A General Software Readability Model

Jonathan Dorn
December 18, 2012
Software Maintenance Costs

- Maintenance may cost up to $9x$ all other development costs.

Reading and Maintenance

“A central activity in software maintenance is reading.”*

Reading and Maintenance

“A central activity in software maintenance is reading.”*

readability, *n.*

The **ease** with which a text may be **scanned** or **read**; the quality in a book, etc., of being easy to understand and enjoyable to read.

www.oed.com
Making Code More Readable

• **Programming languages**
  • Literate Programming (e.g. CWEB) [Knuth 1984]
  • Python [Van Rossum 1996]

• **Development Process**
  • Readability development phase [Elshoff & Marcotty 1982]
  • Readability review phase [Knight & Myers 1993]
  • Readability team [Haneef 1998]
Is It Working?

- **90%** in survey desire readability metric.
- **3rd** most requested metric.

Parallels: English Readability

- Flesch-Kincaid Grade Level
- Government mandated
  - Military manuals: 9th grade
    DOD MIL-M-28784B
  - Insurance policies: 10th grade
    C.R.S 10-16-107.3 (1)(a)
Flesch-Kincaid Grade Level

$$0.39 \left( \frac{\text{total words}}{\text{total sentences}} \right) + 11.8 \left( \frac{\text{total syllables}}{\text{total words}} \right) - 15.59$$

- Simple surface-level features (syllables, words, sentences).
- Weights calculated using regression analysis.
Flesch-Kincaid Grade Level

- Simple **surface-level features** (syllables, words, sentences).

- Weights calculated using regression analysis.

\[
0.39 \left( \frac{\text{total words}}{\text{total sentences}} \right) + 11.8 \left( \frac{\text{total syllables}}{\text{total words}} \right) - 15.59
\]
Flesch-Kincaid Grade Level

- Simple surface-level features (syllables, words, sentences).
- **Weights** calculated using regression analysis.

\[
0.39 \left( \frac{\text{total words}}{\text{total sentences}} \right) + 11.8 \left( \frac{\text{total syllables}}{\text{total words}} \right) - 15.59
\]
Learning a Readability Metric

Regression

Metric

\[ f = \beta_1 x_1 + \beta_0 \]
Source Code Readability

• Buse & Weimer 2008
  • 25 surface features (max line length, average whitespace, etc.)

• Posnett, et al. 2011
  $8.87 - 0.033 \text{(Halstead volume)} + 0.4 \text{(total lines)} - 1.5 \text{(token entropy)}$
Problem solved?
def handleBlockQuote(node):
    result = BlockQuoteDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleList(node):
    result = ListDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleListItem(node):
    result = ListItemDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleTable(node):
    result = TableDitem(node.nodeName)
    # Ignore table contents that are not tr
    result.children = [x
                      for x in processChildren(node) if x.type=="tr"]
    return result

def handleTr(node):
    result = TrDitem(node.nodeName)
    # Ignore tr contents that are not th or td
    result.children = [x
                       for x in processChildren(node) if x.type in ("th", "td")]
    return result
//float *attenuationIntegralPlaneArray_d;  // stores partial integral on planes parallel to the camera
//CUDA_SAFE_CALL(cudamalloc((void **) attenuationIntegralPlaneArray_d, img->dim[1]*img->dim[3]*sizeof(float)));

et_line_integral_attenuated_gpu_kernel <<<G1,B1>>> (*d_activity, *d_attenuation, currentCamPointer);
CUDA_SAFE_CALL(cudaThreadSynchronize());
}
Example Readability

```python
def handleBlockQuote(node):
    result = BlockQuoteDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleList(node):
    result = ListDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleListItem(node):
    result = ListItemDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleTable(node):
    result = TableDitem(node.nodeName)
    # Ignore table contents that are not tr
    result.children = [x for x in processChildren(node) if x.type=='tr']
    return result

def handleTr(node):
    result = TrDitem(node.nodeName)
    # Ignore tr contents that are not th or td
    result.children = [x for x in processChildren(node) if x.type in ('th', 'td')]
    return result
```

