Imperial College London

ICDSS Imperial College Data Science Society

Intelligent Web Crawling

Jakob Torben, Ricardo Mokhtari and Yousef Nami

Advanced Data Science Team, Imperial College London

The Team

Dr. Ovidiu Serban

Supervisor
Research Fellow, DSI
Imperial College London

Jakob Torben

Imperial College London
BSc Physics
MSc Applied Computational Science

Ricardo Mokhtari

Imperial College London
MEng Molecular Bioengineering

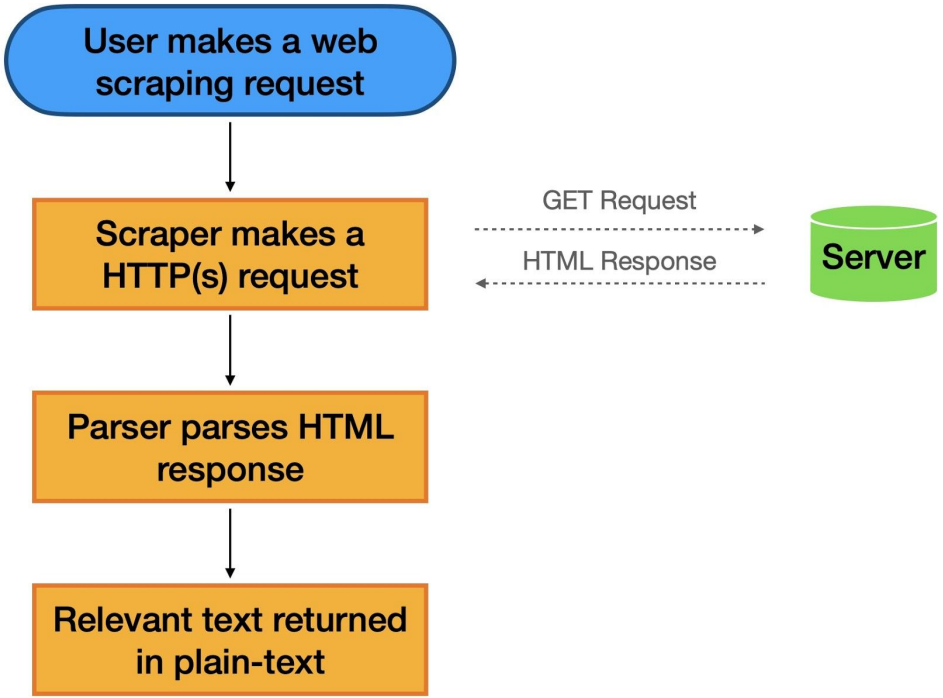
Yousef Nami

Imperial College London
MEng Mechanical Engineering

ICDSS
Imperial College Data Science Society

Problem Definition

Problem Definition



Temporality:

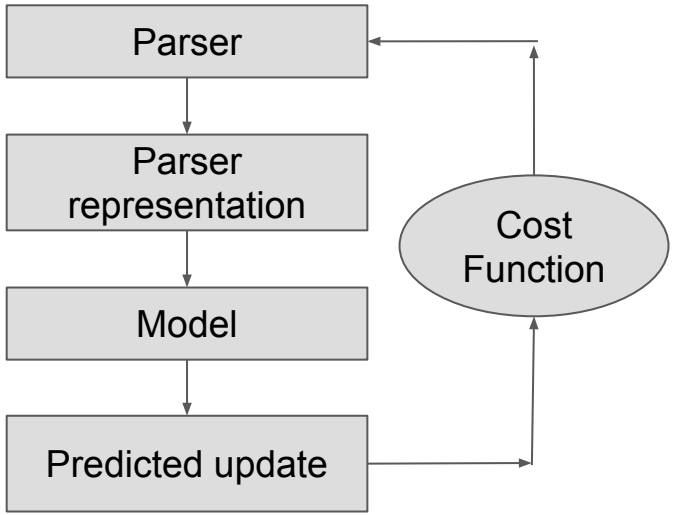
- websites change over time
- Changes in the HTML structure
- Changes in domain names

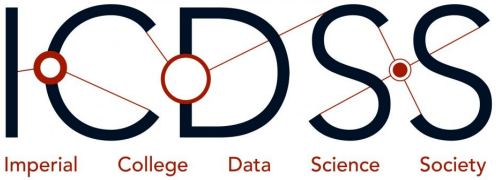
Scale:

- 100 companies
- All HTML structures different
- Different subdomains for leadership pages
- Duplicate names

Problem Definition

- Rule based methods not very effective due to very high variability
- Data driven approach could work
- Requires a ‘Cost Function’ to determine the similarity/difference between any two parsers





Case Study

Case Study: GSK

Corporate Executive Team

The Chief Executive Officer is responsible for the management of the business and is assisted by the Corporate Executive Team (CET). The CET manages our activities, and each member is responsible for a specific part of the business.

