Lecture 6: Web Scraping

Pilsung Kang

School of Industrial Management Engineering

Korea University

AGENDA

01 Prerequisites

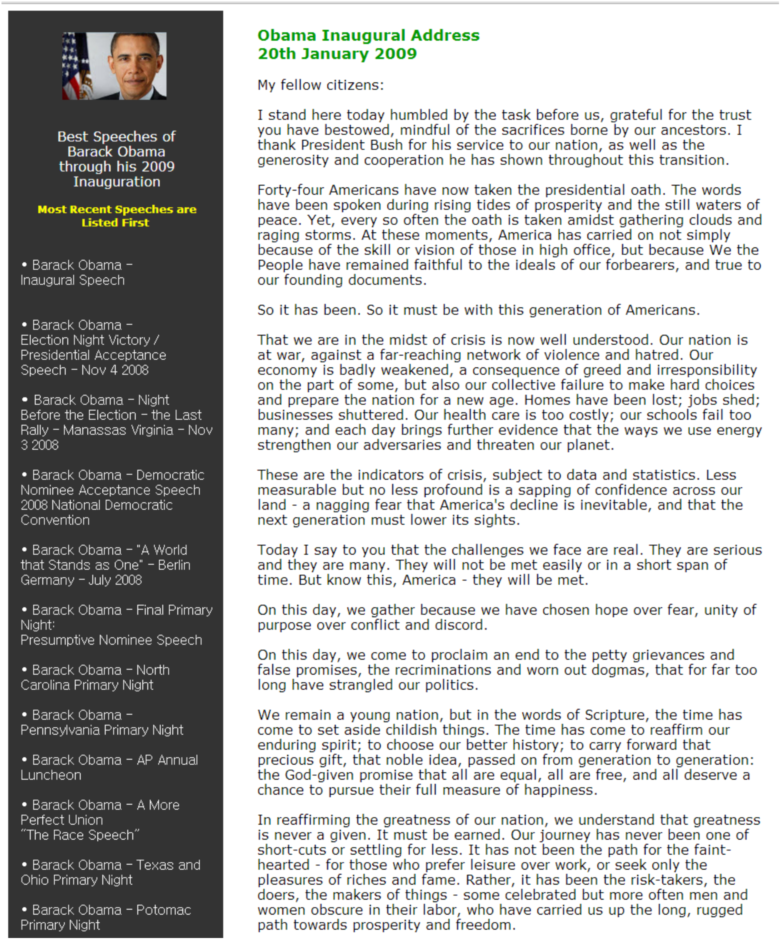
02 Web Scraping: arXiv Research Papers

03 Web Scraping: Movie Reviews

Web Scraping

- Need to understand HTML/XML structures

What we see with a browser



Best Speeches of Barack Obama through his 2009 Inauguration

Most Recent Speeches are Listed First

- Barack Obama - Inaugural Speech
- Barack Obama - Election Night Victory / Presidential Acceptance Speech - Nov 4 2008
- Barack Obama - Night Before the Election - the Last Rally - Manassas Virginia - Nov 3 2008
- Barack Obama - Democratic Nominee Acceptance Speech 2008 National Democratic Convention
- Barack Obama - "A World that Stands as One" - Berlin Germany - July 2008
- Barack Obama - Final Primary Night: Presumptive Nominee Speech
- Barack Obama - North Carolina Primary Night
- Barack Obama - Pennsylvania Primary Night
- Barack Obama - AP Annual Luncheon
- Barack Obama - A More Perfect Union "The Race Speech"
- Barack Obama - Texas and Ohio Primary Night
- Barack Obama - Potomac Primary Night

Obama Inaugural Address
20th January 2009

My fellow citizens:

I stand here today humbled by the task before us, grateful for the trust you have bestowed, mindful of the sacrifices borne by our ancestors. I thank President Bush for his service to our nation, as well as the generosity and cooperation he has shown throughout this transition.

Forty-four Americans have now taken the presidential oath. The words have been spoken during rising tides of prosperity and the still waters of peace. Yet, every so often the oath is taken amidst gathering clouds and raging storms. At these moments, America has carried on not simply because of the skill or vision of those in high office, but because We the People have remained faithful to the ideals of our forbearers, and true to our founding documents.

So it has been. So it must be with this generation of Americans.

That we are in the midst of crisis is now well understood. Our nation is at war, against a far-reaching network of violence and hatred. Our economy is badly weakened, a consequence of greed and irresponsibility on the part of some, but also our collective failure to make hard choices and prepare the nation for a new age. Homes have been lost; jobs shed; businesses shuttered. Our health care is too costly; our schools fail too many; and each day brings further evidence that the ways we use energy strengthen our adversaries and threaten our planet.

These are the indicators of crisis, subject to data and statistics. Less measurable but no less profound is a sapping of confidence across our land - a nagging fear that America's decline is inevitable, and that the next generation must lower its sights.

Today I say to you that the challenges we face are real. They are serious and they are many. They will not be met easily or in a short span of time. But know this, America - they will be met.

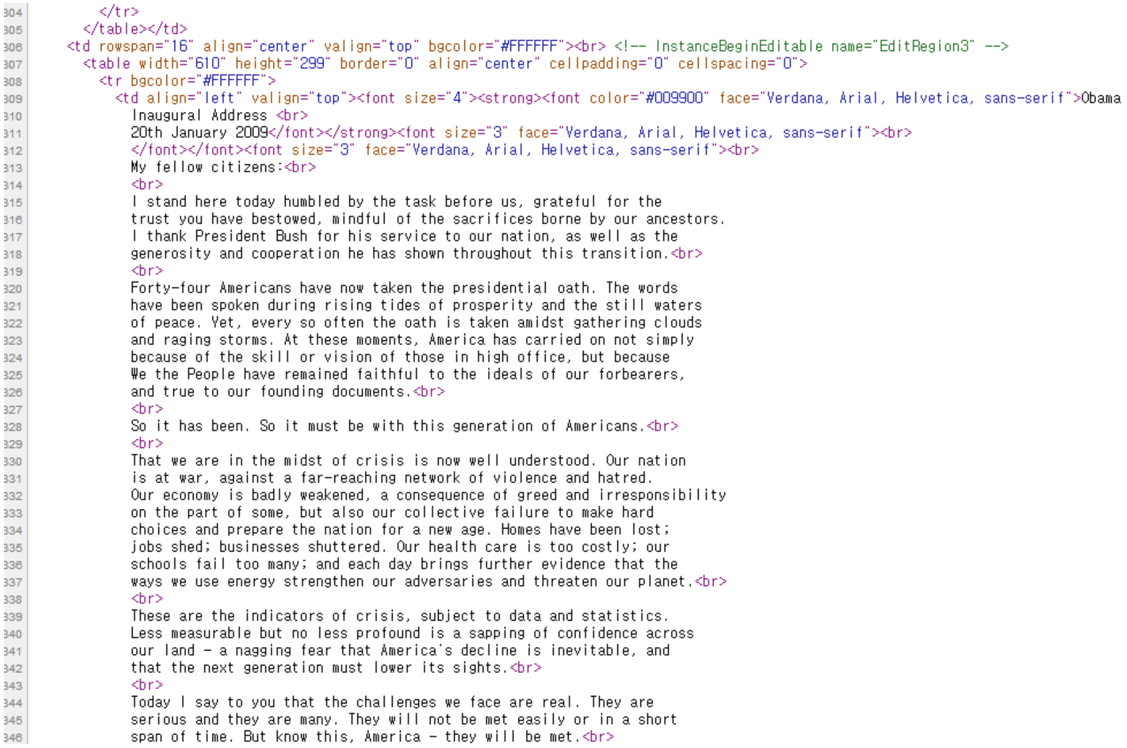
On this day, we gather because we have chosen hope over fear, unity of purpose over conflict and discord.

On this day, we come to proclaim an end to the petty grievances and false promises, the recriminations and worn out dogmas, that for far too long have strangled our politics.

We remain a young nation, but in the words of Scripture, the time has come to set aside childish things. The time has come to reaffirm our enduring spirit; to choose our better history; to carry forward that precious gift, that noble idea, passed on from generation to generation: the God-given promise that all are equal, all are free, and all deserve a chance to pursue their full measure of happiness.

In reaffirming the greatness of our nation, we understand that greatness is never a given. It must be earned. Our journey has never been one of short-cuts or settling for less. It has not been the path for the faint-hearted - for those who prefer leisure over work, or seek only the pleasures of riches and fame. Rather, it has been the risk-takers, the doers, the makers of things - some celebrated but more often men and women obscure in their labor, who have carried us up the long, rugged path towards prosperity and freedom.

What we need to make a web page



```
    </tr>
  </table></td>
<td rowspan="16" align="center" valign="top" bgcolor="#FFFFFF"><br> <!-- InstanceBeginEditable name="EditRegion3" -->
  <table width="610" height="299" border="0" align="center" cellpadding="0" cellspacing="0">
    <tr bgcolor="#FFFFFF">
      <td align="left" valign="top"><font size="4"><strong><font color="#009900" face="Verdana, Arial, Helvetica, sans-serif">Obama
        Inaugural Address <br>
        20th January 2009</font></strong><font size="3" face="Verdana, Arial, Helvetica, sans-serif"><br>
        </font></font><font size="3" face="Verdana, Arial, Helvetica, sans-serif"><br>
        My fellow citizens:<br>
        <br>
        I stand here today humbled by the task before us, grateful for the
        trust you have bestowed, mindful of the sacrifices borne by our ancestors.
        I thank President Bush for his service to our nation, as well as the
        generosity and cooperation he has shown throughout this transition.<br>
        <br>
        Forty-four Americans have now taken the presidential oath. The words
        have been spoken during rising tides of prosperity and the still waters
        of peace. Yet, every so often the oath is taken amidst gathering clouds
        and raging storms. At these moments, America has carried on not simply
        because of the skill or vision of those in high office, but because
        We the People have remained faithful to the ideals of our forbearers,
        and true to our founding documents.<br>
        <br>
        So it has been. So it must be with this generation of Americans.<br>
        <br>
        That we are in the midst of crisis is now well understood. Our nation
        is at war, against a far-reaching network of violence and hatred.
        Our economy is badly weakened, a consequence of greed and irresponsibility
        on the part of some, but also our collective failure to make hard
        choices and prepare the nation for a new age. Homes have been lost;
        jobs shed; businesses shuttered. Our health care is too costly; our
        schools fail too many; and each day brings further evidence that the
        ways we use energy strengthen our adversaries and threaten our planet.<br>
        <br>
        These are the indicators of crisis, subject to data and statistics.
        Less measurable but no less profound is a sapping of confidence across
        our land - a nagging fear that America's decline is inevitable, and
        that the next generation must lower its sights.<br>
        <br>
        Today I say to you that the challenges we face are real. They are
        serious and they are many. They will not be met easily or in a short
        span of time. But know this, America - they will be met.<br>
```

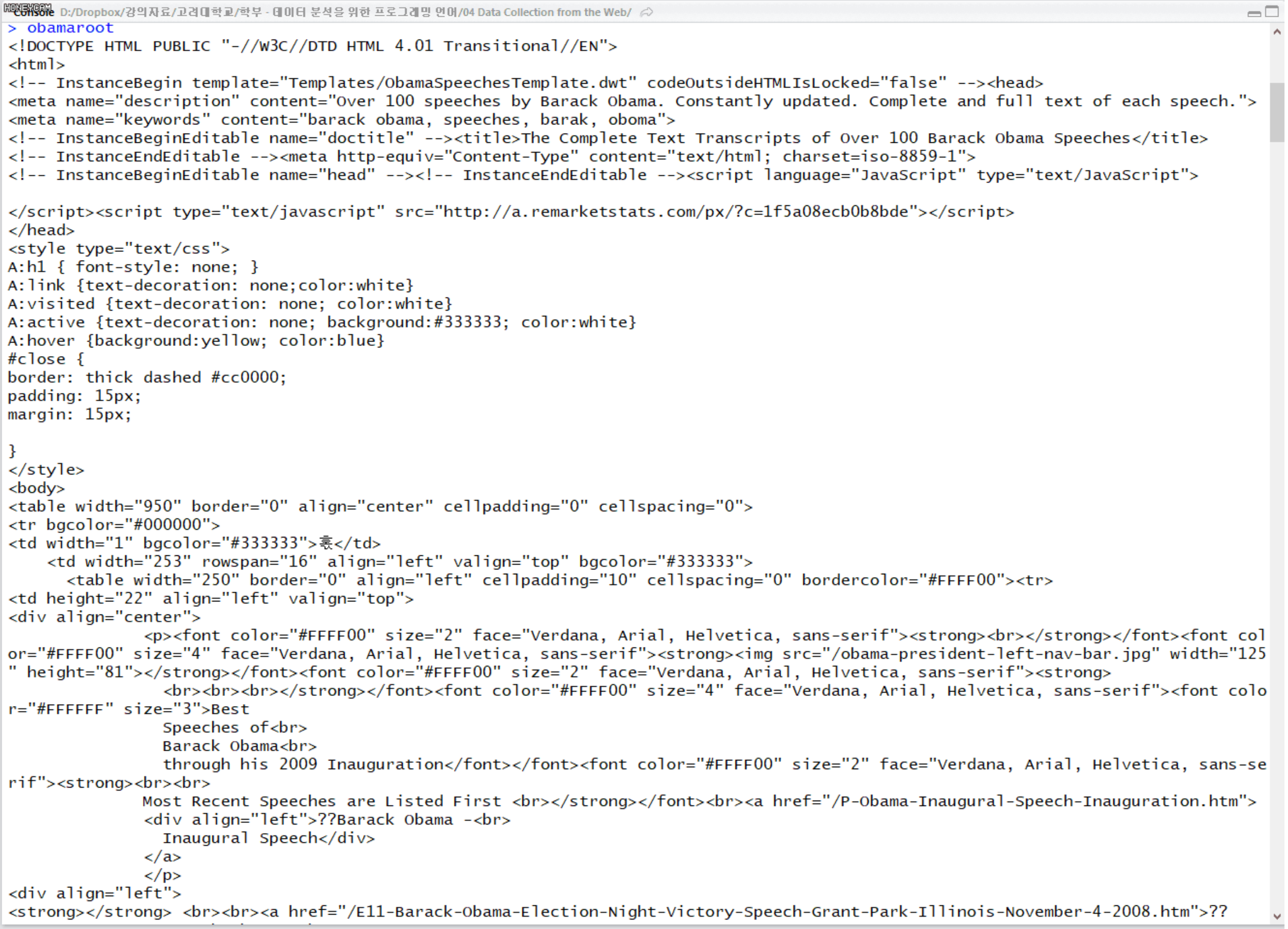
Web Scraping

- Parsing
  - ✓ The process of analyzing a string of symbols, either in natural language or in computer languages (HTML/XML), conforming to the rules of a formal grammar

```
# Case 3: XPath with XML ------------------------------------------
install.packages("XML")
library("XML")

# XML/HTML parsing
obamaurl <- "http://www.obamaspeeches.com/"
obamaroot <- htmlParse(obamaurl)
obamaroot
```

Web Scraping

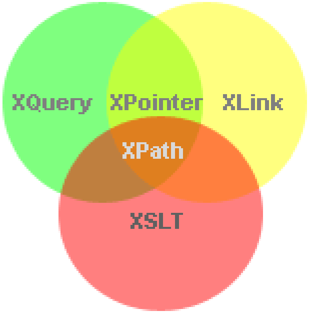
- Parsing result


Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Data Collection from the Web/

```
> obamaroot
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<!-- InstanceBegin template="Templates/ObamaSpeechesTemplate.dwt" codeOutsideHTMLIsLocked="false" --><head>
<meta name="description" content="Over 100 speeches by Barack Obama. Constantly updated. Complete and full text of each speech.">
<meta name="keywords" content="barack obama, speeches, barak, oboma">
<!-- InstanceBeginEditable name="doctitle" --><title>The Complete Text Transcripts of Over 100 Barack Obama Speeches</title>
<!-- InstanceEndEditable --><meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<!-- InstanceBeginEditable name="head" --><!-- InstanceEndEditable --><script language="JavaScript" type="text/JavaScript">

</script><script type="text/javascript" src="http://a.remarketstats.com/px/?c=1f5a08ecb0b8bde"></script>
</head>
<style type="text/css">
A:h1 { font-style: none; }
A:link {text-decoration: none;color:white}
A:visited {text-decoration: none; color:white}
A:active {text-decoration: none; background:#333333; color:white}
A:hover {background:yellow; color:blue}
#close {
border: thick dashed #cc0000;
padding: 15px;
margin: 15px;

}
</style>
<body>
<table width="950" border="0" align="center" cellpadding="0" cellspacing="0">
<tr bgcolor="#000000">
<td width="1" bgcolor="#333333">좋</td>
    <td width="253" rowspan="16" align="left" valign="top" bgcolor="#333333">
      <table width="250" border="0" align="left" cellpadding="10" cellspacing="0" bordercolor="#FFFF00"><tr>
<td height="22" align="left" valign="top">
<div align="center">
              <p><font color="#FFFF00" size="2" face="Verdana, Arial, Helvetica, sans-serif"><strong><br></strong></font><font col
or="#FFFF00" size="4" face="Verdana, Arial, Helvetica, sans-serif"><strong><img src="/obama-president-left-nav-bar.jpg" width="125
" height="81"></strong></font><font color="#FFFF00" size="2" face="Verdana, Arial, Helvetica, sans-serif"><strong>
                <br><br><br></strong></font><font color="#FFFF00" size="4" face="Verdana, Arial, Helvetica, sans-serif"><font colo
r="#FFFFFF" size="3">Best
                Speeches of<br>
                Barack Obama<br>
                through his 2009 Inauguration</font></font><font color="#FFFF00" size="2" face="Verdana, Arial, Helvetica, sans-se
rif"><strong><br><br>
              Most Recent Speeches are Listed First <br></strong></font><br><a href="/P-Obama-Inaugural-Speech-Inauguration.htm">
              <div align="left">??Barack Obama -<br>
                Inaugural Speech</div>
              </a>
              </p>
<div align="left">
<strong></strong> <br><br><a href="/E11-Barack-Obama-Election-Night-Victory-Speech-Grant-Park-Illinois-November-4-2008.htm">??
```

Web Scraping

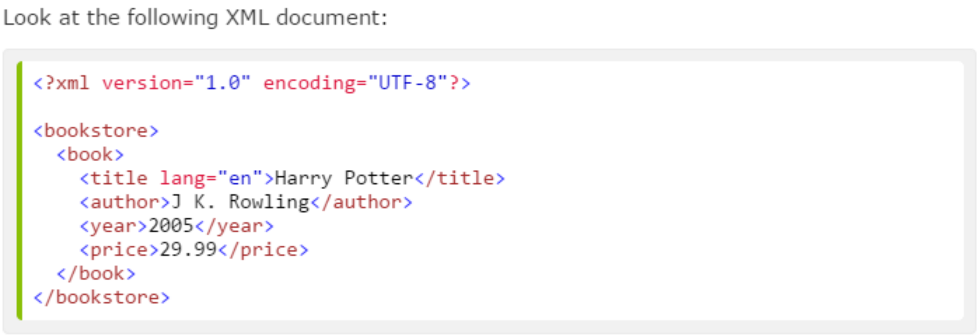
- To extract information that we need from HTML/XML documents, we should also understand Xpath expressions