1  Readability Rating  5
   Humans

//float *attenuationIntegralPlaneArray_d;  //stores partial integral on planes parallel to the camera
//CUDA_SAFE_CALL(cudaMalloc((void **)&attenuationIntegralPlaneArray_d, img->dim[1]*img->dim[2]*sizeof(float)));

et_line_integral_attenuated_gpu_kernel <<<G1,B1>>> (*d_activity, *d_attenuation, currentCamPointer);
CUDA_SAFE_CALL(cudaThreadSynchronize());
Metric Mismatch

```python
def handleBlockQuote(node):
    result = BlockQuoteDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleList(node):
    result = ListDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleListItem(node):
    result = ListItemDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleTable(node):
    result = TableDitem(node.nodeName)
    # Ignore table contents that are not tr
    result.children = [x
        for x in processChildren(node) if x.type=='tr']
    return result

def handleTr(node):
    result = TrDitem(node.nodeName)
    # Ignore tr contents that are not th or td
    result.children = [x
        for x in processChildren(node) if x.type in ('th', 'td')]
    return result
```
What happened?
What Happened?

**Model**

• Character features only.
• Missing:
  • Structural patterns.
  • Line-to-line variation.
  • Spatial layout.
  • Syntax highlighting.

**Ground Truth**

• Small survey
  • 120 participants.
• Similar backgrounds
  • All UVa students.
• One programming language
  • Java.
• Short code samples
  • 4 – 13 lines.
General Readability Metric

1. New model.
   - Buse baseline features
   - Additional visual features
2. Ground truth from a large human study.
3. Combine and evaluate.
General Readability Metric

1. New model.
   - Buse baseline features
   - Additional visual features
2. Ground truth from a large human study.
3. Combine and evaluate
Visual Structural Features
Visual Structural Features

• Line-to-line periodic structure
  • E.g. indentation.

• How can we measure periodicity?
Fourier Series

- **Idea**: periodic functions can be written as the sum of a series of sines.

\[ \sum_{n=-\infty}^{\infty} c_n (\cos(nx) + i \sin(nx)) \]
Discrete Fourier Transforms

- The **Discrete Fourier Transform** (DFT) computes the coefficients.
- **Bandwidth**: the range of important coefficients.
- Common in signal processing.
Visual Structural Features

- Sample at each line.
- Take DFT of samples.
- Record bandwidth.
DFT Example (indentation)
DFT Example (indentation)
DFT Example (indentation)
DFT Example (indentation)
Spatial Layout Features

```python
# Update weights
if (i < H * W) {
    u = n / samples;
    int w = i % n + i;
    cudaFloat learningRate = UpdateLearningRate(learningRate, lastDeltaWithoutLearningMomentumW, deltaW, u, w, d);
    UpdateWeight(learningRate, momentum, deltaW, lastDelta, lastDeltaWithoutLearningMomentumW, weights, v);
}
if (i < H * W) {
    if (i == 0) {
        delta = 1.0 / samples;
    }
    cudaFloat learningRate = UpdateLearningRate(learningRate, lastDeltaWithoutLearningMomentumA, deltaA, i, u, w, d);
    UpdateWeight(learningRate, momentum, deltaA, lastDelta, lastDeltaWithoutLearningMomentumA, a, i);
}
# Update b
if (i == 0) {
    deltaB /= samples;
}
```
Spatial Layout Features

• Fraction of screen occupied by each color.
  • Count area highlighted with each color.
  • Record ratios between colors.