To view all our images in high resolution, visit our GSK Flickr page.

h2 325.05 × 26.4

Dame Emma Walmsley

Chief Executive Officer

Read more

Iain Mackay

Chief Financial Officer

Read more

```
<li class="grid-listing__item">
  ::marker
  <a class="grid-listing__link" href="/en-gb/ab
    <div class="grid-listing__img">…</div>
    <div class="grid-listing__content">
      <h2>Dame Emma Walmsley</h2>
      <p>Chief Executive Officer</p>
      <span>Read more</span>
    </div>
  </a>
</li>
<li class="grid-listing__item">…</li>
<li class="grid-listing__item">…</li>
<li class="grid-listing__item">…</li>
<li class="grid-listing__item">…</li>
```

Case Study: GSK Parser

Imperial College Data Science Society

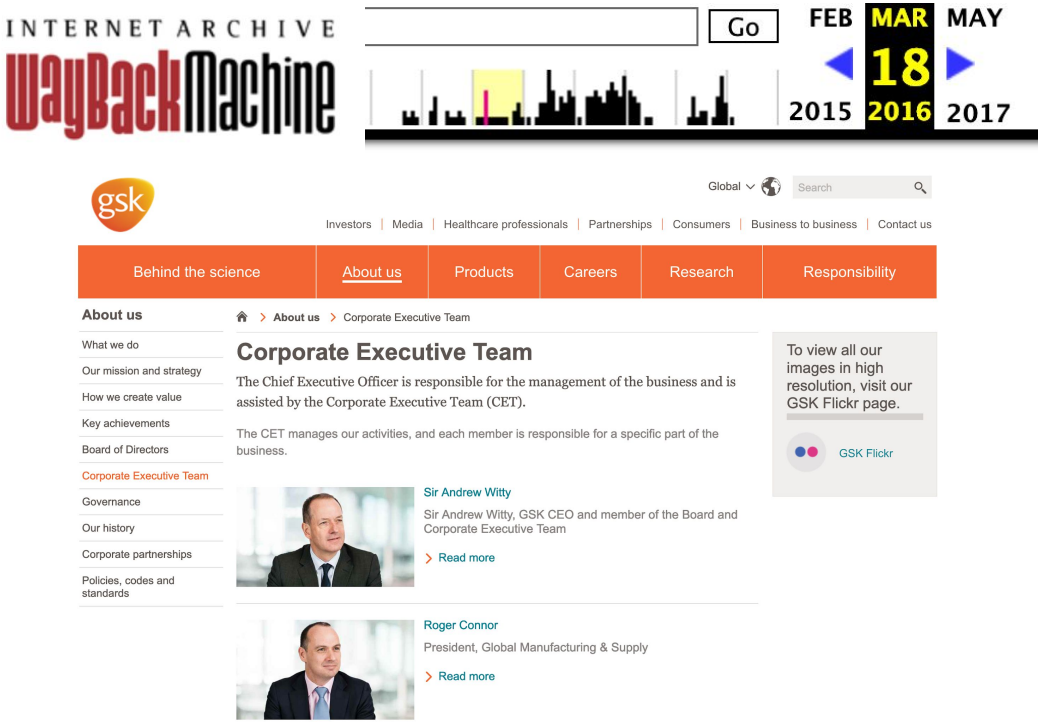

```python
# parse the current GSK board
def parse_current(self, response):
    # define selector that contains all items
    all_people = response.css("li.grid-listing__item")
    print(all_people)
    # iterate through items
    for person in all_people:
        # manually parse name
        name = person.css("a>div>h2::text").get()
        # manually parse title
        title = person.css("a>div>p::text").get()

        now = datetime.datetime.now()
        year = now.year

        # Return item
        yield self.create_board(name,title,year)
```

Case Study: Historical Information

3 Different parsers:

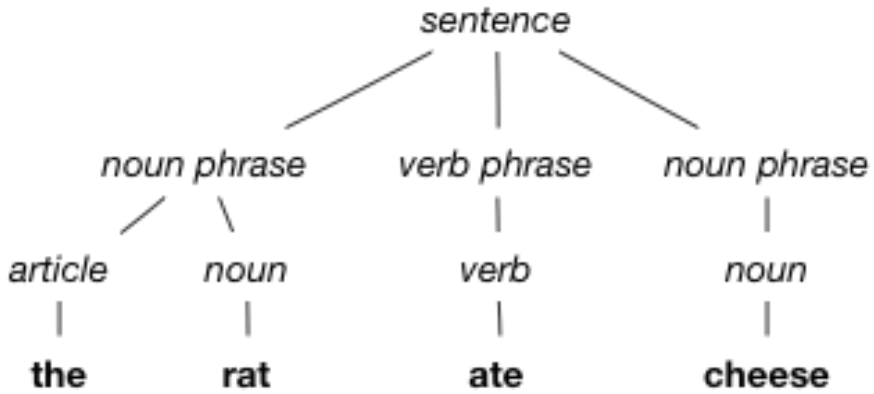
- Current web page
- 2 archived versions

How can we compute the similarity between these parsers?



Case Study: What are ASTs?

