



CENTER FOR TEXTUAL STUDIES AND DIGITAL HUMANITIES

DIGH 400 - Introduction to Digital Humanities Research

Fall Semester 2013

Week 12

Today's Class

- UK Web Archive
- Group Project Work
- XSLT (continued)
- Stylistic analysis etc...

DIGH 400 - Introduction to Digital Humanities Research

[UK Web Archive](#)

Group Project Work - Using Drupal 7

Getting started with Drupal 7

- consider content you might want to include in the site
 - categories, sub-categories
 - latest news, conferences, journals, websites, job listings...
 - software available for DH
 - original content including reviews, essays, articles, photos, videos...
- think about any modules you need to extend the functionality of the site to fit your content
- consider site structure for menus, blocks, layout of content
- adding polls, forums, blogs to the site
- who is responsible for each section of the site?
- Google Groups discussion available for group chats

Group Project Work

<http://www.digital-humanities.com>

DIGH 400 - Introduction to Digital Humanities Research

You will need a laptop tomorrow!

DIGH 400 - Introduction to Digital Humanities Research

XSLT Demos

XSLT - <xsl:if>

- conditional test within the template for certain conditions in the XML

```
<xsl:if test="year > 1929">
```

```
</xsl:if>
```

- value of 'test' attribute contains the expression to be tested

DIGH 400 - Introduction to Digital Humanities Research

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
  <body>
  <h2>Collection</h2>
  <table>
    <tr>
      <th>Title</th>
      <th>Author</th>
      <th>Year</th>
    </tr>
    <xsl:for-each select="catalogue/book">
      <xsl:sort select="title"/>
      <xsl:if test="year > 1937">
        <tr>
          <td><xsl:value-of select="title"/></td>
          <td><xsl:value-of select="author"/></td>
          <td><xsl:value-of select="year"/></td>
        </tr>
      </xsl:if>
    </xsl:for-each>
  </table>
  </body>
  </html>
</xsl:template>

</xsl:stylesheet>
```

[Example](#)

XSLT - <xsl:choose>

- we can use <xsl:choose> with <xsl:when> or <xsl:otherwise> to test multiple conditions

```
<xsl:choose>
  <xsl:when test="expression">
    ... some output ...
  </xsl:when>
  <xsl:otherwise>
    ... some output ....
  </xsl:otherwise>
</xsl:choose>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<catalogue>
  <book>
    <title>Evil Under the Sun</title>
    <author>Agatha Christie</author>
    <country>UK</country>
    <publisher>Collins Crime
Club</publisher>
    <price>7 shillings and sixpence</price>
    <year>1941</year>
  </book>
  .
  .
</catalogue>
```

DIGH 400 - Introduction to Digital Humanities Research

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
  <body>
  <h2>Collection</h2>
  <table>
    <tr>
      <th>Title</th>
      <th>Author</th>
    </tr>
    <xsl:for-each select="catalogue/book">
      <tr>
        <td><xsl:value-of select="title"/></td>
        <xsl:choose>
          <xsl:when test="year > 1941">
            <td class="post">After: <xsl:value-of select="author"/></td>
          </xsl:when>
          <xsl:otherwise>
            <td class="pre">Before: <xsl:value-of select="author"/></td>
          </xsl:otherwise>
        </xsl:choose>
      </tr>
    </xsl:for-each>
  </table>
  </body>
</html>
</xsl:template>

</xsl:stylesheet>
```

[Example](#)

XSLT - <xsl:apply-templates>

- add a template to a current element or its child nodes
- a 'select' attribute will only process the child element specified in the value
- use the 'select' attribute to specify order of child node processing

DIGH 400 - Introduction to Digital Humanities Research

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

```
<xsl:template match="/">
  <html>
  <body>
  <h2>Collection</h2>
  <xsl:apply-templates/>
  </body>
  </html>
</xsl:template>
```

```
<xsl:template match="book">
  <p>
  <xsl:apply-templates select="title"/>
  <xsl:apply-templates select="author"/>
  </p>
</xsl:template>
```

```
<xsl:template match="title">
  Title: <span class="title"><xsl:value-of select="."/></span>
  <br />
</xsl:template>
```

```
<xsl:template match="author">
  Author: <span class="author"><xsl:value-of select="."/></span>
  <br />
</xsl:template>
```

```
</xsl:stylesheet>
```