  - ✓ A syntax for defining parts of an XML document
  - ✓ Uses path expressions to navigate in XML documents
    - ▪ To select nodes or node-sets in an XML document
    - ▪ Path expressions look very much like the expressions you see when you work with a traditional computer file system
  - ✓ Contains a library of standard functions
    - ▪ Include over 100 built-in functions (string values, numeric values, date and time comparison, etc.)
  - ✓ For more information, visit https://www.w3schools.com/xml/xpath_intro.asp

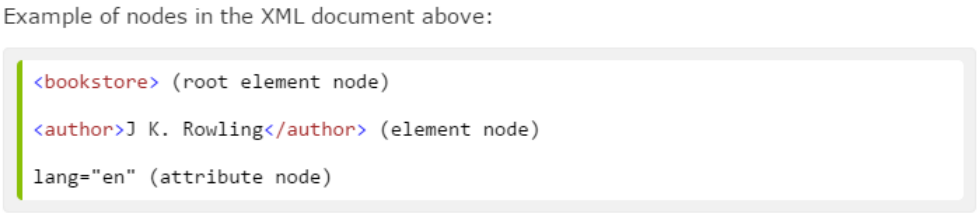
Web Scraping

- Xpath terminology
  - ✓ Nodes: element, attribute, text, namespace, processing-instruction, comment, document
    - ▪ XML documents are treated as trees of nodes
    - ▪ Root node: the topmost element of the tree
  - ✓ Atomic values: nodes with no children or parent
  - ✓ Items: atomic values or nodes



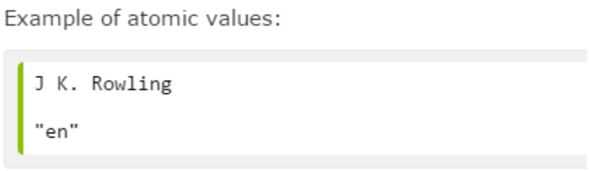
Look at the following XML document:

```
<?xml version="1.0" encoding="UTF-8"?>

<bookstore>
  <book>
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
</bookstore>
```



Example of nodes in the XML document above:

```
<bookstore> (root element node)

<author>J K. Rowling</author> (element node)

lang="en" (attribute node)
```

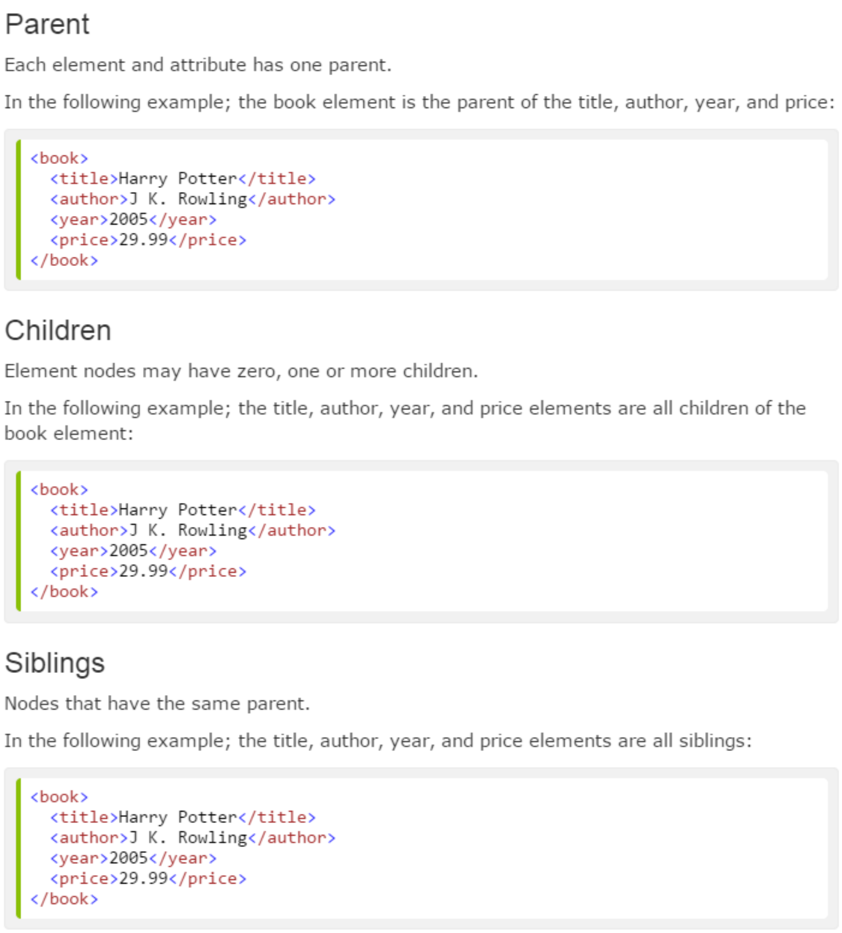


Example of atomic values:

```
J K. Rowling

"en"
```

Web Scraping

- Xpath terminology
  - ✓ Relationship of Nodes: Parent, children, siblings, ancestors, descendants



Parent

Each element and attribute has one parent.

In the following example; the book element is the parent of the title, author, year, and price:

```
<book>
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
```

Children

Element nodes may have zero, one or more children.

In the following example; the title, author, year, and price elements are all children of the book element:

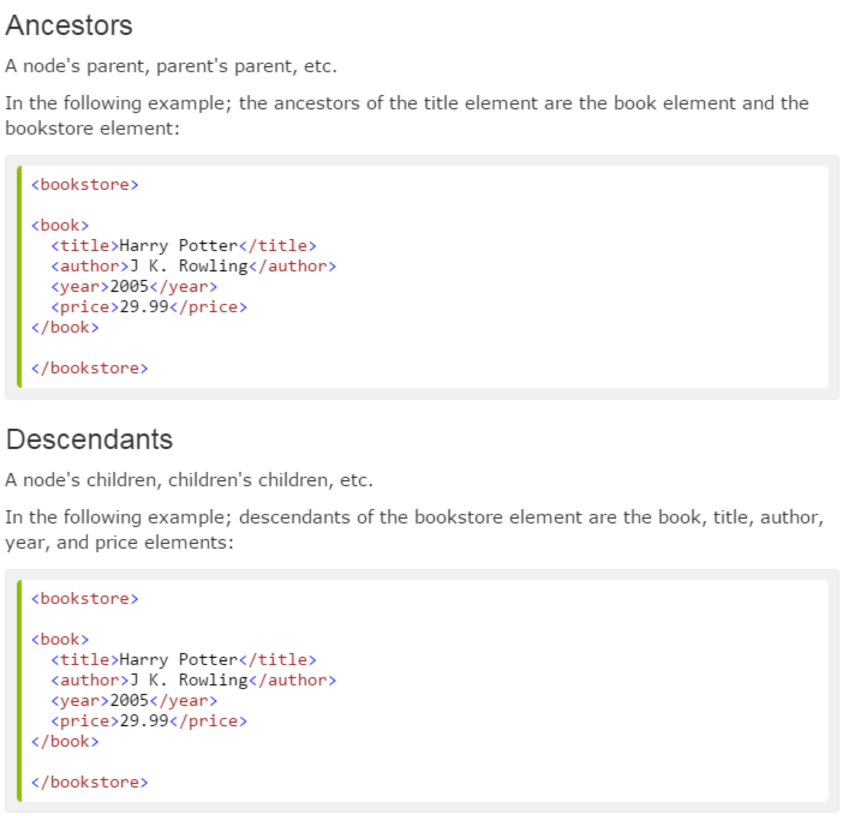
```
<book>
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
```

Siblings

Nodes that have the same parent.

In the following example; the title, author, year, and price elements are all siblings:

```
<book>
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
```



Ancestors

A node's parent, parent's parent, etc.

In the following example; the ancestors of the title element are the book element and the bookstore element:

```
<bookstore>

<book>
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

</bookstore>
```

Descendants

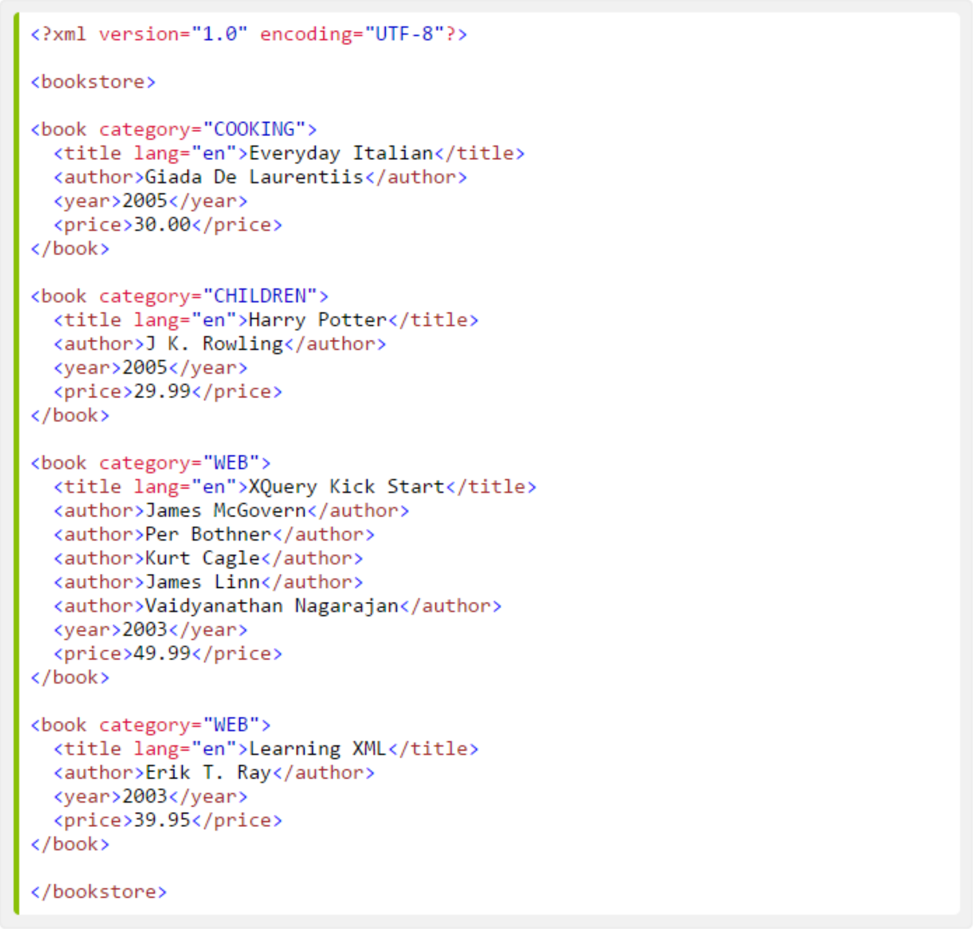
A node's children, children's children, etc.

In the following example; descendants of the bookstore element are the book, title, author, year, and price elements:

```
<bookstore>

<book>
  <title>Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

</bookstore>
```

Web Scraping

- Xpath Syntax
  - ✓ Example document:

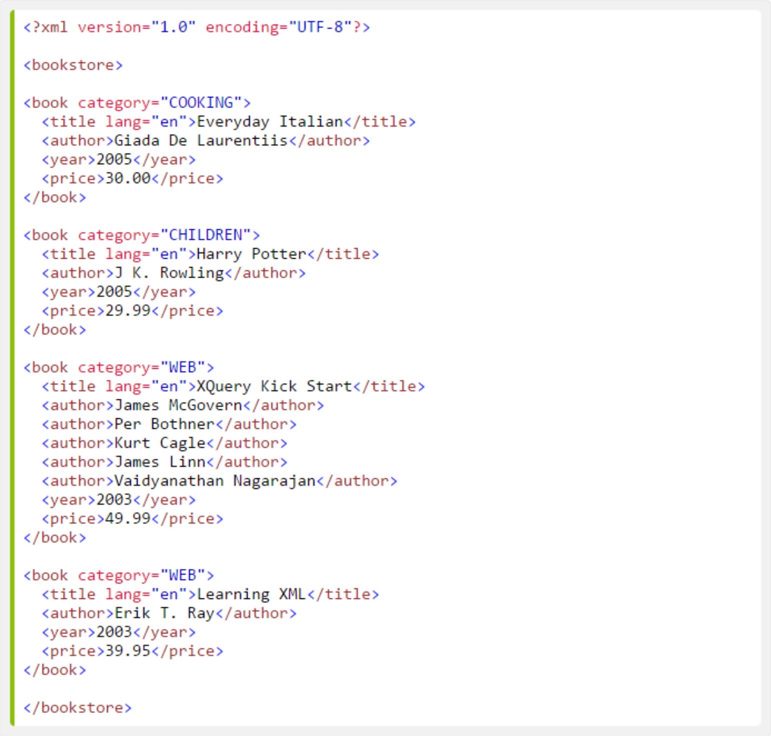


```
<?xml version="1.0" encoding="UTF-8"?>

<bookstore>

<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>

<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>

<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>

</bookstore>
```

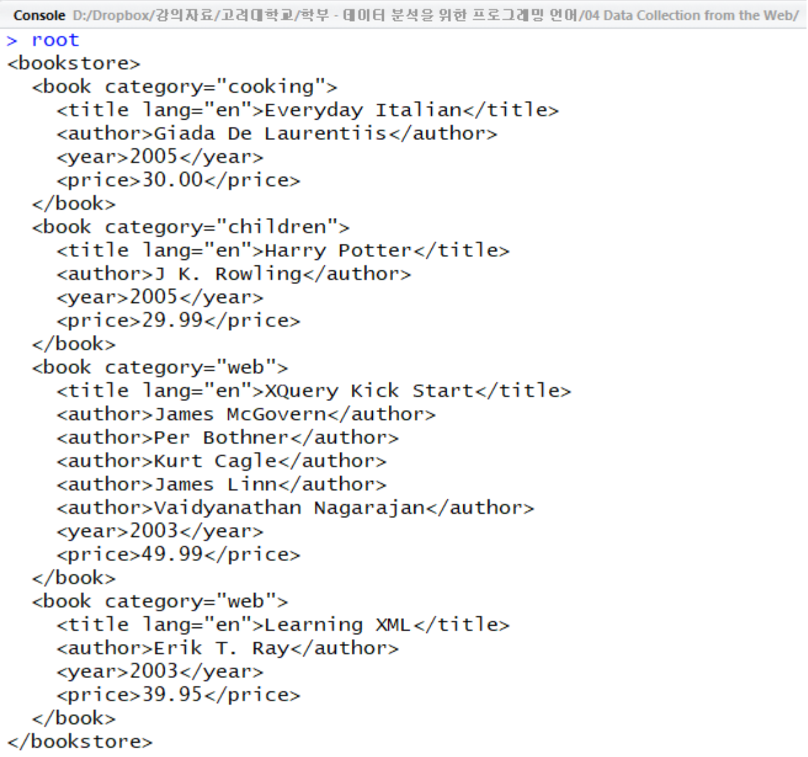
Web Scraping

- Xpath Syntax
  - ✓ Example document:

```r
# Xpath example
xmlfile <- "xml_example.xml"
tmpxml <- xmlParse(xmlfile)
root <- xmlRoot(tmpxml)
root
```



```xml
<?xml version="1.0" encoding="UTF-8"?>

<bookstore>

<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>

<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>

<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>

</bookstore>
```



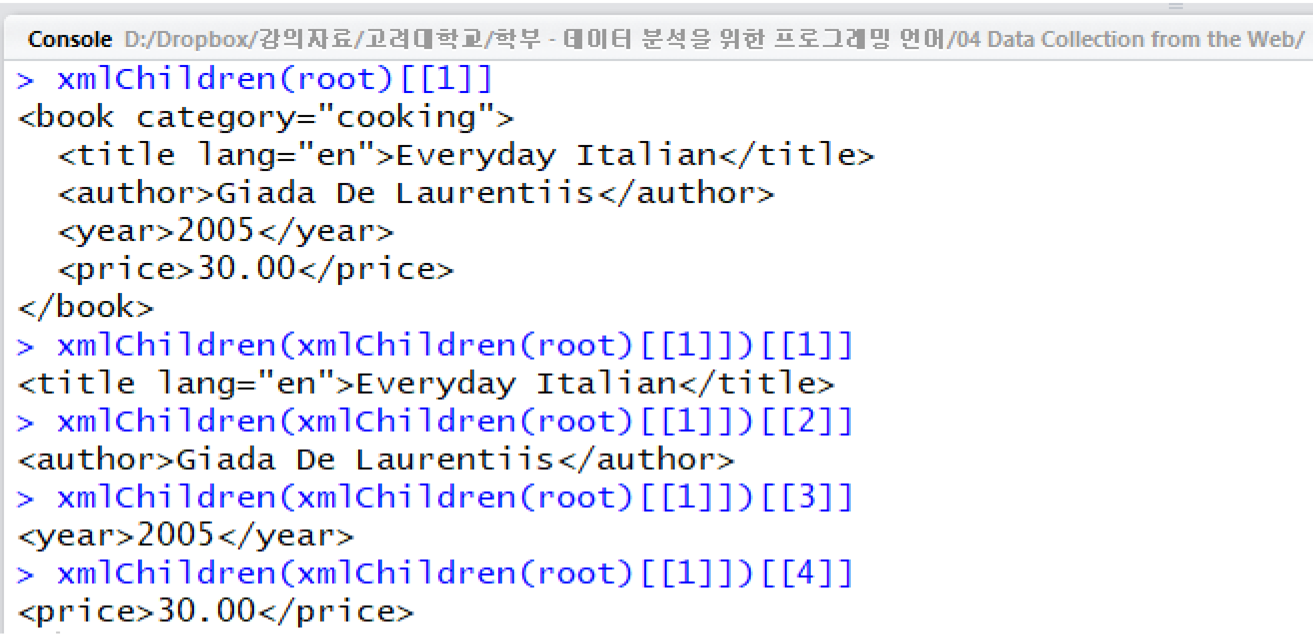
Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Data Collection from the Web/

```
> root
<bookstore>
  <book category="cooking">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="web">
    <title lang="en">XQuery Kick Start</title>
    <author>James McGovern</author>
    <author>Per Bothner</author>
    <author>Kurt Cagle</author>
    <author>James Linn</author>
    <author>Vaidyanathan Nagarajan</author>
    <year>2003</year>
    <price>49.99</price>
  </book>
  <book category="web">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

Web Scraping

- Xpath Syntax
  - ✓ Selecting nodes with node index

