N-Gram Language Models for Offline Handwritten Text Recognition

IWFHR 2004 Tokyo, Japan

Matthias Zimmermann and Horst Bunke

Computer Science Department

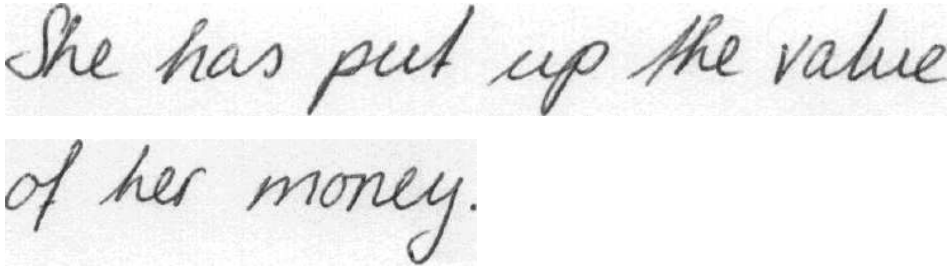
University of Bern, Switzerland

Organization

1. Introduction
2. N-Gram Language Models
3. Random Sentence Generation
4. Language Model Performance
5. Recognition System Performance
6. Setup, Optimization and Evaluation
7. Conclusions

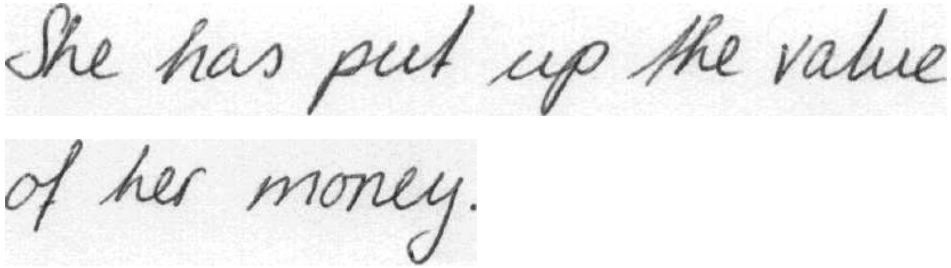
1. Introduction

Recognition of Handwritten Sentences



1. Introduction

Recognition of Handwritten Sentences

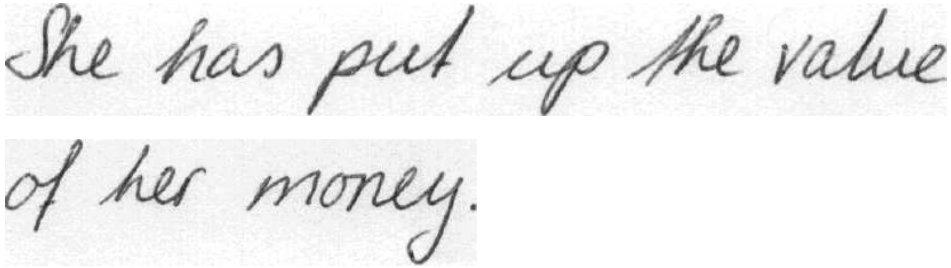


↓

Recognizer

1. Introduction

Recognition of Handwritten Sentences



↓

Recognizer

↓

She has put up the value of her money .

1. Introduction

Handwriting Recognition and Language Modeling

- recognition of handwritten text is investigated since many years

1. Introduction

Handwriting Recognition and Language Modeling

- recognition of handwritten text is investigated since many years
- the use of statistical LM is relatively new

1. Introduction

Handwriting Recognition and Language Modeling

- recognition of handwritten text is investigated since many years
- the use of statistical LM is relatively new
- the use of LM consistently improves baseline recognizer

1. Introduction

Handwriting Recognition and Language Modeling

- recognition of handwritten text is investigated since many years
- the use of statistical LM is relatively new
- the use of LM consistently improves baseline recognizer
- LM are predominantly used within the HMM framework

1. Introduction

Handwriting Recognition and Language Modeling

- recognition of handwritten text is investigated since many years
- the use of statistical LM is relatively new
- the use of LM consistently improves baseline recognizer
- LM are predominantly used within the HMM framework
- so far, mostly bigram LM were used

1. Introduction

General Principle

$$\hat{W} = \underset{W}{argmax}\ \ p(X|W)p(W)$$

1. Introduction

General Principle

$$\hat{W} = \underset{W}{argmax}\ p(X|W)p(W)$$

$p(X|W)$ provided by HMM (optical model)

1. Introduction

General Principle

$$\hat{W} = \underset{W}{argmax}\ p(X|W)p(W)$$

$p(X\|W)$	provided by HMM (optical model)
$p(W)$	provided by the Language Model (LM)

1. Introduction

General Principle

$$\hat{W} = \underset{W}{argmax}\ p(X|W)p(W)$$

$p(X\|W)$	provided by HMM (optical model)
$p(W)$	provided by the Language Model (LM)

$p(W) =$ $p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

Approximation by N-Gram LM

$$p(W) = \prod_{i=1}^{n} p(w_i | w_{i-n+1}^{i-1})$$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

Approximation by N-Gram LM

$$p(W) = \prod_{i=1}^{n} p(w_i | w_{i-n+1}^{i-1})$$

Trigram Example ($n = 3$)

$$p(W) = p(She | \diamond \diamond)$$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

Approximation by N-Gram LM

$$p(W) = \prod_{i=1}^{n} p(w_i | w_{i-n+1}^{i-1})$$

Trigram Example ($n = 3$)

$$p(W) = p(She|\diamond\diamond)\ p(has|\diamond\ She)$$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

Approximation by N-Gram LM

$$p(W) = \prod_{i=1}^{n} p(w_i | w_{i-n+1}^{i-1})$$

Trigram Example ($n = 3$)

$$p(W) = p(She|\diamond\diamond)\ p(has|\diamond\ She)\ p(put|She\ has)$$

2. N-Gram Language Models

Statistical LM provide probabilities for given sentences

$$p(W) = p(She\ has\ put\ up\ the\ value\ of\ her\ money\ .)$$

Approximation by N-Gram LM

$$p(W) = \prod_{i=1}^{n} p(w_i | w_{i-n+1}^{i-1})$$

Trigram Example ($n = 3$)

$$p(W) = p(She|\diamond\diamond)\ p(has|\diamond\ She)\ p(put|She\ has)\ ...\ \ p(.|her\ money)$$

2. N-Gram Language Models

This Talk

- comparison of bigram and trigram LM
- artificial and natural text to smooth LM

2. N-Gram Language Models

This Talk

- comparison of bigram and trigram LM
- artificial and natural text to smooth LM

Bigram and Trigram LM

- positive effect is reported for bigram LM
- trigram LM typically outperform bigram LM in the speech domain

2. N-Gram Language Models

This Talk

- comparison of bigram and trigram LM
- artificial and natural text to smooth LM

Bigram and Trigram LM

- positive effect is reported for bigram LM
- trigram LM typically outperform bigram LM in the speech domain

Artificial Text

- SCFG to produce random sentences to smooth n-gram LM
- grammar extracted from Lancaster parsed corpus

2. N-Gram Language Models

This Talk

- comparison of bigram and trigram LM
- artificial and natural text to smooth LM

Bigram and Trigram LM

- positive effect is reported for bigram LM
- trigram LM typically outperform bigram LM in the speech domain