• Patterns of color.
  • Construct matrix of 0s (whitespace) and 1s (highlighted text).
  • Compute 2D DFT of matrix.
  • Record average bandwidth in X and Y dimensions.
DFT Example (comments)

def deltaM == vj(threadIdx.x) * hj(threadIdx.y) - Vj(threadIdx.x) * hj(threadIdx.y):
    
    if 1 < threadIdx.x < J:
        deltaM /= samples;
    int w = threadIdx.x;

cudafloat learningRate = UpdateLearningRate(learningRateDA, lastDeltaWithoutLearningMomentumA, deltaM, w, u, d);
    UpdateWeight(learningRate, momentum, deltaM, lastDeltaA, lastDeltaWithoutLearningMomentumA, weights, w);
    
    if threadIdx.x == 0:
        error += 1 == error;

    for update:
        if threadIdx.x == 0:
            deltaM /= samples;

        if threadIdx.x == 0:
            deltaM /= samples;

        if threadIdx.x == 0:
            deltaM /= samples;

        if threadIdx.x == 0:
            deltaM /= samples;
DFT Example (comments)
DFT Example (comments)
Alignment Features

• Identify 3+ lines with same token/token or token/whitespace transitions.

• Record number and length of matches.

wxSCHEDULER_DAILY
wxSCHEDULER_WEEKLY
wxSCHEDULER_MONTHLY
wxSCHEDULER_TODAY
wxSCHEDULER_TO_DAY
wxSCHEDULER_PREV
wxSCHEDULER_NEXT
wxSCHEDULER_PREVIEW
Linguistic Features

• Average dictionary words in identifiers
  • Underscore-separated words
  • CamelCase
  • Prefix and suffix
General Readability Metric

1. New model.
   - Buse baseline features
   - Additional visual features

2. Ground truth from a large human study.

3. Combine and evaluate.
Ground-Truth Survey

• Similar backgrounds (all UVa students).

• Single programming language (Java).

• Short code samples (4 – 13 lines).
Ground-Truth Survey

- Similar backgrounds (all UVa students).
- Single programming language (Java).
- Short code samples (4 – 13 lines).
Ground-Truth Survey

• Similar backgrounds (all UVa students).
• Diverse backgrounds:
  • Udacity students: beginners, professionals learning Python
  • reddit users: forum on programming

• Single programming language (Java).

• Short code samples (4 – 13 lines).
Ground-Truth Survey

• Diverse backgrounds: Udacity students, reddit users.

• **Single programming language** (Java).

• Short code samples (4 – 13 lines).
Ground-Truth Survey

- Diverse backgrounds: Udacity students, reddit users.
- Single programming language (Java).
- Multiple languages: Java, Python, CUDA.
- Short code samples (4 – 13 lines).
Ground-Truth Survey

• Diverse backgrounds: Udacity students, reddit users.

• Multiple languages: Java, Python, CUDA.

• Short code samples (4—13 lines).
Ground-Truth Survey

- Diverse backgrounds: Udacity students, reddit users.

- Multiple languages: Java, Python, CUDA.

- Short code samples (4—13 lines).
- Three code sample lengths: 10, 30, and 50 lines.
Code Samples

• Top-ten most recently updated projects in SourceForge.

• 360 total code samples.
  • 120 samples from each language.
  • 120 samples of each length.

• Survey takers rated 20 randomly selected samples.
  • Syntax pre-highlighted on server.
Survey Summary

- Over **76,000** individual ratings (**6x larger**).
- Over **2,600** completed surveys (**21x larger**).

<table>
<thead>
<tr>
<th>Category</th>
<th>Median (yrs)</th>
<th>&gt; 1 year</th>
<th>&gt; 5 years</th>
<th>&gt; 10 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>8</td>
<td>2598</td>
<td>1972</td>
<td>1242</td>
</tr>
<tr>
<td>Java</td>
<td>2</td>
<td>1896</td>
<td>646</td>
<td>247</td>
</tr>
<tr>
<td>Python</td>
<td>1</td>
<td>1655</td>
<td>253</td>
<td>59</td>
</tr>
<tr>
<td>CUDA</td>
<td>0</td>
<td>181</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>School</td>
<td>3</td>
<td>2118</td>
<td>522</td>
<td>28</td>
</tr>
<tr>
<td>Industry</td>
<td>3</td>
<td>1808</td>
<td>1091</td>
<td>655</td>
</tr>
</tbody>
</table>
Survey Summary

• Over 76,000 individual ratings (6x larger).
• Over 2,600 completed surveys (21x larger).

