

Jeff Sherwood, Programmer.  
Anjanette Young, Systems Librarian.  
University of Washington, Libraries.

# DSpace Repository

**Goal** Ingest Metadata and PDF's for ETD's received from UMI into a DSpace repository.

The screenshot shows the University of Washington Libraries ResearchWorks Archive website. The header includes the university logo and name, a search bar with a 'Go' button, and a 'Login' link. The main navigation bar features 'ResearchWorks' and 'Dissertations and Theses' tabs. The left sidebar contains sections for 'Browse', 'All of ResearchWorks', 'This Community', and 'My Account'. The main content area is titled 'Dissertations and Theses' and includes a description, a 'Copyright and License' section with the text 'Copyright is held by the individual authors.', and a list of 'Collections in this community' with their respective counts.

University of Washington Libraries FAQ

**UNIVERSITY LIBRARIES**  
UNIVERSITY of WASHINGTON Login

Search    Search ResearchWorks  
 This Community [Advanced Search](#)

[ResearchWorks](#) [Dissertations and Theses](#)

**Browse**

**All of ResearchWorks**

- [Communities & Collections](#)
- [By Issue Date](#)
- [Authors](#)
- [Titles](#)
- [Subjects](#)

**This Community**

- [By Issue Date](#)
- [Authors](#)
- [Titles](#)
- [Subjects](#)

**My Account**

- [Login](#)
- [Register](#)

**Related Information**

- [Issues in Scholarly Communication](#)
- [Publication Agreement Author Rights Addendum](#)

## Dissertations and Theses

Most dissertations and theses are available via University of Washington IP address only. For access outside the UW community, please request titles via Interlibrary Loan from your local academic or public library.