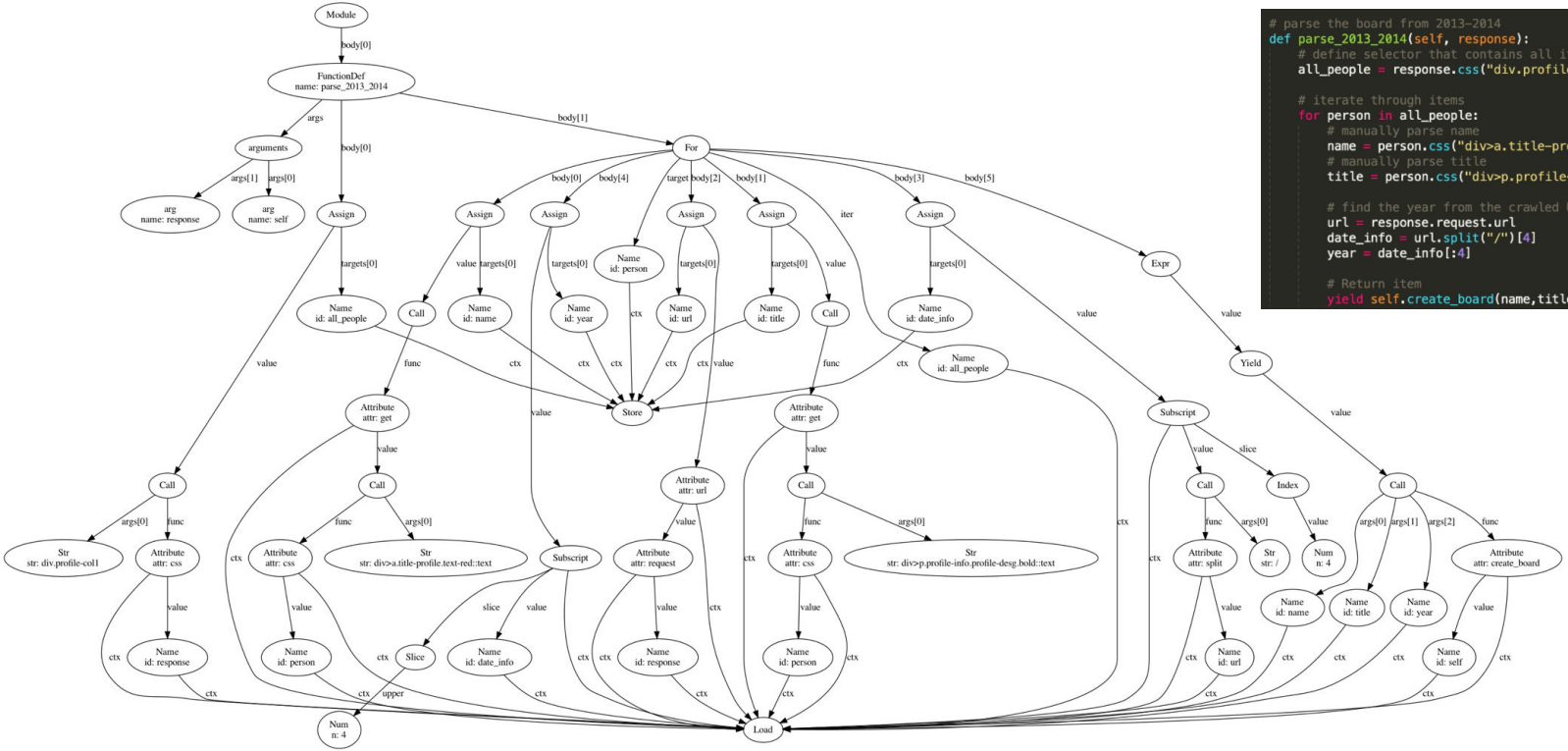
- Recognize code elements



- Understand code syntax

Case Study: What are ASTs?

Imperial College Data Science Society




```python
# parse the board from 2013-2014
def parse_2013_2014(self, response):
    # define selector that contains all items
    all_people = response.css("div.profile-col1")

    # iterate through items
    for person in all_people:
        # manually parse name
        name = person.css("div>a.title-profile.text-red::text").get()
        # manually parse title
        title = person.css("div>p.profile-info.profile-desg.bold::text").get()

        # find the year from the crawled URL
        url = response.request.url
        date_info = url.split("/")[4]
        year = date_info[:4]

        # Return item
        yield self.create_board(name,title,year)
```

Case Study: Similarity of ASTs

- Plagiarism detection
- ASTs better captures intent of code
- Tree edit distance
  - Weighted number of edit operations (insert, delete, and modify) to transform one tree to another
- Removing node normalisation

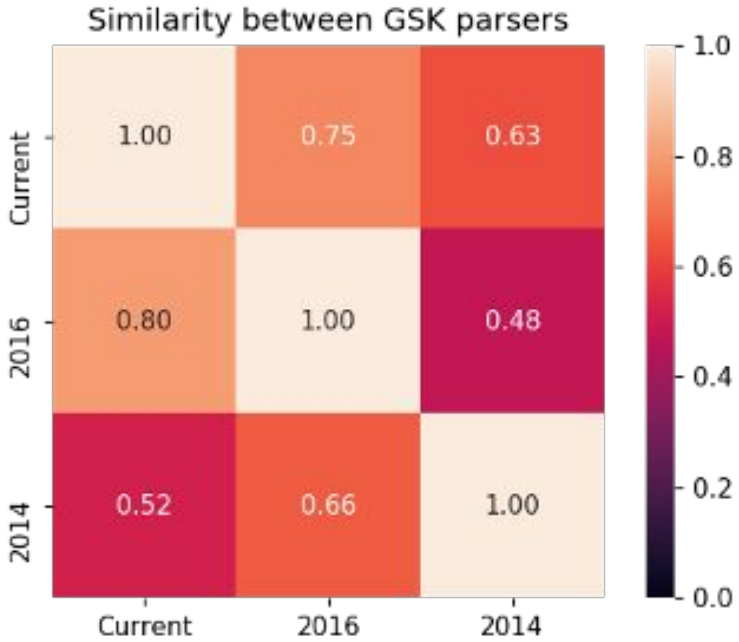
GSK: 2016

```
all_people = response.css("article.listing-item.with-image")
```