<table>
<thead>
<tr>
<th>Category</th>
<th>Median (yrs)</th>
<th>&gt; 1 year</th>
<th>&gt; 5 years</th>
<th>&gt; 10 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>8</td>
<td>2598</td>
<td>1972</td>
<td>1242</td>
</tr>
<tr>
<td>Java</td>
<td>2</td>
<td>1896</td>
<td>646</td>
<td>247</td>
</tr>
<tr>
<td>Python</td>
<td>1</td>
<td>1655</td>
<td>253</td>
<td>59</td>
</tr>
<tr>
<td>CUDA</td>
<td>0</td>
<td>181</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>School</td>
<td>3</td>
<td>2118</td>
<td>522</td>
<td>28</td>
</tr>
<tr>
<td>Industry</td>
<td>3</td>
<td>1808</td>
<td>1091</td>
<td>655</td>
</tr>
</tbody>
</table>
Survey Summary

- Over 76,000 individual ratings (6x larger).
- Over 2,600 completed surveys (21x larger).

<table>
<thead>
<tr>
<th>Category</th>
<th>Median (yrs)</th>
<th>&gt; 1 year</th>
<th>&gt; 5 years</th>
<th>&gt; 10 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>8</td>
<td>2598</td>
<td>1972</td>
<td>1242</td>
</tr>
<tr>
<td>Java</td>
<td>2</td>
<td>1896</td>
<td>646</td>
<td>247</td>
</tr>
<tr>
<td>Python</td>
<td>1</td>
<td>1655</td>
<td>253</td>
<td>59</td>
</tr>
<tr>
<td>CUDA</td>
<td>0</td>
<td>181</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>School</td>
<td>3</td>
<td>2118</td>
<td>522</td>
<td>28</td>
</tr>
<tr>
<td>Industry</td>
<td>3</td>
<td>1808</td>
<td>1091</td>
<td>655</td>
</tr>
</tbody>
</table>
Survey Summary

- Over 76,000 individual ratings (6x larger).
- Over 2,600 completed surveys (21x larger).

<table>
<thead>
<tr>
<th>Category</th>
<th>Median (yrs)</th>
<th>&gt; 1 year</th>
<th>&gt; 5 years</th>
<th>&gt; 10 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>8</td>
<td>2598</td>
<td>1972</td>
<td>1242</td>
</tr>
<tr>
<td>Java</td>
<td>2</td>
<td>1896</td>
<td>646</td>
<td>247</td>
</tr>
<tr>
<td>Python</td>
<td>1</td>
<td>1655</td>
<td>253</td>
<td>59</td>
</tr>
<tr>
<td>CUDA</td>
<td>0</td>
<td>181</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>School</td>
<td>3</td>
<td>2118</td>
<td>522</td>
<td>28</td>
</tr>
<tr>
<td>Industry</td>
<td>3</td>
<td>1808</td>
<td>1091</td>
<td>655</td>
</tr>
</tbody>
</table>
General Readability Metric

1. New model.
   • Buse baseline features
   • Additional visual features

2. Ground truth from a large human study.

3. Combine and evaluate.
Example Readability

```python
def handleBlockQuote(node):
    result = BlockQuoteDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleList(node):
    result = ListDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleListItem(node):
    result = ListItemDitem(node.nodeName)
    result.children = processChildren(node)
    return result

def handleTable(node):
    result = TableDitem(node.nodeName)
    # Ignore table contents that are not tr
    result.children = [x
        for x in processChildren(node) if x.type=='tr']
    return result

def handleTr(node):
    result = TrDitem(node.nodeName)
    # Ignore tr contents that are not th or td
    result.children = [x
        for x in processChildren(node) if x.type in ('th', 'td')]
    return result
```

