INTERNATIONAL CONFERENCE

COMMUNICATIONS 2008

June 05 – 07, 2008
BUCHAREST, ROMANIA

PROCEEDINGS

Organized by:
Military Technical Academy
“Politehnica” University of Bucharest
“Electronica 2000” Foundation
and
IEEE Romanian Section

Editura PRINTECH
2008
Honorary Chairperson:

Adelaida MATEESCU,  
"Politehnica" University of Bucharest

Conference Chairpersons:

Mihai RADU,  
Military Technical Academy  
Alexandru ȘERBĂNESCU,  
Military Technical Academy  
Victor CROITORU,  
"Politehnica" University of Bucharest

Honorary Committee:

Doru SAFTA,  
Rector of Military Technical Academy, Bucharest  
Ecaterina ANDRONESCU,  
Rector of "Politehnica" University of Bucharest

Organizing Committee

"Politehnica" University of Bucharest  
Military Technical Academy

Theodor PETRESCU  
Iocan NICOLAESCU
Silviu CIOCHINĂ  
Adrian STOICA
Ion BĂNICĂ  
Victor PATRICIU
Ion MARGHESCU  
Doru MUNTEANU
Eugen BORCOCI  
Cristian MOLDER
Constantin PALEOLOGU  
Cristian RÎNCU
Călin VLĂDEANU
Descrierea CIP a Bibliotecii Naționale a României

**COLECTIV AUTORI**
COMMUNICATIONS 2008

București, Printech, 2008
Bibliogr., Index
**ISBN** 978-606-521-008-0

© Copyright 2008
Toate drepturile prezentei ediții sunt rezervate autorilor. Nici o parte din această lucrare nu poate fi reprodusă, stocată sau transmisă indiferent prin ce formă, fără acordul prealabil scris al autorilor.
NATURAL LANGUAGE AND SPEECH PROCESSING
STATISTICAL LANGUAGE MODELING ON SUSANNE CORPUS

Domokos József1,2, Todorcean Gavriliţa2, Ovidiu Buza2

1Faculty of Technical and Human Sciences/Electrical Engineering Department, Sapientia University
Sighetu Marmatiei, str. 1C, 540485, Targu-Mures/Corunca, Romania
phone: +40265-208.170, fax: +40265-206.211, email: coml@rs.sapientia.ro
web: www.rs.sapientia.ro

2Faculty of Electronics, Telecommunications and Information Technology/Communication Department, Technical University of Cluj-Napoca
Cheile Băiului street, 26-28, 400027, Cluj-Napoca, Romania
phone: +40264-596.285, fax: +40264-591.689
email: todorceas@prosoft.ro
email: Ovidiu.Buza@com.utcluj.ro
web: www.eti.utcluj.ro

ABSTRACT
In this paper are presented the theoretical background and some experiments for statistical language modeling which is a main part of a statistical speech recognition system. There are presented the unigram, bigram and trigram language model as well as the Good-Turing estimator based back-off smoothing algorithm. The experiments were made on Susanne Corpus, a grammatically tagged speech corpus, based on Brown corpus. The built language models are tested and compared using perplexity measure.

1. STATISTICAL SPEECH RECOGNITION
Statistical speech recognition is based on Hidden Markov Models (HMM’s). Such a system, depicted in Fig. 1, is built using multiple chained HMM’s for acoustic modeling and language modeling.

The above system can be described mathematically as follows [3] [4] [5] [8]; we have a set of acoustic vectors \( A = \{a_1, a_2, ..., a_n\} \) and we are searching the most prob-

\[
W^* = \arg \max \left\{ \frac{P(A | W) \cdot P(W)}{P(A)} \right\}
\]

We know, that probability of acoustic vector \( P(A) \) is constant, and we have:

\[
W^* = \arg \max \left\{ P(A | W) \cdot P(W) \right\}
\]

In Eq. 3 we have:
- \( P(W) \) – the language model
- \( P(A|W) \) – the acoustic model

The acoustic modeling part of the speech recognition system can be developed using HMMs, Gaussian Mixture Models (GMMs) or Artificial Neural Networks (ANNs).

The language modeling part of the system can be: [1] [3] [4] [5]:
- statistical language model
- context free grammar (CFG)
- probabilistic context free grammar (PCFG).

In this paper we want to present the statistical n-gram type language model which is the most powerful and the most widely used one.

