Enhancing Search for complex historical texts

Desmond Schmidt

University of Queensland

June 23, 2016
The importance of search

- The consumer of an online edition is now a user.
- The user wants to search and browse more than read.
- Search and browse are the first functions implemented in any digital scholarly edition (e.g. TAPAS, SADE, ELEC, EVT)
- Allows the user to explore the edition in depth
What is ‘search’?

- Searching the content of transcribed historical documents
- *Not* searching of metadata about whole documents
- *Not* about sharing search results between sites
What’s a ‘digital scholarly edition’?

- Originally it meant *critical* editions created by scholars.
- Now it tends to mean digital editions of literary texts that facilitate scholarly interaction.
- **Examples:** an edition of Mary Shelley’s *Frankenstein* or Goethe’s *Faust.*
Methods of searching

1. Actually searching the content of each file
   – e.g. Unix rgrep
   Cons: Too slow, uses too many resources

2. Searching the XML structure directly
   – e.g. XQuery (used in eXist, MarcLogic)
   Cons: Harder to use for mixed content,
   users mostly don’t need to search XML structure

3. Indexed search of content using some XML structure
   – e.g. Lucene (used in Solr, ElasticSearch)
   Pros: Fast, needs fewer resources
Criterion 1: Find words split into parts

1. XML
2. plain text with spaces between former elements
3. plain text without spaces between former elements

1. <hi rend="dul">T</hi>empest
2. T empest
3. Tempest

1. strai<del>gh</del>tended
2. strai gh tened
3. straightened
Criterion 2: Find words broken over lines

Problems:
1. Sometimes a hyphen at line-end should be removed
2. Sometimes not
3. Not always clear what to do
4. Hyphenation is not always marked consistently

Two possible strategies:
1. Do we delete line-breaks and lose information? (60% do)
2. Or encode them and join them up when indexing? (40% do)
Criterion 3: Find deleted text

XML: in the author's <del>early</del> youth
Find: "in the author's early youth"
Criterion 4: Find inline-variants in their correct context

**XML:** Who into fame by martial deeds <subst><del>shall</del><add>would</add></subst> rise

1. Find: "Who into fame by martial deeds shall rise"
2. Find: "Who into fame by martial deeds would rise"
3. *Not* find: "Who into fame by martial deeds shall would rise"
Criterion 5: Perform literal search

1. **Keyword** search: boolean AND of terms
e.g. Find: “Wet the dusty pall with tears”
Finds all documents containing “Wet” AND “the” AND “dusty”
AND “pall” AND “with” AND “tears”
– 3 matches in Shelley-Godwin archive

2. **Literal** search: “Wet the dusty pall with tears”
Finds all documents that contain the phrase:
“Wet the dusty pall with tears”
– 1 match in Shelley-Godwin archive
Criterion 6: Highlight text found in the document

Search

"Wet the dusty pall with tears"

Result (digest)

Match highlighted in document

Strew oh strew
Hair, not yew!
Wet the dusty pall with tears, not dew!
Be the faded flowers
Of Deaths bare bowers
Spread on the corpse of the King of Hours!
Support for search criteria in 30 select DSEs
Looking for a technical solution

To fix these problems we need to ensure that:

1. *All* the text in the document is findable
2. All the text that can be found is *in the document*
Looking for a technical solution

To fix these problems we need to ensure that:

1. *All* the text in the document is findable
2. All the text that can be found is *in the document*

Although most of these problems can be solved through better communication between humanists and technologists, some fundamental problems remain . . .
How to search many versions without duplicate matches?

Ideally we want one match for “We list the laborings”, not six.
And knowing now his truth, shee weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps

And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps
And knowing now his truth, she weeps

Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire
Do bitterest penance in her ire

The waywardness of will
The waywardness of will
The waywardness of will
The waywardness of will
The waywardness of will
The waywardness of will
The waywardness of will
The waywardness of will

Against her wayward will
Against her wayward will
Against her wayward will
Against her wayward will
Against her wayward will
Against her wayward will
Against her wayward will
Against her wayward will

That could
That could
That could
That could
That could
That could
That could
That could

So try
So try
So try
So try
So try
So try
So try
So try

As with fire!
As with fire!
As with fire!
As with fire!
As with fire!
As with fire!
As with fire!
As with fire!
And knowing now his truth, shee weeps would
And blaming, with introverted ire fosters a keen inward ire
Do bitterest penance in her ire,
The waywardness of will
Against her wayward will self, that could
So try it him, as with fire!
Idea: represent the text non-linearly (variant-graph)

And knowing now his truth, shee would weeps.

Do bitterest penance in her ire,
fosters a keen inward ire
blames, with introverted ire

Against her wayward self, that could will
The waywardness of will

So try him, as with fire!
Multi-version document

The same graph can be represented linearly as a ‘Multi-version document’ (Schmidt-Colomb 2009):

1-3: And knowing now his truth, shee; 1-2: weeps; 3: would; 1-2: And; 1: blames, with introverted ire; 2: fosters a keen inward ire; 3: Do bitterest penance in her ire,,; 1: The waywardness of will; 2-3: Against her wayward; 2: will; 3: self; 1-3: that could; 1-3: So try; 1: it; 2-3: him; 1-3: , as with fire!

NB: This example has been simplified by removing formatting (which is represented separately from the text)
Specifying matches via IETF RFC5147

1. In plain text: http://site.org/doc.txt#char=51,96
   And knowing now his truth, shee weeps
   And blames, with introverted ire
   The waywardness of will that could
   So try it, as with fire!

2. In an MVD: http://site.org/doc.mvd#char=190,230
   1-3: And knowing now his truth, shee; 1-2: weeps; 3: would; 1-2: And; 1: blames, with introverted ire; 2: fosters a keen inward ire; 3: Do bitterest penance in her ire,; 1: The waywardness of will; 2-3: Against her wayward; 2: will; 3: self; 1-3: that could; 1-3: So try ; 1: it ; 2-3: him ; 1-3: , as with fire!

   The versions of the match are those shared by the start and end of the range, which contain the search expression. So the match here will be either: ‘that could So try him, as with fire!’ (2-3) or ‘that could So try it, as with fire!’ (1).
Advantages of this approach

1. Precise specification of matches even for complex texts
2. Standards-compliant
3. Can be implemented using existing search engines, e.g. Lucene
4. Produces a minimum of matches with no redundancy