Artificial Text

- SCFG to produce random sentences to smooth n-gram LM
- grammar extracted from Lancaster parsed corpus

Natural Text

- Brown corpus for American English
- Wellington corpus for New Zealand English

3. Random Sentence Generation

Extraction of Stochastic Context-Free Grammars (SCFG)

- extract all productions from parsed corpus

3. Random Sentence Generation

Extraction of Stochastic Context-Free Grammars (SCFG)

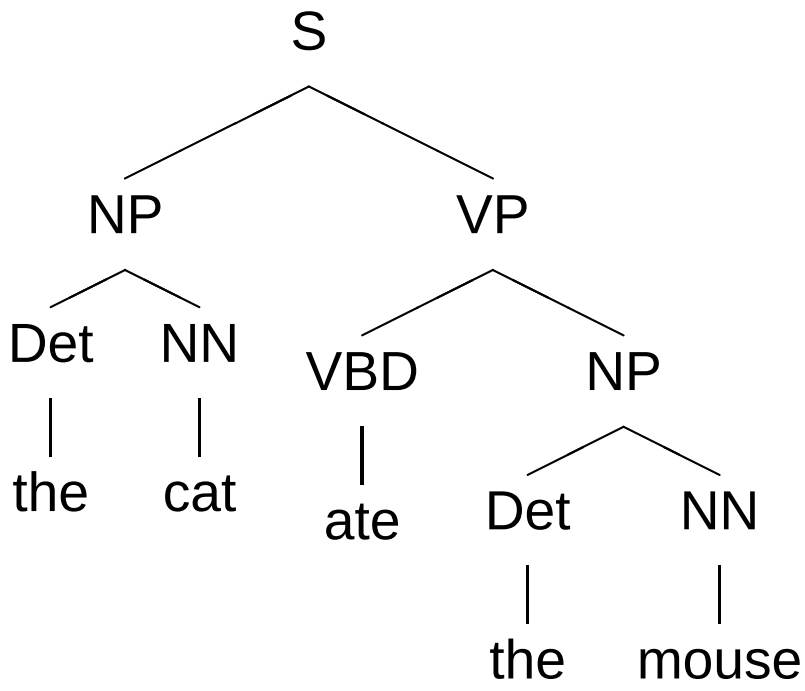
- extract all productions from parsed corpus
- estimate production probabilities from relative frequencies

3. Random Sentence Generation

Extraction of Stochastic Context-Free Grammars (SCFG)

- extract all productions from parsed corpus
- estimate production probabilities from relative frequencies

Sentence from Parsed Corpus



3. Random Sentence Generation

Extraction of Productions

A	$\rightarrow$	α
S	$\rightarrow$	NP VP .

3. Random Sentence Generation

Extraction of Productions

A	$\rightarrow$	α
S	$\rightarrow$	NP VP .
NP	$\rightarrow$	Det NN

3. Random Sentence Generation

Extraction of Productions

A	$\rightarrow$	α
S	$\rightarrow$	NP VP .
NP	$\rightarrow$	Det NN
VP	$\rightarrow$	VBD NP

3. Random Sentence Generation

Extraction of Productions

A	$\rightarrow$	α
S	$\rightarrow$	NP VP .
NP	$\rightarrow$	Det NN
VP	$\rightarrow$	VBD NP
Det	$\rightarrow$	the
...		

3. Random Sentence Generation

Extraction of Productions

A	$\rightarrow$	α
S	$\rightarrow$	NP VP .
NP	$\rightarrow$	Det NN
VP	$\rightarrow$	VBD NP
Det	$\rightarrow$	the
...		

Production Probabilities

$$p(A \rightarrow \alpha) = \frac{N(A \rightarrow \alpha)}{\sum_{\beta} N(A \rightarrow \beta)}$$

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP $\Rightarrow$ Det NN VP

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP $\Rightarrow$ Det NN VP $\Rightarrow$ the NN VP

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP $\Rightarrow$ Det NN VP $\Rightarrow$ the NN VP $\Rightarrow$ the mouse VP

$\Rightarrow \ldots \Rightarrow$ the mouse ate the cat

- sentences grammatically correct

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP $\Rightarrow$ Det NN VP $\Rightarrow$ the NN VP $\Rightarrow$ the mouse VP

$\Rightarrow \ldots \Rightarrow$ the mouse ate the cat

- sentences grammatically correct
- sentences normally not meaningful

3. Random Sentence Generation

SCFG based Sentence Generation

- start with start symbol S
- select a nonterminal symbol A
- choose production with a left hand side A (according to the production probabilities)

Example

S $\Rightarrow$ NP VP $\Rightarrow$ Det NN VP $\Rightarrow$ the NN VP $\Rightarrow$ the mouse VP

$\Rightarrow \ldots \Rightarrow$ the mouse ate the cat

- sentences grammatically correct
- sentences normally not meaningful
- arbitrary number of sentences can be produced

4. Language Model Performance

Perplexity is standard Performance Measure

- measures fitness of a given LM to explain a provided test text
- test text is a set of sentences $(s_1, s_2, \ldots, s_n)$

4. Language Model Performance

Perplexity is standard Performance Measure

- measures fitness of a given LM to explain a provided test text
- test text is a set of sentences $(s_1, s_2, \ldots, s_n)$

$$
\begin{aligned}
PP_T(M) &= 2^{H_T(M)} \\
H_T(M) &= -\frac{1}{n} \sum_{i=1}^{n} \log p(s_i)
\end{aligned}
$$

- $H_T(M)$ is the (cross)entropy of the LM M for text T

4. Language Model Performance

Perplexity is standard Performance Measure

- measures fitness of a given LM to explain a provided test text
- test text is a set of sentences $(s_1, s_2, \ldots, s_n)$

$$
\begin{aligned}
PP_T(M) &= 2^{H_T(M)} \\
H_T(M) &= -\frac{1}{n} \sum_{i=1}^{n} \log p(s_i)
\end{aligned}
$$

- $H_T(M)$ is the (cross)entropy of the LM M for text T
- $p(s_i)$ is the probability of the sentence s_i for the given LM

4. Language Model Performance

Perplexity is standard Performance Measure

- measures fitness of a given LM to explain a provided test text
- test text is a set of sentences $(s_1, s_2, \ldots, s_n)$

$$
\begin{aligned}
PP_T(M) &= 2^{H_T(M)} \\
H_T(M) &= -\frac{1}{n}\sum_{i=1}^{n} \log p(s_i)
\end{aligned}
$$

- $H_T(M)$ is the (cross)entropy of the LM M for text T
- $p(s_i)$ is the probability of the sentence s_i for the given LM
- perplexity values correspond intuitively to the number of relevant words in the lexicon

4. Language Model Performance

Perplexity is standard Performance Measure

- measures fitness of a given LM to explain a provided test text
- test text is a set of sentences $(s_1, s_2, \ldots, s_n)$

$$
\begin{aligned}
PP_T(M) &= 2^{H_T(M)} \\
H_T(M) &= -\frac{1}{n}\sum_{i=1}^{n} \log p(s_i)
\end{aligned}
$$

- $H_T(M)$ is the (cross)entropy of the LM M for text T
- $p(s_i)$ is the probability of the sentence s_i for the given LM
- perplexity values correspond intuitively to the number of relevant words in the lexicon
- the lower the value, the better the model