GSK: 2014

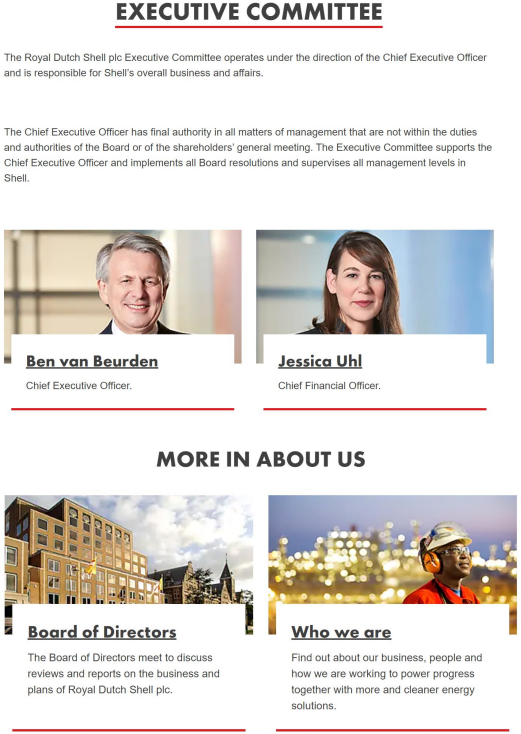
```
all_people = response.css("a.titleLink.textDecorateNone")
```

Case Study: Results



Limitations

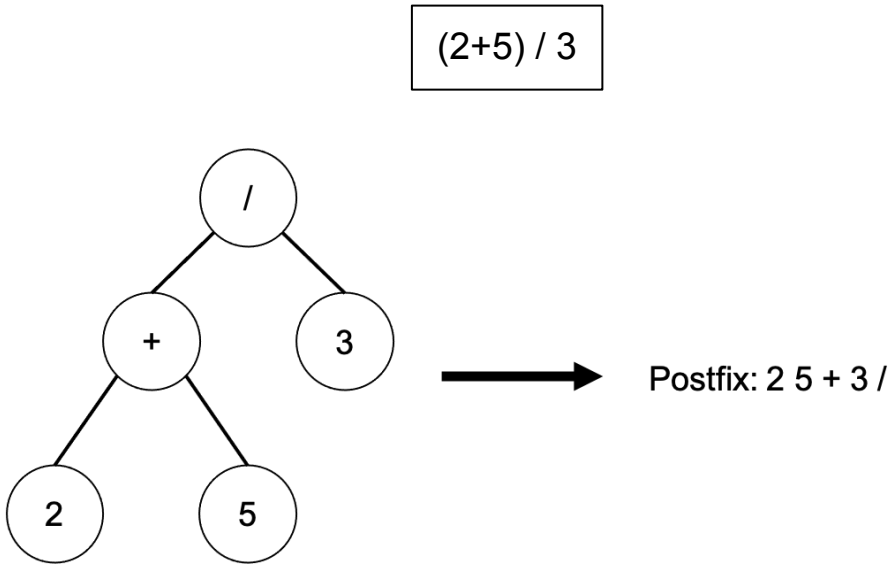
- Historical sites might have completely different structure
- Only works for relatively similar parsers
- Websites lacking structure
- Example: Shell



Alternative Method

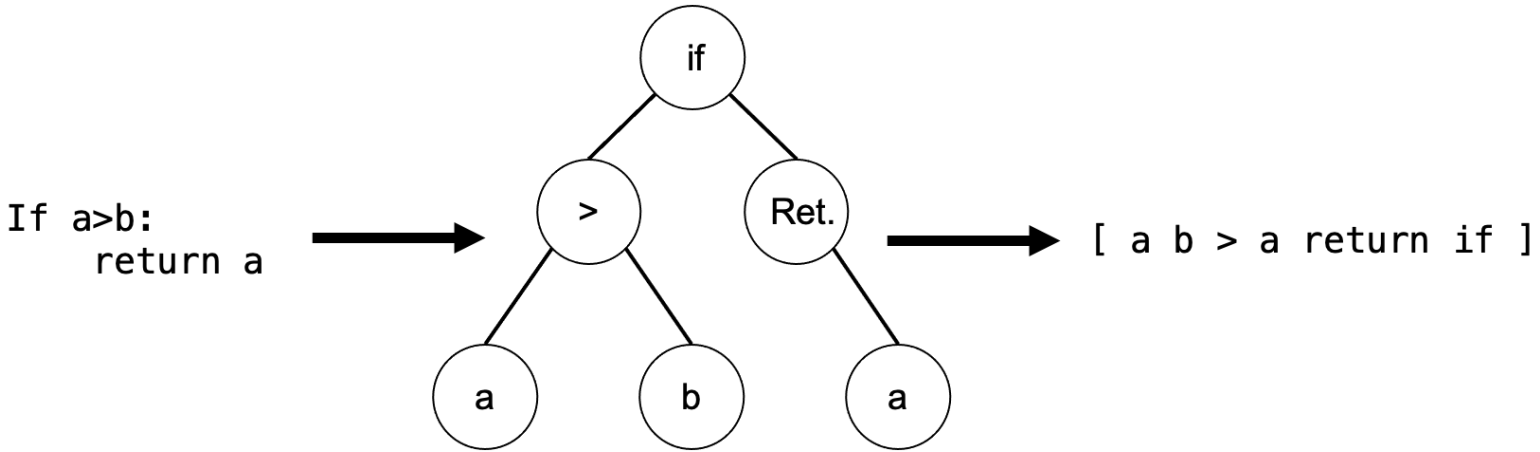
Vectorial Representation of ASTs

- Computing vector similarity is straightforward
- Standard way of representing expression trees as vectors: Postfix representation



Vectorial Representation of ASTs

- Convert code to AST, traverse AST to produce vector representation
- To compute similarity: Jaccard, Levenshtein, Dice