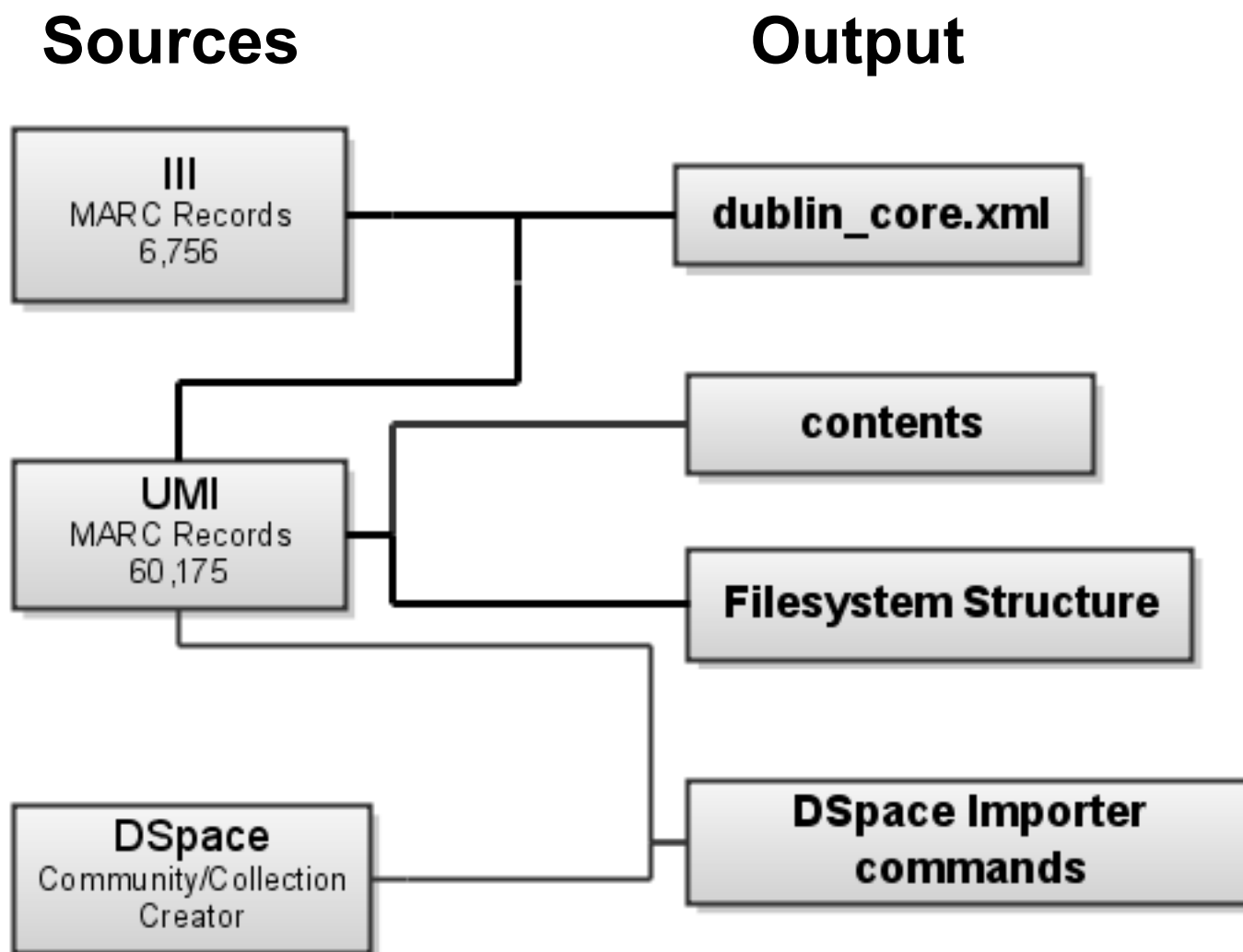
**Copyright and License**

*Copyright is held by the individual authors.*

**Collections in this community**

- Aeronautics and astronautics [56]
- Anthropology [180]
- Applied mathematics [46]
- Astronomy [38]
- Atmospheric sciences [88]
- Behavioral neuroscience [49]
- Bioengineering [147]
- Biological chemistry [79]
- Biological structure [16]
- Biology [168]
- Biomedical and health informatics [3]
- Biostatistics [83]
- Built environment [1]
- Business administration [136]
- Chemical engineering [118]
- Chemistry [319]
- Civil engineering [111]
- Classical languages and literature [35]
- Communications [74]
- Comparative literature [82]

# Electronic Theses & Dissertations



# MARC Fields

## UMI Records

=001 (Filename)  
=520 (Abstract)

## III Records

=001 (OCLC number)  
=100 (Author)  
=245 (Title)  
=260 (Date published)  
=502 (type and date)  
=695 (Department)  
=941 (Local identifier)

# dublin\_core.xml

```
<dublin_core>
  <dcvalue element="identifier" qualifier="other">
    iii[941]</dcvalue>
  <dcvalue element="title" qualifier="none">
    iii[245][a][b]</dcvalue>
  <dcvalue element="contributor" qualifier="author">
    iii[100][a][b][c]</dcvalue>
  <dcvalue element="description" qualifier="abstract">
    umi[520][a]</dcvalue>
  <dcvalue element="subject" qualifier="other">
    iii[655][a][x]</dcvalue>
</dublin_core>
```

# MARC Loader . . . No.

|||0|0| | |0|n|G|0|@ov\_action="o"

|||0|0| | |0|n|G|0

|@ov\_protect="b=V0123456789d(690,695:d)

hn(590:d)y(099,249,852,856:d)y(910,925,  
980,981)F26"

035|001 | +|0|0|b|o|0|y|N|0|%001(start="1-9",char="!-~")

245|| +|0|0|b|t|0|y|N|0|%bracket="h"