5. Recognition System Performance

The following two Performance Measures are used

- sentence recognition rate
- word level accuracy

5. Recognition System Performance

The following two Performance Measures are used

- sentence recognition rate
- word level accuracy

Sentence Recognition Rate

- percentage of correctly recognized sentences

5. Recognition System Performance

The following two Performance Measures are used

- sentence recognition rate
- word level accuracy

Sentence Recognition Rate

- percentage of correctly recognized sentences
- sentence is considered to be correct if recognition result exactly matches the transcription

5. Recognition System Performance

The following two Performance Measures are used

- sentence recognition rate
- word level accuracy

Sentence Recognition Rate

- percentage of correctly recognized sentences
- sentence is considered to be correct if recognition result exactly matches the transcription

Word Level Accuracy

- measures the accuracy of the recognition result

5. Recognition System Performance

The following two Performance Measures are used

- sentence recognition rate
- word level accuracy

Sentence Recognition Rate

- percentage of correctly recognized sentences
- sentence is considered to be correct if recognition result exactly matches the transcription

Word Level Accuracy

- measures the accuracy of the recognition result
- percentage of correctly recognized words minus the percentage of inserted words compared to the transcription
  (H - I) / N

6. Experimental Setup

The Recognizer

- text line normalization (slant, writing regions, contrast)

6. Experimental Setup

The Recognizer

- text line normalization (slant, writing regions, contrast)
- feature extraction (sliding window)

6. Experimental Setup

The Recognizer

- text line normalization (slant, writing regions, contrast)
- feature extraction (sliding window)
- HMM for each character
  - linear topology
  - variable character length
  - multi Gaussian

6. Experimental Setup

The Recognizer

- text line normalization (slant, writing regions, contrast)
- feature extraction (sliding window)
- HMM for each character
  - linear topology
  - variable character length
  - multi Gaussian
- Baum-Welch training

6. Experimental Setup

The Recognizer

- text line normalization (slant, writing regions, contrast)
- feature extraction (sliding window)
- HMM for each character
  - linear topology
  - variable character length
  - multi Gaussian
- Baum-Welch training
- Viterbi decoding

6. Experimental Setup

Assumptions

- recognition of isolated sentences
- closed lexicon: no new words in test set

6. Experimental Setup

Assumptions

- recognition of isolated sentences
- closed lexicon: no new words in test set

Training of Character HMM

- IAM database
- 5,799 lines of handwritten text (33,993 words)
- written by 448 persons

6. Experimental Setup

Assumptions

- recognition of isolated sentences
- closed lexicon: no new words in test set

Training of Character HMM

- IAM database
- 5,799 lines of handwritten text (33,993 words)
- written by 448 persons

Training of Baseline Bigram/Trigram LM

- LOB corpus (tagged)
- 50’000 sentences (excluding test/validation set)

6. Experimental Setup

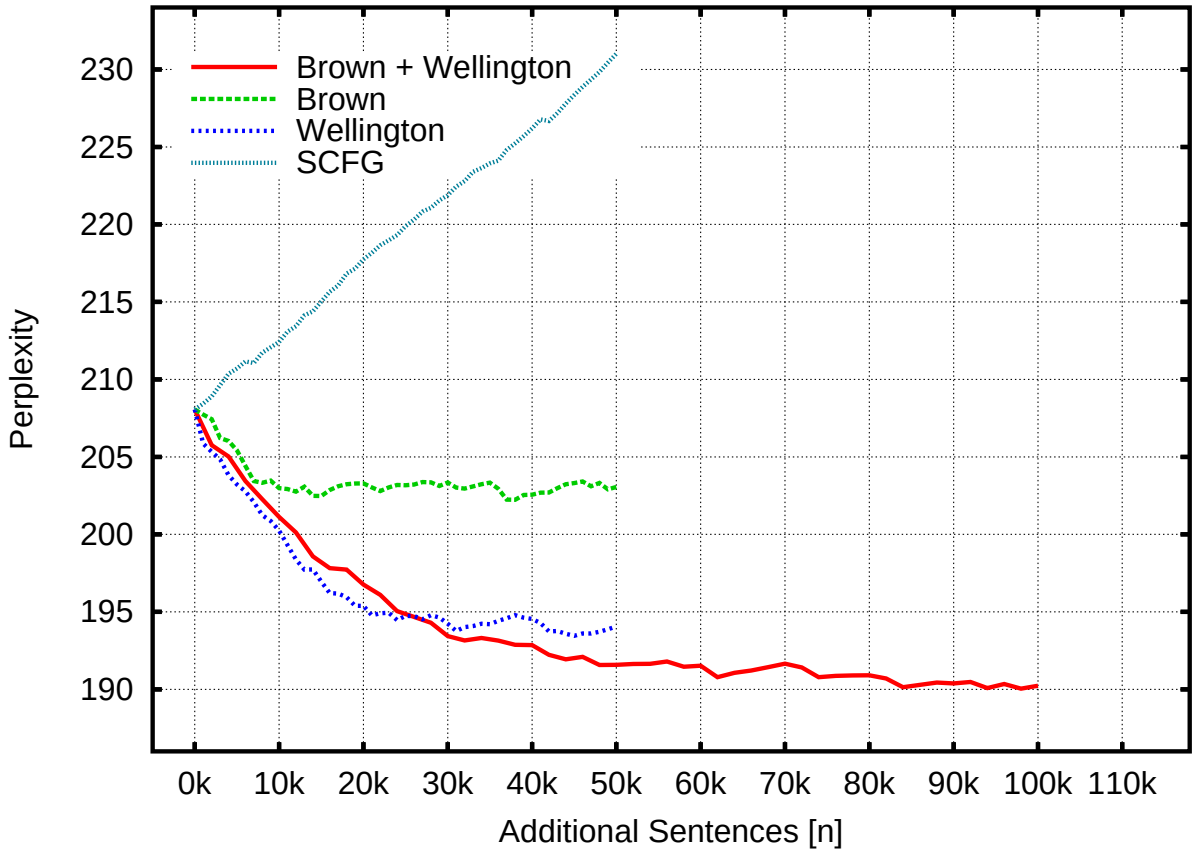
Recognition Task* Specifications

	validation	test
sentences	200	200
text lines	582	575
words	4,094	3,956
lexicon	8,827	8,821
writers	100	100

