# Improved Dehyphenation of Line Breaks for PDF Text Extraction

Examiner: Prof. Dr. Hannah Bast

Adviser: Claudius Korzen

Albert-Ludwigs-Universität Freiburg

**Mari Sverresdatter Hernæs**

Department of Computer Science
Chair of Algorithms and Data Structures

UNI
FREIBURG

# Problem 1/3

- Motivation
  - PDF only stores information about individual characters
  - This makes it difficult to extract text correctly

In this paper, we describe our approach how to construct a high-
quality benchmark from both TeX and PDF data. We create a bench-
mark from 12,000 scientific articles.

- Problem
  - Words split between two lines are two separate parts
  - How to assemble these parts?

# Problem 2/3

- There are four approaches

1. Do not merge the parts

   bench-   mark, high-   quality

2. Always merge the parts without a hyphen

   **benchmark,** highquality

3. Always merge the parts with a hyphen

   bench-mark, **high-quality**

4. Merge the parts with or without a hyphen depending on whether the actual word has a hyphen or not

   **benchmark, high-quality**

# Problem 3/3

- **Problem definition**
  - Given a sequence of characters, ordered by left-to-right reading order, with a line-break hyphen on position i
    - $S = [c_1, c_2, \ldots, c_{i-1}, — , c_{i+1}, \ldots, c_n]$
  - Decide if this hyphen should be deleted or kept…
    - $S^* = [c_1, c_2, \ldots, c_{i-1}, c_{i+1}, \ldots, c_n] \vee S^* = [c_1, c_2, \ldots, c_{i-1}, — , c_{i+1}, \ldots, c_n]$
  - …so that S* is identical to the expected output

UNI FREIBURG

Questions?

# Solution 1/9

- Three approaches
  - Vocabulary-based baseline
  - Logistic regression
  - Language model

# Solution 2/9

- First approach, **vocabulary-based baseline**
  - Look up the word parts with and without a hyphen
  - and choose the most common word
  - for S = [e,l,e,-,p,h,a,n,t], look up ‘ele-phant’ and ‘elephant’ ‘ele-phant’ is not a word - remove the hyphen!
  - Works well for most cases.. but not for:
    - Misspellings *(such as elhe-phant)*
    - Plural or conjugated words *(elephant**s**, run**ning**)*, unless these are explicitly added to the vocabulary
    - Not always for multiple correct spellings *(e-mail, email)*

# Solution 3/9

- First approach, **vocabulary-based baseline**
  - Not always for words with two different meanings

motion planning and control.

Several simulation cases are set up to evaluate the effectiveness of the LSM method. In Section 2, the concept of LSM is introduced and the procedure to obtain the $LSM_m$ is given. Based on a mechanical model shown in Section 3 as well as the gait and computed leg-end forces described in Section 4, simulation results are obtained in Section 5. Four conditions are considered: walking on flat terrain without or with external forces; walking on slope with or without external forces. Conclusions are given in Section 6.

## 2 Leg-end Supporting Moment (LSM)

### 2.1 The Concept of LSM

As shown in Fig. 1, during the walk of a $n$-legged robot, when legs $i$ and $u$, $v$ are the supporting legs, the moment generated by the leg-ends around the line $uv$ can be expressed as:

$$M_{uv} = \sum_{\substack{i \le n \\ i \ne u \ne v \ne j}} \left\{ \left[ \left( \frac{|r_{iu}||r_{iv}|\sin\alpha}{|r_{iu} - r_{iv}|} \hat{r}_{i\perp uv} \right) \times F_i \right] \bullet \hat{r}_{uv} \right\}. \quad (3)$$

the sum of the generaliz

acceleration forces of th

in Eqs. (3-4) are the forc

Furthermore, the relatio

the LSM method is t

external forces and acce

by the forces acting on leg-ends ($F_i$, $F_u$, $F_v$). As mentioned in reference [6], ($M$, $F$) can be obtained by using the leg-end forces, which can be sensed by the force sensors mounted on leg-ends. But in the SLM method, the leg-end forces are directly used to compute the stability margin. It can be then concluded that the LSM method is more direct and practical than the TSM method.