500-599|| +|0|0|b|n|0|y|N|0|

600-651|| -w|0|0|b|d|0|y|N|0|

653-657|| +|0|0|b|d|0|y|N|0|

690-699|| -w|0|0|b|d|0|y|N|0|

700-715|| -w|0|0|b|b|0|y|N|0|

730-740|| -w|0|0|b|f|0|y|N|0|

# Matching overview

## Ham-fisted Method

1. Exact Title + Exact Author
2. Exact Title + Shortened Author

## Cool Math Method

Calculate Similarity of Title

Calculate Similarity of Author

1. Exact Title + Fuzzy Author
2. Fuzzy Title + Fuzzy Author
3. Fuzzy Title or Fuzzy Author

# Pymarc - the MARC Hammer

```
umi_dict = {
```

```
    Alaskan Bootlegger: {author: Leon Kania, umi_count = 1},
```

```
    title2_value: {author: author2_value, umi_count = index2},
```

```
    ...
```

```
}
```

```
iii_dict = {
```

```
    Alaskan Bootlegger: {author: Leon W. Kania, iii_count = 9},
```

```
    title2_value: {author: author2_value, iii_count = index2},
```

```
    ...
```

```
}
```



# Exact title + exact author

```
# Exact Title
```

```
# Create sets out of the dictionary keys
```

```
umi_set = set(umi_dict.iterkeys())
```

```
iii_set = set(iii_dict.iterkeys())
```

```
# Find the Intersection of sets.
```

```
title_match = umi_set & iii_set
```

```
# Verify Intersection with Exact Author
```


```
for x in title_match:
```

```
    if umi_dict[x][author] == iii_dict[x][author]:
```

```
        . . . do stuff.
```

# Exact title + Truncated author

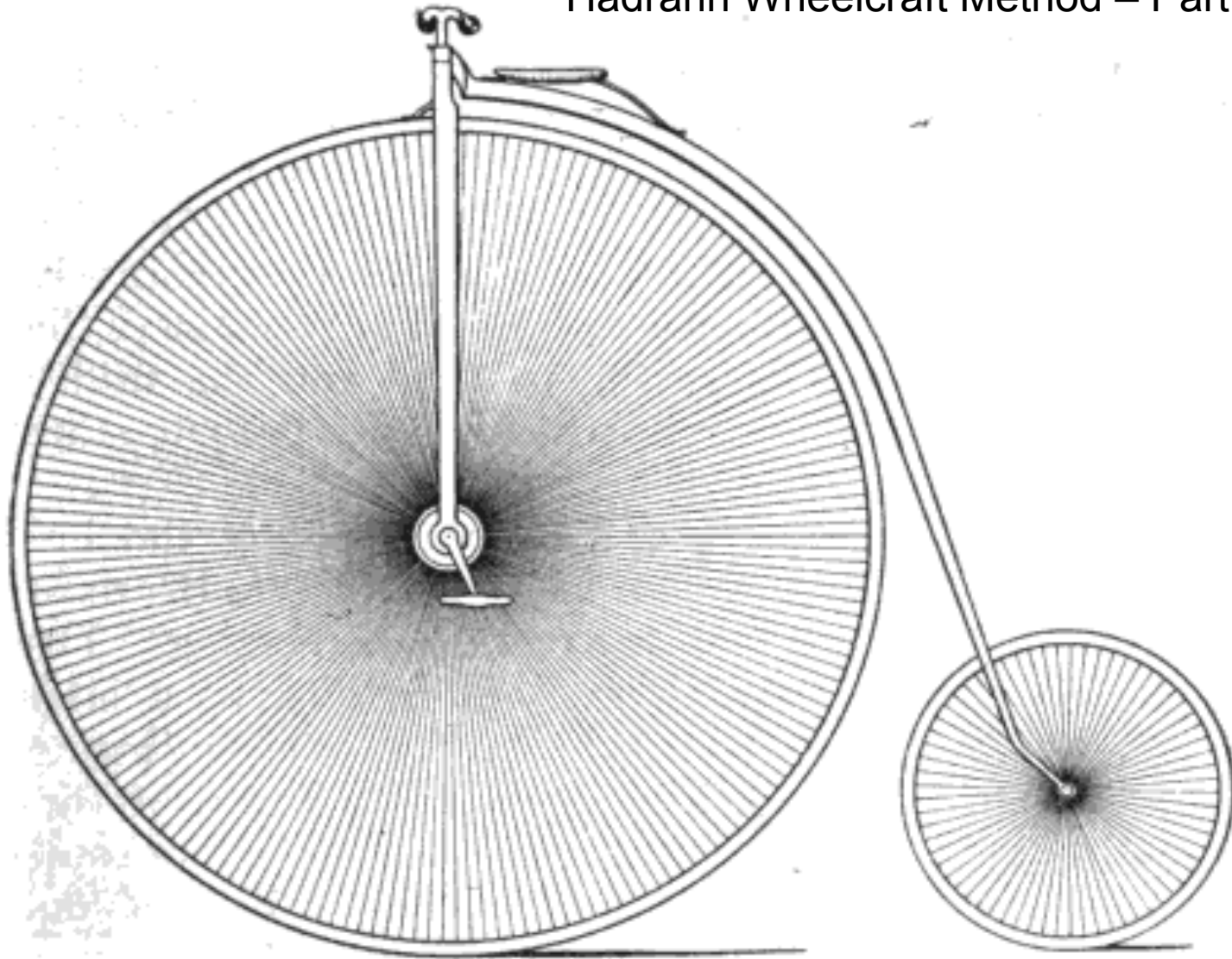
```
def shortenAuthorName(name):  
    #Leon W. Kania    ->    [Leon, W., Kania]  
    namelist = str(name).split()  
    if len(namelist) > 2:  
        shortname = "%s %s" % (namelist[0], namelist[-1])  
    else:  
        shortname = name  
    return shortname
```