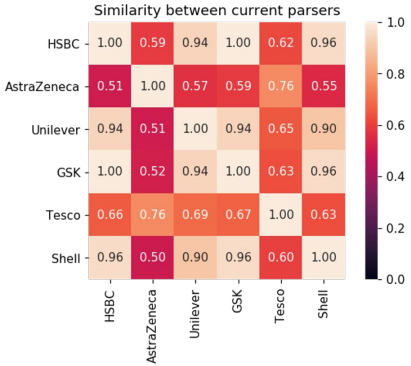


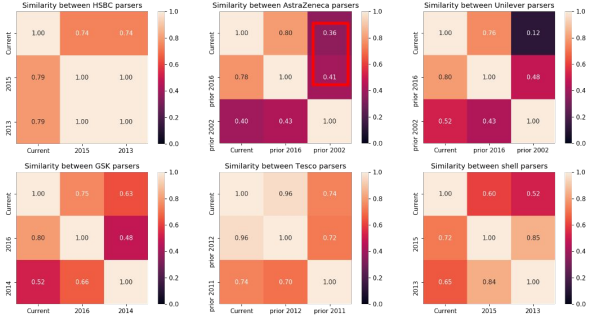
ICDSS
Imperial College Data Science Society

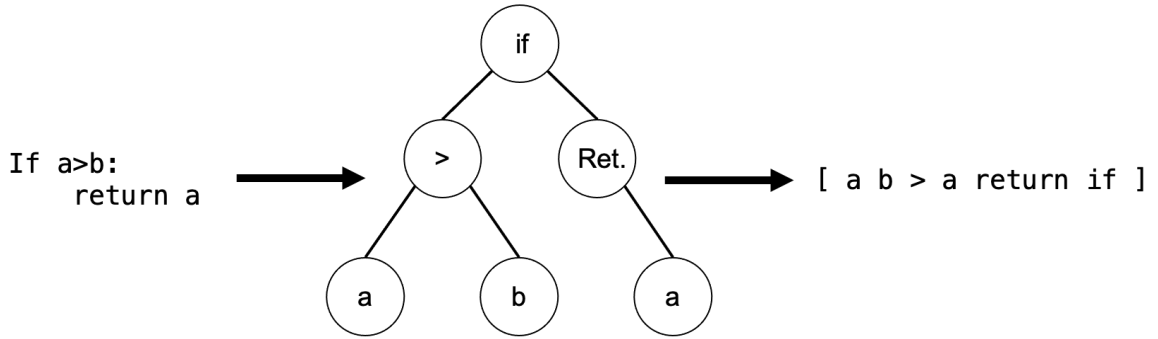
Key Takeaways & Future Work

Key Takeaways

- 18 parsers created in total
- Explored Cost Function based on Abstract Syntax Tree representation
- Post-fix representation of ASTs in vector format







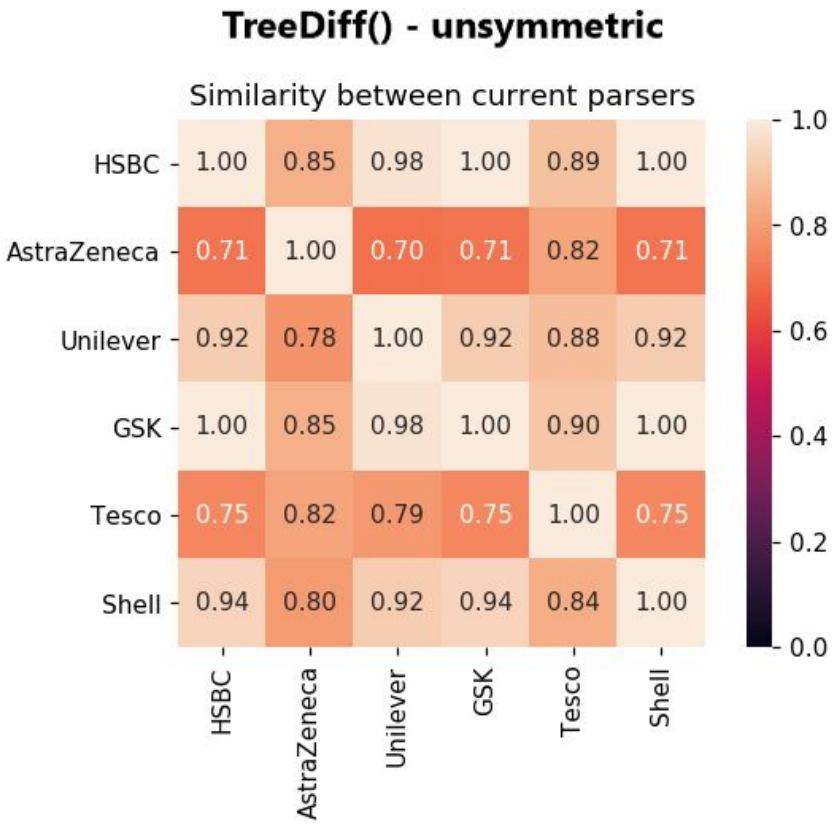
Future work

- Analysis of post-fix represented parsers using common vector comparison metrics (i.e. Levenshtein, Jaccard)
- Re-visit direct AST comparison by reproducing examples found in the literature
- Identify models that can automate the parser update (RL, triplet loss)
- Opportunities:
  - Reduce labour costs
  - Reduce time to update/create new parsers