Find

legend

Previous Next

### 2.2 Computation of LSM

Four coordinate frames are defined in Fig. 1 for modeling purposes: the *Terrain Fixed Frame* (TFF) ($O_t$ $X_t$ $Y_t$ $Z_t$), the *Body-Terrain Frame* (BTF) ($O_{bt}$ $X_{bt}$ $Y_{bt}$ $Z_{bt}$), the *Body Fixed Frame* (BFF) ($O_b$ $X_b$ $Y_b$ $Z_b$) and the *Hip-Body frame* (HBF) ($O_{i0}$ $X_{i0}$ $Y_{i0}$ $Z_{i0}$). The TFF is fixed on the ground. The orientation of the BTF, which is a point-attached frame, is always the same as the TFF while the

  - the sentence should be "…can be obtained using the **leg-end** forces…."
  - not "…can be obtained using **the legend forces**…"

# Solution 4/9

- Second approach, **logistic regression**
  - Goal: recognise misspellings, conjugations and unknown words
  - Statistical classification: “learn” to recognise hyphenated words

    For example, ‘high’ is often followed by a hyphen

    (but not always, such as in ‘higher’)
  - Features for the *prefix* (**before** the hyphen) and for the *suffix* (**after** the hyphen)
  - Three last bigrams of the prefix and three first of the suffix

    *high- quality,* bigrams: **hi**,**ig**,**gh**, **qu**,**ua**,**al**
  - isUppercase, isDigit, hasHyphen, lowercase (‘high’ and ‘quality’), word-shape (xxxx-xxxxxxx)

# Solution 5/9

- Third approach, **bi-LSTM Language Model**
  - bidirectional Long short-term memory (bi-LSTM) Network
  - Can learn to connect information in (for example) a sentence and make predictions, despite large distances to the necessary information.

    "I grew up in Norway…. I speak fluent *Norwegian*"

  - We use a bi-LSTM on the **character** level
  - Designed by Matthias Hertel for tokenization repair (a special type of spelling correction)
  - Finds the most probable character (not word) at each position

# Solution 6/9

- Third approach, **bi-LSTM Language Model**
  - Procedure:
    - Make two sentences; one with the line break hyphen, one without

      1)…to make a high-quality…2)…to make a highquality…
    - Predict the two sentences **separately**
    - Sum up the probabilities of (each character in) 'high-quality' in the first sentence and 'highquality' in the second sentence

# Solution 7/9

- Techniques, **data sets**
  - Sentences from ClueWeb12, Ontonotes 5.0 and Wikipedia
  - We inserted hyphens with the TeX hyphenation procedure
  - Roughly one or two new hyphens per sentence

  For example: “we **de·scribe** a high-quality **bench·mark**” for “we describe a high-quality benchmark”

  <hyphenated sentence> TAB <original sentence>

| Hyphenated data set | ClueWeb12 Large | ClueWeb12 Small | Ontonotes Release 5.0 | Wikipedia Extract |
|---|---|---|---|---|
| size | 104GB | 152MB | 23MB | 63MB |
| sentences | 2·370,958,448 | 2·529,933 | 2·117,450 | - |
| words | 2·10,182,157,839 | 2·14,553,968 | 2·2,234,528 | 2·5,294,052 |
| new hyphens | 543,538,945 | 776,700 | 126,028 | 300,009 |

# Solution 8/9

- Techniques, **vocabulary**
  - We needed a good vocabulary for the baseline
  - Solution: take **all words** from either ClueWeb12 or Ontonotes
  - And register their **frequencies**
  - Problem: many words have numbers

    For example: 17-years-old, 19th-century
  - But we wanted reasonable frequency scores
  - Therefore, we replaced all numbers with X (chi)

| Word | Frequency |
|---|---|
| the | 576565079 |
| and | 289254192 |
| of | 277112975 |
| in | 228629057 |
| to | 210044965 |
| a | 169434632 |
| for | 88935175 |
| is | 85552512 |
| … | … |
| business | 4317176 |
| XXth | 4286474 |
| team | 4281688 |
| ... | ... |
| XXXXs | 1604329 |
| valley | 1600795 |
| … | … |

# Solution 9/9

- Techniques, **vocabulary**
  - There was an obvious risk of vocabulary bias
  - We were going to evaluate the baseline on ClueWeb12, with a vocabulary made from the same sentences
  - Another collection of words: from IMDB (Maas et Al.'s Large Movie Review Dataset)
  - This vocabulary did **not** include words with numbers

Questions?