2. STATISTICAL LANGUAGE MODELING
The speech can be considered a stochastic process and every linguistic unit (phone, syllable, or word) can be considered a random variable with a random probability distribution. The n-gram language models try to estimate the probability of the next word based on the history (the last n-1 preceding words) [4] [5] [7] [8].

The language model must estimate the probability of word sequence \( w^*_n = (w_1, w_2, ..., w_n) \), which is:

\[
p(w^*_n) = p(w_1) \cdot p(w_2 | w_1) \cdot p(w_3 | w_1, w_2) \cdot ... \cdot p(w_n | w_1^{n-1})
\]

\[
p(w^*_n) = p(w_1) \cdot \prod_{k=2}^{n} p(w_k | w_{k-1}^{k-1})
\]

Using Markov assumption, the history can be reduced to the last n-1 words, and we have:
\( P(w_n \mid w_{n-1}^{n-1}) = P(w_n \mid w_{k+1}^{k+1}) \)  

Even Eq. 5 is hard to compute for \( n > 3 \) because we need a huge training corpus to properly evaluate the probabilities. For \( n = 1 \cdots 3 \) we have:
- Unigram language model \((n=1)\)
- Bigram language model \((n=2)\)
- Trigram language model \((n=3)\)

### 2.1 Unigram language model

The unigram language model considers all the words independent. This means that no history information is involved.

\[ P(w_k \mid w_{k-1}^{k-1}) = P(w_k) \]  

If we use Eq. 4, we have the probability estimation for the unigram model.

\[ p(w_k) = \prod_{k=2}^{n} p(w_k) \]  

### 2.2 Bigram language model

The bigram language model takes in consideration only one word for history.

\[ P(w_k \mid w_{k-1}^{k-2}) = P(w_k \mid w_{k-1}) \]  

If we put Eq. 8 in Eq. 4, we have the formula for probability estimation for bigram language model:

\[ p(w_k) = p(w_k) \cdot \prod_{k=2}^{n} p(w_k \mid w_{k-1}) \]  

### 2.3 Trigram language model

The trigram language model uses two word history.

\[ P(w_k \mid w_{k-1}^{k-2}) = P(w_k \mid w_{k-2}, w_{k-3}) \]  

The formula for probability estimation is given in Eq. 11.

\[ p(w_k) = p(w_k) \cdot p(w_k \mid w_{k-1}) \cdot \prod_{k=2}^{n} p(w_k \mid w_{k-3}, w_{k-2}) \]  

### 3. PROBABILITY ESTIMATION AND SMOOTHING

The probabilities for Eq. 7, 9, and 11 can be simply calculated using MLE (Maximum Likelihood Expectation) algorithm [5]. Thus we have for unigram, bigram and trigram language models the following MLE estimators.

\[ p(w_k) = \frac{n_k}{N} \]  

\[ p(w_k \mid w_{k-1}) = \frac{N_{r_k}(w_{k-1} \mid w_{k-2})}{N_{r_k}(w_{k-1})} \]  

\[ p(w_k \mid w_{k-2}, w_{k-3}) = \frac{N_{r_k}(w_{k-2}, w_{k-3})}{N_{r_k}(w_{k-2})} \]  

where \( n_k \) is the number of occurrence of word \( k \);
\( N \) is the number of words which occurs exactly \( r \) times in the training corpus;
\( N \) is the total number of words from the training corpus;
\( E \) is an estimation function for \( N \);
\( r^* \) is the adjusted number of occurrence.

The total value of probability calculated using Good-Turing estimator is always smaller than \( l \). The remaining probability mass is reallocated to the unseen words from the vocabulary. The simplest way to choose the estimation function \( E \) is presented in Eq. 16.

\[ E(n+1) = \frac{n}{n+1} \cdot (1 - \frac{E(1)}{N}) \]  

### 3.2. Back-off smoothing

Back-off smoothing was firstly introduced by Katz. He showed that MLE estimation of probabilities is good enough if the number of occurrences of a word is bigger than a threshold value \( K = 6 \) [5].

All the probabilities for \( n \)-gram word sequences which have an occurrence number between 0 and \( K \) will be smoothed using Good-Turing estimator to serve probability mass for unseen word sequences.

If a word sequence has zero occurrences we try to estimate its probability using the inferior \((n-1)\)-gram model. If the occurrence is still zero for this inferior model we continue to back-off to a lower model. Finally if we reach the unigram model, we have the relative frequency of a word bigger than zero.

