# Automated Techniques for Finding and Maintaining Traces

Group 1

28th November 2016

# Overview

# The Need for Automatisation

Traditional requirements tracing is

- slow
- expensive
- prone to non-completeness (missing links)

$\Rightarrow$ Indicators for automation potential

# How to Find Links?

- Relevancy likelihood estimation is a basic task in computational linguistics.
- Basically a search engine where everything's a query and everything's a document.
- One possible technique: term vectors.

# Term Vectors

How relevant is a specific term to an artifact?

## TF-IDF

$$r_{i,j} = tf_{i,j} \cdot \log \frac{1}{df_j}$$

# Term Vectors

How relevant is a specific term to an artifact?

## TF-IDF

$$r_{i,j} = tf_{i,j} \cdot \log \frac{1}{df_j}$$

## Term Vector structure

$$\vec{r_i} = \begin{pmatrix} r_{i,\text{turnstile}} \\ r_{i,\text{card}} \\ r_{i,\text{the}} \\ \vdots \\ r_{i,\text{operator}} \end{pmatrix} = \begin{pmatrix} 0.22 \\ 0.68 \\ 0.02 \\ \vdots \\ 0.00 \end{pmatrix}$$

# Term Vectors

How relevant is a specific term to an artifact?

## TF-IDF

$$r_{i,j} = tf_{i,j} \cdot \log \frac{1}{df_j}$$

## Term Vector structure

$$\vec{r_i} = \begin{pmatrix} r_{i,\text{turnstile}} \\ r_{i,\text{card}} \\ r_{i,\text{the}} \\ \vdots \\ r_{i,\text{operator}} \end{pmatrix} = \begin{pmatrix} 0.22 \\ 0.68 \\ 0.02 \\ \vdots \\ 0.00 \end{pmatrix}$$

How relevant are two artifacts to each other?

## Cosine Similarity

$$rel_{\vec{r_1},\vec{r_2}} = \cos(\vec{r_1} \angle \vec{r_2}) = \frac{\vec{r_1} \cdot \vec{r_2}}{|\vec{r_1}| \cdot |\vec{r_2}|}$$

# What to do with this information?

- We have an automatically generated relevancy likelihood extimation for each pair of trace artifacts.
- Two options:
  1. Register all traces with a similarity $> \theta$ and be done with it.
  2. Present higher-probability links to human analyst for validation.
- Option 2. Definitely option 2.
- Therefore recall must be optimised over precision.

# Learning

- Automatic relevancy vectors are unreliable.
- Let's steal some stuff from computational linguists again!
  - Weight matrices are useful!

## Adding a weight vector

$$\vec{r}_{\text{doc}} \cdot \vec{w} = \begin{pmatrix} r_{\text{turnstile}} \\ r_{\text{card}} \\ r_{\text{the}} \\ \vdots \\ r_{\text{operator}} \end{pmatrix} \cdot \begin{pmatrix} w_{\text{turnstile}} \\ w_{\text{card}} \\ w_{\text{the}} \\ \vdots \\ w_{\text{operator}} \end{pmatrix} = \begin{pmatrix} 0.22 \\ 0.68 \\ 0.02 \\ \vdots \\ 0.00 \end{pmatrix} \cdot \begin{pmatrix} 0.75 \\ 2.00 \\ 0.25 \\ \vdots \\ 1.00 \end{pmatrix}$$

- Weights can be uniformly initalised and later modified according to feedback from human analysts.

# Hayes et al. 2007

- RETRO system
  - Facilitates tracing for human analysts
  - Intended for vertical tracing ('high-level' to 'low-level' artifacts)

| | Recall | Precision | Time [min] |
|---|---|---|---|
| Manual group | 0.33 | 0.24 | 120.67 |
| RETRO group | 0.70 | 0.13 | 41.88 |
| T test (p value) | 0.001 | 0.01 | 0.0004 |

# Huang et al. 2007

- Poirot System
  - Smarter parsing
  - (Programming) stopword removal
  - Normalisation of `conventionallyStyledNames();`
- Achieved 90% recall and about 30% precision
- Deduced nine ‘best practices’

# Huang's Best Practices

1. Trace for a purpose
2. Define a suitable trace granularity
3. Support in-place traceability
4. Use a well-defined project glossary
5. Write quality requirements
6. Construct a meaningful hierarchy
7. Bridge the intradomain semantic gap
8. Create rich content
9. **Use a process improvement plan**

# Discussion