# Evaluation 1/5

- Setup, **ground truth**
  - Remember: the data set format was

    <hyphenated sentence> TAB <original sentence>
  - We looked at the words with **new** hyphens in the hyphenated sentence (bench·mark)
  - Ground truth: the word in the original sentence (benchmark)
  - Notice: nothing guaranteed that the ground truth did not contain any unusual spellings or misspellings.
  - Also: there were many words with several valid spellings

    For example: e-mail / email

# Evaluation 2/5

- Setup, **metrics**
  - Common evaluation metrics: Precison, Recall, F1 score
  - But we needed good positive **and** negative classifications

| | expected 0 | expected 1 | |
|---|---|---|---|
| predicted 0 | 50.000 (TN) | 100 (FN) | |
| predicted 1 | 1000 (FP) | 500 (TP) | Precision<br>33.3%<br>TP/(TP+FP) |
| | | Recall<br>83.3%<br>TP/(TP+FN) | F1-Score<br>47.2%<br>2·P·R/(P+R) |

# Evaluation 3/5

- Setup, **metrics**
  - We used: Accuracy, Recall (accuracy for expected hyphen), Specificity (accuracy for expected non-hyphen), and balanced Accuracy (bACC)

| | expected 0 | expected 1 | |
|---|---|---|---|
| predicted 0 | 50.000 (TN) | 100 (FN) | Accuracy 97.8% |
| predicted 1 | 1000 (FP) | 500 (TP) | |
| | Specificity 98.0% TN/(TN+FP) | Recall 83.3% TP/(FN+TP) | bACC 90.7% (S+R) / 2 |

# Evaluation 4/5

- Main results, **ClueWeb12 Small**

776,700 hyphenated words. 13,112 expected hyphen

| Model | Version | Accuracy | Specificity | Recall | bACC |
|---|---|---|---|---|---|
| Baseline | Vocabulary Ontonotes | 98.79% | **99.91%** | 33.83% | 66.87% |
| Logistic regression | 11.64% expected hyphen | 98.75% | 98.98% | **85.78%** | **92.38%** |
| bi-LSTM | - | **99.25%** | 99.56% | 80.71% | 90.14% |

# Evaluation 5/5

- Main results, **“maximum achievable result”**

| Model | Dataset | Accuracy | Specificity | Recall | bACC |
|---|---|---|---|---|---|
| Baseline | ClueWeb12 Large | 99.66% | 99.94% | 83.55% | 91.74% |
| Logistic regression | Wikipedia Extract | 99.52% | 99.60% | 95.01% | 97.31% |

- Typical errors:
  - Several valid spellings: email / e-mail, southeast / south-east, nonprofit / non-profit
  - Abbreviations: ADAC, APS-C
  - Words in camelCase: LocalWiki

Thank you for your attention

# Liang's Hyphenation Algorithm

- Odd numbers: hyphen
- Even number: no hyphen

```
. h y   p h e   n   a   t e .
          h e 2 n
          h e   n   a 4
          h e   n 5 a   t
  h y 3 p h
              1 n   a
                n 2 a   t
                      4 t e .
-------------------------------
. h y 3 p h e 2 n 5 a 4 t e .
  h y - p h e   n - a   t e
```