```
# Select children node
xmlChildren(root)[[1]]

xmlChildren(xmlChildren(root)[[1]])[[1]]
xmlChildren(xmlChildren(root)[[1]])[[2]]
xmlChildren(xmlChildren(root)[[1]])[[3]]
xmlChildren(xmlChildren(root)[[1]])[[4]]
```

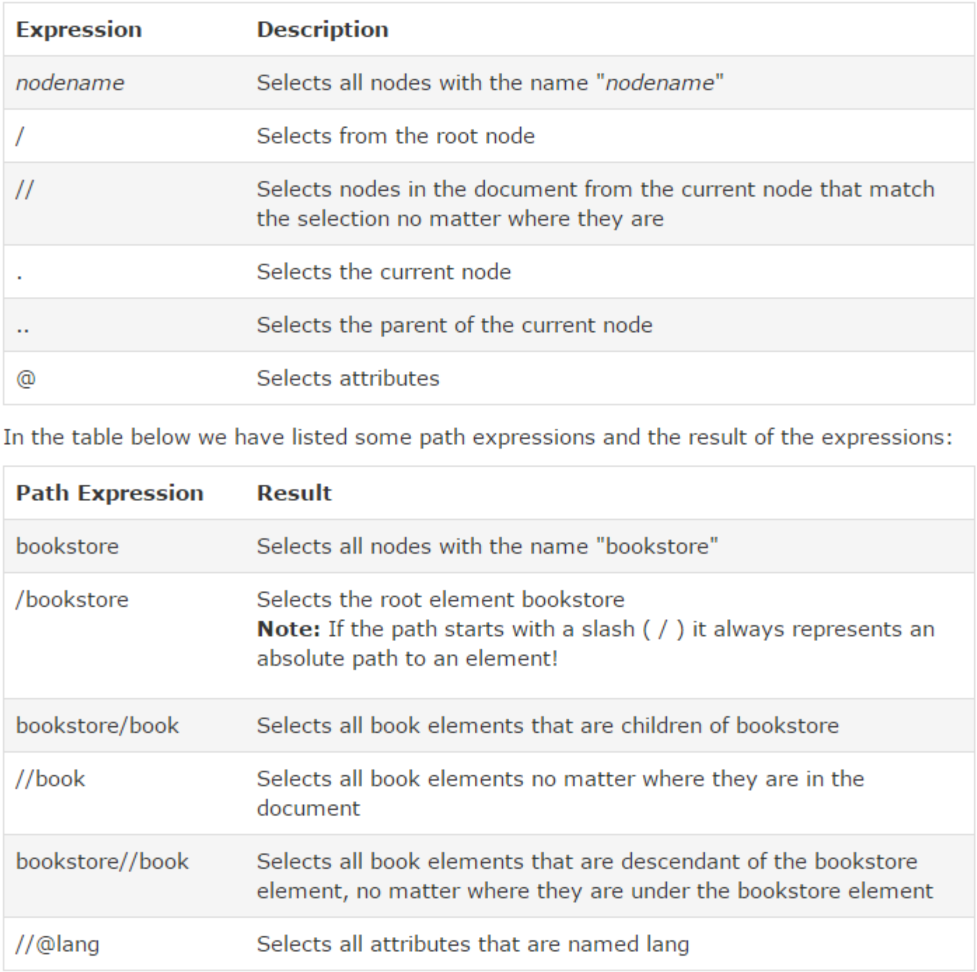


Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Data Collection from the Web/

```
> xmlChildren(root)[[1]]
<book category="cooking">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>
> xmlChildren(xmlChildren(root)[[1]])[[1]]
<title lang="en">Everyday Italian</title>
> xmlChildren(xmlChildren(root)[[1]])[[2]]
<author>Giada De Laurentiis</author>
> xmlChildren(xmlChildren(root)[[1]])[[3]]
<year>2005</year>
> xmlChildren(xmlChildren(root)[[1]])[[4]]
<price>30.00</price>
```

Web Scraping

- Xpath Syntax
  - ✓ Selecting nodes: some useful path expressions



Expression	Description
nodename	Selects all nodes with the name "*nodename*"
/	Selects from the root node
//	Selects nodes in the document from the current node that match the selection no matter where they are
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes

In the table below we have listed some path expressions and the result of the expressions:

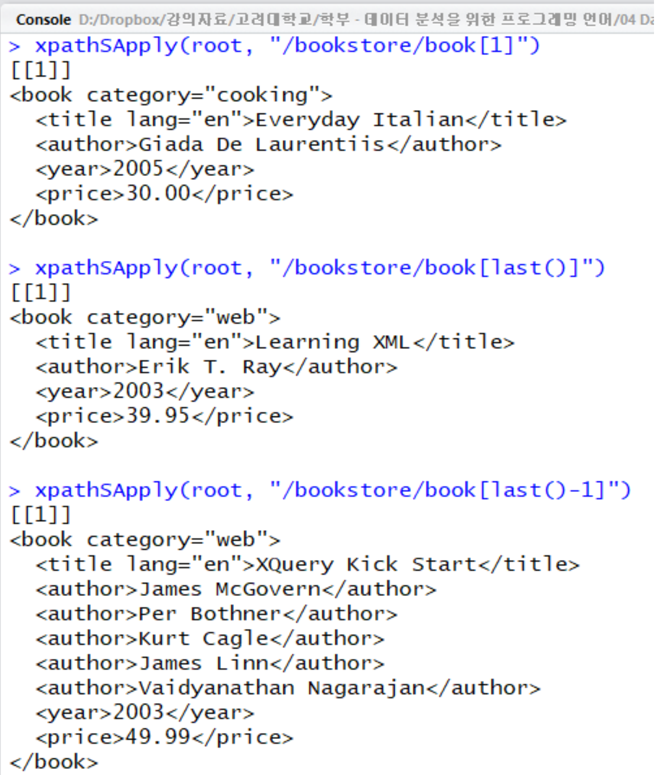
Path Expression	Result
bookstore	Selects all nodes with the name "bookstore"
/bookstore	Selects the root element bookstore **Note:** If the path starts with a slash (/) it always represents an absolute path to an element!
bookstore/book	Selects all book elements that are children of bookstore
//book	Selects all book elements no matter where they are in the document
bookstore//book	Selects all book elements that are descendant of the bookstore element, no matter where they are under the bookstore element
//@lang	Selects all attributes that are named lang

Web Scraping

- Xpath Syntax

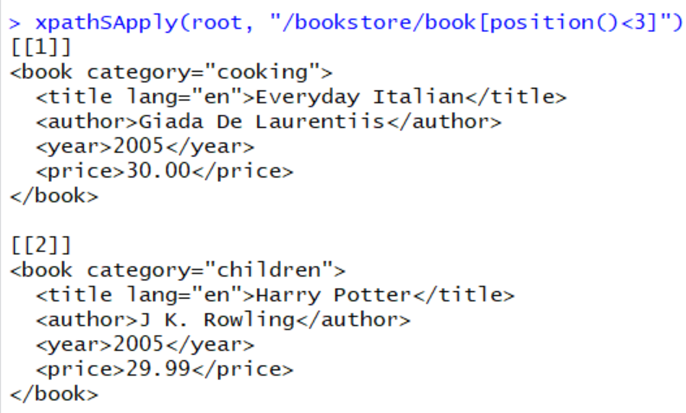
  ✓ Selecting nodes: some useful path expressions

```
# Selecting nodes
xpathSApply(root, "/bookstore/book[1]")
xpathSApply(root, "/bookstore/book[last()]")
xpathSApply(root, "/bookstore/book[last()-1]")
xpathSApply(root, "/bookstore/book[position()<3]")
```



Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Da

```
> xpathSApply(root, "/bookstore/book[1]")
[[1]]
<book category="cooking">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>

> xpathSApply(root, "/bookstore/book[last()]")
[[1]]
<book category="web">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>

> xpathSApply(root, "/bookstore/book[last()-1]")
[[1]]
<book category="web">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>
```

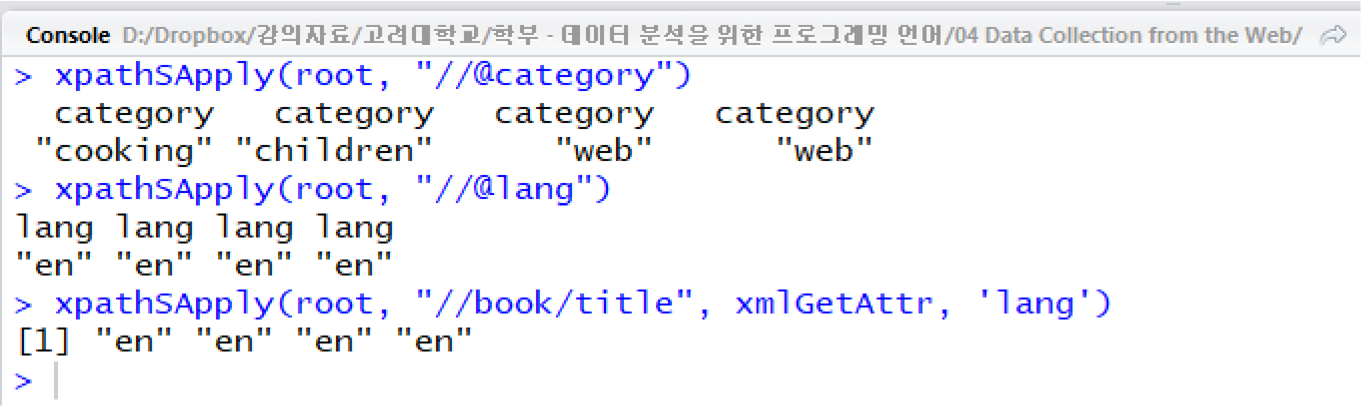


```
> xpathSApply(root, "/bookstore/book[position()<3]")
[[1]]
<book category="cooking">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>

[[2]]
<book category="children">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
```

Web Scraping

- Xpath Syntax
  - ✓ Selecting attributes: some useful path expressions

```
# Selecting attributes
xpathSApply(root, "//@category")
xpathSApply(root, "//@lang")
xpathSApply(root, "//book/title", xmlGetAttr, 'lang')
```

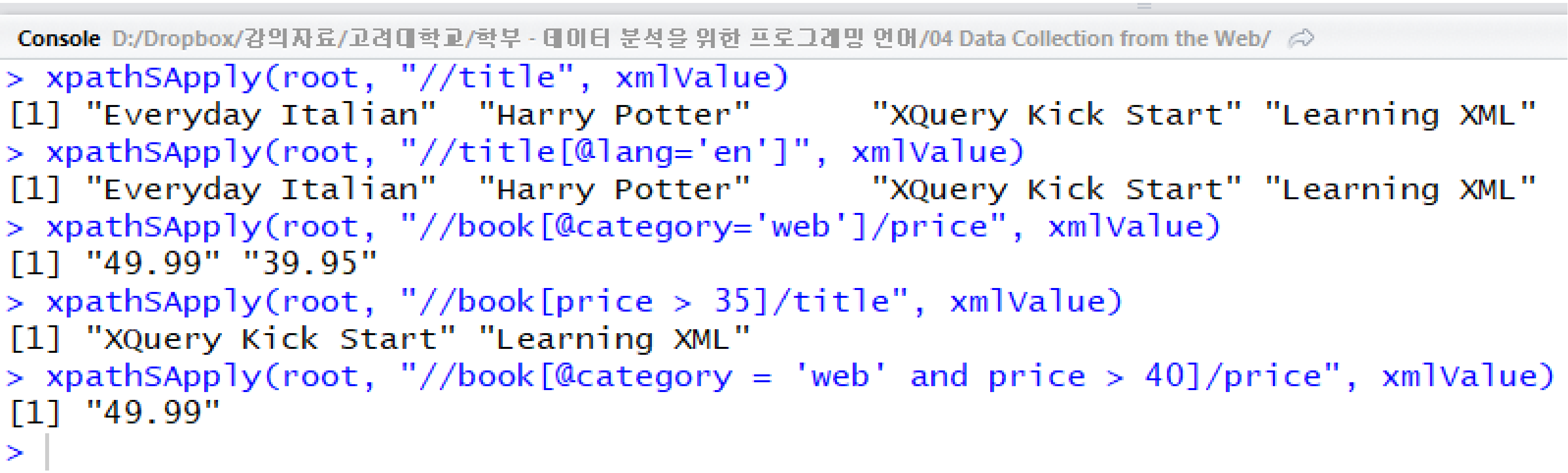

Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Data Collection from the Web/

```
> xpathSApply(root, "//@category")
  category   category   category   category 
 "cooking" "children"      "web"      "web" 
> xpathSApply(root, "//@lang")
lang lang lang lang 
"en" "en" "en" "en" 
> xpathSApply(root, "//book/title", xmlGetAttr, 'lang')
[1] "en" "en" "en" "en"
> 
```

Web Scraping

- Xpath Syntax
  - ✓ Selecting atomic values: some useful path expressions

```
# Selecting atomic values
xpathSApply(root, "//title", xmlValue)
xpathSApply(root, "//title[@lang='en']", xmlValue)
xpathSApply(root, "//book[@category=‘web']/price", xmlValue)
xpathSApply(root, "//book[price > 35]/title", xmlValue)
xpathSApply(root, "//book[@category = ‘web' and price > 40]/price", xmlValue)
```

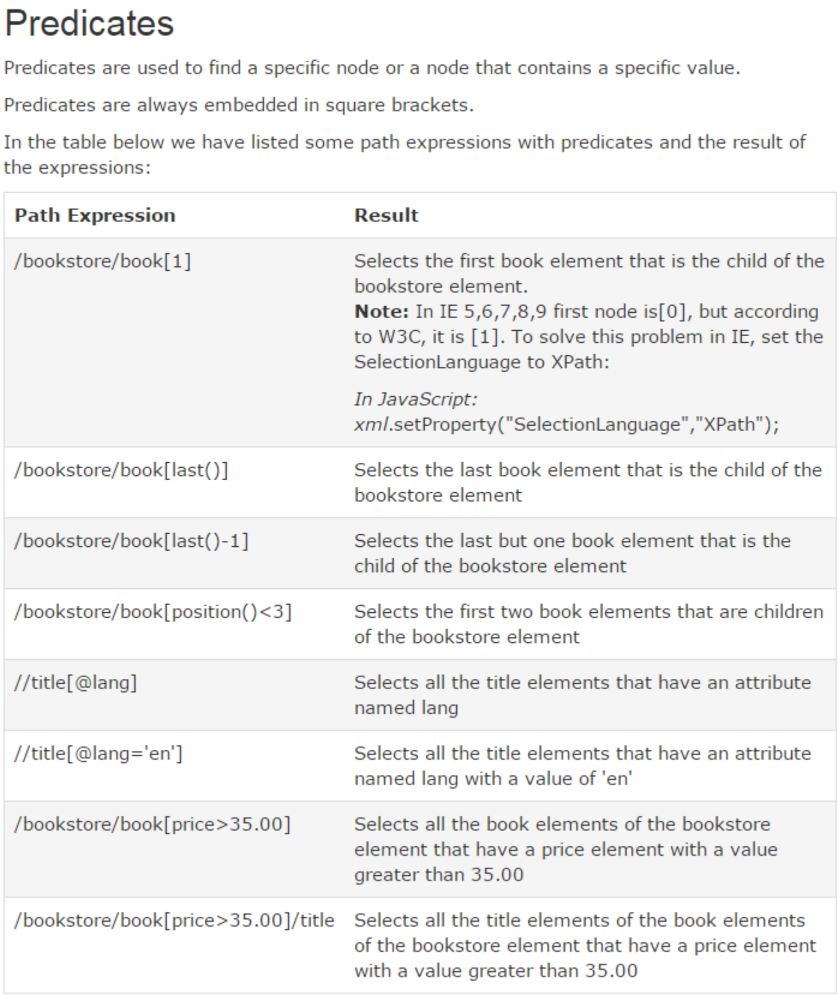

Console D:/Dropbox/강의자료/고려대학교/학부 - 데이터 분석을 위한 프로그래밍 언어/04 Data Collection from the Web/