[Example](#)

```
<xsl:template match="title">
  <xsl:choose>
    <xsl:when test="../price<10">
      <span style="color:#ff00ff">
        <xsl:value-of select="."/>
      </span>
    </xsl:when>
    <xsl:otherwise>
      <span>Price too high: <xsl:value-of select="."
/></span>
    </xsl:otherwise>
  </xsl:choose>
</xsl:template>
```

[Example](#)

XSLT - client side

- XML and XSL can be transformed with a browser
- javascript can also be used to perform the transformation
 - allows browser-specific testing
 - can apply different style sheets as needed
- sample javascript process may include
 - load defined XML and XSL files
 - test current browser type
 - perform specified functions relative to browser type
 - output styled document to specified container
- javascript caveat: will not work in a browser lacking XML parser
- XML parsers in PHP such as SAX parser

[Example](#)

XSLT - server side

- transform the XML to XHTML on the server
- often known as server-side processing
- provides a cross-browser solution
- many different languages include support for XSLT including PHP
- transform offline and upload to web server
- integrated development environment (IDE) for editing and transformation

Information on TEI stylesheets can be found at

<http://www.tei-c.org/Tools/Stylesheets/>

Stylistic Analysis

Overview

- consideration of patterns in style
- influence of style on readers' perceptions
- disciplinary concerns of literary and linguistic interpretation
- patterns using computational stylistics

Stylistic Analysis

Analysing Shakespeare (Hugh Craig, 'A Companion to Digital Humanities')

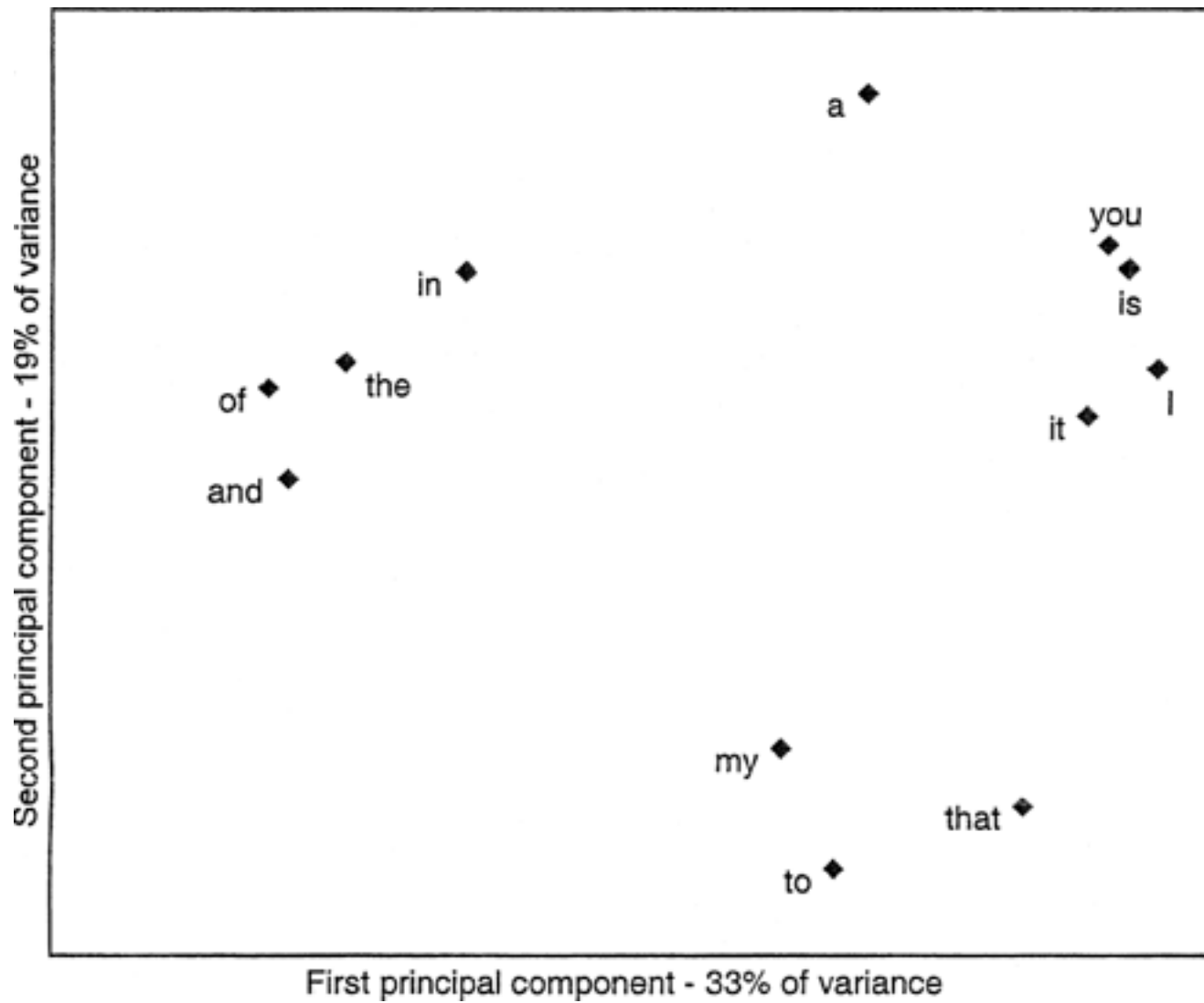
- analysis of 25 of 38 plays in standard Complete Shakespeare
- conduct analysis of the 12 most commonly used words
- Principal Components Analysis (PCA)
- PCA simplifies data by finding new variables that represent most relationships
- new composite variables still represent the variation in a data set
- PCA vectors are an extension of this principle

