SAND2016-4133C

A Text Mining and Information Extraction Tool for Unstructured Data

Arthur McDonald Lanny Gilbertson Tim C'de Baca

Advanced Software Engineering
Sandia National Laboratories

National Laboratories Information Technology Summit, 2016

Outline

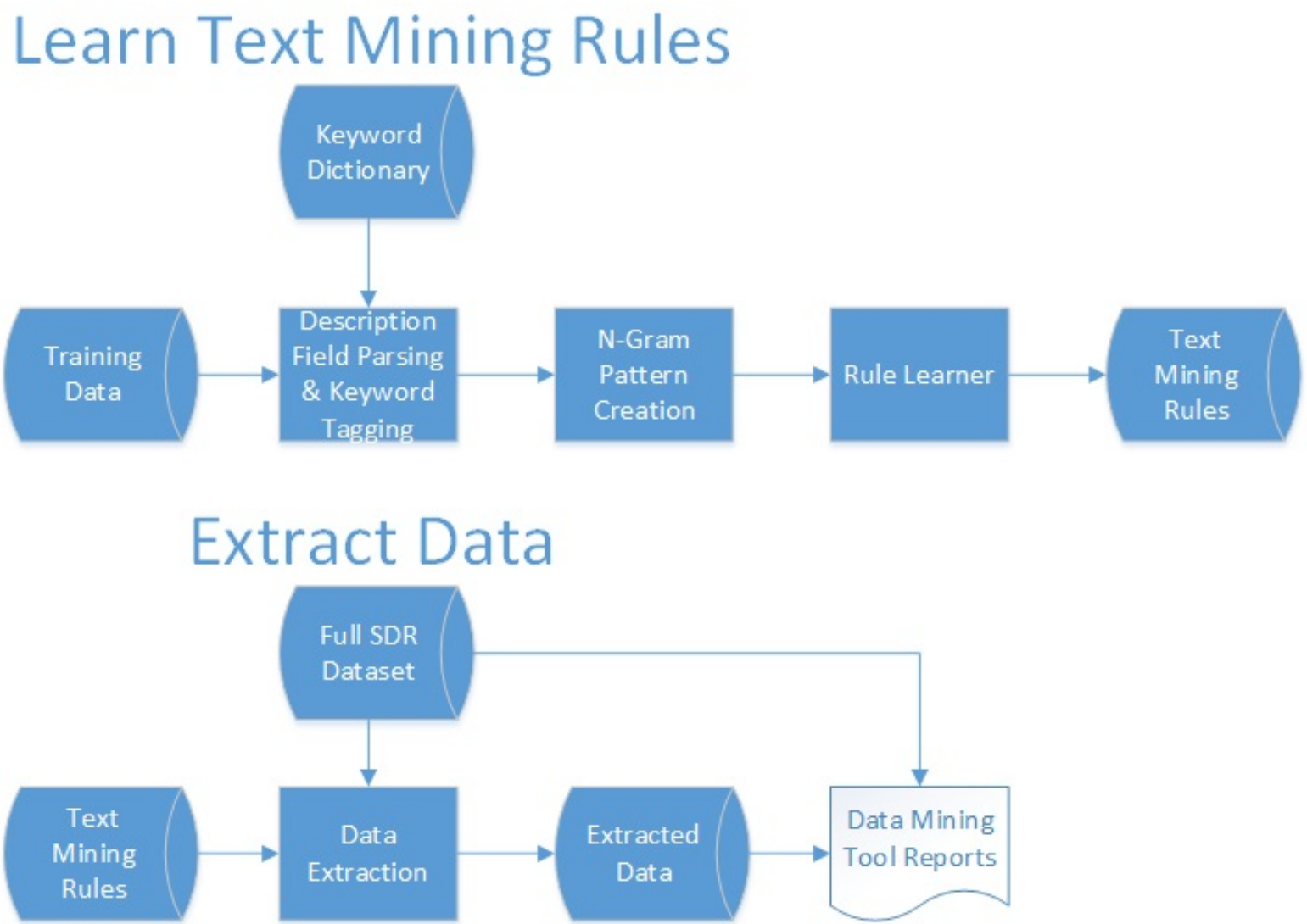
The Problem - Transport Aircraft Risk Assessment

- Defines a process for calculating risk in transport aircraft design.
- Uses the conditional probability that the occurrence of a fatigue crack will Not be Detected (ND).
- Defects (crack, corrosion, dent, etc.) are submitted to FAA by carriers in Service Difficulty Reports (SDRs).
- Approx. 1.4 million SDRs submitted since 1974.
- Report entry has no standardized required format. Technicians enter the data in an unstructured description field (text).
- Initial work involved manually sorting/searching SDR records for crack information - tedious and time consuming.
- FAA approached Sandia Labs for a software solution.