```
> xpathSApply(root, "//title", xmlValue)
[1] "Everyday Italian"  "Harry Potter"      "XQuery Kick Start" "Learning XML"
> xpathSApply(root, "//title[@lang='en']", xmlValue)
[1] "Everyday Italian"  "Harry Potter"      "XQuery Kick Start" "Learning XML"
> xpathSApply(root, "//book[@category='web']/price", xmlValue)
[1] "49.99" "39.95"
> xpathSApply(root, "//book[price > 35]/title", xmlValue)
[1] "XQuery Kick Start" "Learning XML"
> xpathSApply(root, "//book[@category = 'web' and price > 40]/price", xmlValue)
[1] "49.99"
> 
```

Web Scraping

- Xpath Syntax

  ✓ Predicates, unknown nodes, and several paths



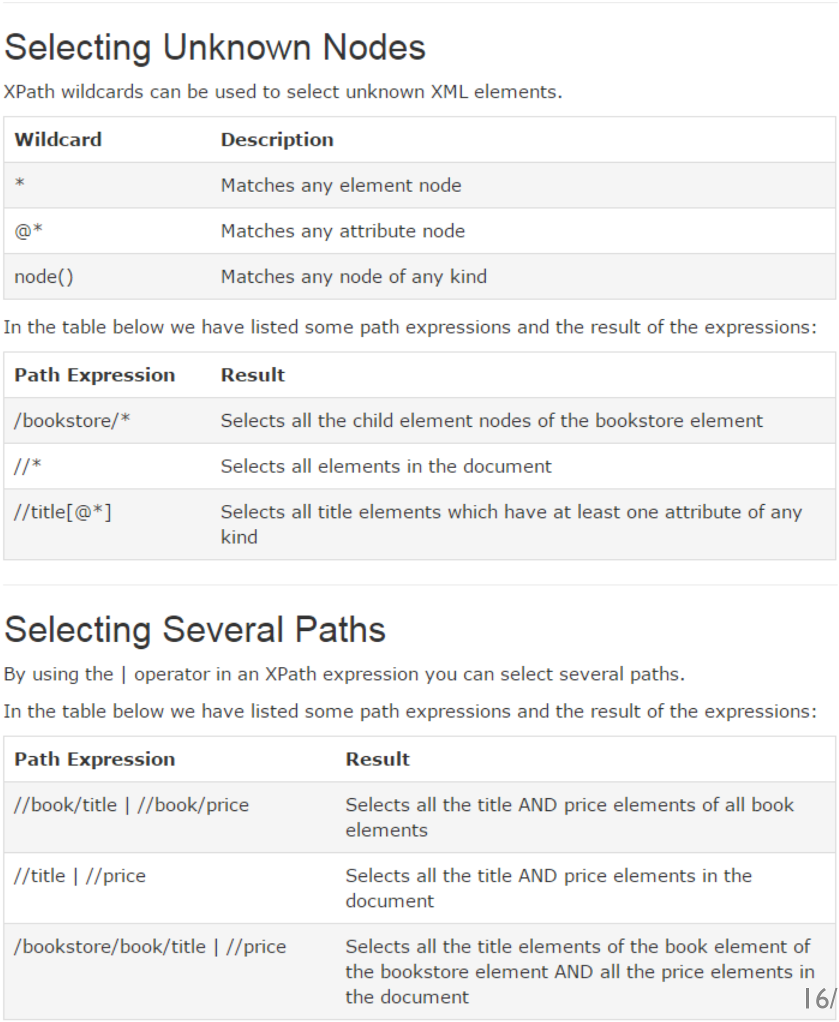
Predicates

Predicates are used to find a specific node or a node that contains a specific value.

Predicates are always embedded in square brackets.

In the table below we have listed some path expressions with predicates and the result of the expressions:

Path Expression	Result
/bookstore/book[1]	Selects the first book element that is the child of the bookstore element. **Note:** In IE 5,6,7,8,9 first node is[0], but according to W3C, it is [1]. To solve this problem in IE, set the SelectionLanguage to XPath: *In JavaScript:* *xml*.setProperty("SelectionLanguage","XPath");
/bookstore/book[last()]	Selects the last book element that is the child of the bookstore element
/bookstore/book[last()-1]	Selects the last but one book element that is the child of the bookstore element
/bookstore/book[position()<3]	Selects the first two book elements that are children of the bookstore element
//title[@lang]	Selects all the title elements that have an attribute named lang
//title[@lang='en']	Selects all the title elements that have an attribute named lang with a value of 'en'
/bookstore/book[price>35.00]	Selects all the book elements of the bookstore element that have a price element with a value greater than 35.00
/bookstore/book[price>35.00]/title	Selects all the title elements of the book elements of the bookstore element that have a price element with a value greater than 35.00



Selecting Unknown Nodes

XPath wildcards can be used to select unknown XML elements.

Wildcard	Description
*	Matches any element node
@*	Matches any attribute node
node()	Matches any node of any kind

In the table below we have listed some path expressions and the result of the expressions:

Path Expression	Result
/bookstore/*	Selects all the child element nodes of the bookstore element
//*	Selects all elements in the document
//title[@*]	Selects all title elements which have at least one attribute of any kind

Selecting Several Paths

By using the | operator in an XPath expression you can select several paths.

In the table below we have listed some path expressions and the result of the expressions:

Path Expression	Result
//book/title | //book/price	Selects all the title AND price elements of all book elements
//title | //price	Selects all the title AND price elements in the document
/bookstore/book/title | //price	Selects all the title elements of the book element of the bookstore element AND all the price elements in the document

AGENDA

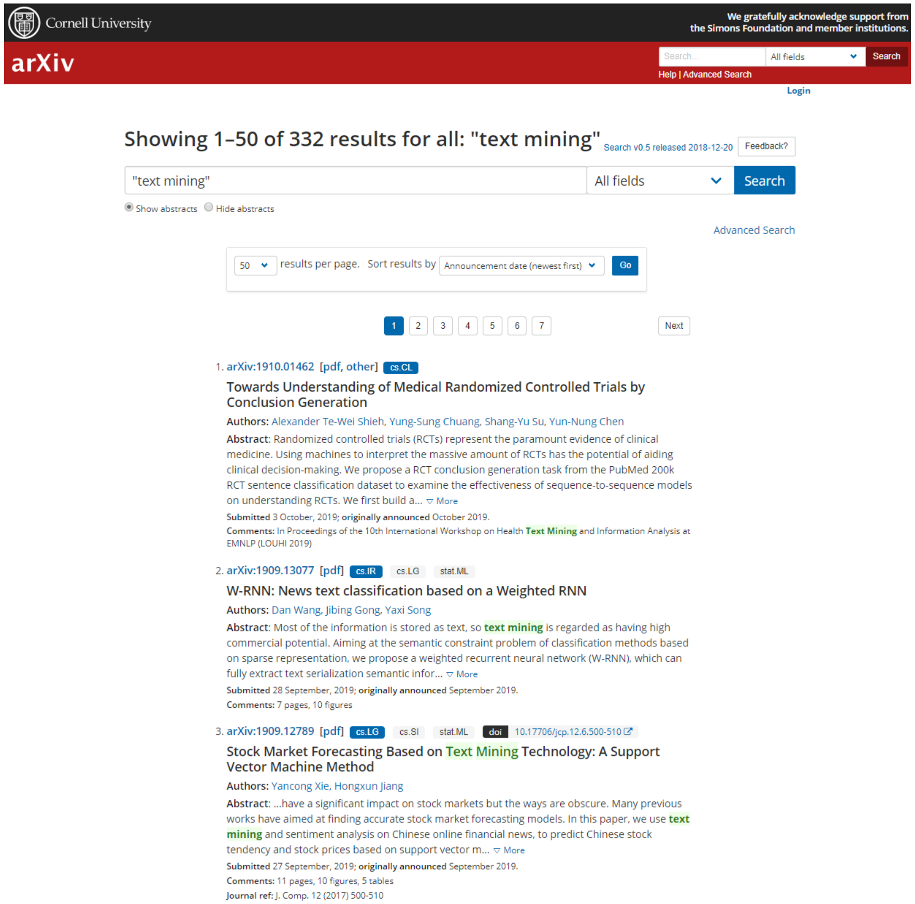
01 Prerequisites

02 Web Scraping: arXiv Research Papers

03 Web Scraping: Movie Reviews

Web Scraping: arXiv Papers

- Web scraping example 1: arXiv papers about “Text Mining”
  - ✓ arXiv website: http://arxiv.org/
  - ✓ Collect Title, Authors, Subjects, Abstracts, and Meta Information



Showing 1–50 of 332 results for all: "text mining"

"text mining"

● Show abstracts ○ Hide abstracts

50 results per page. Sort results by Announcement date (newest first)

1. arXiv:1910.01462 [pdf, other] cs.CL

Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Authors: Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

Abstract: Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a... ▽ More

Submitted 3 October, 2019; **originally announced** October 2019.

Comments: In Proceedings of the 10th International Workshop on Health **Text Mining** and Information Analysis at EMNLP (LOUHI 2019)

2. arXiv:1909.13077 [pdf] cs.IR cs.LG stat.ML

W-RNN: News text classification based on a Weighted RNN

Authors: Dan Wang, Jibing Gong, Yaxi Song

Abstract: Most of the information is stored as text, so **text mining** is regarded as having high commercial potential. Aiming at the semantic constraint problem of classification methods based on sparse representation, we propose a weighted recurrent neural network (W-RNN), which can fully extract text serialization semantic infor... ▽ More

Submitted 28 September, 2019; **originally announced** September 2019.

Comments: 7 pages, 10 figures

3. arXiv:1909.12789 [pdf] cs.LG cs.SI stat.ML doi 10.17706/jcp.12.6.500-510

Stock Market Forecasting Based on Text Mining Technology: A Support Vector Machine Method

Authors: Yancong Xie, Hongxun Jiang

Abstract: ...have a significant impact on stock markets but the ways are obscure. Many previous works have aimed at finding accurate stock market forecasting models. In this paper, we use **text mining** and sentiment analysis on Chinese online financial news, to predict Chinese stock tendency and stock prices based on support vector m... ▽ More

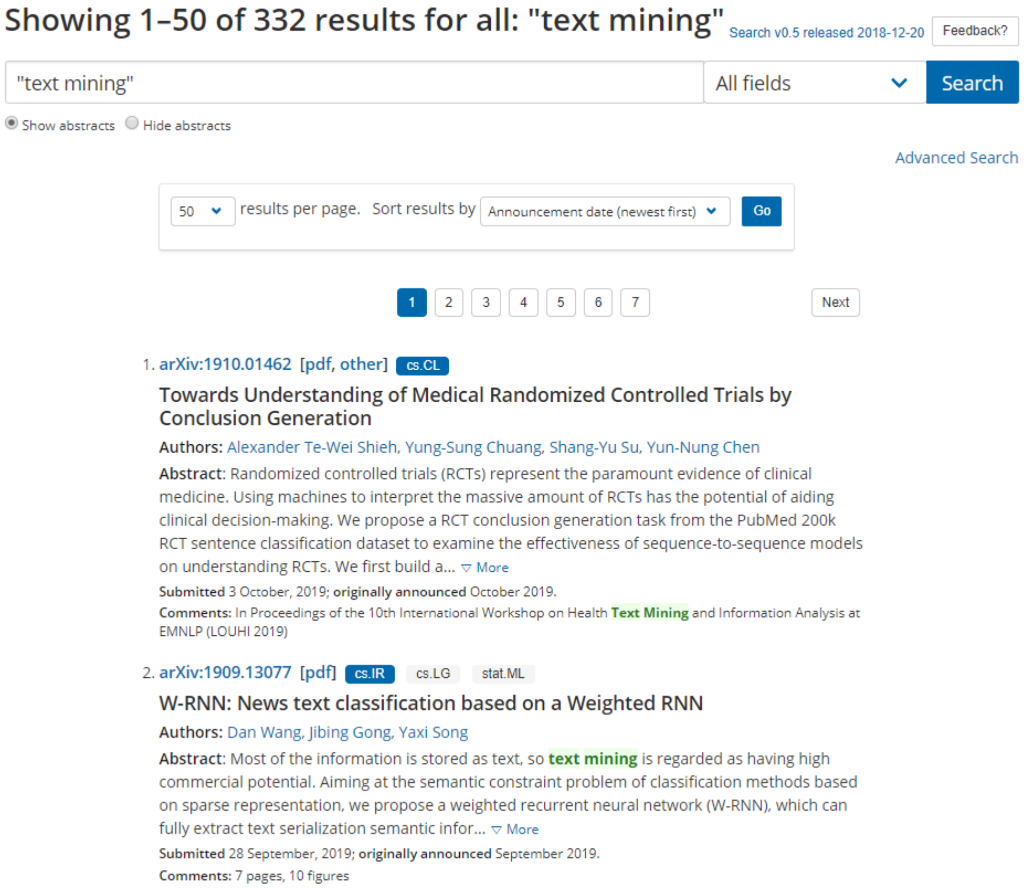
Submitted 27 September, 2019; **originally announced** September 2019.

Comments: 11 pages, 10 figures, 5 tables

Journal ref: J. Comp. 12 (2017) 500-510

Web Scraping: arXiv Papers

- Step 1: Understand the basic structure
  - ✓ A total of 332 papers are returned (2019-10-07), each page contains 50 papers
  - ✓ Each paper has a unique ID



Showing 1–50 of 332 results for all: "text mining"

"text mining" All fields

● Show abstracts ○ Hide abstracts

50 results per page. Sort results by Announcement date (newest first)

1. arXiv:1910.01462 [pdf, other] cs.CL

Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Authors: Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

Abstract: Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a… ▽ More

Submitted 3 October, 2019; **originally announced** October 2019.

Comments: In Proceedings of the 10th International Workshop on Health **Text Mining** and Information Analysis at EMNLP (LOUHI 2019)

2. arXiv:1909.13077 [pdf] cs.IR cs.LG stat.ML

W-RNN: News text classification based on a Weighted RNN

Authors: Dan Wang, Jibing Gong, Yaxi Song

Abstract: Most of the information is stored as text, so **text mining** is regarded as having high commercial potential. Aiming at the semantic constraint problem of classification methods based on sparse representation, we propose a weighted recurrent neural network (W-RNN), which can fully extract text serialization semantic infor… ▽ More

Submitted 28 September, 2019; **originally announced** September 2019.

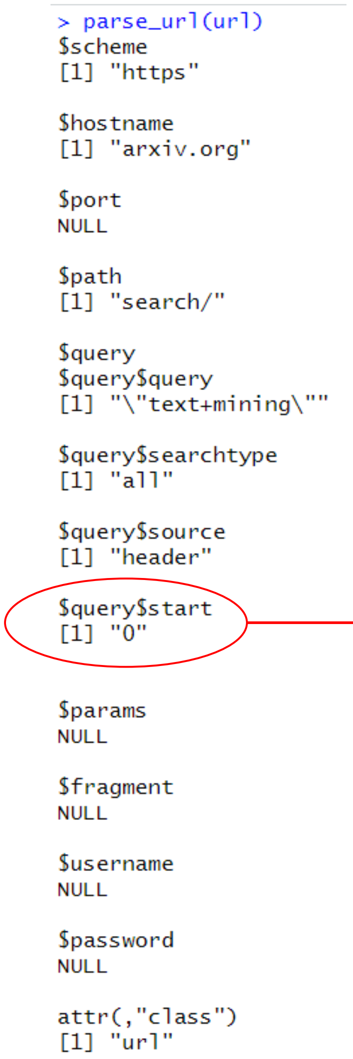
Comments: 7 pages, 10 figures

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure
  - ✓ First page URL
    - https://arxiv.org/search/?query=%22text+mining%22&searchtype=all&source=header&start=0
  - ✓ Second page URL
    - https://arxiv.org/search/?query=%22text+mining%22&searchtype=all&source=header&start=50
  - ✓ Third page URL
    - https://arxiv.org/search/?query=%22text+mining%22&searchtype=all&source=header&start=100

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure
  - ✓ URL Parsing



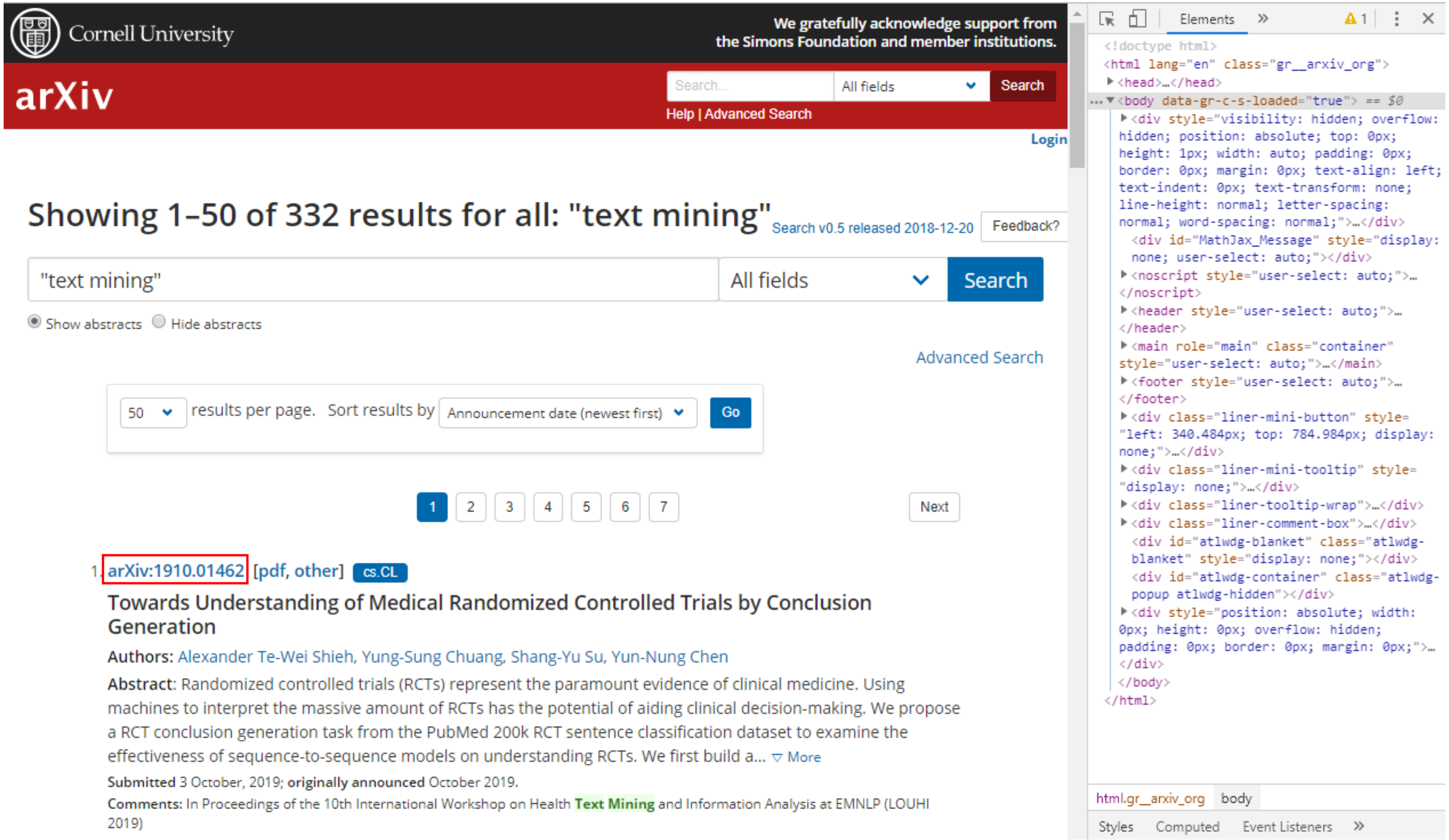
```
> parse_url(url)
$scheme
[1] "https"

$hostname
[1] "arxiv.org"

$port
NULL

$path
[1] "search/"

$query
$query$query
[1] "\"text+mining\""

$query$searchtype
[1] "all"

$query$source
[1] "header"

$query$start
[1] "0"


$params
NULL

$fragment
NULL

$username
NULL

$password
NULL

attr(,"class")
[1] "url"
```

```r
tmp_url <- modify_url(url, query = list(start = i))
```

The only part that actually changes

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure (Press F12 in Chrome browser)
  - ✓ Find the node that contains the necessary links



Cornell University
We gratefully acknowledge support from the Simons Foundation and member institutions.
arXiv
Search...
All fields
Search
Help | Advanced Search
Login

Showing 1–50 of 332 results for all: "text mining"

Search v0.5 released 2018-12-20
Feedback?

"text mining" | All fields | Search

● Show abstracts ○ Hide abstracts

50 results per page. Sort results by Announcement date (newest first) Go

1. arXiv:1910.01462 [pdf, other] cs.CL

Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Authors: Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

Abstract: Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a... ▽ More

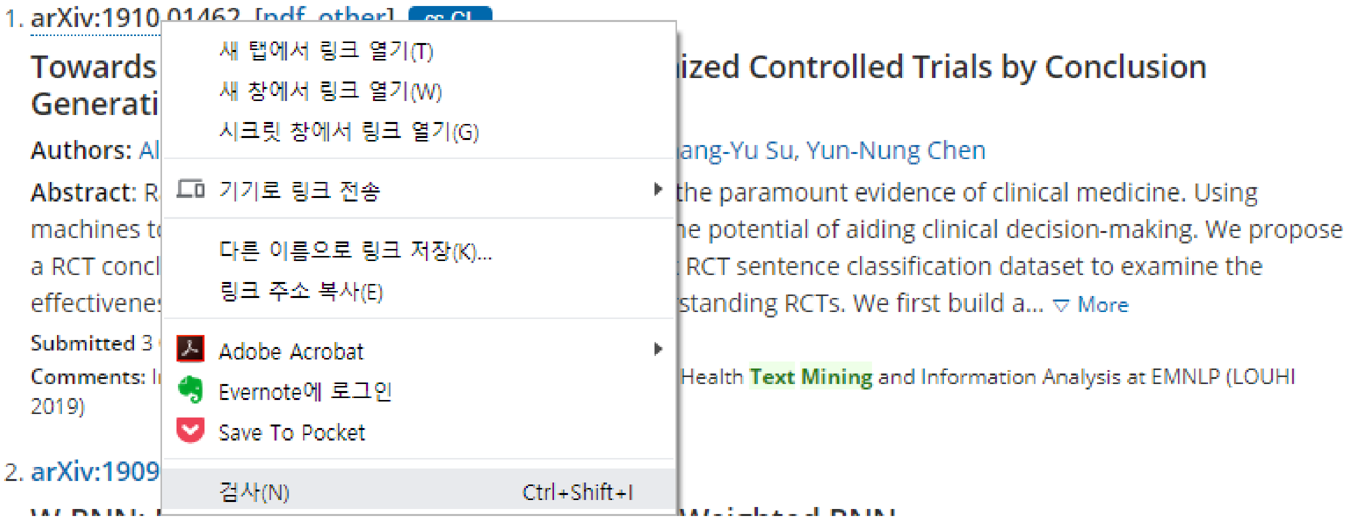
Submitted 3 October, 2019; **originally announced** October 2019.

Comments: In Proceedings of the 10th International Workshop on Health **Text Mining** and Information Analysis at EMNLP (LOUHI 2019)