The diagram consists of two blue arrows. The first arrow originates from the first element of the list `namelist[0]` in the code and points to the first `%s` placeholder in the format string `"%s %s"`. The second arrow originates from the last element of the list `namelist[-1]` and points to the second `%s` placeholder in the same format string. This illustrates how the function uses the first and last parts of a multi-part name to create a truncated author name.

# "If you break three spokes, it is time for a rebuild"

Charles Hadrann,  
"Hadrann Wheelcraft Method – Part 1 Lacing"



# Rogues Gallery

# USE OF CROWN LENGTH TO DEFINE STEM FORM: SEGMENTED TAPER EQUATION (DOUGLAS FIR)

Use of crown length to define stem form  
:: segmented taper equation

Towards an understanding of seismic performance of three-dimensional structures: Stability and reliability

Towards an understanding of seismic performance of 3D structures :: stability & reliability

Hoekstra, Hopi Danielle Elisabeth

Hoekstra, Danielle E

Arnason, Halldor

Halldór Árnason





# Levenshtein Edit Distance

**Edit distance** is the number of operations required to transform one string of characters into the another.

How many steps to turn

**kitten**

into **sitting?**

**3**

**kitten → sitten** (k changes to s)

**sitten → sittin** (e changes to i)

**sittin → sitting** (insert g)

# LD is Always...

- $\geq$  difference in string lengths
- $\leq$  length of the longer string
- $= 0$  if the strings are identical

# Similarity Score

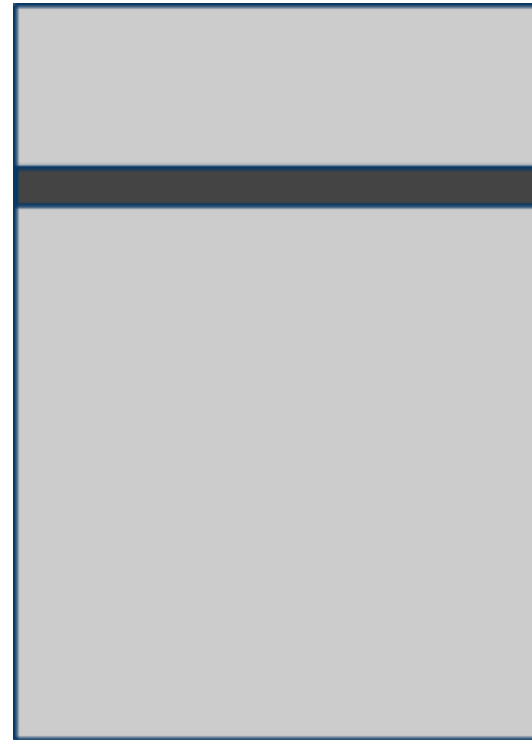
$$\textit{similarity} = 1 - \frac{d_L}{\max(|s_1|, |s_2|)}$$