The Problem - Information Extraction

- Approximately 5% of the SDR records have CrackLength column populated in database.
- Other 95% may or may not have crack length data in the unstructured Description field.
- How to find and extract this data?
  - Not an easy task - can not simply use SQL 'LIKE' statements or Regular Expressions.
  - Description field written in natural language by human technicians
    - Different variations of short-hand, misspellings, different units of measurement, etc.
- Solution: use natural language processing, machine learning, and text mining.

The Process


Learn Text Mining Rules
Keyword Dictionary
Training Data
Description Field Parsing & Keyword Tagging
N-Gram Pattern Creation
Rule Learner
Text Mining Rules
Extract Data
Full SDR Dataset
Text Mining Rules
Data Extraction
Extracted Data
Data Mining Tool Reports

Sentence Parsing

- Split the sentence into tokens, using a set of delimiter characters.
- Remove stop words
- Tag each token based on a keyword dictionary.
  - Tokens are considered "unknown" if not in dictionary.

Sentence Parsing

- Create N-Grams from the sentence.
  - 4-Grams for the sentence "The quick fox jumped over the dog":
  - The quick fox jumped
  - quick fox jumped over
  - fox jumped over the
  - jumped over the dog
- From each N-Gram, we now have a pattern using the tagged tokens.

Sentence Parsing

Example:

FOUND A 2 INCH CRACK AT OUTBOARD END OF RT HORIZONTAL STABILIZER

⟨found⟩FOUND⟨/found⟩ ⟨num⟩2⟨/num⟩ ⟨inch⟩INCH⟨/inch⟩
⟨crack⟩CRACK⟨/crack⟩ ⟨loc⟩OUTBOARD⟨/loc⟩ END
⟨loc⟩RT⟨/loc⟩HORIZONTAL⟨struct⟩STABILIZER⟨/struct⟩

Other Sentence Pre-processing

- Convert number words into values:
  ONE $\longrightarrow$ 1
- Parsing tokens for inch markers:
  1" $\longrightarrow$ ⟨num⟩1⟨/num⟩ ⟨inch⟩INCH⟨/inch⟩
- Converting fractions to doubles:
  $1/4 \longrightarrow 0.25$
- Metric to U.S. standard conversion:
  x millimeters to inches $(\frac{x}{25.4})$
  x centimeters to inches $(\frac{x}{2.54})$

Pattern Rule Learning

- 16,000 SDR records with CrackLength column populated - use as training examples.
- Parse training examples, tag words found in dictionary, and create N-Grams.
- For each N-Gram in the training example, if the N-Gram contains the CrackLength value, create a rule from that pattern.
  - If the rule already exists, then increase the occurrence count of that rule.
- Rule patterns created from FOUND A 2 INCH CRACK AT OUTBOARD END OF RT HORIZONTAL STABILIZER:
  - found num inch crack
  - num inch crack loc

Rule Table:

	nlp_rule_id	pattern	type	score	created_from_sdr_id	occurs
277	4190	crack num inch struct	CRACKLEN	0.96784	110745	873
278	4498	unknown found num loc	CRACKLEN	0.96000	204569	24
279	5311	found crack crack num	CRACKLEN	0.96000	855961	24
280	4477	crack sizeattr equals num	CRACKLEN	0.95652	194913	22
281	4688	struct found crack num	CRACKLEN	0.95652	258101	22
282	4307	num inch struct num	CRACKLEN	0.95360	159059	740
283	4348	found num inch sizeattr	CRACKLEN	0.95238	161595	40
284	4893	crack crack num inch	CRACKLEN	0.94285	335273	33
285	4411	found unknown crack num	CRACKLEN	0.94244	173624	131
286	4853	crack num struct unknown	CRACKLEN	0.94230	320573	49
287	4327	found struct crack num	CRACKLEN	0.93769	160550	602
288	4316	crack num sizeattr at	CRACKLEN	0.93750	160087	15

Crack Length Rule Learning

Algorithm 1 Crack Length Rule Learning Algorithm

```
1: procedure LearnLengthRule(Training example s, Sentence s)
2:     crackLength ⟵ sdr.CrackLength
3:     for each N–Gram n ∈ s.ngrams do
4:         if crackLength ∈ n then
5:             newRule ⟵ n.pattern
6:             if newRule ∈ RuleSet then
7:                 RuleSet(newRule).Occurs + +
8:             else
9:                 RuleSet ⟵ RuleSet ∪ newRule
10:            end if
11:        end if
12:    end for
13: end procedure
```

Rule Scoring

- After rule pattern discovery, run each rule on the training examples to check if it covers the example.
- Assign a score for each learned rule:

$$Score(rule) = \frac{rule.Pos}{rule.Pos + rule.Neg}$$

- A rule positively covers if the pattern is found in the example, and the value found in the pattern equals the CrackLength value.
- A rule negatively covers if the pattern is found in the example but the value found in the pattern does not equal the CrackLength value.