Stylistic Analysis

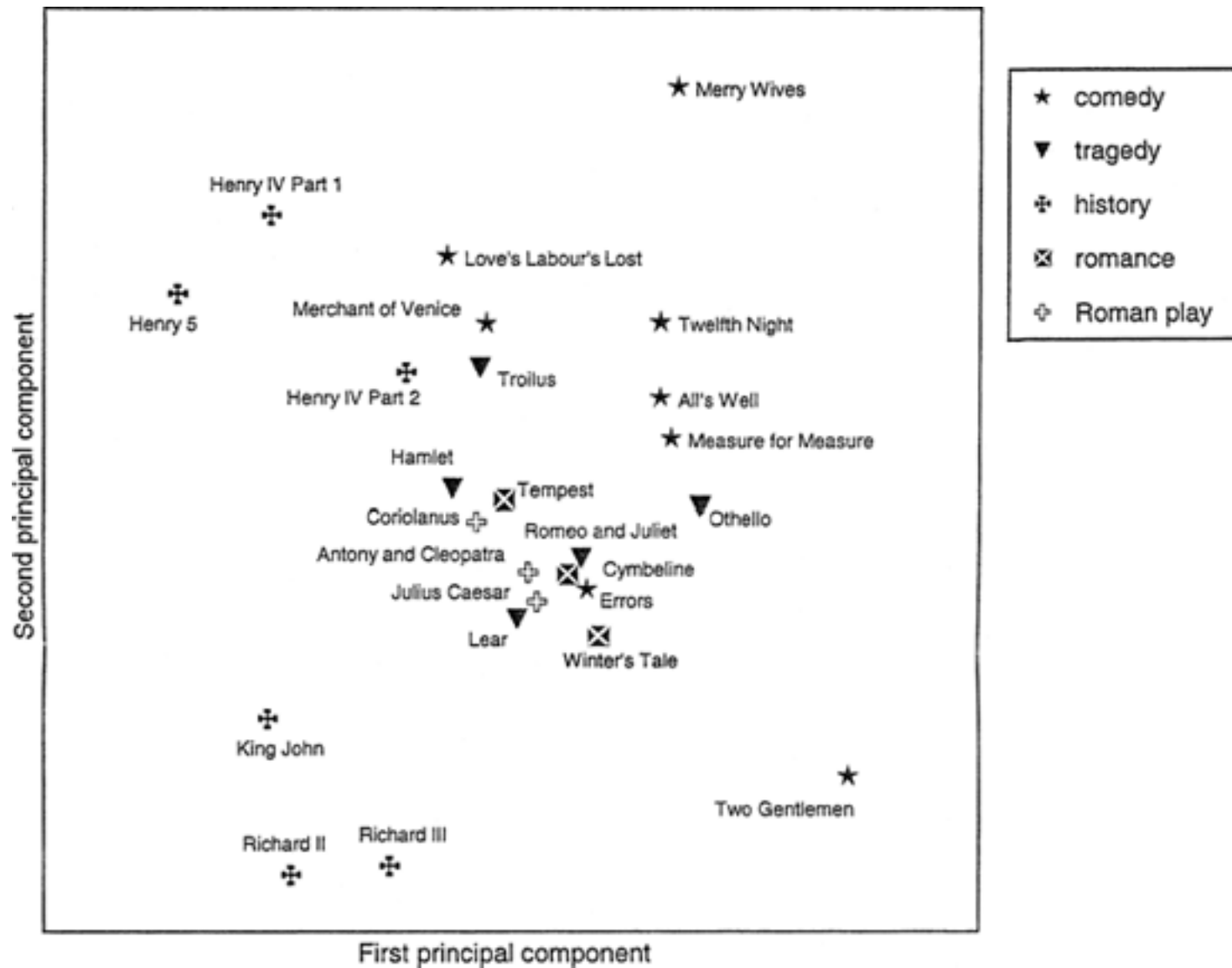
Analysing Shakespeare (Hugh Craig, 'A Companion to Digital Humanities')

- relative weightings used to create vector which accounts for greatest proportion of variance
- then the second vector, and so on...
- strong associations between variables will lead to the first few composite variables accounting for most of the results

DIGH 400 - Introduction to Digital Humanities Research



DIGH 400 - Introduction to Digital Humanities Research



Stylistic Analysis

Considerations

- consider how constant such patterns are when we remove or add a play
- effect of modifying variables on the results
- why did we choose the given parameters for this particular study?

Stylistic Analysis

Computational Stylistics

- extensive and perhaps best suited to large scale comparisons
- changes in language over time, eg: a writer's career, historical period...
- provision of a class of evidence not otherwise accessible
- not a solution by itself, requires knowledge of humanities and statistical techniques