* writer independent: different writers in training, validation and test set

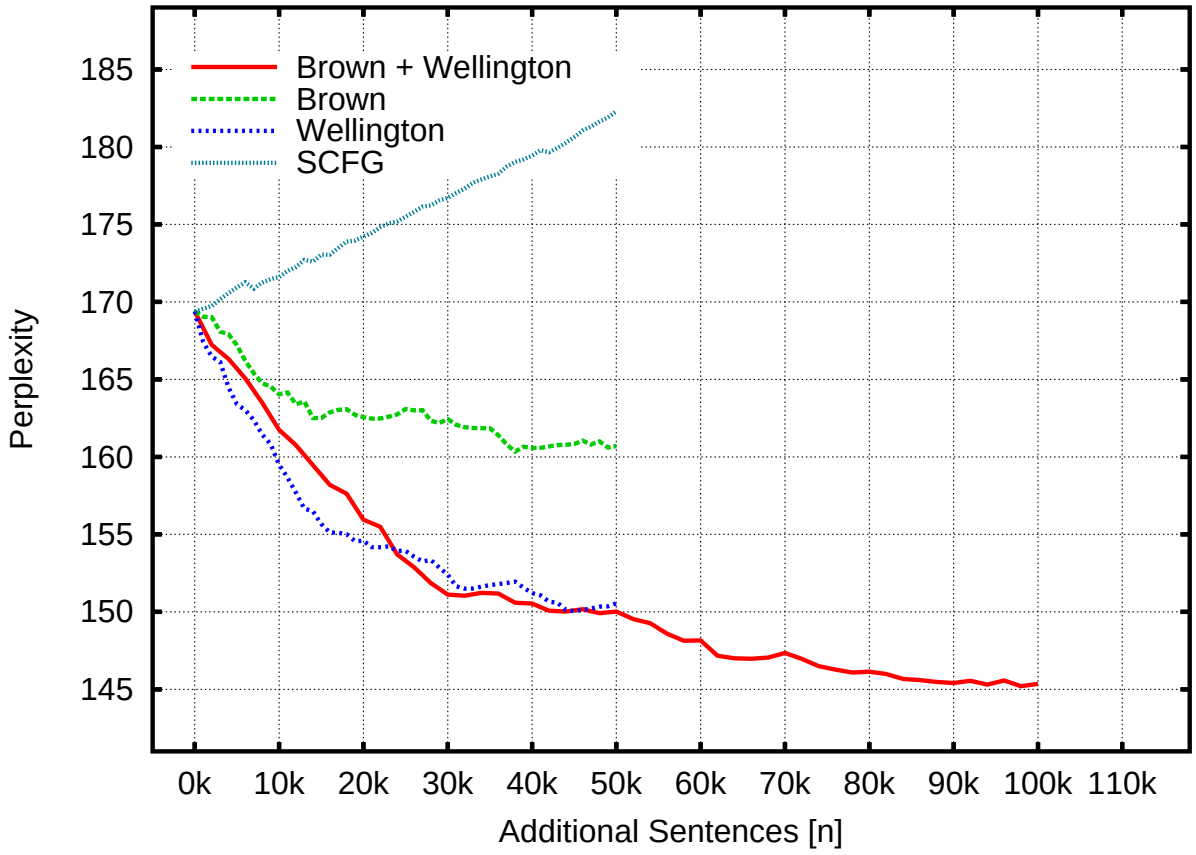
6. Optimization

Bigram LM: Optimizing the Perplexity



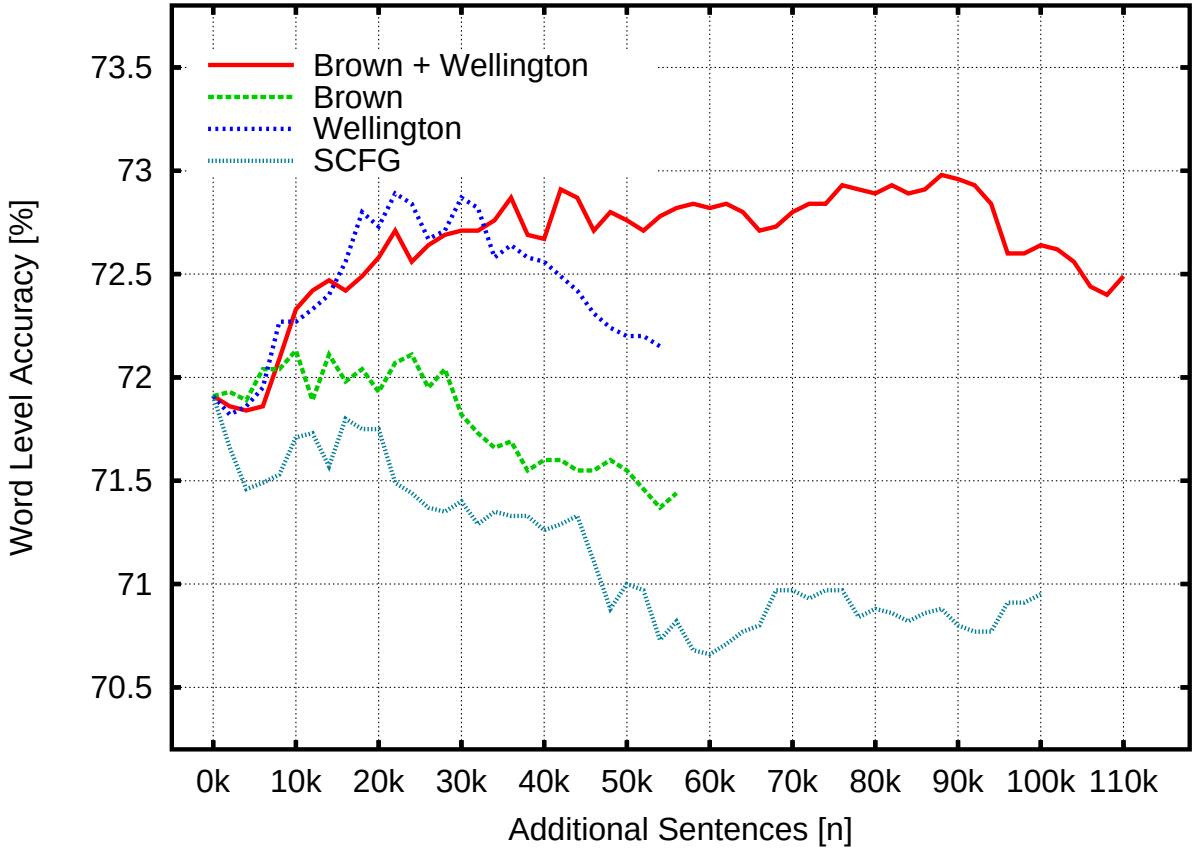
6. Optimization

Trigram LM: Optimizing the Perplexity



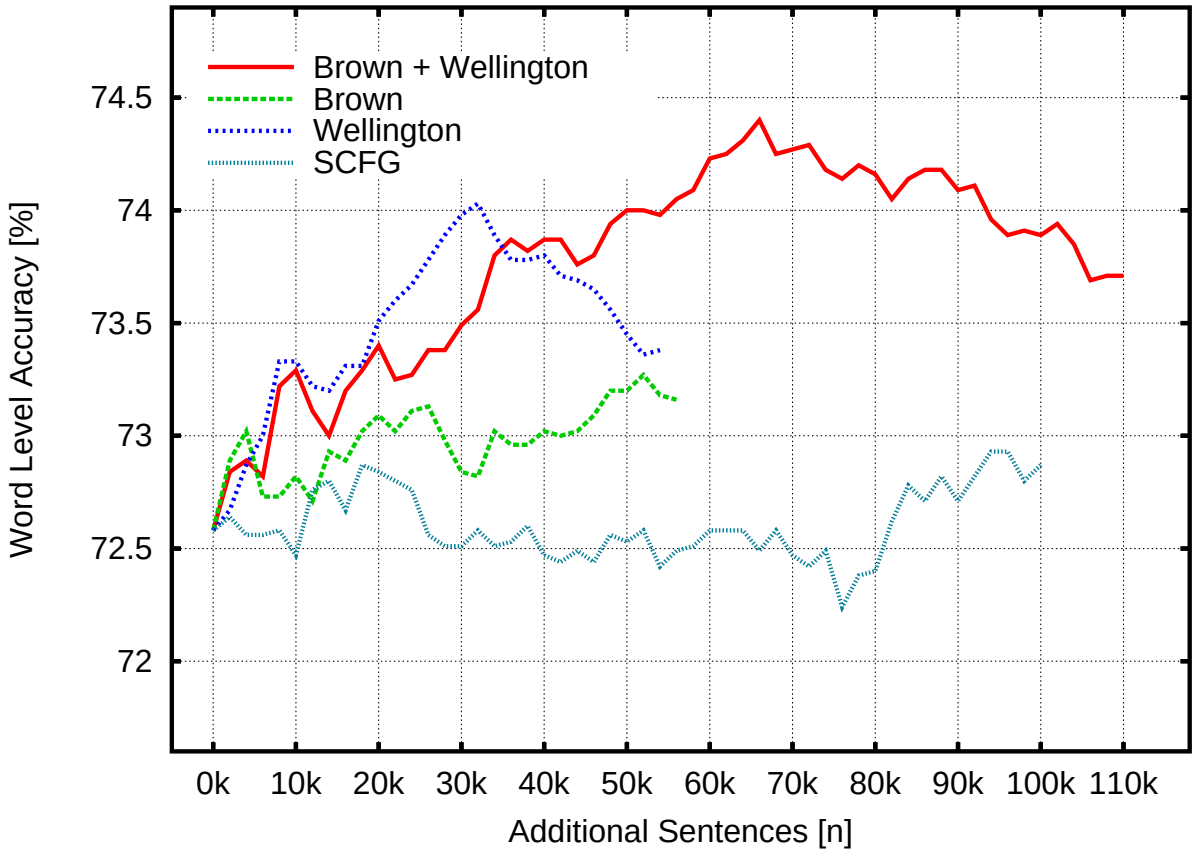
6. Optimization

Bigram LM: Optimizing the Word Level Accuracy (H-I)/N



6. Optimization

Trigram LM: Optimizing the Word Level Accuracy (H-I)/N



6. System Evaluation

Test Set Results

Language Model	Sentence Rec. Rate	Word Level Acc.
Bigram	11%	76.3%

6. System Evaluation

Test Set Results

Language Model	Sentence Rec. Rate	Word Level Acc.
Bigram	11%	76.3%
Trigram	12%	78.2%

6. System Evaluation

Test Set Results

Language Model	Sentence Rec. Rate	Word Level Acc.
Bigram	11%	76.3%
Trigram	12%	78.2%
Smoothed* Trigram	14%	79.9%

* LOB plus first 65’000 sentences from combined Brown + Wellington corpus

7. Conclusions

Main Conclusions

- texts to train LM should match recognition task

7. Conclusions

Main Conclusions

- texts to train LM should match recognition task
- random sentences produced by SCFG did not help

7. Conclusions

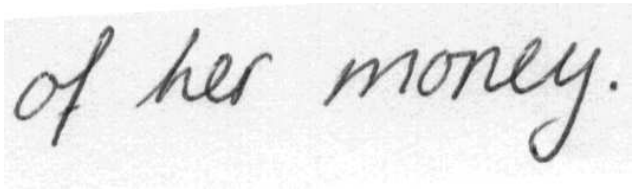
Main Conclusions

- texts to train LM should match recognition task
- random sentences produced by SCFG did not help
- trigram LM outperformed bigram LM significantly