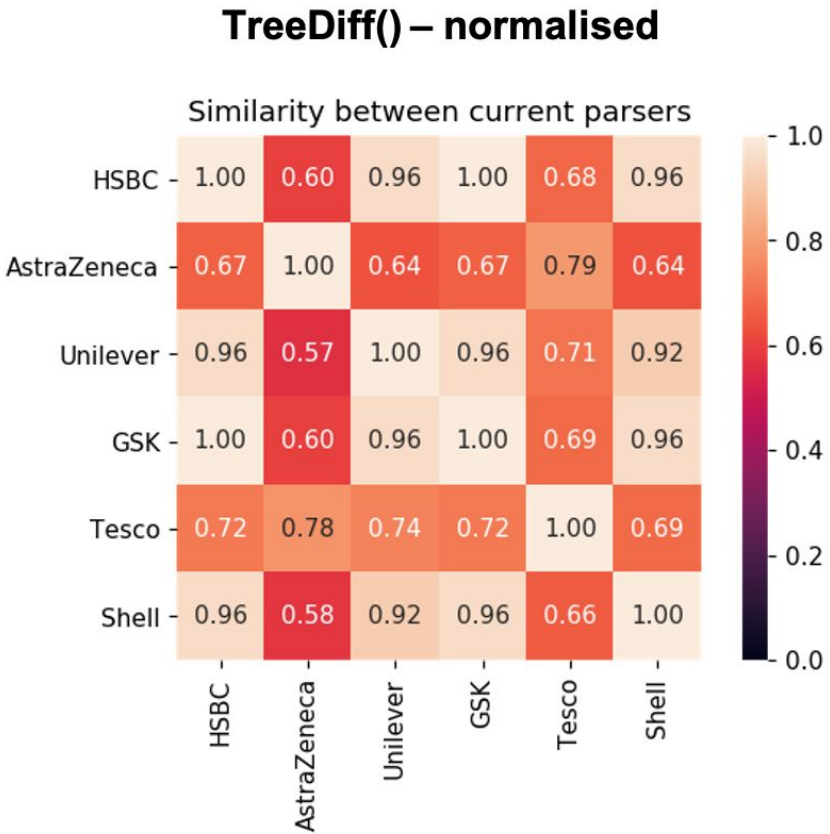
ICDSS
Imperial College Data Science Society

Thank you for listening!