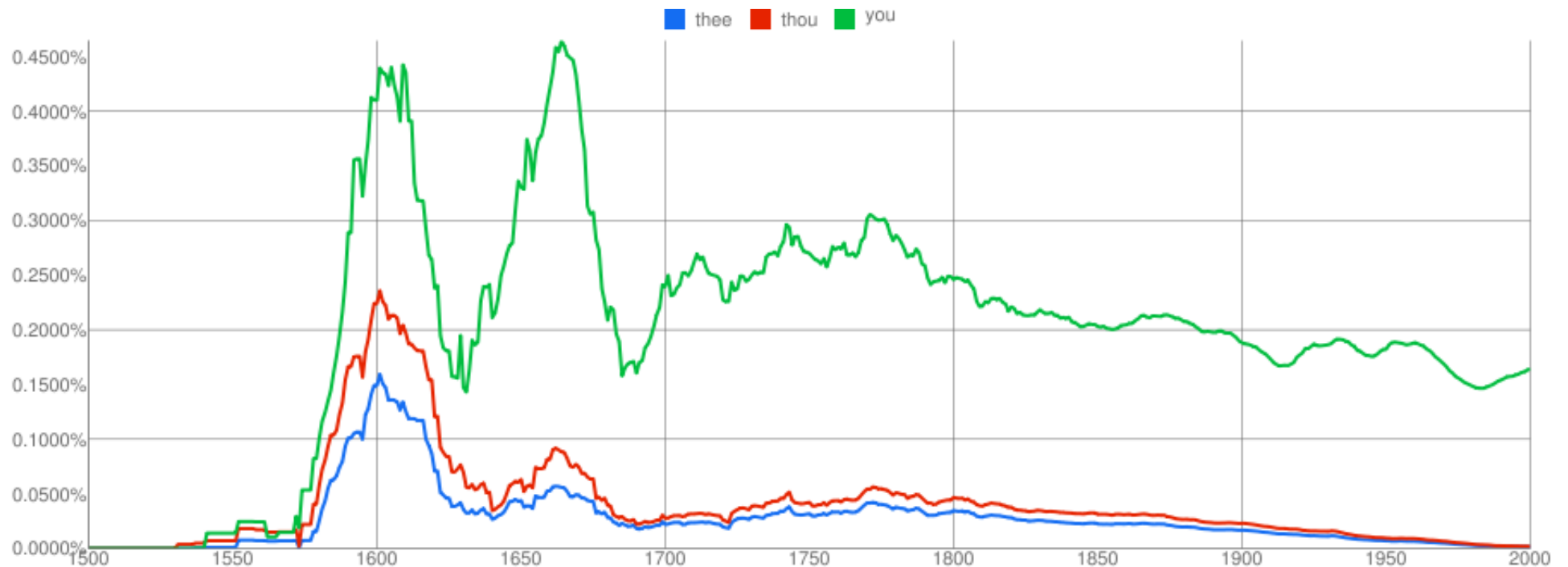
Google Books Ngram Viewer

What does it actually do?

- searches a selected corpus of books for a user selected set of phrases
- select years for search