```
<!doctype html>
<html lang="en" class="gr__arxiv_org">
 ▶<head>…</head>
…▼<body data-gr-c-s-loaded="true"> == $0
   ▶<div style="visibility: hidden; overflow:
   hidden; position: absolute; top: 0px;
   height: 1px; width: auto; padding: 0px;
   border: 0px; margin: 0px; text-align: left;
   text-indent: 0px; text-transform: none;
   line-height: normal; letter-spacing:
   normal; word-spacing: normal;">…</div>
     <div id="MathJax_Message" style="display:
     none; user-select: auto;"></div>
   ▶<noscript style="user-select: auto;">…
   </noscript>
   ▶<header style="user-select: auto;">…
   </header>
   ▶<main role="main" class="container"
   style="user-select: auto;">…</main>
   ▶<footer style="user-select: auto;">…
   </footer>
   ▶<div class="liner-mini-button" style=
   "left: 340.484px; top: 784.984px; display:
   none;">…</div>
   ▶<div class="liner-mini-tooltip" style=
   "display: none;">…</div>
   ▶<div class="liner-tooltip-wrap">…</div>
   ▶<div class="liner-comment-box">…</div>
     <div id="atlwdg-blanket" class="atlwdg-
     blanket" style="display: none;"></div>
     <div id="atlwdg-container" class="atlwdg-
     popup atlwdg-hidden"></div>
   ▶<div style="position: absolute; width:
   0px; height: 0px; overflow: hidden;
   padding: 0px; border: 0px; margin: 0px;">…
   </div>
   </body>
</html>
```

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure (Mouse right click ->)
  - ✓ Find the node that contains the necessary links



1. arXiv:1910.01462 [pdf, other] cs.CL

Towards ... ized Controlled Trials by Conclusion
Generati

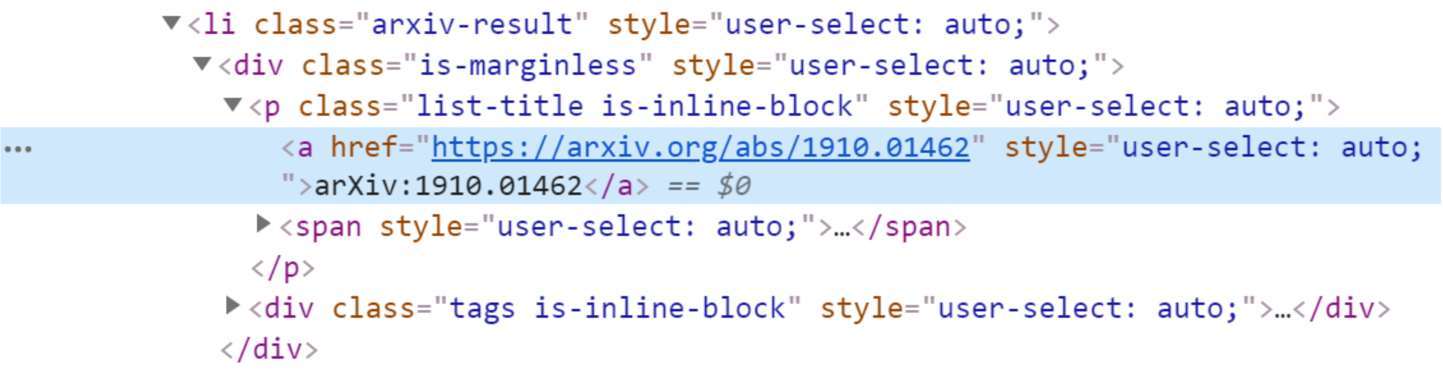
Authors: Al ... ang-Yu Su, Yun-Nung Chen

Abstract: R ... the paramount evidence of clinical medicine. Using
machines t ... e potential of aiding clinical decision-making. We propose
a RCT concl ... RCT sentence classification dataset to examine the
effectivene ... standing RCTs. We first build a... ▽ More

Submitted 3

Comments: I ... Health Text Mining and Information Analysis at EMNLP (LOUHI
2019)

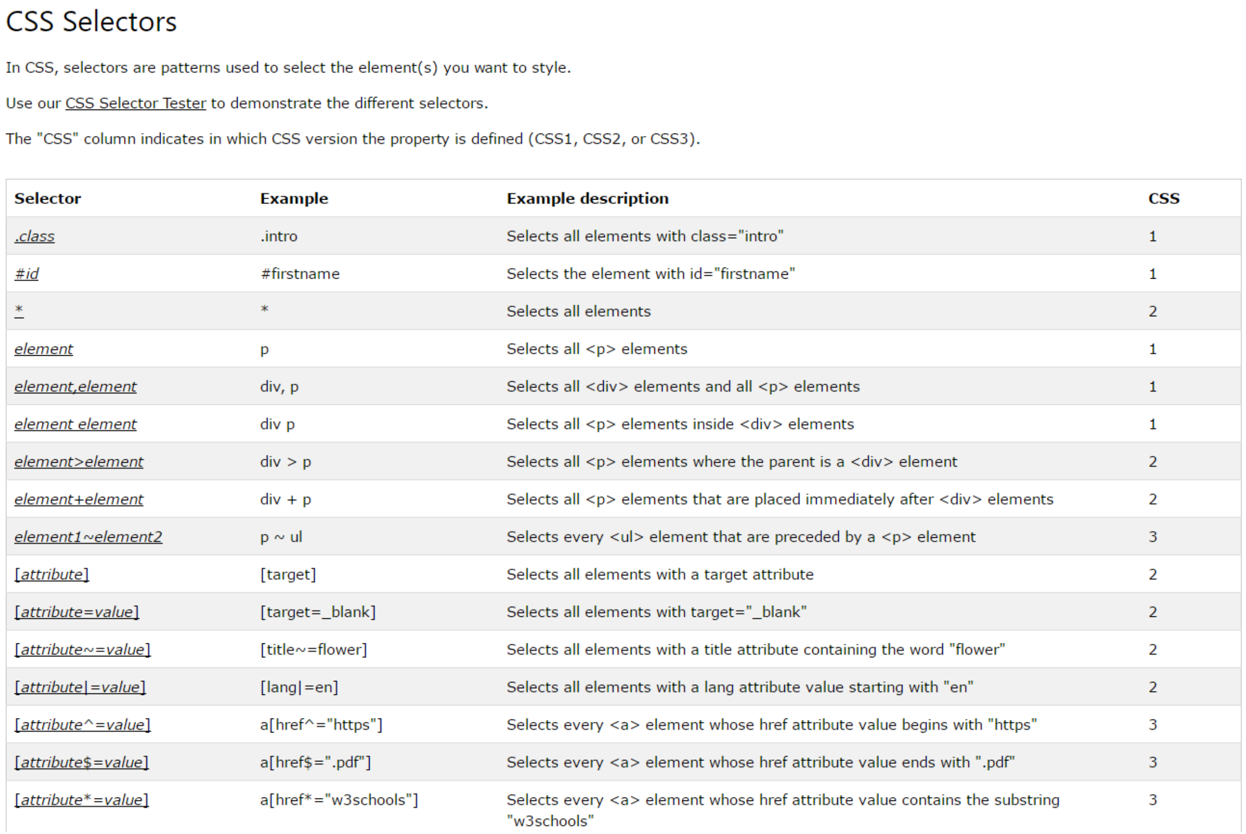
2. arXiv:1909



```
▼<li class="arxiv-result" style="user-select: auto;">
  ▼<div class="is-marginless" style="user-select: auto;">
    ▼<p class="list-title is-inline-block" style="user-select: auto;">
...     <a href="https://arxiv.org/abs/1910.01462" style="user-select: auto;
        ">arXiv:1910.01462</a> == $0
      ▶<span style="user-select: auto;">…</span>
      </p>
    ▶<div class="tags is-inline-block" style="user-select: auto;">…</div>
    </div>
```

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure
  - ✓ Extract the link information
  - ✓ Should be familiar to the usage of CSS Selector
    - http://www.w3schools.com/cssref/css_selectors.asp



CSS Selectors

In CSS, selectors are patterns used to select the element(s) you want to style.

Use our CSS Selector Tester to demonstrate the different selectors.

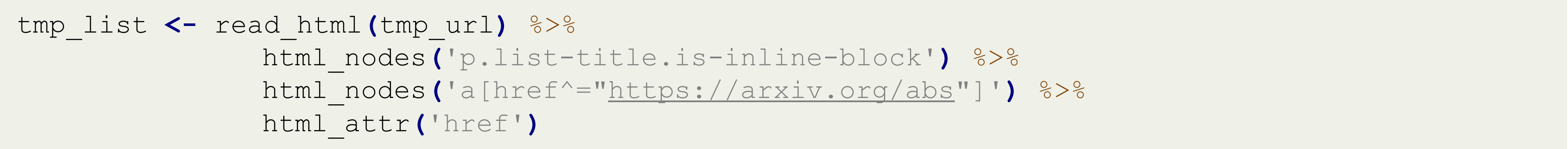
The "CSS" column indicates in which CSS version the property is defined (CSS1, CSS2, or CSS3).

Selector	Example	Example description	CSS
.class	.intro	Selects all elements with class="intro"	1
#id	#firstname	Selects the element with id="firstname"	1
*	*	Selects all elements	2
element	p	Selects all <p> elements	1
element,element	div, p	Selects all <div> elements and all <p> elements	1
element element	div p	Selects all <p> elements inside <div> elements	1
element>element	div > p	Selects all <p> elements where the parent is a <div> element	2
element+element	div + p	Selects all <p> elements that are placed immediately after <div> elements	2
element1~element2	p ~ ul	Selects every <ul> element that are preceded by a <p> element	3
[attribute]	[target]	Selects all elements with a target attribute	2
[attribute=value]	[target=_blank]	Selects all elements with target="_blank"	2
[attribute~=value]	[title~=flower]	Selects all elements with a title attribute containing the word "flower"	2
[attribute|=value]	[lang|=en]	Selects all elements with a lang attribute value starting with "en"	2
[attribute^=value]	a[href^="https"]	Selects every <a> element whose href attribute value begins with "https"	3
[attribute$=value]	a[href$=".pdf"]	Selects every <a> element whose href attribute value ends with ".pdf"	3
[attribute*=value]	a[href*="w3schools"]	Selects every <a> element whose href attribute value contains the substring "w3schools"	3

Web Scraping: arXiv Papers

- Step 2: Analyzing the HTML Structure

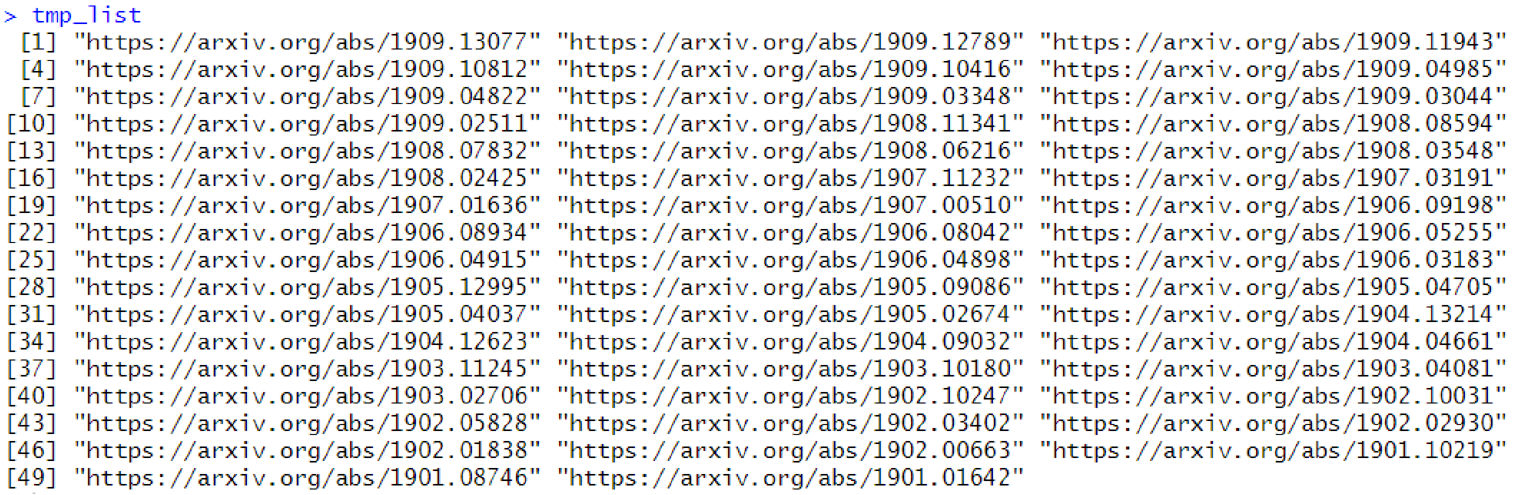
✓ Extract the link information



```
tmp_list <- read_html(tmp_url) %>%
               html_nodes('p.list-title.is-inline-block') %>%
               html_nodes('a[href^="https://arxiv.org/abs"]') %>%
               html_attr('href')
```

▪ find the node (p class = “list-title is –inline-block”) → find the node whose href attribute begins with https://arxiv.org/abs→ Store the attribute value of ‘href’ to the tmp_list

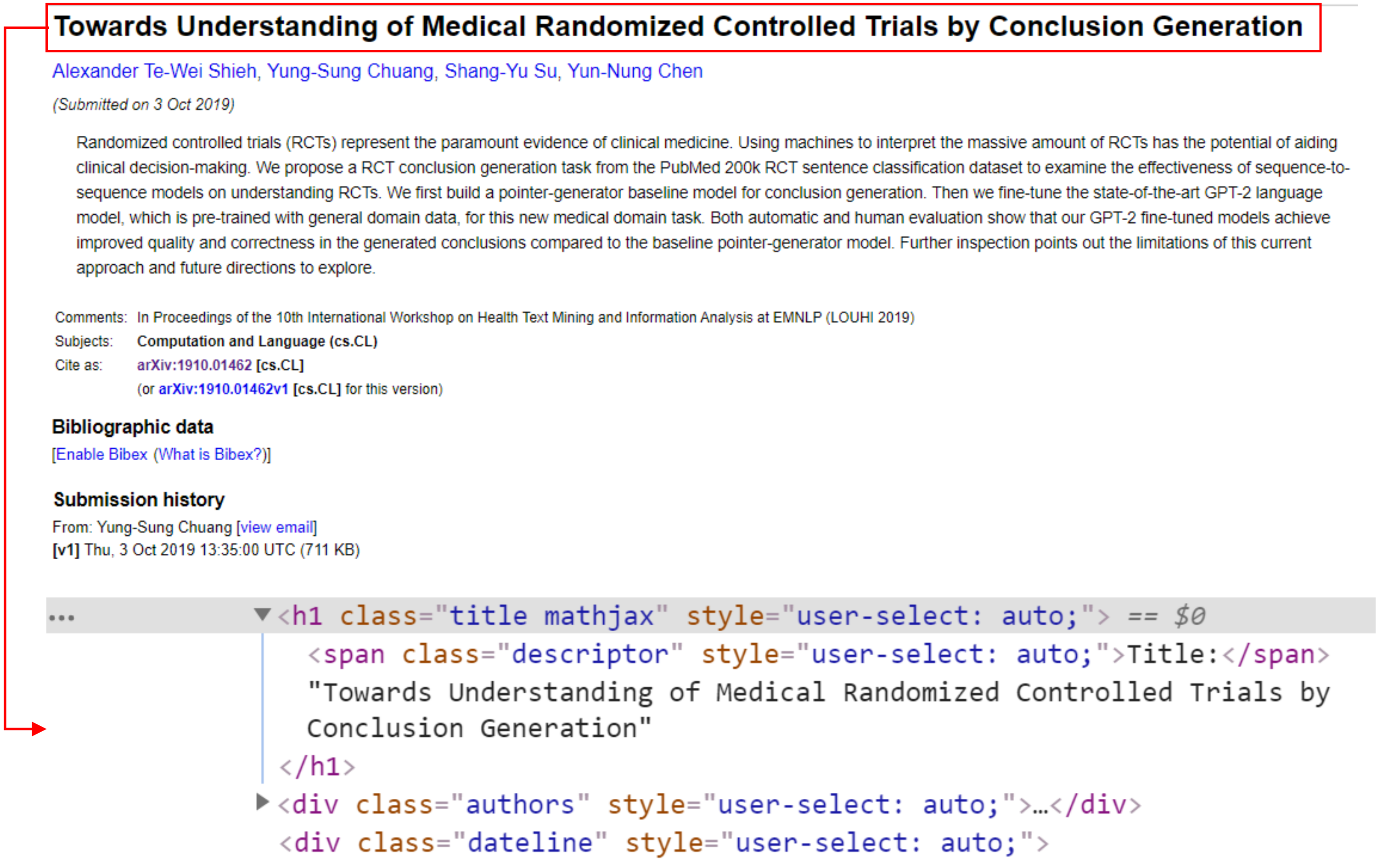
✓ Values that are stored in the “tmp_list”



```
> tmp_list
 [1] "https://arxiv.org/abs/1909.13077" "https://arxiv.org/abs/1909.12789" "https://arxiv.org/abs/1909.11943"
 [4] "https://arxiv.org/abs/1909.10812" "https://arxiv.org/abs/1909.10416" "https://arxiv.org/abs/1909.04985"
 [7] "https://arxiv.org/abs/1909.04822" "https://arxiv.org/abs/1909.03348" "https://arxiv.org/abs/1909.03044"
[10] "https://arxiv.org/abs/1909.02511" "https://arxiv.org/abs/1908.11341" "https://arxiv.org/abs/1908.08594"
[13] "https://arxiv.org/abs/1908.07832" "https://arxiv.org/abs/1908.06216" "https://arxiv.org/abs/1908.03548"
[16] "https://arxiv.org/abs/1908.02425" "https://arxiv.org/abs/1907.11232" "https://arxiv.org/abs/1907.03191"
[19] "https://arxiv.org/abs/1907.01636" "https://arxiv.org/abs/1907.00510" "https://arxiv.org/abs/1906.09198"
[22] "https://arxiv.org/abs/1906.08934" "https://arxiv.org/abs/1906.08042" "https://arxiv.org/abs/1906.05255"
[25] "https://arxiv.org/abs/1906.04915" "https://arxiv.org/abs/1906.04898" "https://arxiv.org/abs/1906.03183"
[28] "https://arxiv.org/abs/1905.12995" "https://arxiv.org/abs/1905.09086" "https://arxiv.org/abs/1905.04705"
[31] "https://arxiv.org/abs/1905.04037" "https://arxiv.org/abs/1905.02674" "https://arxiv.org/abs/1904.13214"
[34] "https://arxiv.org/abs/1904.12623" "https://arxiv.org/abs/1904.09032" "https://arxiv.org/abs/1904.04661"
[37] "https://arxiv.org/abs/1903.11245" "https://arxiv.org/abs/1903.10180" "https://arxiv.org/abs/1903.04081"
[40] "https://arxiv.org/abs/1903.02706" "https://arxiv.org/abs/1902.10247" "https://arxiv.org/abs/1902.10031"
[43] "https://arxiv.org/abs/1902.05828" "https://arxiv.org/abs/1902.03402" "https://arxiv.org/abs/1902.02930"
[46] "https://arxiv.org/abs/1902.01838" "https://arxiv.org/abs/1902.00663" "https://arxiv.org/abs/1901.10219"
[49] "https://arxiv.org/abs/1901.08746" "https://arxiv.org/abs/1901.01642"
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-1: Extract Title



Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

(Submitted on 3 Oct 2019)

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore.

