synthesized texts

- "W czasie suszy szosa sucha"

Sample synthesized texts

- "W czasie suszy szosa sucha"
- "Za górami za lasami znajduję się wysoka wieża strzeżona przez smoka"

Sample synthesized texts

- "W czasie suszy szosa sucha"
- "Za górami za lasami znajduję się wysoka wieża strzeżona przez smoka"
- "Litwo Ojczyzno moja ty jesteś jak zdrowie Ile cię trzeba cenić ten tylko się dowie Kto cię stracił Dziś piękność twą w całej ozdobie Widzę i opisuję bo tęsknię po tobie"

Sample synthesized texts

- "W czasie suszy szosa sucha"
- "Za górami za lasami znajduję się wysoka wieża strzeżona przez smoka"
- "Litwo Ojczyzno moja ty jesteś jak zdrowie Ile cię trzeba cenić ten tylko się dowie Kto cię stracił Dziś piękność twą w całej ozdobie Widzę i opisuję bo tęsknię po tobie"
- "Chrząszcz brzmi w trzcinie w Szczebrzeszynie W szczękach chrząszcza trzeszczy miąższ Czcza szczypawka czka w Szczecinie"

Sample synthesized texts

- "W czasie suszy szosa sucha"
- "Za górami za lasami znajduję się wysoka wieża strzeżona przez smoka"
- "Litwo Ojczyzno moja ty jesteś jak zdrowie Ile cię trzeba cenić ten tylko się dowie Kto cię stracił Dziś piękność twą w całej ozdobie Widzę i opisuję bo tęsknię po tobie"
- "Chrząszcz brzmi w trzcinie w Szczebrzeszynie W szczękach chrząszcza trzeszczy miąższ Czcza szczypawka czka w Szczecinie"
- "Rosja przedwojenna była wymarzoną areną dorobku dla ludzi tego typu zwłaszcza pochodzących z Królestwa"

Conclusions

I believe that I managed to show in this thesis, that word alignment can be done without much apriori knowledge. My algorithm was able to return a decent alignment for an input audio file and recorded text with only knowledge about:

1. punctuation marks and their relationship to pauses
2. a bit of knowledge about relationship between graphemes and phonemes (conversions grammars)
3. a human anatomy and capabilities, especially about speech frequency ranges
4. assumption that the input text has a high accuracy