- apply smoothing to specify moving average of results returned
- search Google Books from returned set of results
- use the returned raw data to create your own visualisations, tests...

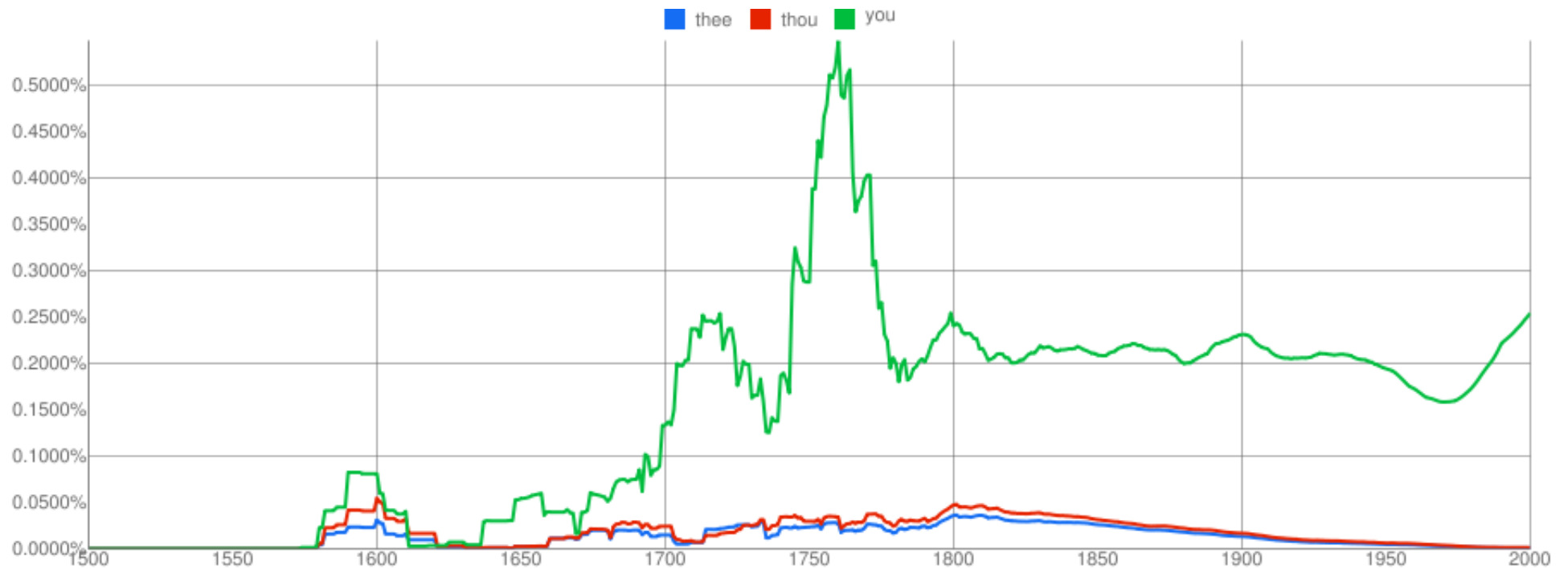
[Try a few tests](#)

