Requirements Metrics for Requirements Statements

Chao Y. Din, David C. Rine
05/28/2013
cdin08@gmail.com, davidcrine@yahoo.com
Constrained Research Question

“Is there a way to identify low quality requirements statements in a requirements document or in a database?”
Constraints/Assumptions

- Systematic method is used (e.g., VORD)
- Grammar and spelling are checked
- Guidelines to avoid ambiguous terms and weak phrases are applied (e.g., ARM)
- Domain thesaurus or term definitions are supplied
- Requirements inspection methods are adopted
Related to Our Research Question

- **Requirements Document Complexity Metrics**
  - Ricker (1995)
    - Developed based on Information Retrieval techniques
    - Correlated complexity metrics to understandability (a quality factor)
      - Cohesion, Context, Coupling

- **Readability Metrics**
  - Readability Index
    - Based on number of words, number of syllables, word length, ... etc.
    - Flesch Reading Ease Score
    - Flesch-Kincaid Grade Level Score
  - Coh-Metrix (2002-present)
    - Measuring the cohesion of texts
Our Approach and Answers

- IR + NLP + NP chunks
- What is NP chunks
- How to acquire NP chunks
- A suite of metrics

NP Chunks

➢ Short History

  ▪ Abney, 1991
    - “I read (a sentence) a chunk at a time.”
    - Partial parsing

  ▪ Brill, 1993
    - Part-of-speech tagging
    - Supervised machine learning tagger
    - Statistical approach
      - “He will race/VB the car”
      - “When will the race/NN end”

  ▪ Ramshaw & Marcus, 1995
    - Using Brill’s tagger for chunk parsing (a partial parsing technique)

➢ Summary

  ▪ Non-recursive, non-overlapping, non-exhaustive

  ▪ Parsing by chunks is feasible and can be automated

  ▪ Closely related to the concept of objects in object-oriented programming paradigm
Acquiring NP Chunks

➢ Steps
  ▪ Tagging
  ▪ Partial Parsing
  ▪ Post-Processing

➢ Techniques Used
  ▪ Information Retrieval Techniques
  ▪ Statistical Natural Language Processing
  ▪ Chunk Parsing

➢ Tools Used
  ▪ Natural Language Tool Kit (NLTK) for Python
Three Core Metrics

- **NPC-Count**
  - NPC-Sentence *(metric 1 – example 1, 4)*
  - NPC-Req *(metric 2 – example 5)*

- **NPC-Cohesion** *(metric 3 – example 2)*

- **NPC-Coupling** *(metric 4 – example 3)*
  - NPC-Composite *(metric 5)*
NPC-Sentence Metric

\[ NPC - Sentence(sentence_j) = \sum_{1 \leq i \leq N} \frac{\text{entry}(i, j)}{\sum_{1 \leq j \leq C} \text{entry}(i, j)} \]

where \( \text{entry}(i, j) \) is the number of occurrence of NP chunk \( NP_i \) in sentence \( sentence_j \), \( 1 \leq i \leq N \), \( 1 \leq j \leq C \), \( N \) is the total number of NP chunks, \( C \) is the total number of sentences, and \( N \geq C \) before post-processing.
Example 1: Calculating NPC-Sentence

IM2-WKASSIGN-4: Assign Users to WAAs
IMAGS II shall track and maintain users’ assignment to WAAs. A user can be assigned to one or more WAAs, and a WAA can have more than one user assigned.

IM2-WKASSIGN-7: Assign Incoming Addresses on WAAs
IMAGS II shall assign incoming addresses to users based upon their WAAs.

IM2-WKASSIGN-8: Assign Multiple WAAs to Multiple Users
IMAGS II shall provide a way to assign a list of WAAs to multiple users at once.