For a trigram back-off model we have the following relations:

Smoothing means that a probability mass is retained from high probabilities to be reallocated to zero or small probability values. There are a lot of useful smoothing techniques:

- Add one or Laplace smoothing
- Good-Turing estimator
- Back-off or Katz smoothing
- Kneser - Ney smoothing
- Jelinek - Mercer smoothing or interpolation

For the experiments we used Good - Turing estimator and back-off smoothing.

### 3.1. Good-Turing estimator

The Good-Turing estimator comes from biology where it was used for species estimation.

The general form of the estimator is [6]:

\[ P(X) = \frac{r^*}{N} \]  

where:
\( r^* = (r + 1) \cdot \frac{E(N_{r+1})}{E(N_{r})} \)

where:
\( r \) is the number of occurrence of word \( k \);
\( N_{r} \) is the number of words which occurs exactly \( r \) times in the training corpus;
\( N \) is the total number of words from the training corpus;
\( E \) is an estimation function for \( N \);
\( r^* \) is the adjusted number of occurrence.

The total value of probability calculated using Good-Turing estimator is always smaller than \( l \). The remaining probability mass is reallocated to the unseen words from the vocabulary. The simplest way to choose the estimation function \( E \) is presented in Eq. 16.

\[ E(n+1) = \frac{n}{n+1} \cdot (1 - \frac{E(1)}{N}) \]
4. LANGUAGE MODEL EVALUATION

Language model evaluation can be done in different ways [1] [2] [4] [5]:
- random sentence generation
- words reordering in sentences
- perplexity

For the experiments we used perplexity to measure the quality of language models. Perplexity is the most used measure for language model evaluation.

Perplexity can be defined using entropy from information theory. For a random variable $X = \{x_1, x_2, ..., x_L\}$, the entropy can be defined:

$$H(X) = -\sum_{x \in X} p(x) \cdot \log_2 p(x)$$  

(17)

Instead of entropy, we use the entropy rate calculated as follows:

$$\frac{1}{L} H(w^*_1) = -\frac{1}{L} \sum_{w \in \mathcal{V}} p(w^*_1) \cdot \log_2 p(w^*_1)$$

(18)

For a real language we should consider infinitely long word sequences:

$$\frac{1}{L} H(w^*_1) = -\frac{1}{L} \sum_{w \in \mathcal{V}} p(w^*_1) \cdot \log_2 p(w^*_1)$$

(19)

Using Shannon-McMillan-Breiman theorem, if the language is stationary and ergodic, the above formula can be simplified:

$$H(L) = \lim_{n \to \infty} -\frac{1}{N} \log_2 p(w^*_1)$$

(20)

Finally, we use a large training corpus to estimate probabilities $p^*$ and we have the logprob value:

$$LP = -\frac{1}{N} \log_2 p^*(w^*_1)$$

(21)

Perplexity is defined:

$$PP = 2^{LP}$$

(22)

5. EXPERIMENTAL RESULTS

We try to create and compare the unigram, bigram and trigram Katz back-off language models on Susanne Corpus, developed at University of Sussex, England [8]. This corpus contains 64 grammatically tagged files from Brown Corpus, each of them with more than 2000 words. The 64 files are grouped in 4 categories (each with 16 files) noted as follows [8]:

- A – press reportage
- G – belles letters, biography, memoirs
- I – learned (mainly scientific and technical) writing
- N – adventure and Western fiction

We divide the corpus in two sets: one training set and one test set. The test set consists of one file from each category. The rest of files were used for training the language models. In our first experiment, we have build unigram, bigram and trigram language models for each category using the most probable 5000 words for dictionary and test them using the corresponding test file.

We calculate the perplexity of language models to evaluate and compare them.

The results are presented in Tables 1 to 4.

For $A$ type model we have used 33175 words in the training set and 2226 words from 85 sentences in the test set. The number of out-of-vocabulary words was 505, that means 23.59%.

Table 1. Perplexity results for language models built using $A$ category files from Susanne Corpus

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Words predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>525.85</td>
<td>1513</td>
</tr>
<tr>
<td>Bigram</td>
<td>368.07</td>
<td>1277</td>
</tr>
<tr>
<td>Trigram</td>
<td>376.24</td>
<td>959</td>
</tr>
</tbody>
</table>

For $G$ type texts model we used 33866 words for training and 78 sentences with a number of 2232 words for testing. The number of out-of-vocabulary words was 394 (18.29%).