DIGH 400 - Introduction to Digital Humanities Research



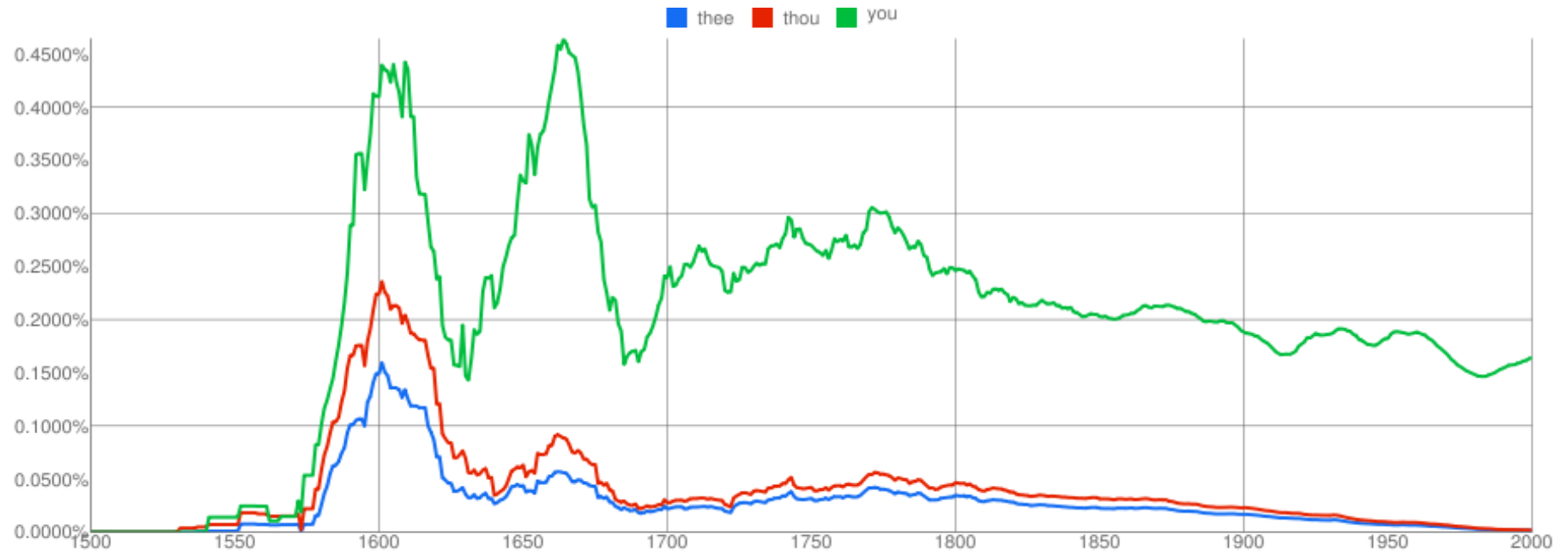
British English Corpus results for 1500 -
2000