Comments:	In Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis at EMNLP (LOUHI 2019)
Subjects:	**Computation and Language (cs.CL)**
Cite as:	**arXiv:1910.01462 [cs.CL]**
	(or **arXiv:1910.01462v1 [cs.CL]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

Submission history

From: Yung-Sung Chuang [view email]
[v1] Thu, 3 Oct 2019 13:35:00 UTC (711 KB)

```
▼<h1 class="title mathjax" style="user-select: auto;"> == $0
   <span class="descriptor" style="user-select: auto;">Title:</span>
   "Towards Understanding of Medical Randomized Controlled Trials by
   Conclusion Generation"
 </h1>
▶<div class="authors" style="user-select: auto;">…</div>
 <div class="dateline" style="user-select: auto;">
```

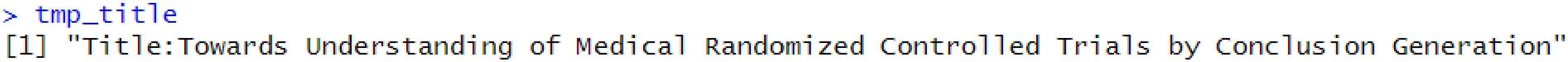
Web Scraping: arXiv Papers

- Step 3: Extract necessary information

  ✓ Step 3-1: Extract Title

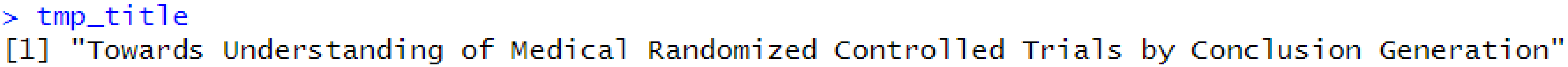
```r
# title
tmp_title <- tmp_paragraph %>% html_nodes('h1.title.mathjax') %>% html_text(T)
tmp_title <- gsub('Title:', '', tmp_title)
title <- c(title, tmp_title)
```

  - From tmp_paragraph → find the node whose h1 class name is "title mathjax" → extract the html text and store in to tmp_title



```
> tmp_title
[1] "Title:Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation"
```

  - Remove "Title:" from the tmp_title

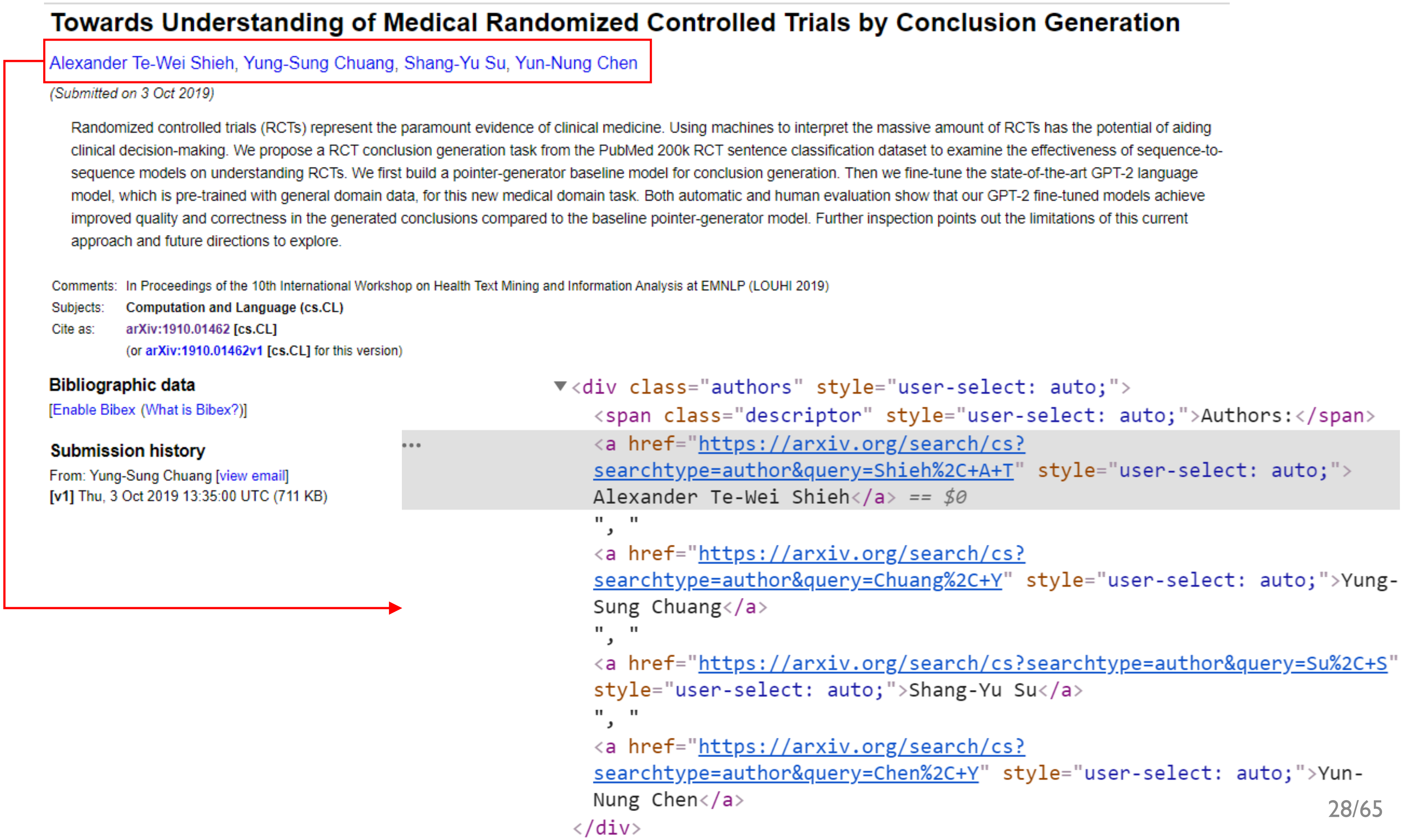


```
> tmp_title
[1] "Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation"
```

  - Append the tmp_title to title

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-2: Extract Authors



Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

(Submitted on 3 Oct 2019)

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore.

Comments:	In Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis at EMNLP (LOUHI 2019)
Subjects:	**Computation and Language (cs.CL)**
Cite as:	**arXiv:1910.01462 [cs.CL]**
	(or **arXiv:1910.01462v1 [cs.CL]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

Submission history

From: Yung-Sung Chuang [view email]
[v1] Thu, 3 Oct 2019 13:35:00 UTC (711 KB)

```
▼<div class="authors" style="user-select: auto;">
   <span class="descriptor" style="user-select: auto;">Authors:</span>
…  <a href="https://arxiv.org/search/cs?
   searchtype=author&query=Shieh%2C+A+T" style="user-select: auto;">
   Alexander Te-Wei Shieh</a> == $0
   ", "
   <a href="https://arxiv.org/search/cs?
   searchtype=author&query=Chuang%2C+Y" style="user-select: auto;">Yung-
   Sung Chuang</a>
   ", "
   <a href="https://arxiv.org/search/cs?searchtype=author&query=Su%2C+S"
   style="user-select: auto;">Shang-Yu Su</a>
   ", "
   <a href="https://arxiv.org/search/cs?
   searchtype=author&query=Chen%2C+Y" style="user-select: auto;">Yun-
   Nung Chen</a>
 </div>
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information

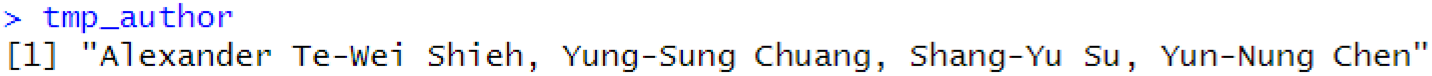
  ✓ Step 3-2: Extract Authors

```r
# author
tmp_author <- tmp_paragraph %>% html_nodes('div.authors') %>% html_text
tmp_author <- gsub('\\s+',' ',tmp_author)
tmp_author <- gsub('Authors:','',tmp_author) %>% str_trim
author <- c(author, tmp_author)
```

▪ From tmp_paragraph → Select node whose div class = “authors” → Store the html text

▪ Replace various spaces (space, tab, etc.) by a single space

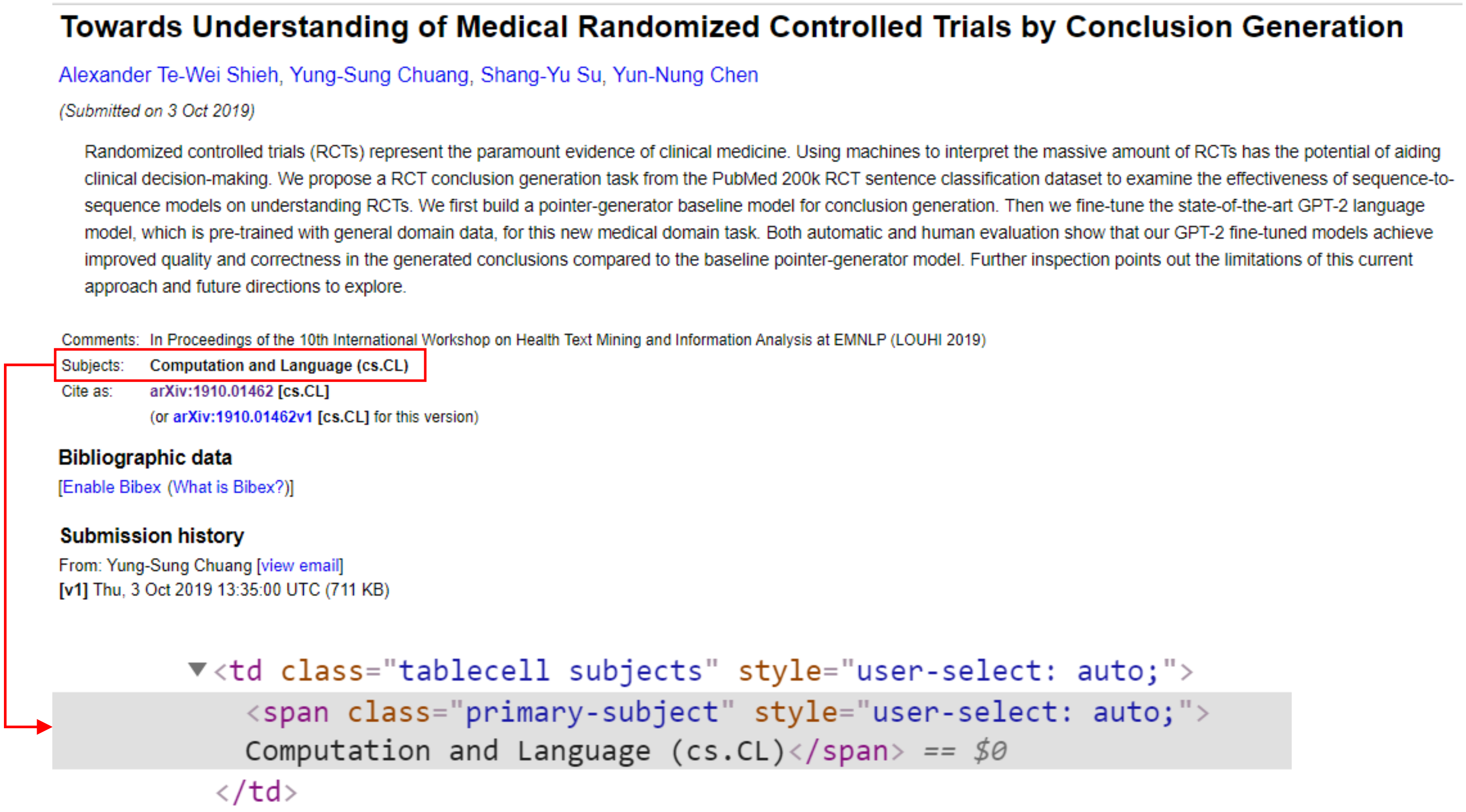
▪ Remove ‘Authors:“ and trim the string



```
> tmp_author
[1] "Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen"
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-3: Extract Subjects



Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

(Submitted on 3 Oct 2019)

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore.

Comments:	In Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis at EMNLP (LOUHI 2019)
Subjects:	**Computation and Language (cs.CL)**
Cite as:	**arXiv:1910.01462 [cs.CL]**
	(or **arXiv:1910.01462v1 [cs.CL]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

Submission history

From: Yung-Sung Chuang [view email]
[v1] Thu, 3 Oct 2019 13:35:00 UTC (711 KB)

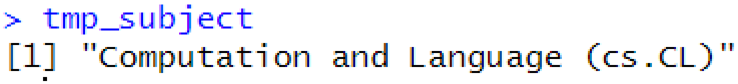
```
▼<td class="tablecell subjects" style="user-select: auto;">
   <span class="primary-subject" style="user-select: auto;">
   Computation and Language (cs.CL)</span> == $0
 </td>
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-3: Extract Subjects

```
# subject
tmp_subject <- tmp_paragraph %>% html_nodes('span.primary-subject') %>% html_text(T)
subject <- c(subject, tmp_subject)
```

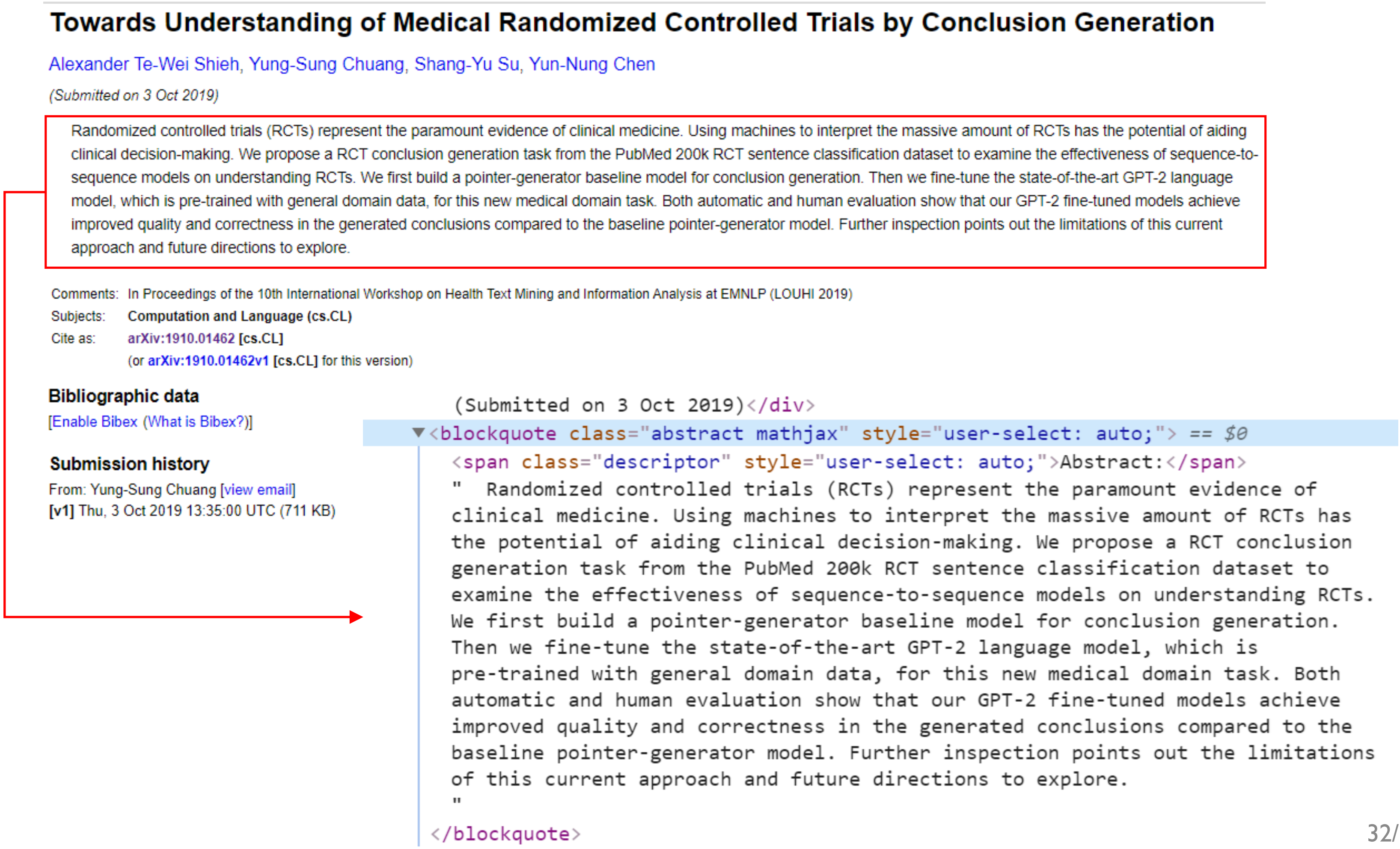
  - ▪ From tmp_paragraph → find the node whose span class = “primary-subject” → store the html text to tmp_subject



```
> tmp_subject
[1] "Computation and Language (cs.CL)"
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-4: Extract Abstract



Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

(Submitted on 3 Oct 2019)

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore.