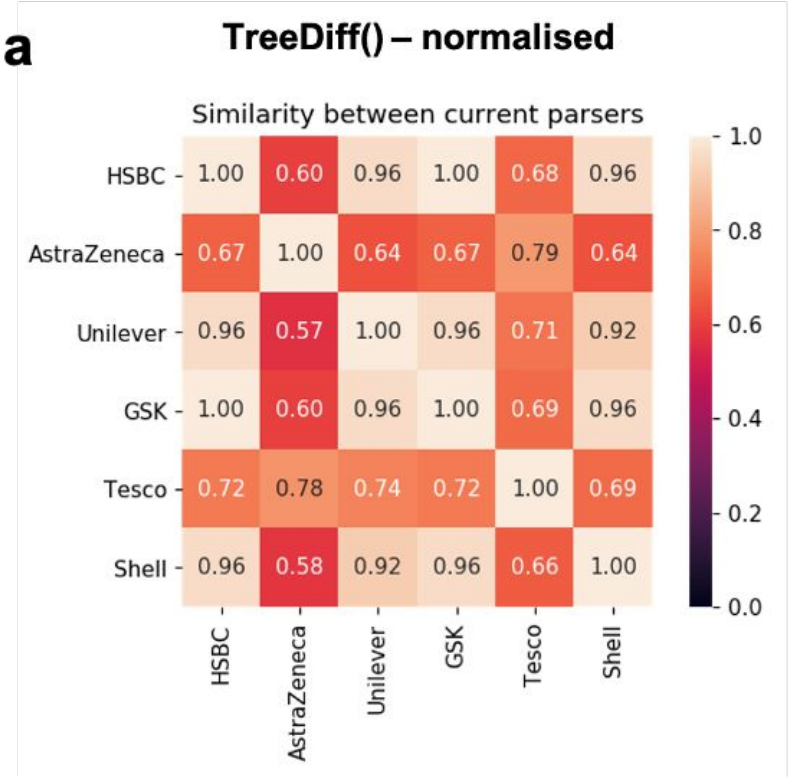
Appendix

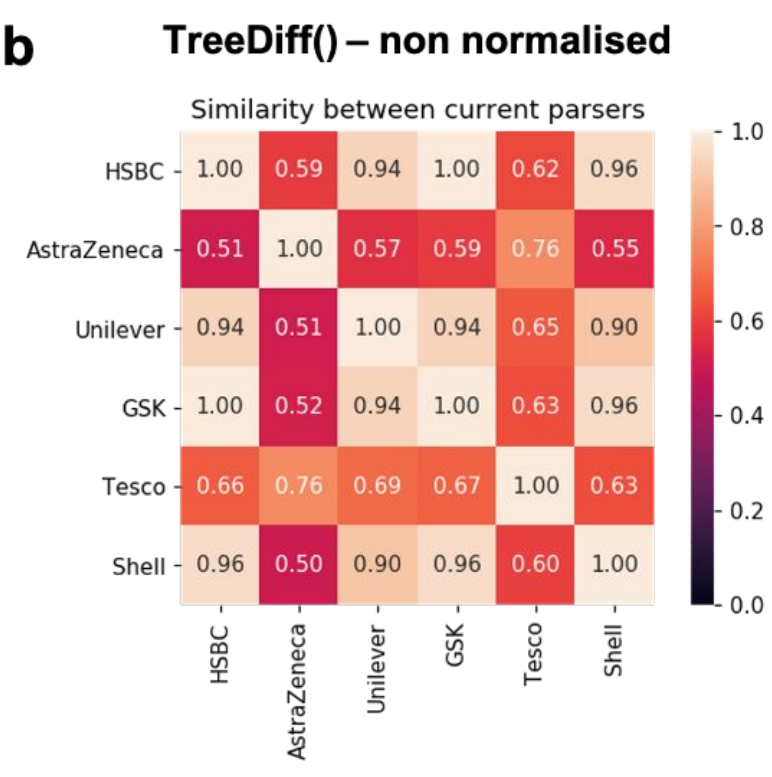

TreeDiff() - unsymmetric
Similarity between current parsers
HSBC 1.00 0.85 0.98 1.00 0.89 1.00
AstraZeneca 0.71 1.00 0.70 0.71 0.82 0.71
Unilever 0.92 0.78 1.00 0.92 0.88 0.92
GSK 1.00 0.85 0.98 1.00 0.90 1.00
Tesco 0.75 0.82 0.79 0.75 1.00 0.75
Shell 0.94 0.80 0.92 0.94 0.84 1.00
HSBC
AstraZeneca
Unilever
GSK
Tesco
Shell
1.0
0.8
0.6
0.4
0.2
0.0

Appendix


TreeDiff() – normalised
Similarity between current parsers
HSBC 1.00 0.60 0.96 1.00 0.68 0.96
AstraZeneca 0.67 1.00 0.64 0.67 0.79 0.64
Unilever 0.96 0.57 1.00 0.96 0.71 0.92
GSK 1.00 0.60 0.96 1.00 0.69 0.96
Tesco 0.72 0.78 0.74 0.72 1.00 0.69
Shell 0.96 0.58 0.92 0.96 0.66 1.00
HSBC
AstraZeneca
Unilever
GSK
Tesco
Shell
1.0
0.8
0.6
0.4
0.2
0.0

Appendix

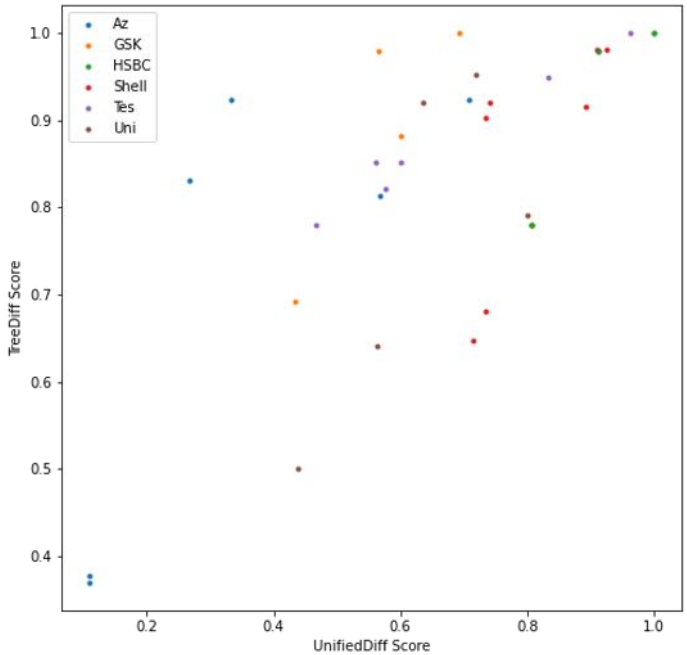

a
TreeDiff() – normalised
Similarity between current parsers
HSBC 1.00 0.60 0.96 1.00 0.68 0.96
AstraZeneca 0.67 1.00 0.64 0.67 0.79 0.64
Unilever 0.96 0.57 1.00 0.96 0.71 0.92
GSK 1.00 0.60 0.96 1.00 0.69 0.96
Tesco 0.72 0.78 0.74 0.72 1.00 0.69
Shell 0.96 0.58 0.92 0.96 0.66 1.00
HSBC
AstraZeneca
Unilever
GSK
Tesco
Shell
1.0
0.8
0.6
0.4
0.2
0.0


b
TreeDiff() – non normalised
Similarity between current parsers
HSBC 1.00 0.59 0.94 1.00 0.62 0.96
AstraZeneca 0.51 1.00 0.57 0.59 0.76 0.55
Unilever 0.94 0.51 1.00 0.94 0.65 0.90
GSK 1.00 0.52 0.94 1.00 0.63 0.96
Tesco 0.66 0.76 0.69 0.67 1.00 0.63
Shell 0.96 0.50 0.90 0.96 0.60 1.00
HSBC
AstraZeneca
Unilever
GSK
Tesco
Shell
1.0
0.8
0.6
0.4
0.2
0.0