DIGH 400 - Introduction to Digital Humanities Research

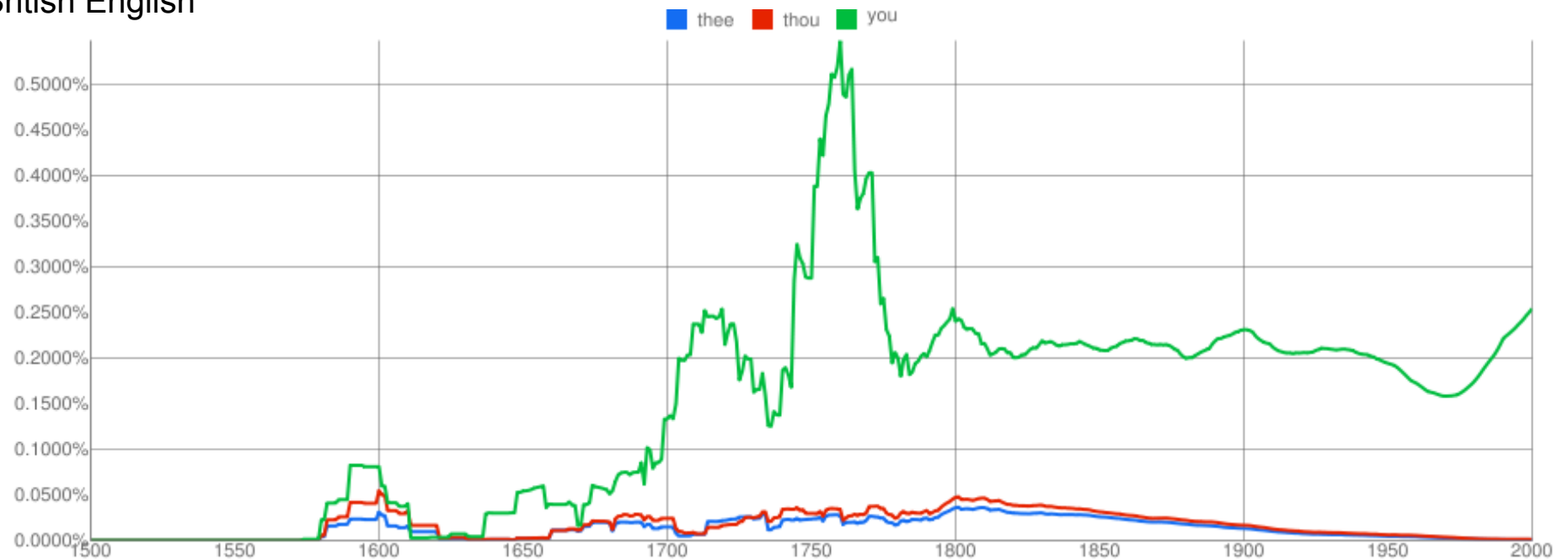


American English Corpus results for 1500 -
2000

DIGH 400 - Introduction to Digital Humanities Research



British English



American English

Google+ Ripples

- allows users to quickly and easily visualise post sharing
- describes itself as

“a way to visualize the impact of any public post.”

- nested circles show generations of the shared post
- use the circles to view how the post was shared over time

[Further Info](#)

Gephi

What does it actually do? (see <https://gephi.org/about/>)

- software for network visualisation and analysis
- helps data analysts to intuitively reveal patterns and trends...
- highlight outliers and tell stories with data
- displays large graphs in real time to speed up exploration
- built-in functionalities and flexible architecture for networks to
 - explore, analyse, spatialise, filter, cluster, manipulate & export