Comments: In Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis at EMNLP (LOUHI 2019)
Subjects: **Computation and Language (cs.CL)**
Cite as: **arXiv:1910.01462 [cs.CL]**
(or **arXiv:1910.01462v1 [cs.CL]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

Submission history

From: Yung-Sung Chuang [view email]
[v1] Thu, 3 Oct 2019 13:35:00 UTC (711 KB)

```
  (Submitted on 3 Oct 2019)</div>
▼<blockquote class="abstract mathjax" style="user-select: auto;"> == $0
  <span class="descriptor" style="user-select: auto;">Abstract:</span>
  "  Randomized controlled trials (RCTs) represent the paramount evidence of
  clinical medicine. Using machines to interpret the massive amount of RCTs has
  the potential of aiding clinical decision-making. We propose a RCT conclusion
  generation task from the PubMed 200k RCT sentence classification dataset to
  examine the effectiveness of sequence-to-sequence models on understanding RCTs.
  We first build a pointer-generator baseline model for conclusion generation.
  Then we fine-tune the state-of-the-art GPT-2 language model, which is
  pre-trained with general domain data, for this new medical domain task. Both
  automatic and human evaluation show that our GPT-2 fine-tuned models achieve
  improved quality and correctness in the generated conclusions compared to the
  baseline pointer-generator model. Further inspection points out the limitations
  of this current approach and future directions to explore.
  "
 </blockquote>
```

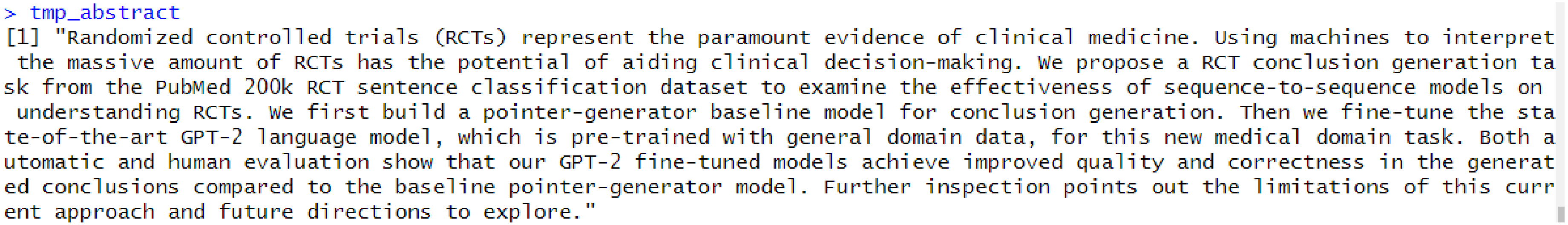
Web Scraping: arXiv Papers

- Step 3: Extract necessary information

  ✓ Step 3-4: Extract Abstract

```
# abstract
tmp_abstract <- tmp_paragraph %>% html_nodes('blockquote.abstract.mathjax') %>% html_text(T)
tmp_abstract <- gsub('\\s+',' ',tmp_abstract)
tmp_abstract <- sub('Abstract:','',tmp_abstract) %>% str_trim
abstract <- c(abstract, tmp_abstract)
```

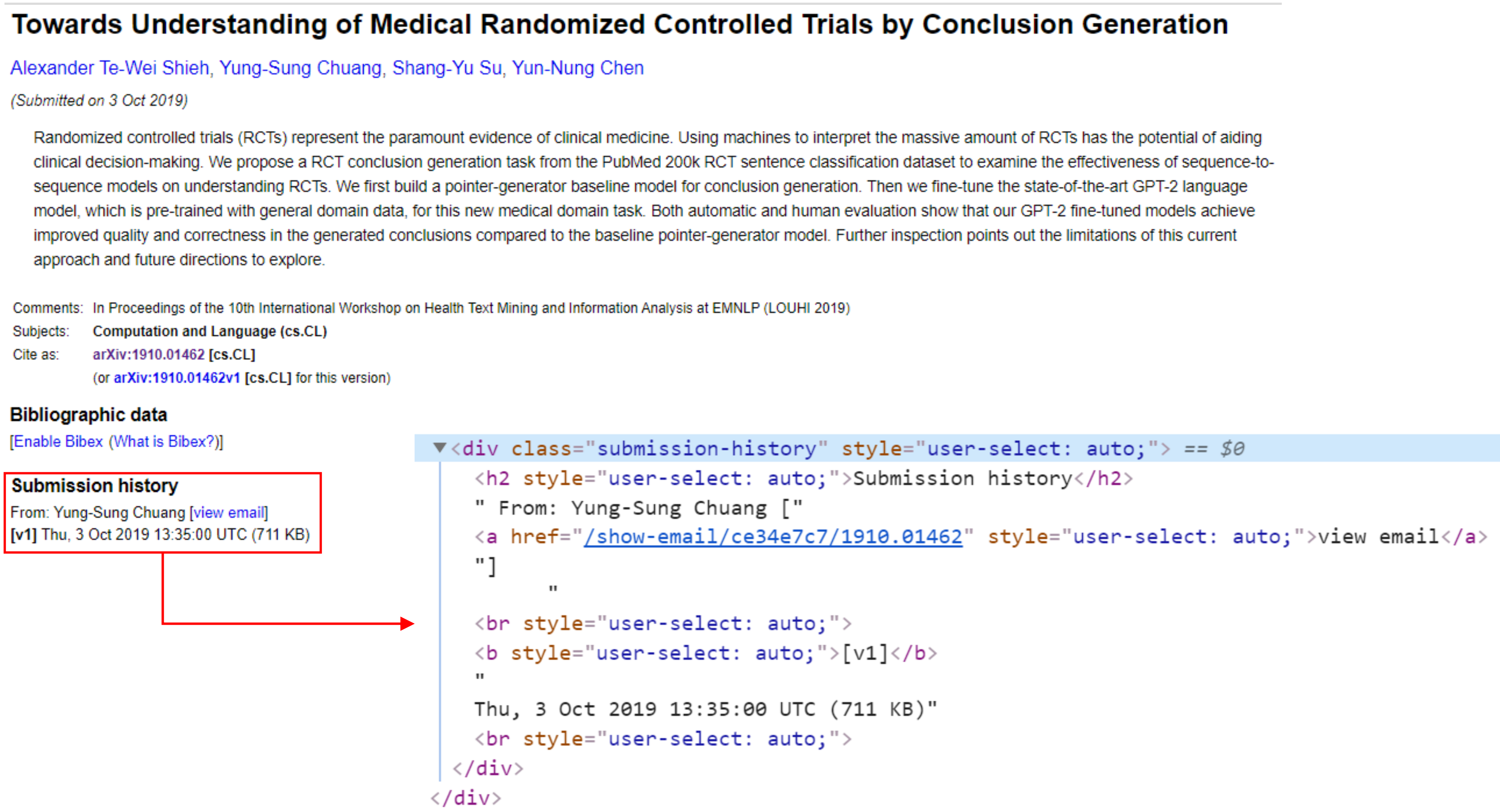
  - From tmp_paragraph → find the node whose blockquote class = "abstract mathjax" → Store the html text to tmp_abstract
  - Remove "Abstract:" and trim the text



```
> tmp_abstract
[1] "Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret
 the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation ta
sk from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on
 understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the sta
te-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both a
utomatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generat
ed conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this curr
ent approach and future directions to explore."
```

Web Scraping: arXiv Papers

- Step 3: Extract necessary information
  - ✓ Step 3-5: Extract Meta information



Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, Yun-Nung Chen

(Submitted on 3 Oct 2019)

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore.

Comments:	In Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis at EMNLP (LOUHI 2019)
Subjects:	**Computation and Language (cs.CL)**
Cite as:	arXiv:1910.01462 **[cs.CL]**
	(or **arXiv:1910.01462v1 [cs.CL]** for this version)

Bibliographic data

[Enable Bibex (What is Bibex?)]

Submission history

From: Yung-Sung Chuang [view email]

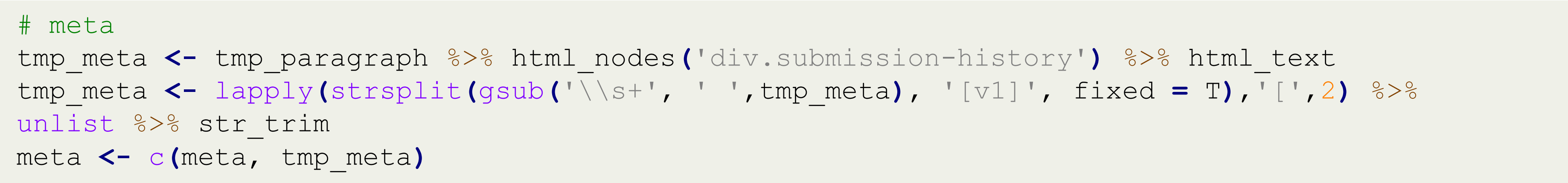
[v1] Thu, 3 Oct 2019 13:35:00 UTC (711 KB)

```
▼<div class="submission-history" style="user-select: auto;"> == $0
   <h2 style="user-select: auto;">Submission history</h2>
   " From: Yung-Sung Chuang ["
   <a href="/show-email/ce34e7c7/1910.01462" style="user-select: auto;">view email</a>
   "]
         "
   <br style="user-select: auto;">
   <b style="user-select: auto;">[v1]</b>
   "
   Thu, 3 Oct 2019 13:35:00 UTC (711 KB)"
   <br style="user-select: auto;">
 </div>
</div>
```

Web Scraping: arXiv Papers

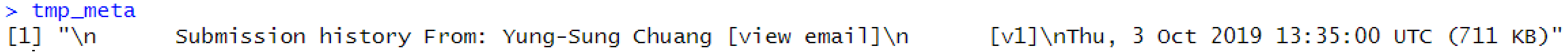
- Step 3: Extract necessary information

  ✓ Step 3-5: Extract Meta information



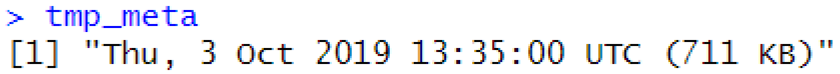
```r
# meta
tmp_meta <- tmp_paragraph %>% html_nodes('div.submission-history') %>% html_text
tmp_meta <- lapply(strsplit(gsub('\\s+', ' ',tmp_meta), '[v1]', fixed = T),'[',2) %>%
unlist %>% str_trim
meta <- c(meta, tmp_meta)
```

- From tmp_paragraph → find the node whose div class name is "submission-history" → Store the html text to tmp_meta



```
> tmp_meta
[1] "\n      Submission history From: Yung-Sung Chuang [view email]\n      [v1]\nThu, 3 Oct 2019 13:35:00 UTC (711 KB)"
```

- Replace all spaces by a single space → Split the text (split point = [v1]) → Take the second element → Unlist it → trim the text

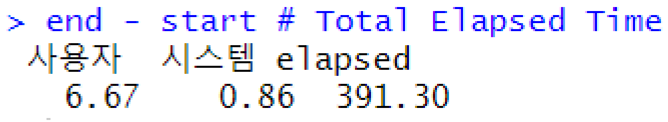


```
> tmp_meta
[1] "Thu, 3 Oct 2019 13:35:00 UTC (711 KB)"
```

Web Scraping: arXiv Papers

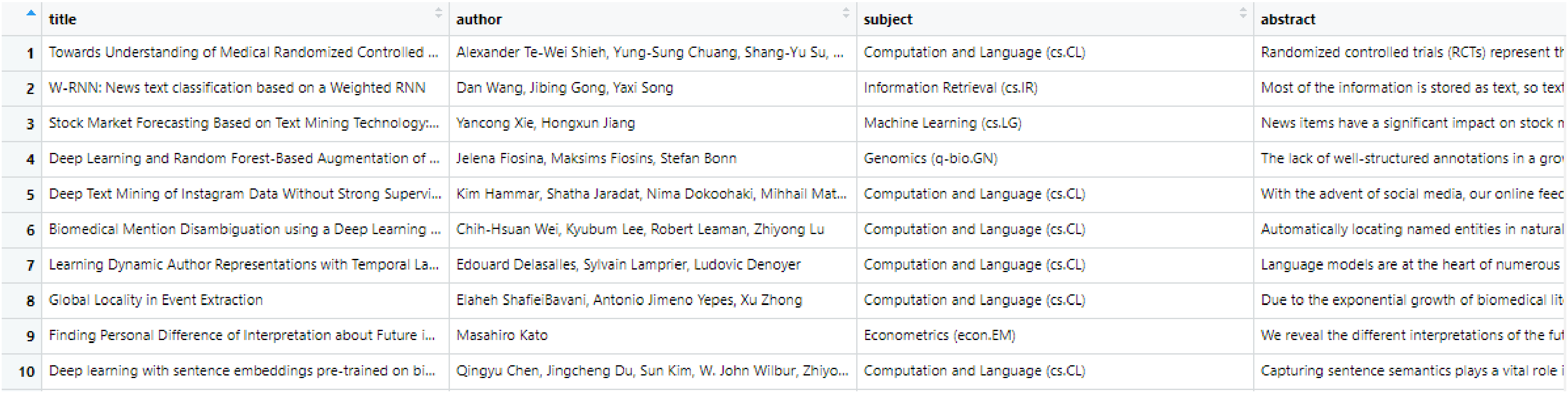
- Step 4: Repeat the process and export the data

  ✓ Elapsed time for data collection



```
> end - start # Total Elapsed Time
 사용자  시스템 elapsed 
   6.67    0.86  391.30 
```

  ✓ Check the dataset



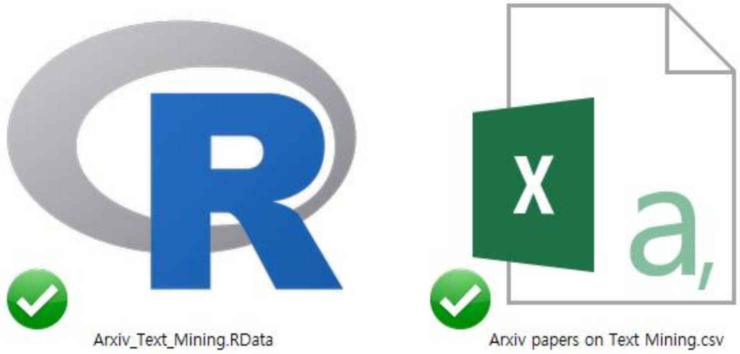
	title	author	subject	abstract
1	Towards Understanding of Medical Randomized Controlled ...	Alexander Te-Wei Shieh, Yung-Sung Chuang, Shang-Yu Su, ...	Computation and Language (cs.CL)	Randomized controlled trials (RCTs) represent th
2	W-RNN: News text classification based on a Weighted RNN	Dan Wang, Jibing Gong, Yaxi Song	Information Retrieval (cs.IR)	Most of the information is stored as text, so text
3	Stock Market Forecasting Based on Text Mining Technology:...	Yancong Xie, Hongxun Jiang	Machine Learning (cs.LG)	News items have a significant impact on stock m
4	Deep Learning and Random Forest-Based Augmentation of ...	Jelena Fiosina, Maksims Fiosins, Stefan Bonn	Genomics (q-bio.GN)	The lack of well-structured annotations in a grow
5	Deep Text Mining of Instagram Data Without Strong Supervi...	Kim Hammar, Shatha Jaradat, Nima Dokoohaki, Mihhail Mat...	Computation and Language (cs.CL)	With the advent of social media, our online feec
6	Biomedical Mention Disambiguation using a Deep Learning ...	Chih-Hsuan Wei, Kyubum Lee, Robert Leaman, Zhiyong Lu	Computation and Language (cs.CL)	Automatically locating named entities in natural
7	Learning Dynamic Author Representations with Temporal La...	Edouard Delasalles, Sylvain Lamprier, Ludovic Denoyer	Computation and Language (cs.CL)	Language models are at the heart of numerous
8	Global Locality in Event Extraction	Elaheh ShafieiBavani, Antonio Jimeno Yepes, Xu Zhong	Computation and Language (cs.CL)	Due to the exponential growth of biomedical lit
9	Finding Personal Difference of Interpretation about Future i...	Masahiro Kato	Econometrics (econ.EM)	We reveal the different interpretations of the fut
10	Deep learning with sentence embeddings pre-trained on bi...	Qingyu Chen, Jingcheng Du, Sun Kim, W. John Wilbur, Zhiyo...	Computation and Language (cs.CL)	Capturing sentence semantics plays a vital role i

Web Scraping: arXiv Papers

- Step 4: Repeat the process and export the data
  - ✓ Store the dataframe as an RData format or export it as a csv file

```r
# Export the result
save(final, file = "Arxiv_Text_Mining.RData")
write.csv(papers, file = "Arxiv papers on Text Mining.csv")
```

  - ✓ You can find the following two files in your working directory



AGENDA

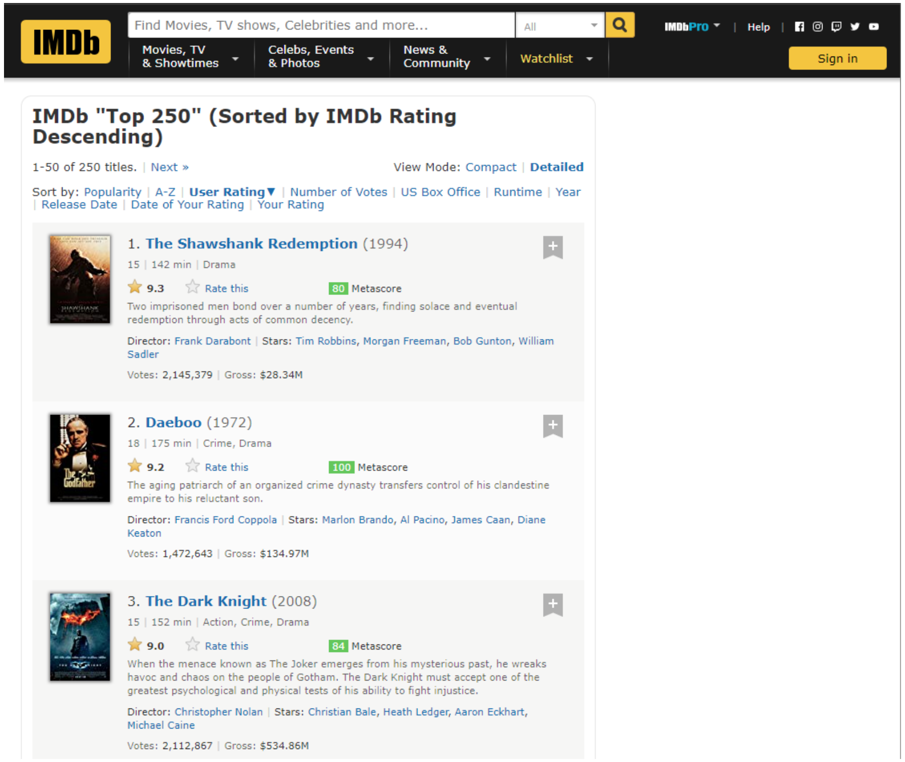
01 Prerequisites

02 Web Scraping: arXiv Research Papers

03 Web Scraping: Movie Reviews

Web Scraping: IMDB Top 50 Movie Reviews

- Web scraping example 2: Movie review scraping (IMDB Top 50 Movies)
  - ✓ IMDB Top 250: https://www.imdb.com/search/title/?groups=top_250&sort=user_rating
  - ✓ Collect Title, Year, Average Rating, Total Number of Ratings, Summary, Director, Writer, Review Rating, Review Title, Review Text



IMDb "Top 250" (Sorted by IMDb Rating Descending)

1-50 of 250 titles. | Next »

View Mode: Compact | **Detailed**

Sort by: Popularity | A-Z | **User Rating▼** | Number of Votes | US Box Office | Runtime | Year | Release Date | Date of Your Rating | Your Rating

1. **The Shawshank Redemption** (1994)

15 | 142 min | Drama

★ 9.3 ☆ Rate this 80 Metascore

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont | Stars: Tim Robbins, Morgan Freeman, Bob Gunton, William Sadler

Votes: 2,145,379 | Gross: $28.34M

2. **Daeboo** (1972)

18 | 175 min | Crime, Drama

★ 9.2 ☆ Rate this 100 Metascore

The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.