Table 2. Perplexity results for language models built using $G$ category files from Susanne Corpus

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Words predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>525.61</td>
<td>1760</td>
</tr>
<tr>
<td>Bigram</td>
<td>403.48</td>
<td>1427</td>
</tr>
<tr>
<td>Trigram</td>
<td>409.81</td>
<td>1180</td>
</tr>
</tbody>
</table>

For $I$ type texts model we used 33373 words for training and 62 sentences with a number of 2164 words for testing. The number of out of vocabulary words was 441 (20.98%).

Table 3. Perplexity results for language models built using $I$ category files from Susanne Corpus

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Words predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>559.74</td>
<td>1637</td>
</tr>
<tr>
<td>Bigram</td>
<td>397.37</td>
<td>1277</td>
</tr>
<tr>
<td>Trigram</td>
<td>419.52</td>
<td>1011</td>
</tr>
</tbody>
</table>

For $N$ type texts model we used 34845 words for training and 103 sentences with a number of 2230 words for testing. The number of out of vocabulary words was 267 (12.55%).

Table 4. Perplexity results for language models built using $N$ category files from Susanne Corpus

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Words predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>410.7</td>
<td>1836</td>
</tr>
<tr>
<td>Bigram</td>
<td>260.52</td>
<td>1612</td>
</tr>
<tr>
<td>Trigram</td>
<td>261.48</td>
<td>1448</td>
</tr>
</tbody>
</table>
In the second experiment we have built unigram, bigram and trigram models for the whole corpus using three different dictionaries (5000, 10000, 15000 words) and test them with all test set. The 5000 word dictionary is useful to compare this model with those adapted to text genre. The model built based on 10000 words is relevant because appreciatively 10000 words occur more than once in training set. The last model built on a 15000 word dictionary includes almost all of the vocabulary of the training set of Susanne corpus. Thus we trained the models using 135259 words (15444 distinct words) from 6259 sentences. The test set contains 8855 words from 328 sentences. The results are resumed in Table 5.

Table 5. Perplexity results for language models built using the entire Susanne Corpus

<table>
<thead>
<tr>
<th>Vocabulary</th>
<th>Model</th>
<th>Perplexity</th>
<th>Words predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000</td>
<td>Unigram</td>
<td>625.03</td>
<td>7126</td>
</tr>
<tr>
<td></td>
<td>Bigram</td>
<td>555.64</td>
<td>5957</td>
</tr>
<tr>
<td></td>
<td>Trigram</td>
<td>352.7</td>
<td>5078</td>
</tr>
<tr>
<td>10000</td>
<td>Unigram</td>
<td>815.53</td>
<td>7542</td>
</tr>
<tr>
<td></td>
<td>Bigram</td>
<td>476.64</td>
<td>6673</td>
</tr>
<tr>
<td></td>
<td>Trigram</td>
<td>470.62</td>
<td>5984</td>
</tr>
<tr>
<td>15000</td>
<td>Unigram</td>
<td>936.22</td>
<td>7753</td>
</tr>
<tr>
<td></td>
<td>Bigram</td>
<td>550.26</td>
<td>7065</td>
</tr>
<tr>
<td></td>
<td>Trigram</td>
<td>545.62</td>
<td>6502</td>
</tr>
</tbody>
</table>

5. CONCLUSIONS AND FURTHER WORKS

In the first experiment the best results are achieved using bigram model. The trigram model can't improve the results because there are insufficient data for training the models. We can see from the results that if the number of words increase, the perplexity of the model increase too, and the model has weaker quality. The models built by eliminating the words with occurrences smaller than a threshold are simpler and performs better. The best perplexity achieved for N type files from Susanne corpus outperforms those reported in the literature (i.e. [9]). The words which occur once does not improve the model quality, they rise perplexity and they should be eliminated from vocabulary. In the second experiment the trigram models performs better than bigram models. This performance can be explained by the large amount of available training data. In conclusion the used n-gram model should be chosen considering the amount of training data.