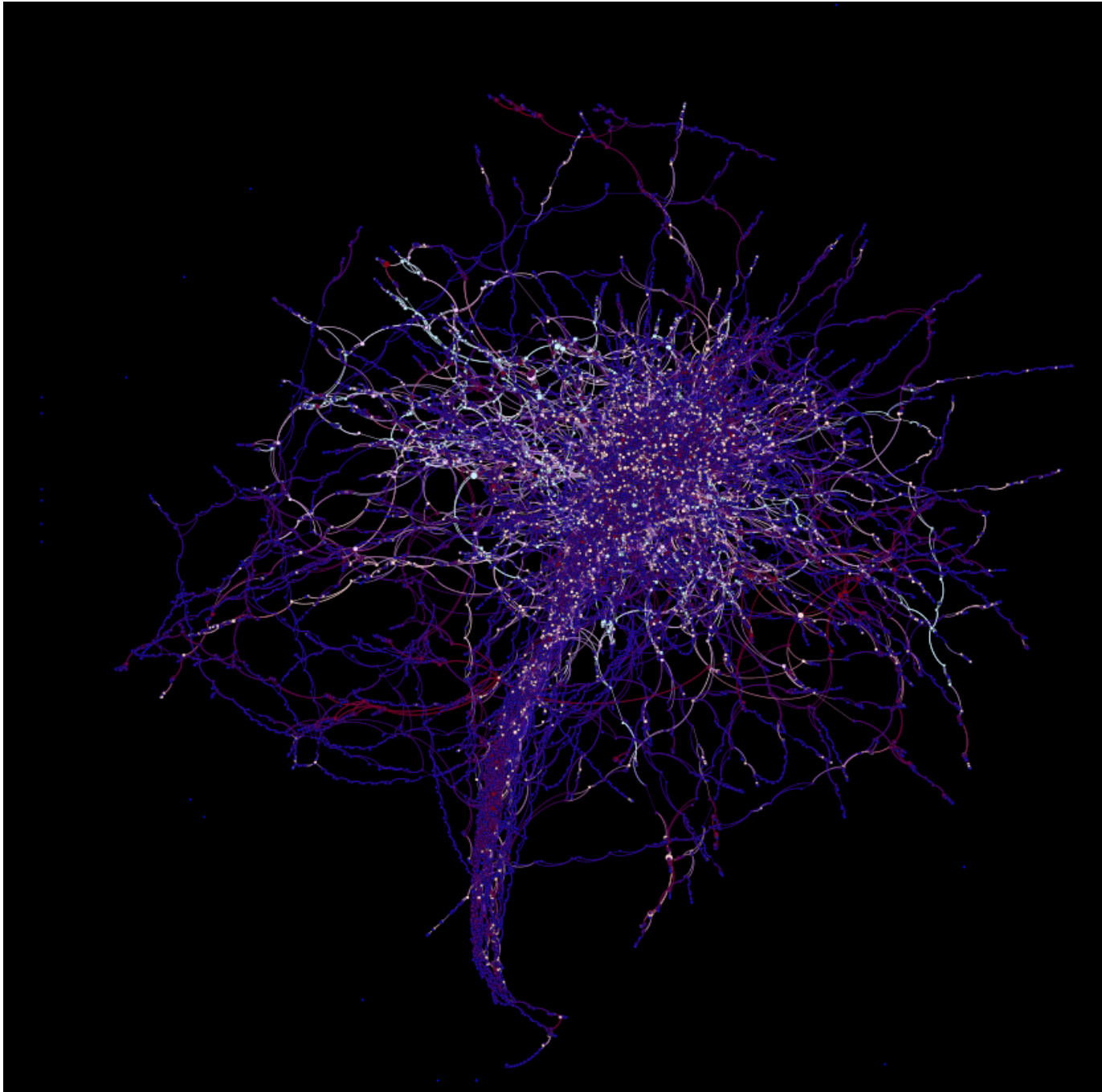
DIGH 400 - Introduction to Digital Humanities Research



[Spreadsheet data](#)

DIGH 400 - Introduction to Digital Humanities Research

[Further Info](#)



And now for something completely different...but somewhat relevant, and a lot of fun

Twitterology (?) - a bit of fun...

- used in diverse fields such as linguistics, sociology, and psychology
- analyse and examine language usage, patterns, location specific terms...
- immediacy and immensity
- University of Texas research into tweets and streams emanating from Libya and Egypt
- noticeable increase in usage following certain political events
- language patterns could also be discerned relative to such events

NYTimes Article on "[Twitterology: A New Science](#)"

[Carnegie Mellon Article](#)