(S: (0: <imags/NN> <ii/NN>) <shall/MD> <track/VB> <and/CC> <maintain/VB> (1: <users/NNS> <'/POS> <assignment/NN>) <to/TO> <waas/VB> <./.>)
(S: (2: <a/DT> <user/NN>) <can/MD> <be/VB> <assigned/VBN> <to/TO> (3: <one/CD>) <or/CC> (4: <more/JJR> <waas/NNS>) <./.> <and/CC> (5: <a/DT> <waa/NN>) <can/MD> <have/VB> <more/JJR> <than/IN> (6: <one/CD> <user/NN>) <assigned/VBN> <./.>)
(S: (7: <imags/NN> <ii/NN>) <shall/MD> <assign/VB> <incoming/VBG> (8: <addresses/NNS>) <to/TO> (9: <users/NNS>) <based/VBN> <upon/IN> (10: <their/PRP$> <waas/NNS>) <to/TO> <once/RB> <./.>)
(S: (11: <imags/NN> <ii/NN>) <shall/MD> <provide/VB> <to/TO> <assign/VB> (12: <a/DT> <way/NN>) <to/TO> (13: <a/DT> <list/NNS>) <of/IN> (14: <waas/NNS>) <at/IN> <users/NNS> <./.>)
Example 1 NPs in Vector Space (cont.)

<table>
<thead>
<tr>
<th>Stop words (or NP chunks): (a user), (one), (users), (a way), (a list)</th>
<th>req 1</th>
<th>req 2</th>
<th>req 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;imgs/NN&gt; &lt;ii/NN&gt;</td>
<td>0,7,11</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>&lt;users/NNS&gt; &lt;/POS&gt; &lt;assignment/NN&gt;</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>&lt;more/JJR&gt; &lt;waas/NNS&gt;</td>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&lt;a/DT&gt; &lt;waa/NN&gt;</td>
<td>5</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&lt;one/CD&gt; &lt;user/NN&gt;</td>
<td>6</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&lt;addresses/NNS&gt;</td>
<td>8</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>&lt;their/PRP$&gt; &lt;waas/NNS&gt;</td>
<td>10</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>&lt;waas/NNS&gt;</td>
<td>14</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>&lt;multiple/NN&gt; &lt;users/NNS&gt;</td>
<td>15</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Example 1 Text Normalization of NPs (cont.)

<table>
<thead>
<tr>
<th></th>
<th>req 1</th>
<th></th>
<th>req 2</th>
<th></th>
<th>req 3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent. 1</td>
<td>sent. 2</td>
<td>sent. 3</td>
<td>sent. 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;imgs/NN&gt; &lt;ii/NN&gt;</td>
<td>0,7,11</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>&lt;users/NNS&gt; &lt;/POS&gt; &lt;assign/NN&gt;</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>&lt;waa/NN&gt;</td>
<td>4,5,10,14</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>&lt;user/NN&gt;</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>&lt;address/NN&gt;</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>&lt;multiple/NN&gt; &lt;user/NN&gt;</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Example 1 Sentence Level Complexity (cont.)

\[
NPC\text{-Sentence}(\text{sentence}) = \sum_i \left( \frac{\text{entry } (i, j)}{\sum_j \text{entry } (i, j)} \right)
\]

<table>
<thead>
<tr>
<th></th>
<th>req 1</th>
<th>req 2</th>
<th>req 3</th>
<th>req 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent. 1</td>
<td>sent. 2</td>
<td>sent. 3</td>
<td>sent. 4</td>
</tr>
<tr>
<td>&lt;imgs/NN&gt; &lt;ii/NN&gt;</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>&lt;users/NNS&gt;/&lt;/POS&gt; &lt;assign/NN&gt;</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>&lt;waa/NN&gt;</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>&lt;user/NN&gt;</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>&lt;address/NN&gt;</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>&lt;multiple/NN&gt; &lt;user/NN&gt;</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Sentence level complexity</td>
<td>$\frac{1}{3}+1 = 1.33$</td>
<td>$\frac{2}{4}+1 = 1.5$</td>
<td>$\frac{1}{3}+\frac{1}{4}+1 = 1.58$</td>
<td>$\frac{1}{3}+\frac{1}{4}+1 = 1.58$</td>
</tr>
</tbody>
</table>
Three Core Metrics

- **NPC-Count**
  - NPC-Sentence *(metric 1 – example 1, 4)*
  - NPC-Req *(metric 2 – example 5)*