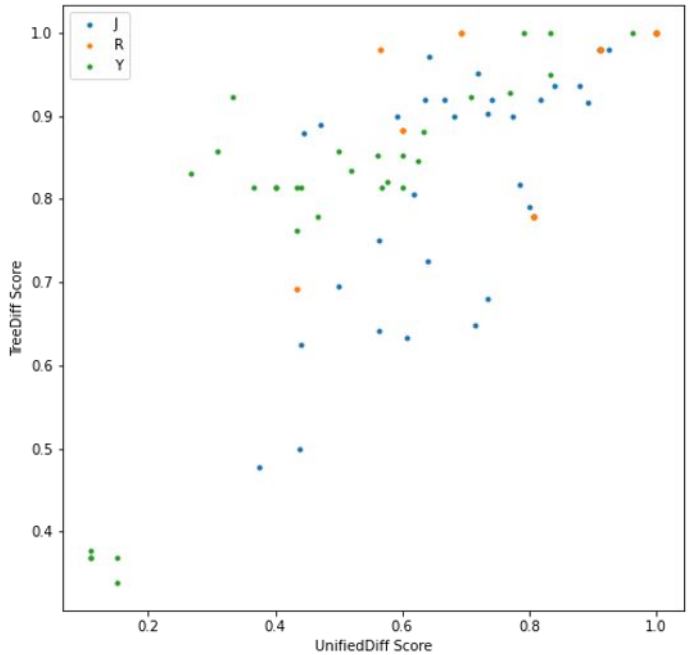
Appendix

Imperial College Data Science Society

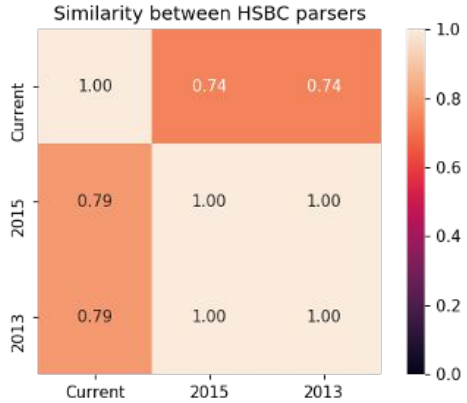
a

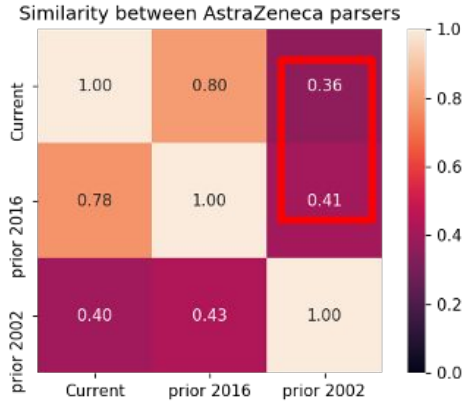


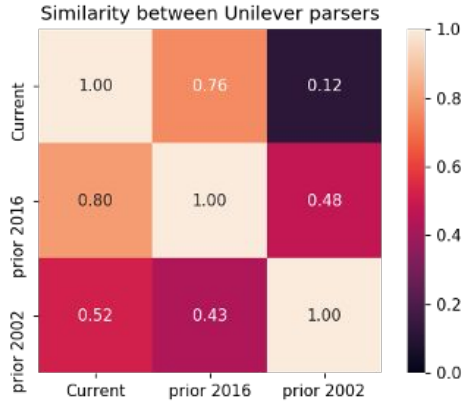
b

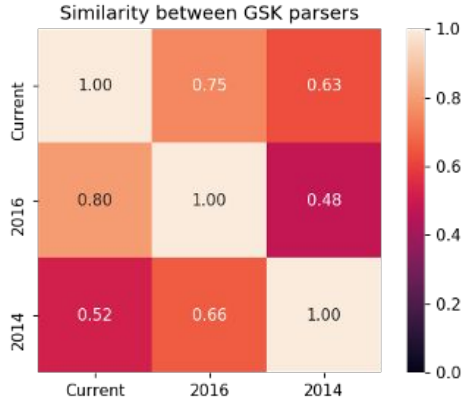


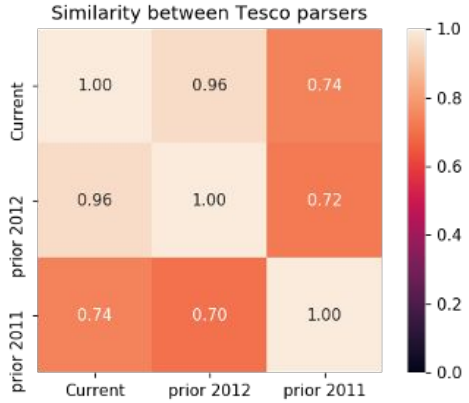
Appendix

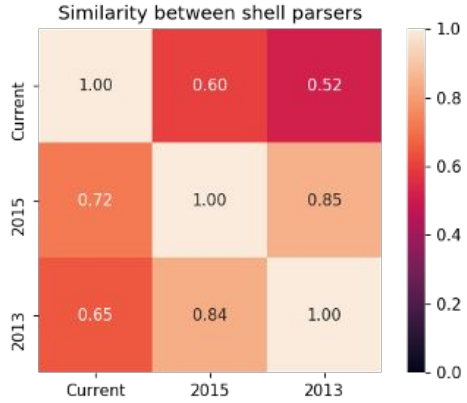

Similarity between HSBC parsers
Current
2015
2013
1.00
0.74
0.74
0.79
1.00
1.00
0.79
1.00
1.00
Current
2015
2013
1.0
0.8
0.6
0.4
0.2
0.0


Similarity between AstraZeneca parsers
Current
prior 2016
prior 2002
1.00
0.80
0.36
0.78
1.00
0.41
0.40
0.43
1.00
Current
prior 2016
prior 2002
1.0
0.8
0.6
0.4
0.2
0.0


Similarity between Unilever parsers
Current
prior 2016
prior 2002
1.00
0.76
0.12
0.80
1.00
0.48
0.52
0.43
1.00
Current
prior 2016
prior 2002
1.0
0.8
0.6
0.4
0.2
0.0


Similarity between GSK parsers
Current
2016
2014
1.00
0.75
0.63
0.80
1.00
0.48
0.52
0.66
1.00
Current
2016
2014
1.0
0.8
0.6
0.4
0.2
0.0


Similarity between Tesco parsers
Current
prior 2012
prior 2011
1.00
0.96
0.74
0.96
1.00
0.72
0.74
0.70
1.00
Current
prior 2012
prior 2011
1.0
0.8
0.6
0.4
0.2
0.0


Similarity between shell parsers
Current
2015
2013
1.00
0.60
0.52
0.72
1.00
0.85
0.65
0.84
1.00
Current
2015
2013
1.0
0.8
0.6
0.4
0.2
0.0