---

1. Readability Rating 5
   - New Metric
   - Humans
   - Buse Metric
Annotator Agreement

- Spearman correlation: Agreement on ordering

<table>
<thead>
<tr>
<th>Score</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>+1</td>
<td>Perfect agreement</td>
</tr>
<tr>
<td>0</td>
<td>No relationship</td>
</tr>
<tr>
<td>-1</td>
<td>Perfect disagreement</td>
</tr>
</tbody>
</table>

- Our metric: 0.724
- Median: 0.551
- Avg: 0.512
- Buse metric: 0.309

Score | Meaning
--- | ---
+1 | Perfect agreement
0 | No relationship
-1 | Perfect disagreement
Impact of New Features

• How much improvement is due to our new features?

• **Re-train** Buse metric with our survey results.

• Compare our metric (**old + new features**) to Buse metric (**old features only**)
Impact of New Features

• Compute **f-measure**: 

\[
f = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
\]

\[
\text{precision} = \frac{TP}{TP + FP}
\]

\[
\text{recall} = \frac{TP}{TP + FN}
\]
Head-to-Head F-Measure

- Multi-language
  - 5% improvement

- Single-language
  - 16-26% improvement
## Predictors of Readability

### All Languages, All Lengths

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Line Length</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Long lines</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>Operator area</td>
<td>-</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of syntax</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>2D DFT of comments</td>
<td>+</td>
</tr>
<tr>
<td>Visual</td>
<td>String area to keyword area</td>
<td>+</td>
</tr>
<tr>
<td>Alignment</td>
<td>Min alignment length</td>
<td>+</td>
</tr>
</tbody>
</table>

### 5+ Years Industry Experience

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Long lines</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Whitespace</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>Comment area</td>
<td>+</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of whitespace</td>
<td>-</td>
</tr>
</tbody>
</table>
# Predictors of Readability

## All Languages, All Lengths

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Line Length</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Long lines</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>Operator area</td>
<td>-</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of syntax</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>2D DFT of comments</td>
<td>+</td>
</tr>
<tr>
<td>Visual</td>
<td>String area to keyword area</td>
<td>+</td>
</tr>
<tr>
<td>Alignment</td>
<td>Min alignment length</td>
<td>+</td>
</tr>
</tbody>
</table>

## 5+ Years Industry Experience

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Long lines</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Whitespace</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>Comment area</td>
<td>+</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of whitespace</td>
<td>-</td>
</tr>
</tbody>
</table>
## Predictors of Readability

### Java

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structural</td>
<td>1D DFT of whitespace</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Long lines</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Lines between identifiers</td>
<td>-</td>
</tr>
<tr>
<td>Syntax</td>
<td>Keywords</td>
<td>+</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of syntax</td>
<td>-</td>
</tr>
</tbody>
</table>

### Python

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>+/-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Identifiers</td>
<td>-</td>
</tr>
<tr>
<td><strong>Linguistic</strong></td>
<td>Identifier components</td>
<td>-</td>
</tr>
<tr>
<td>Visual</td>
<td>Operator area to keyword area</td>
<td>-</td>
</tr>
<tr>
<td>Structural</td>
<td>Operator to identifier tokens</td>
<td>+</td>
</tr>
<tr>
<td>Structural</td>
<td>1D DFT of syntax</td>
<td>-</td>
</tr>
</tbody>
</table>
Conclusion

• **Visual and spatial features** can significantly improve the accuracy of readability metrics.
  • **Different features** are more predictive for **different languages**.

• **Largest** human study of readability ratings to date.
  • Survey data is available **online**.
Questions?