Thank You

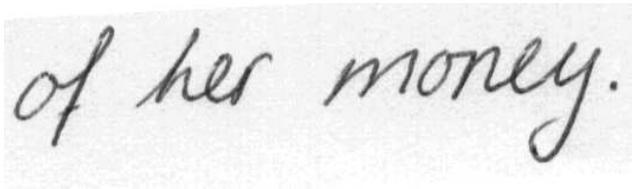
Appendix The Recognizer

Text Line Normalization

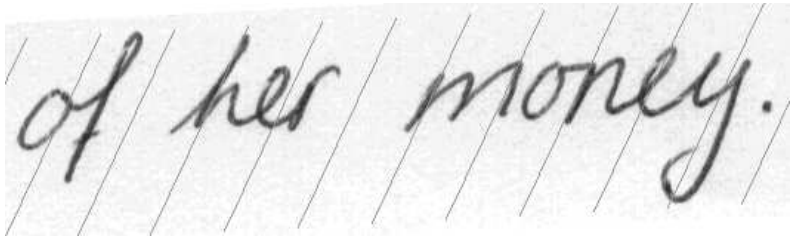


Appendix The Recognizer

Text Line Normalization

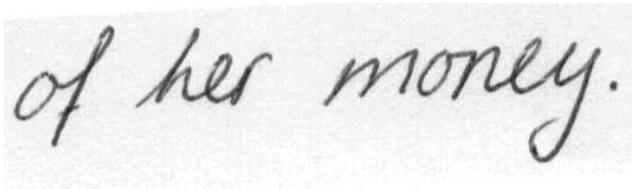


slant estimation

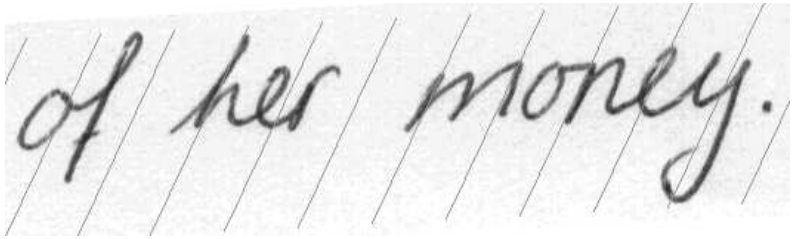


Appendix The Recognizer

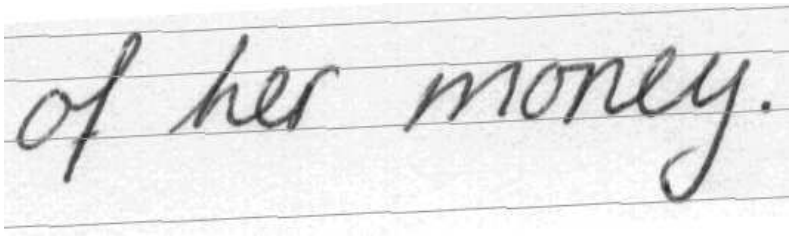
Text Line Normalization



slant estimation

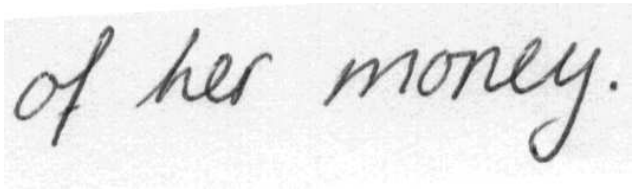


reference line estimation

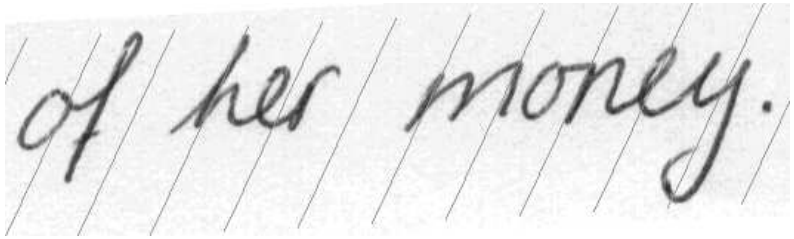


Appendix The Recognizer

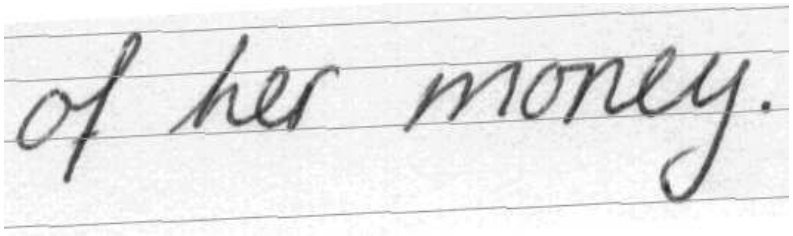
Text Line Normalization



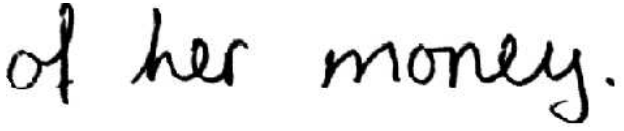
slant estimation



reference line estimation

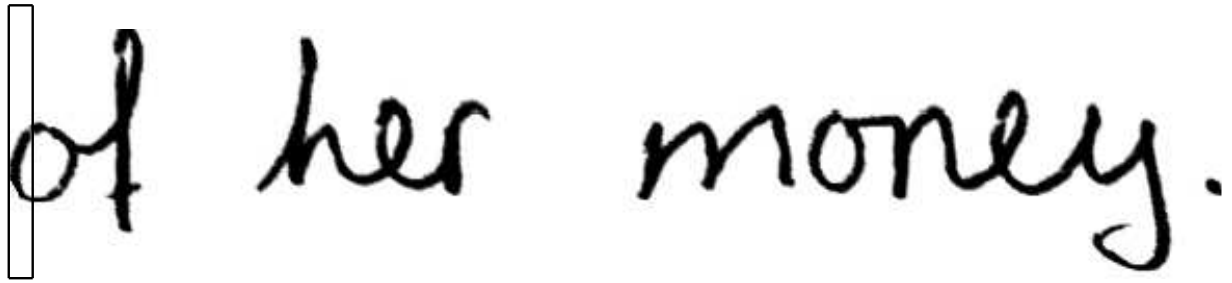


normalized text line



Appendix The Recognizer

Extraction of the Feature Vector Sequence