- **NPC-Cohesion** *(metric 3 – example 2)*

- **NPC-Coupling** *(metric 4 – example 3)*
  - NPC-Composite *(metric 5)*
NPC-Req Metric

\[ NPC-\text{Req}(\text{Req}_j) = \sum_{\text{sentence} \in \text{Req}_j} NPC-\text{Sentence}(\text{sentence}) , \]

where \(1 \leq i \leq L_j\), \(1 \leq j \leq M\), \(L_j\) is the total number of sentences of requirement \(j\), and \(M\) is the total number of requirements.
NPC-Cohesion Metric

- Unit of distance: sentence
- Step 1: cluster analysis
- Step 2: For each section, calculate the size of the clusters by distance

\[
NPC-Cohesion(S_j) = \begin{cases} 
\frac{\sum ClusterSize(i, j)}{L_j - 1}, & L_j > 1 \\
1, & L_j = 1 
\end{cases}
\]

where \( M_j \) is the total number of clusters in section \( S_j \), and \( L_j \) is the total number of sentences in section \( S_j \).

- \( NPC-Cohesion(S_1) = 2 / (3-1) = 1 \),
- \( NPC-Cohesion(S_2) = 1 / (2-1) = 1 \),
- \( NPC-Cohesion(S_3) = 1 \)
NPC-Coupling Metric

- Unit of distance: sentence
- Step 1: Cluster analysis.
- Step 2: Calculate the centroids of each cluster.
- Step 3: for each cluster in a section, calculate the distance from the NP chunk outside of the section to the centroid of the cluster in the section
- Step 4: for each NP chunk, X, that does not belong to a cluster, calculate the distance between the NP chunk, Y, outside the section to the NP chunk X
- Add all the above values together

\[
NPC-Coupling(S_1) = 4 + 3 + 2 + 4 + 3 = 16,
NPC-Coupling(S_2) = 3 + 2 = 5,
NPC-Coupling(S_3) = 4 + 4 + 3 = 11
\]
Evaluation

Case Study Fundamentals

- Three Types of Case Studies
  - Exploratory
  - Descriptive
  - Explanatory

- Case Study Methodology (Yin, 2003)
  - Step 1: A study question
  - Step 2: Study propositions, or hypotheses
  - Step 3: Unit(s) of analysis
  - Step 4: The logic linking of the data to the propositions
  - Step 5: The criteria for interpreting the findings
Example 2: Calculating NPC-Cohesion

![Cohesion Comparison (Din vs. Ricker)]
Example 2 (cont.)

Section Title: Flight Plan Output Data

• Sentence 1:

“The system shall provide flight plan outputs to a variety of operational positions, collocated processors, and remote facilities.”

(S:
(0: <the/DT> <system/NN>)
<shall/MD>
<provide/VB>
(1: <flight/NN> <plan/NN> <outputs/NN>)
<to/TO>
(2: <a/DT> <variety/NN>)
<of/IN>
(3: <operational/JJ> <positions/NNS>)
<./>.
<collocated/VBN>
(4: <processors/NNS>)
<./>.
<and/CC>
(5: <remote/JJ> <facilities/NN>)
<./>)

• Sentence 2:

“The ACCC shall output data periodically, on request, or in accordance with specified criteria (NAS-MD-311 and NAS-MD-314).”

(S:
(6: <the/DT> <accc/NN>)
<shall/MD>
<output/VB>
(7: <data/NNS>)
<periodically/RB>
<./>.
<on/IN>
(8: <request/NN>)
<./>.
<or/CC>
<in/IN>
(9: <accordance/NN>)
<with/IN>
<specified/VBN>
(10: <criteria/NN>)
<./>(
(11: <nas-md-311/NN>)
<and/CC>
(12: <nas-md-314/NN>)
</>)
<./>)
Example 3: Calculating NPC-Coupling

![Graph showing the calculation of NPC-Coupling with Ricker and Din sections.](image-url)
Example 4: Calculating NPC-Sentence
Example 4 (cont.)

![Graph showing sentence level complexity for Version 2 and Version 2 plus changes.](image-url)
Example 5: Calculating NPC-Req
Example 5 (cont.)
NPC Metrics for Database

- NPC-Count
  - NPC-Sentence
  - NPC-Req

- NPC-Cohesion:
  Total # of All NPCs / (Total # of Distinct NPCs * Total # of Sentences)

- NPC-Coupling:
  sum(Total # of External Sentences that refer to the internal NPCs) / Total # of External Sentences
Questions & Answers

❓