REFERENCES
CONTENTS

SESSION OF INVITED PAPERS

CURRENT AND PLANNED IMPLEMENTATIONS OF NETWORK CAPABILITIES WITHIN THE ROMANIAN ARMED FORCES .......................................................... 7
Cristea Dumitru, Costică Pastalache

A NEW MULTICARRIER SYSTEM: A RESEARCH STORY IN THREE CHAPTERS .... 15
Willmut Zschunke

UNDERWATER ACOUSTIC COMMUNICATIONS: OPPORTUNITIES AND RESEARCH CHALLENGES ................................................................................ 23
André Quinquis

SIGNAL PROCESSING 1

NOISE ESTIMATION VARIABLE STEP SIZE (NE-VSS) ADAPTIVE ALGORITHM ..... 29
Sorin Zoican

PERMANENT MAGNET SYNCHRONOUS MOTOR DRIVE SYSTEM ...................... 33
Paul Alexandru Stânescu, Pascal Goureau, Ovidiu Radu, Ana-Maria Manole

A BIPOLAR IMPLEMENTATION OF SECOND ORDER CURRENT MODE ELIN FILTERS WITH INSTANTANEOUS “ARCSINH” COMPANDING ........................................ 37
Radu Gabriel Bosomitu, Iulian Chereches

MULTICOMPONENT FILTERING FOR LEAKAGE DETECTION USING BIDIMENSIONAL THERMOMETRIC DATA ................................................................. 41
Amir Khan, Valeriu Vrabie, Guy D’Urso, Jérôme Mars

SURVEYING THE FRONTIER BETWEEN SIGNAL PROCESSING AND DYNAMICAL SYSTEMS .................................................................................. 45
Felix Tatin, Alexandru Șerbănescu, Corneli Ioana, Jerome Mars, Srdjan Stankovic, Mădălina Mazilu

SIGNAL PROCESSING 2

TANDEM REPEATS DETECTION IN BIOLOGICAL SEQUENCES ..................... 51
Petre G. Pop, Eugen Lupu

A NEW TIME APPROACH ALGORITHM FOR ENGINE KNOCK DETECTION .... 55
Iulian Nîță, Dan Lăzărescu, Rodica Constantinescu

LOSSLESS TEXT COMPRESSION USING THE LIPT TRANSFORM ................... 59
Radu Rădescu

439
AN EFFICIENT STUDY ON THE MODIFICATION OF KAISER WINDOW
Kemal Avci, Arif Nacaroğlu

63

NATURAL LANGUAGE AND SPEECH PROCESSING

STATISTICAL LANGUAGE MODELING ON SUSANNE CORPUS
Jozsef Domokos, Gavril Todorean, Ovidiu Baza

69

PHONETIC SPEAKER RECOGNITION
Margit Antal

73

SOME CONNECTIONS BETWEEN DIGRAM AND WORD STATISTICAL
STRUCTURES IN PRINTED ROMANIAN WHEN ORTHOGRAPHY
AND PUNCTUATION MARKS ARE INCLUDED
Adrian Mitrea, Adriana Vlad, Mihai Mitrea

77

LANGUAGE PRODUCTION - THE LEXICON
Amelia Molea

81

LETTER TO SOUND SYSTEM FOR THE ROMANIAN LANGUAGE
Stefan-Adrian Toma, Doru-Petru Munteanu

85

PROFESSIONAL SPEECH CORPUS
Eugeniu Oancea, Corneliu Burileanu, Amelia Molea

89

ON THE WAY FROM AUTOMATIC SPEECH RECOGNITION-ASR
TO AUTOMATIC SPEECH RECOGNITION AND UNDERSTANDING-ASRU
Inge Gavat, Octavian Dumitru

93

IMAGE PROCESSING 1

MULTIVALUED KNOWLEDGE REPRESENTATION BASED COLOR ANALYSIS
Vasile Pătraşcu

99

BEST MEAN RANK VECTOR FILTER FOR IMPULSIVE NOISE REDUCTION
IN COLOR IMAGES
Valeriu Vrabe, Michel Herbin, Constantin Vertan, Mihai Ciuc

103

AUTOMATIC FINGERPRINT IDENTIFICATION SYSTEM EVALUATION USING
CLASSIFICATION AND MATCHING PARAMETERS
Mohsen Mohammadpour, Mohammad Sadegh Helfroush

107

HIP PROSTHESIS X-RAY IMAGE PROCESSING BY LIP EXPOSURE PREDICTION
Corneliu Florea, Constantin Vertan, Laura Florea

111

IMPROVED NYQUIST FILTER CHARACTERISTIC USING SPLINE FUNCTIONS
INTERPOLATION
Ligia Alexandra Onofrei, Nicolae Dumitru Alexandru

115