- Top 10 mistakes for baseline with ClueWeb vocabulary on ClueWeb12 Large 1/2

False negative cases (predicted merge, expected hyphen):

| frequency | mistake | confidence | score |
|---|---|---|---|
| 66522 | e·mail | 0.73 | |
| 35007 | wal·mart | 0.52 | |
| 18053 | long·time | 0.61 | |
| 14941 | on·line | 0.97 | |
| 12647 | plug·in | 0.55 | |
| 12503 | south·east | 0.89 | |
| 11508 | line·up | 0.7 | |
| 9920 | north·west | 0.92 | |
| 9682 | north·east | 0.89 | |
| 8278 | south·west | 0.92 | |

- Top 10 mistakes for baseline with ClueWeb vocabulary on ClueWeb12 Large 2/2

False positive cases (predicted hyphen, expected merge):

| frequency | mistake | confidence | score |
|---|---|---|---|
| 15057 | non·profit | 0.53 | |
| 12649 | best·selling | 0.59 | |
| 6483 | post·war | 0.56 | |
| 5374 | first·hand | 0.54 | |
| 4279 | non·stop | 0.6 | |
| 4058 | on·site | 0.68 | |
| 3607 | long·standing | 0.6 | |
| 3487 | semi·final | 0.53 | |
| 3114 | re·election | 0.7 | |
| 3019 | spider·man | 0.72 | |

- Top 10 correct for baseline with ClueWeb vocabulary on ClueWeb12 Large 1/2

True positive cases (predicted hyphen, expected hyphen):

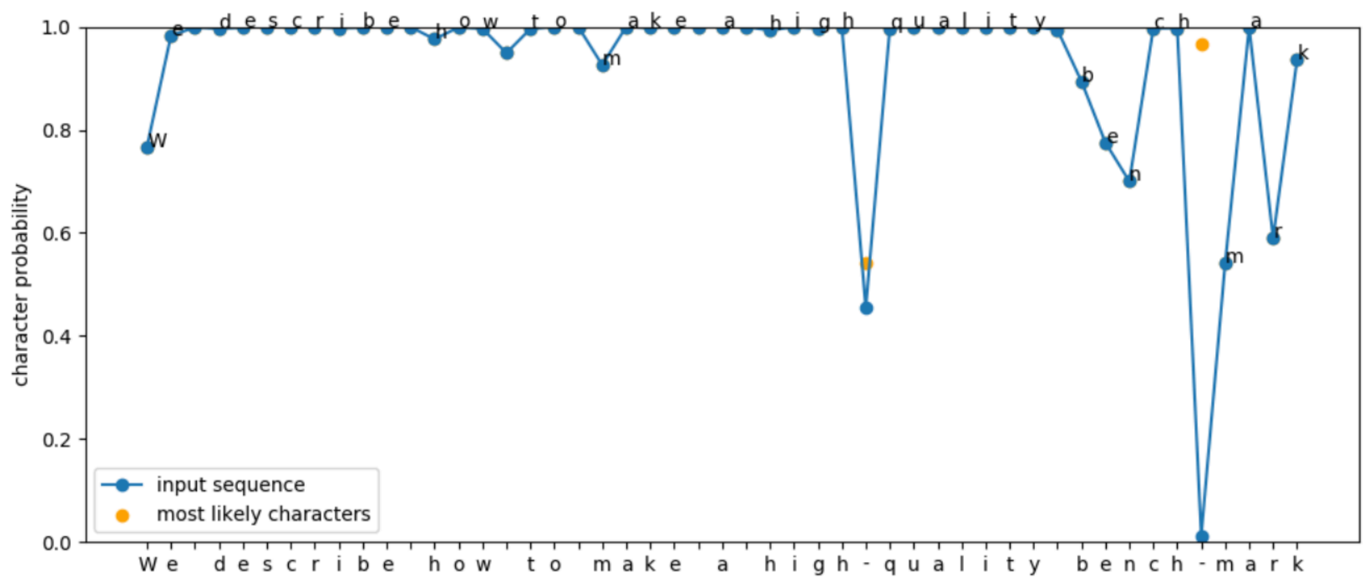
| frequency | mistake | confidence | score |
|---|---|---|---|
| 56985 | so·called | 1.0 | |
| 55825 | long·term | 0.99 | |
| 53386 | well·known | 0.99 | |
| 49720 | wi·fi | 0.59 | |
| 31252 | full·time | 0.95 | |
| 27220 | blu·ray | 0.93 | |
| 26802 | real·time | 0.92 | |
| 26410 | built·in | 0.99 | |
| 26297 | all·star | 0.98 | |
| 24772 | all·time | 0.99 | |