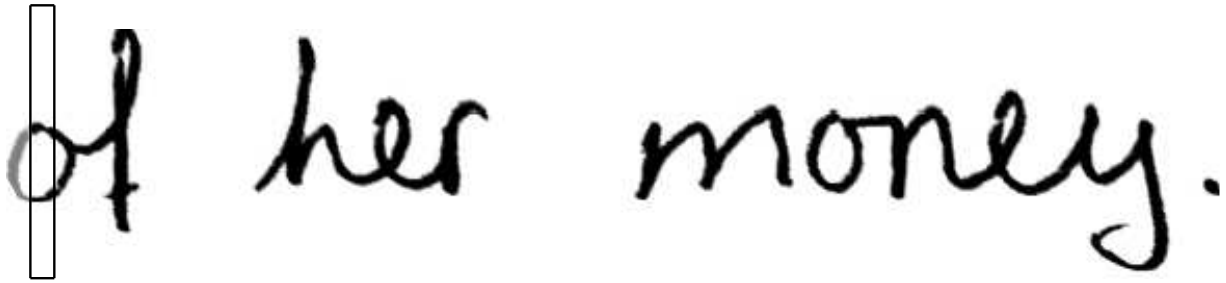
$$\downarrow$$

$$\begin{pmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1m} \end{pmatrix}$$

$$X = (X_1,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence



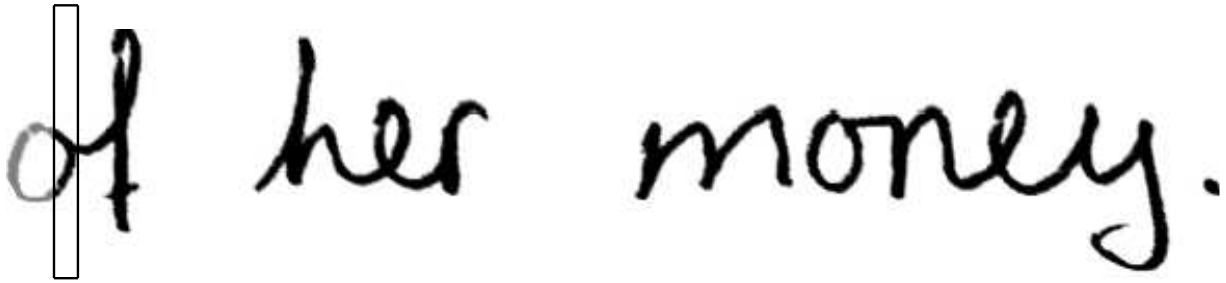
$$\downarrow$$

$$\begin{pmatrix} x_{21} \\ x_{22} \\ \vdots \\ x_{2m} \end{pmatrix}$$

$$X = (X_1, X_2,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence

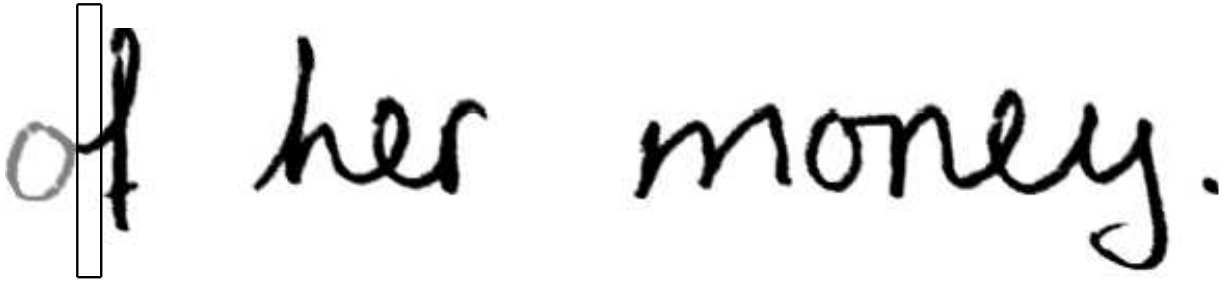


$$\begin{pmatrix} x_{31} \\ x_{32} \\ \vdots \\ x_{3m} \end{pmatrix}$$

$$X = (X_1, X_2, X_3,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence

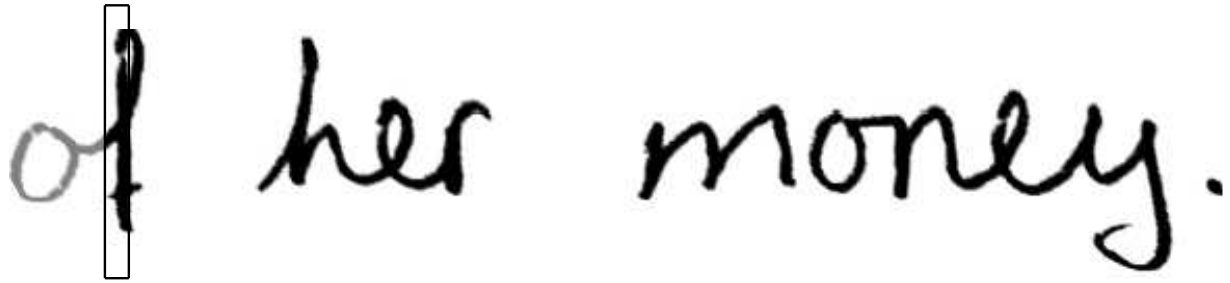


$$\begin{pmatrix} x_{41} \\ x_{42} \\ \vdots \\ x_{4m} \end{pmatrix}$$

$$X = (X_1, X_2, X_3, X_4,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence

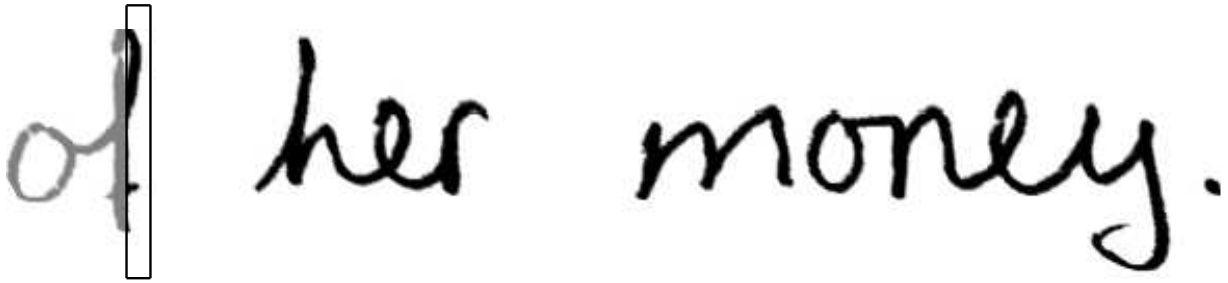


$$\begin{pmatrix} x_{51} \\ x_{52} \\ \vdots \\ x_{5m} \end{pmatrix}$$

$$X = (X_1, X_2, X_3, X_4, X_5,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence

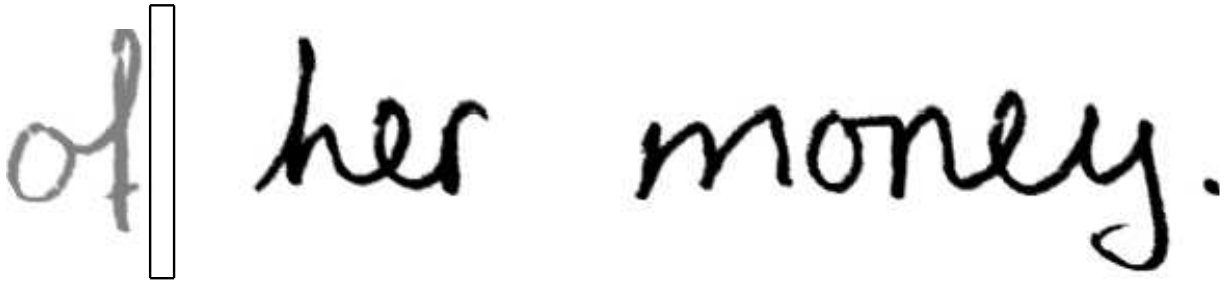


$$\begin{pmatrix} x_{61} \\ x_{62} \\ \vdots \\ x_{6m} \end{pmatrix}$$

$$X = (X_1, X_2, X_3, X_4, X_5, X_6,$$

Appendix The Recognizer

Extraction of the Feature Vector Sequence



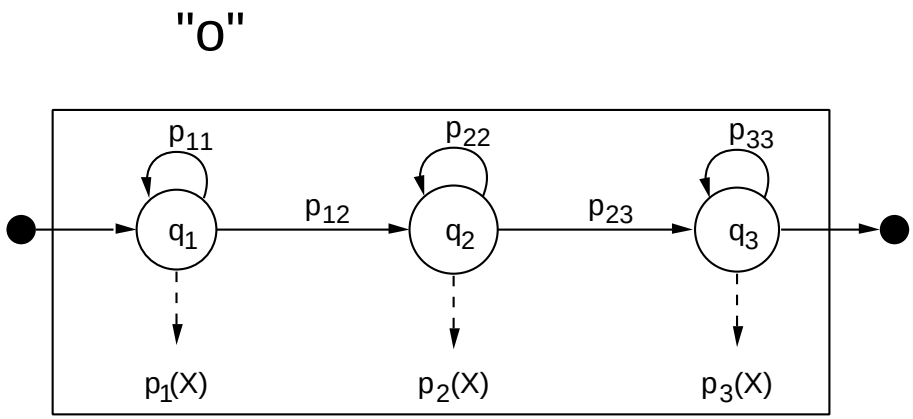
$$\begin{pmatrix} x_{71} \\ x_{72} \\ \vdots \\ x_{7m} \end{pmatrix}$$

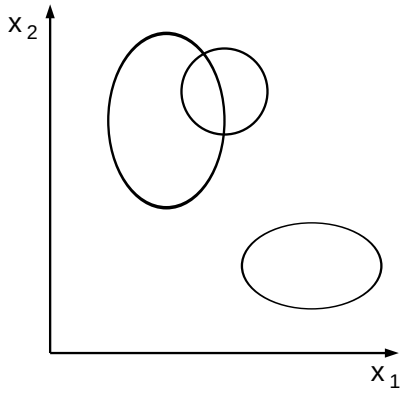
$$X = (X_1, X_2, X_3, X_4, X_5, X_6, X_7, \ldots$$