Scoring Algorithm

Algorithm 2 Rule Confidence Scoring Algorithm

```
1:  procedure ConfidenceScore
2:      for each rule r do
3:          Positives ⟵ 0
4:          Negatives ⟵ 0
5:          for each training example te do
6:              tempdata ⟵ ExtractData(te, r)
7:              if tempdata ≠ null then
8:                  if tempdata.Contains(te.CrackLength) then
9:                      Positives + +
10:                 else
11:                     Negatives + +
12:                 end if
13:             end if
14:         end for
15:         r.Score ⟵ Positives / (Positives + Negatives)
16:     end for
17: end procedure
```

Data Extraction

- Determine "good" ruleset to use based on Score and Occurrence values.
  - Rule score $>$ 0.85 and occurrence $>$ 20.
  - 44 crack length rules.
- Run ruleset against entire SDR database to extract CrackLength values.
  - Over 61,000 crack lengths extracted.
  - What about false positives?
  - Manual validation by an FAA expert.
- Most accurate rules were created from 4-Grams, but missed approximately 15,000 records on first pass.
  - Also include the rule with pattern "num inch crack"
- Multiple rules might extract duplicates - keep track of position of extracted data to avoid extracting the same value.

Extracted Data Table:

	extracted_data_id	extracted_data	data_type	sdr_id	nlp_rule_id	valid	position
129	169517	4	CRACKLEN	3226	4134	NULL	7
130	169518	4	CRACKLEN	3228	4134	NULL	7
131	201493	3	CRACKLEN	3248	7414	NULL	21
132	201494	.125	CRACKLEN	3248	7414	NULL	22
133	184271	1	CRACKLEN	3290	7411	NULL	0
134	184272	.75	CRACKLEN	3290	7411	NULL	11
135	184273	1.75	CRACKLEN	3297	7411	NULL	5
136	184274	.75	CRACKLEN	3299	7411	NULL	5
137	159531	4	CRACKLEN	3377	4101	NULL	11
138	159532	3.75	CRACKLEN	3379	4101	NULL	7

Extracted N-Grams Table:

	extract...	ngram	nlp_rul...	sdr_id
1	247511	<num>.5</num> <num>3</num> <inch>INCH</inch> <long>LONG</long>	7414	35
2	247512	<num>3</num> <inch>INCH</inch> <long>LONG</long> <struct>FASTENERS</str...	7414	35
3	247513	<and>AND</and> <num>3</num> <inch>INCH</inch> <long>LONG</long>	7414	38
4	247514	<num>3</num> <inch>INCH</inch> <long>LONG</long> <crack>CRACKED</crack>	7414	38
5	209303	<and>AND</and> <num>19.5</num> <inch>INCH</inch> <crack>CRACK</crack>	7411	40
6	209304	<num>19.5</num> <inch>INCH</inch> <crack>CRACK</crack> <loc>LT</loc>	7411	40
7	209305	<found>FOUND</found> <num>2</num> <inch>INCH</inch> <crack>CRACK</crac...	7411	47
8	209306	<num>2</num> <inch>INCH</inch> <crack>CRACK</crack> <at>AT</at>	7411	47
9	209325	<num>.75</num> <inch>INCH</inch> <crack>CRACK</crack> <loc>NR</loc>	7411	460

Data Extraction

Algorithm 3 Data Extraction Algorithm

```
1: procedure DataExtract(RuleSet, SDRs)
2:     for each rule r ∈ RuleSet do
3:         for each Sentence s ∈ SDRs do
4:             if Match(r.pattern, s) then
5:                 extracted_data ⟵ token with num tag
6:             end if
7:         end for
8:     end for
9: end procedure
```

Data Validation

7373H4

ATA Code	Largest Cracks	Remarks	Status		
5210					
5210	2	DURING SCHEDULED C3 CHECK, FOUND AFT ENTRY DOOR FWD LOWER CORNER CUTOUT CRACKED 2.0 IN. REPAIRED PER EA.;	VALID		
5210	1.5	DURING SCHEDULED SERVICE CHECK, FOUND FWD ENTRY DOOR EXTERNAL SKIN CRACKED 1.5 INCH LONG AT FWD LOWER CORNER. REPAIRED PER SRM.;		Valid	Invalid
5210	0.375	DURING SCHEDULED B2 CHECK, FOUND AFT ENTRY DOOR UPPER RADIUS HINGE CUTOUT CRACKED .3750 INCH. REPAIRED PER BOEING STRUCTURAL REPAIR MANUAL.;		Valid	Invalid
5230					
5230	437	FORWARD CARGO DOOR SKIN CRACKED BETWEEN BS 415+7" TO BS 437+2". REPAIRED SKIN IAW EA.;	INVALID		

Summary

- Future work:
  - Learn rules for Number of Cracks:
    - INSPECTION FOUND TWO .125 INCH CRACKS IN TOP CENTER PANEL...
  - Learn rules for crack locations and/or structures.
  - Generalize the NLP engine for wider use in text mining and information extraction applications.
  - Better scoring algorithm - reduce number of false positives.
  - Get results from manual validation to determine accuracy of the rules used for extraction.