- Top 10 correct for baseline with ClueWeb vocabulary on ClueWeb12 Large 2/2

True negative cases (predicted merge, expected merge):

| frequency | mistake | confidence | score |
|---|---|---|---|
| 2878874 | oth·er | 1.0 | |
| 1807802 | af·ter | 1.0 | |
| 1714932 | peo·ple | 1.0 | |
| 1687206 | unit·ed | 1.0 | |
| 1156943 | be·fore | 1.0 | |
| 1069576 | be·ing | 1.0 | |
| 990356 | be·tween | 1.0 | |
| 982421 | be·cause | 1.0 | |
| 908616 | coun·try | 1.0 | |
| 906160 | dur·ing | 1.0 | |

- second approach, **bi-LSTM Language Model**



# Backup slides

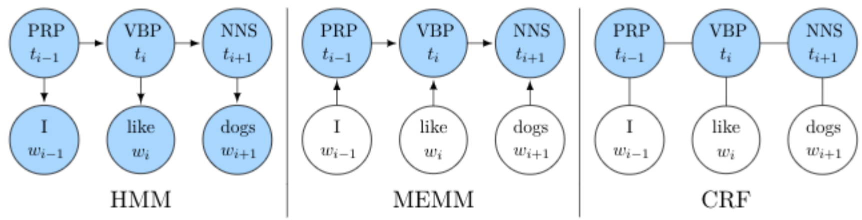



Figure 3.5: **A graphical representation of the sequence models** This figure shows the difference between HMM, MEMM and linear-chain CRF. The first two models are Bayesian networks, with arrows indicating conditional dependencies; the third is an undirected graphical model. A filled circle indicates a variable generated by the model.

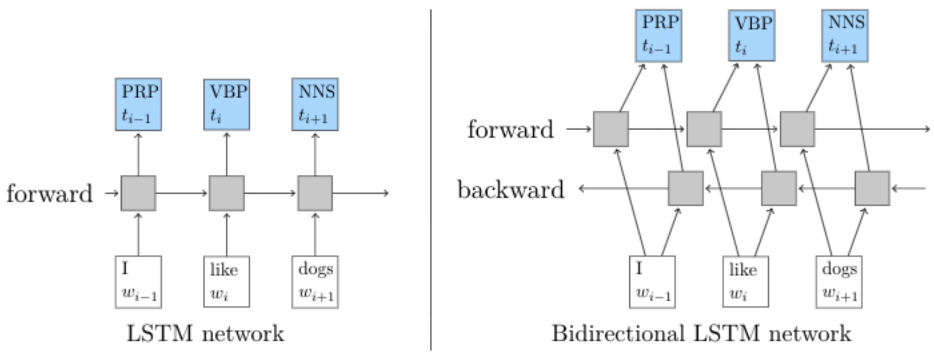



Figure 3.6: **LSTM Networks**. This example, inspired by Huang et Al. [28], shows a unidirectional and a bidirectional LSTM. Each grey box represents a LSTM memory cell. These cells can make use of long-range dependencies in the data.

| Model | Version | Accuracy | Specificity | Recall | bACC |
|---|---|---|---|---|---|
| Baseline | Vocabulary Ontonotes | 98.79% | **99.91%** | 33.83% | 66.87% |
| Logistic regression | 11.64% expected hyphen | 98.75% | 98.98% | **85.78%** | **92.38%** |
| Logistic regression | 4.42% expected hyphen | 99.14% | 99.55% | 75.38% | 87.47% |
| bi-LSTM | - | **99.25%** | 99.56% | 80.71% | 90.14% |