Appendix The Recognizer

Viterbi Decoding

$$(X_1, X_2, X_3, \ldots$$

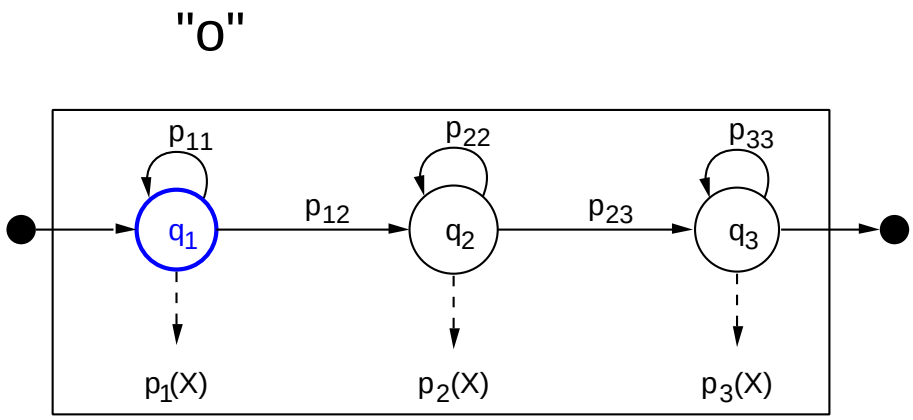


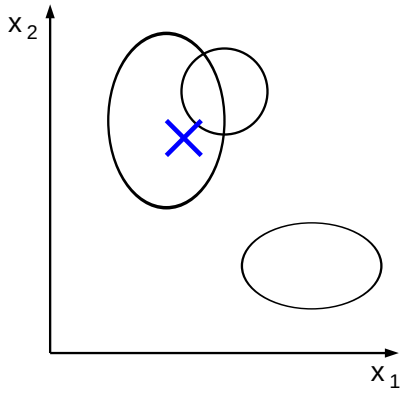


Appendix The Recognizer

Viterbi Decoding

$(X_1, X_2, X_3, \ldots$

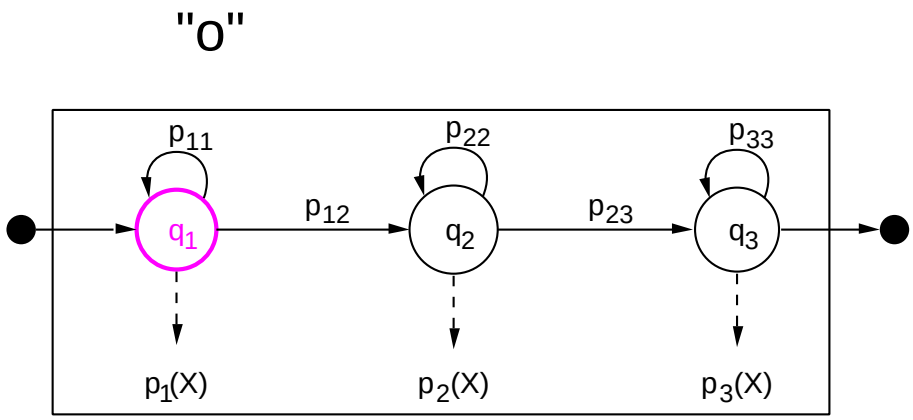


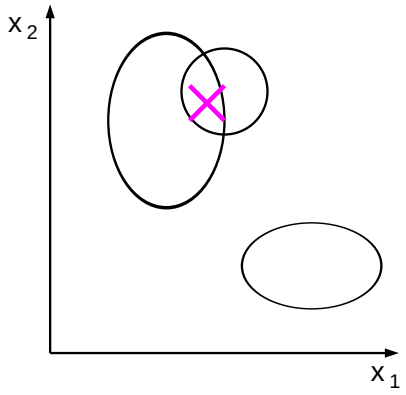


Appendix The Recognizer

Viterbi Decoding

$$(X_1, X_2, X_3, \ldots$$

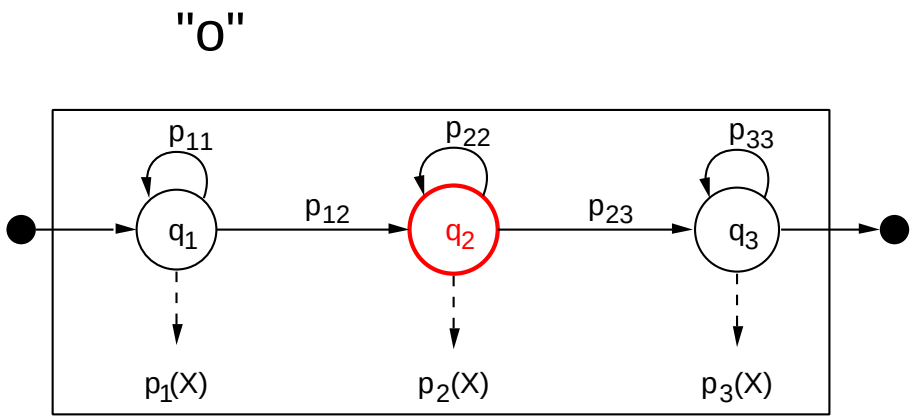


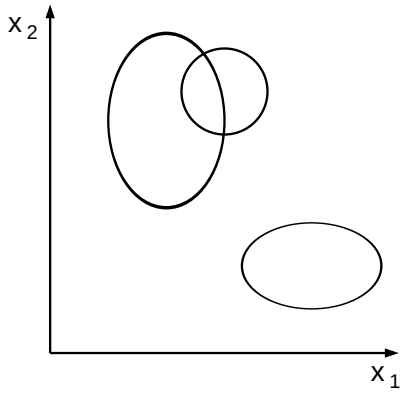


Appendix The Recognizer

Viterbi Decoding

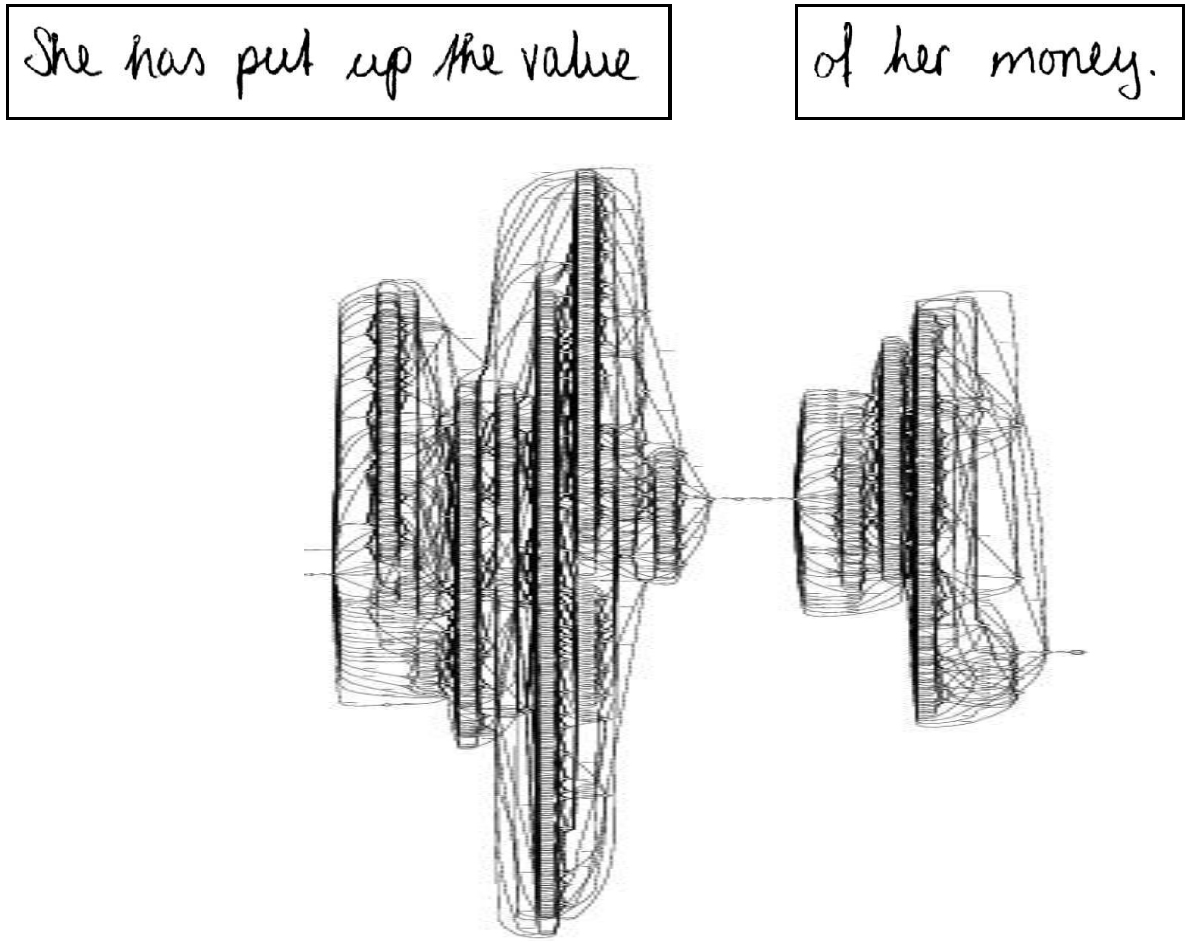
$$(X_1, X_2, X_3, \ldots$$





Appendix The Recognizer

Recognition Lattice Generation



Appendix The Recognizer

Sublattice of 2nd Line