# Optimizations

# Reduce the Search Space

"A stochastic model  
of cyclical interaction  
processes"



**All titles**

# Reduce the Search Space

## Identify Stopwords

the: 24587  
for: 7643  
with: 3323  
effects: 1958  
evaluation: 1073

...

hypoxic: 1  
reduplication: 1  
picaresque: 1  
emperador 1  
heteroduplex 1

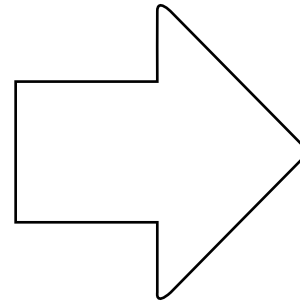
**Throw out common  
words in titles**

**Keep the rarer ones**

# Reduce the Search Space

## Extract Significant Words

"Stochastic  
models for DNA  
sequence data"



stochastic  
dna  
sequence

# Reduce the Search Space

```
rec = {'title': 'Stochastic models...'};
```

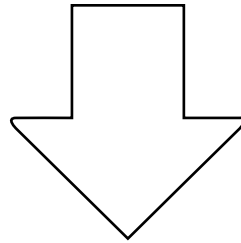
```
index['stochastic'].append(rec)
```

```
index['dna'].append(rec)
```

```
index['sequence'].append(rec)
```

# Reduce the Search Space

**index['stochastic']**



```
{'title': "Stochastic models for DNA sequence data", ...}  
{'title': "A stochastic model of clan systems", ...}  
{'title': "A stochastic model of cyclical interaction processes", ...}  
{'title': "Stochastic reliability models for maintained systems", ...}  
{'title': "Uniform approximation and almost periodicity of doubly stochastic operators", ...}
```

# Normalize Names

Hoekstra, Hopi Danielle Elisabeth

Hoekstra, Danielle E



# Normalize Names

Hoekstra, H

Hoekstra, D



# Normalize Names

Arnason, Halldor

Halldór Árnason



# Normalize Names

Arnason, H

Árnason, H

**Improvements**

# Jaro-Winkler Algorithm

$$d_j = \frac{1}{3} \left( \frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m - t}{m} \right)$$

# What's a "match"?

Two characters match if they are a reasonable distance from one another as defined by:

$$\left\lfloor \frac{\max(|s_1|, |s_2|)}{2} \right\rfloor - 1$$

# Example

$$d_j = \frac{1}{3} \left( \frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right)$$

$s_1 = \text{Martha}$

$s_2 = \text{Marhta}$

$$d_j = \frac{1}{3} \left( \frac{6}{6} + \frac{6}{6} + \frac{6-1}{6} \right)$$

$$\approx 0.944$$

# Example

$$\textit{similarity} = 1 - \frac{d_L}{\max(|s_1|, |s_2|)}$$

s1 = Martha

s2 = Marhta

$$\textit{similarity} = 1 - \frac{2}{6}$$

$$\approx 0.667$$

Jaro-Winkler works  
best for short strings



# Resources

# Levenshtein & Jaro-Winkler

## Background

[http://en.wikipedia.org/wiki/Levenshtein\\_distance](http://en.wikipedia.org/wiki/Levenshtein_distance)

[http://en.wikipedia.org/wiki/Jaro-Winkler\\_distance](http://en.wikipedia.org/wiki/Jaro-Winkler_distance)

## Code

<http://pypi.python.org/pypi/editdist/0.1>

<http://pypi.python.org/pypi/python-Levenshtein/0.10.1>

# Miscellaneous

## **String Comparison Tutorial**

<http://bit.ly/ZGSmF>

## **SecondString - Java text analysis library**

<http://secondstring.sourceforge.net/>

## **MarcXimiL - MARC de-duping package**

<http://marcximil.sourceforge.net/>

**<http://snurl.com/uggtn>**