Director: Francis Ford Coppola | Stars: Marlon Brando, Al Pacino, James Caan, Diane Keaton

Votes: 1,472,643 | Gross: $134.97M

3. **The Dark Knight** (2008)

15 | 152 min | Action, Crime, Drama

★ 9.0 ☆ Rate this 84 Metascore

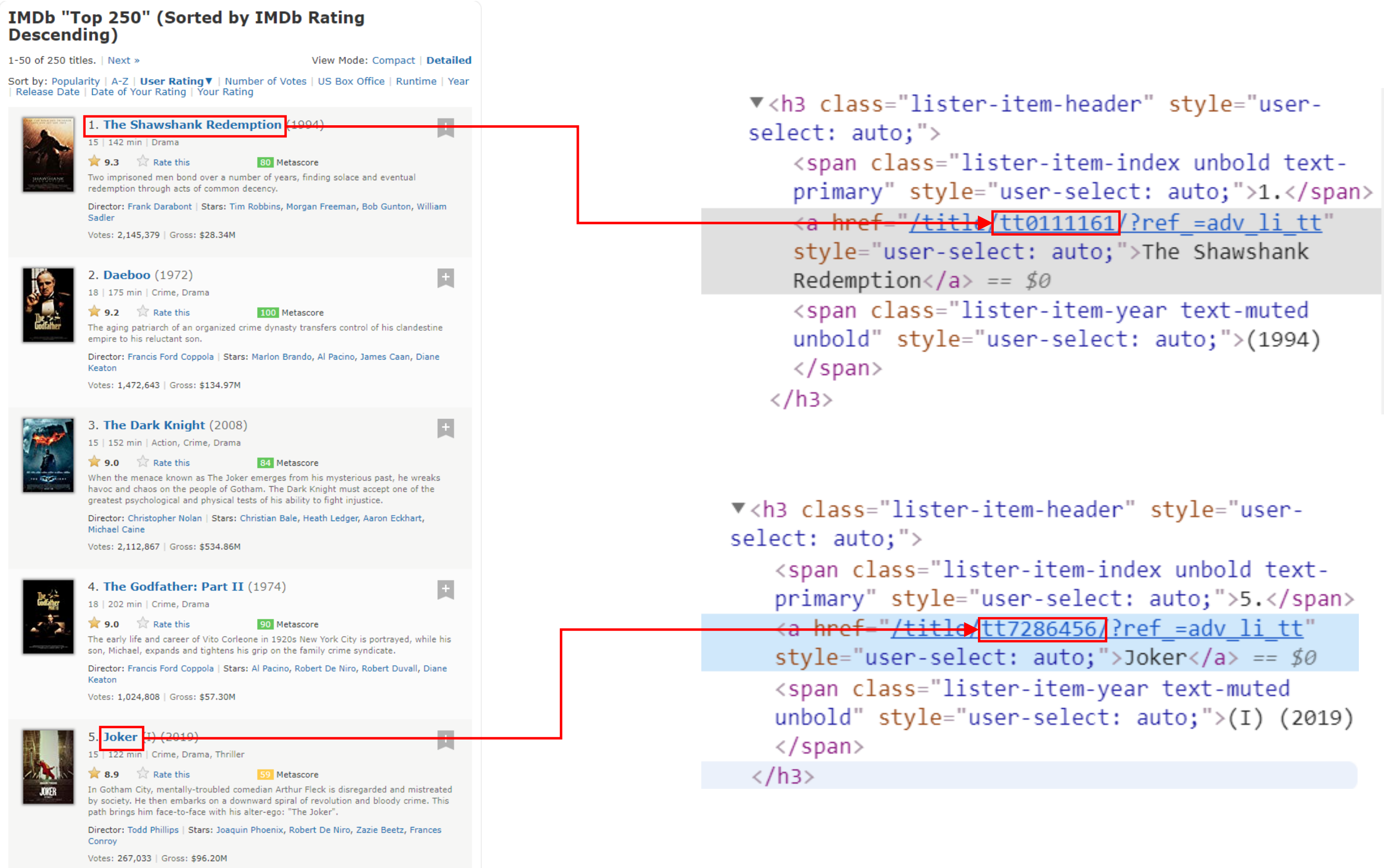
When the menace known as The Joker emerges from his mysterious past, he wreaks havoc and chaos on the people of Gotham. The Dark Knight must accept one of the greatest psychological and physical tests of his ability to fight injustice.

Director: Christopher Nolan | Stars: Christian Bale, Heath Ledger, Aaron Eckhart, Michael Caine

Votes: 2,112,867 | Gross: $534.86M

Web Scraping: IMDB Top 50 Movie Reviews

- Step 1: Understand the basic structure
  - ✓ A total of 250 movies are listed, each page contains 50 movies
  - ✓ Each movie has a unique ID



IMDb "Top 250" (Sorted by IMDb Rating Descending)

1-50 of 250 titles. | Next » View Mode: Compact | **Detailed**

Sort by: Popularity | A-Z | **User Rating▼** | Number of Votes | US Box Office | Runtime | Year | Release Date | Date of Your Rating | Your Rating

1. **The Shawshank Redemption** (1994)
15 | 142 min | Drama
★ 9.3 ☆ Rate this 80 Metascore
Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.
Director: Frank Darabont | Stars: Tim Robbins, Morgan Freeman, Bob Gunton, William Sadler
Votes: 2,145,379 | Gross: $28.34M

2. **Daeboo** (1972)
18 | 175 min | Crime, Drama
★ 9.2 ☆ Rate this 100 Metascore
The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.
Director: Francis Ford Coppola | Stars: Marlon Brando, Al Pacino, James Caan, Diane Keaton
Votes: 1,472,643 | Gross: $134.97M

3. **The Dark Knight** (2008)
15 | 152 min | Action, Crime, Drama
★ 9.0 ☆ Rate this 84 Metascore
When the menace known as The Joker emerges from his mysterious past, he wreaks havoc and chaos on the people of Gotham. The Dark Knight must accept one of the greatest psychological and physical tests of his ability to fight injustice.
Director: Christopher Nolan | Stars: Christian Bale, Heath Ledger, Aaron Eckhart, Michael Caine
Votes: 2,112,867 | Gross: $534.86M

4. **The Godfather: Part II** (1974)
18 | 202 min | Crime, Drama
★ 9.0 ☆ Rate this 90 Metascore
The early life and career of Vito Corleone in 1920s New York City is portrayed, while his son, Michael, expands and tightens his grip on the family crime syndicate.
Director: Francis Ford Coppola | Stars: Al Pacino, Robert De Niro, Robert Duvall, Diane Keaton
Votes: 1,024,808 | Gross: $57.30M

5. **Joker** (I) (2019)
15 | 122 min | Crime, Drama, Thriller
★ 8.9 ☆ Rate this 59 Metascore
In Gotham City, mentally-troubled comedian Arthur Fleck is disregarded and mistreated by society. He then embarks on a downward spiral of revolution and bloody crime. This path brings him face-to-face with his alter-ego: "The Joker".
Director: Todd Phillips | Stars: Joaquin Phoenix, Robert De Niro, Zazie Beetz, Frances Conroy
Votes: 267,033 | Gross: $96.20M

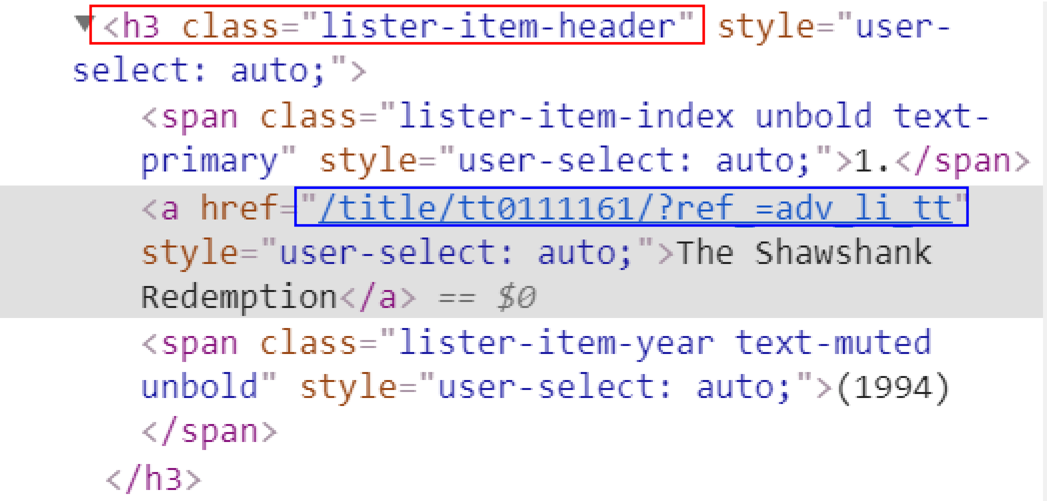
```
▼<h3 class="lister-item-header" style="user-
select: auto;">
   <span class="lister-item-index unbold text-
   primary" style="user-select: auto;">1.</span>
   <a href="/title/tt0111161/?ref_=adv_li_tt"
   style="user-select: auto;">The Shawshank
   Redemption</a> == $0
   <span class="lister-item-year text-muted
   unbold" style="user-select: auto;">(1994)
   </span>
 </h3>
```

```
▼<h3 class="lister-item-header" style="user-
select: auto;">
   <span class="lister-item-index unbold text-
   primary" style="user-select: auto;">5.</span>
   <a href="/title/tt7286456/?ref_=adv_li_tt"
   style="user-select: auto;">Joker</a> == $0
   <span class="lister-item-year text-muted
   unbold" style="user-select: auto;">(I) (2019)
   </span>
 </h3>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Step 2: Get the url of each movie

```
url <- 'https://www.imdb.com/search/title/?groups=top_250&sort=user_rating'
...
tmp_list <- read_html(url) %>% html_nodes('h3.lister-item-header') %>%
                                html_nodes('a[href^="/title"]') %>% html_attr('href')
```

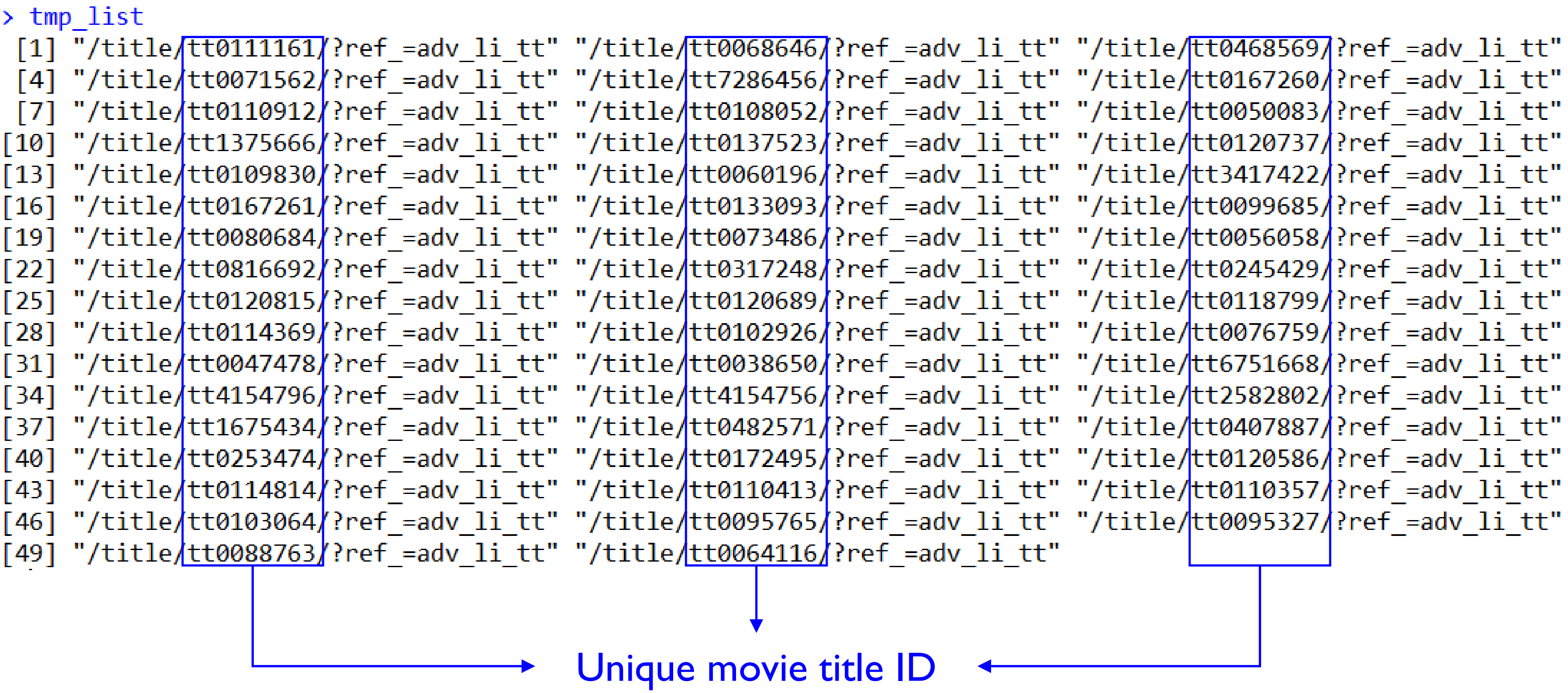


```
▼<h3 class="lister-item-header" style="user-select: auto;">
    <span class="lister-item-index unbold text-primary" style="user-select: auto;">1.</span>
    <a href="/title/tt0111161/?ref_=adv_li_tt" style="user-select: auto;">The Shawshank Redemption</a> == $0
    <span class="lister-item-year text-muted unbold" style="user-select: auto;">(1994)
    </span>
  </h3>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Step 2: Get the url of each movie

```r
url <- 'https://www.imdb.com/search/title/?groups=top_250&sort=user_rating'
...
tmp_list <- read_html(url) %>% html_nodes('h3.lister-item-header') %>%
                              html_nodes('a[href^="/title"]') %>% html_attr('href')
```



```
> tmp_list
 [1] "/title/tt0111161/?ref_=adv_li_tt" "/title/tt0068646/?ref_=adv_li_tt" "/title/tt0468569/?ref_=adv_li_tt"
 [4] "/title/tt0071562/?ref_=adv_li_tt" "/title/tt7286456/?ref_=adv_li_tt" "/title/tt0167260/?ref_=adv_li_tt"
 [7] "/title/tt0110912/?ref_=adv_li_tt" "/title/tt0108052/?ref_=adv_li_tt" "/title/tt0050083/?ref_=adv_li_tt"
[10] "/title/tt1375666/?ref_=adv_li_tt" "/title/tt0137523/?ref_=adv_li_tt" "/title/tt0120737/?ref_=adv_li_tt"
[13] "/title/tt0109830/?ref_=adv_li_tt" "/title/tt0060196/?ref_=adv_li_tt" "/title/tt3417422/?ref_=adv_li_tt"
[16] "/title/tt0167261/?ref_=adv_li_tt" "/title/tt0133093/?ref_=adv_li_tt" "/title/tt0099685/?ref_=adv_li_tt"
[19] "/title/tt0080684/?ref_=adv_li_tt" "/title/tt0073486/?ref_=adv_li_tt" "/title/tt0056058/?ref_=adv_li_tt"
[22] "/title/tt0816692/?ref_=adv_li_tt" "/title/tt0317248/?ref_=adv_li_tt" "/title/tt0245429/?ref_=adv_li_tt"
[25] "/title/tt0120815/?ref_=adv_li_tt" "/title/tt0120689/?ref_=adv_li_tt" "/title/tt0118799/?ref_=adv_li_tt"
[28] "/title/tt0114369/?ref_=adv_li_tt" "/title/tt0102926/?ref_=adv_li_tt" "/title/tt0076759/?ref_=adv_li_tt"
[31] "/title/tt0047478/?ref_=adv_li_tt" "/title/tt0038650/?ref_=adv_li_tt" "/title/tt6751668/?ref_=adv_li_tt"
[34] "/title/tt4154796/?ref_=adv_li_tt" "/title/tt4154756/?ref_=adv_li_tt" "/title/tt2582802/?ref_=adv_li_tt"
[37] "/title/tt1675434/?ref_=adv_li_tt" "/title/tt0482571/?ref_=adv_li_tt" "/title/tt0407887/?ref_=adv_li_tt"
[40] "/title/tt0253474/?ref_=adv_li_tt" "/title/tt0172495/?ref_=adv_li_tt" "/title/tt0120586/?ref_=adv_li_tt"
[43] "/title/tt0114814/?ref_=adv_li_tt" "/title/tt0110413/?ref_=adv_li_tt" "/title/tt0110357/?ref_=adv_li_tt"
[46] "/title/tt0103064/?ref_=adv_li_tt" "/title/tt0095765/?ref_=adv_li_tt" "/title/tt0095327/?ref_=adv_li_tt"
[49] "/title/tt0088763/?ref_=adv_li_tt" "/title/tt0064116/?ref_=adv_li_tt"
```

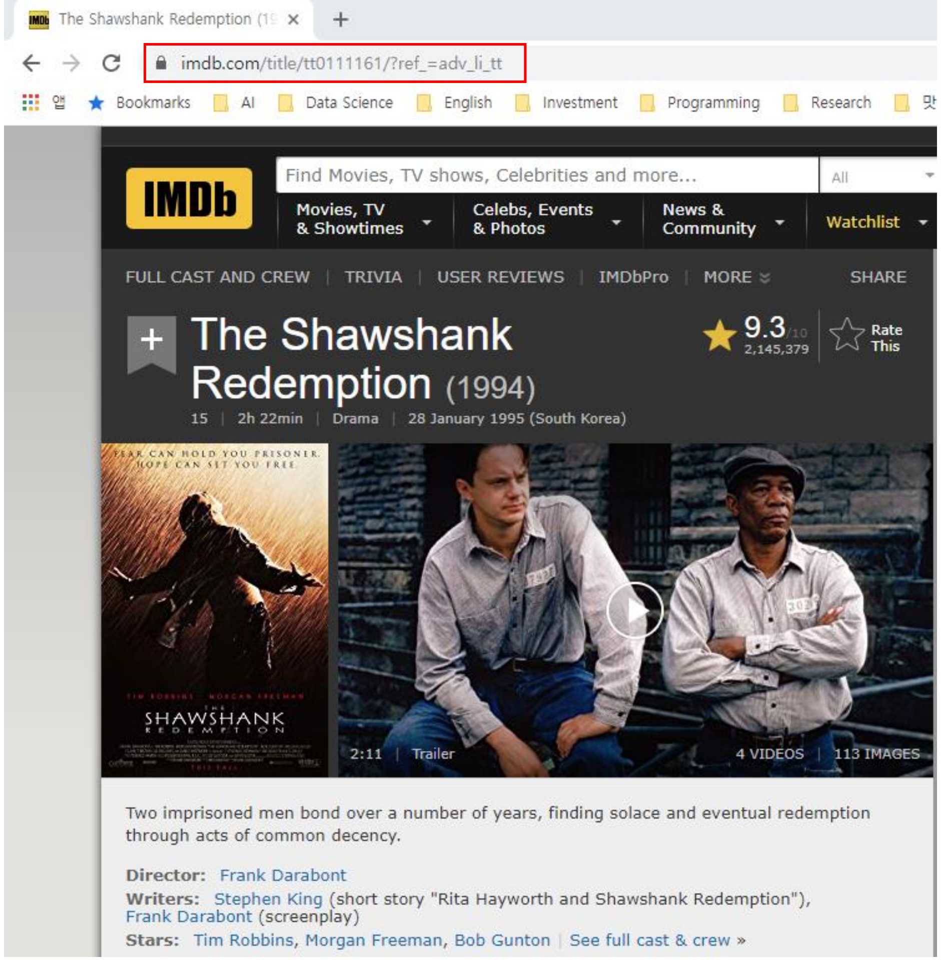
Unique movie title ID

Web Scraping: IMDB Top 50 Movie Reviews

- Meta Information on Each Movie

```
tmp_url <- paste('http://imdb.com', tmp_list[i], sep="")
tmp_content <- read_html(tmp_url)
```

```
> tmp_url
[1] "http://imdb.com/title/tt0111161/?ref_=adv_li_tt"
```



The Shawshank Redemption (1994)

15 | 2h 22min | Drama | 28 January 1995 (South Korea)

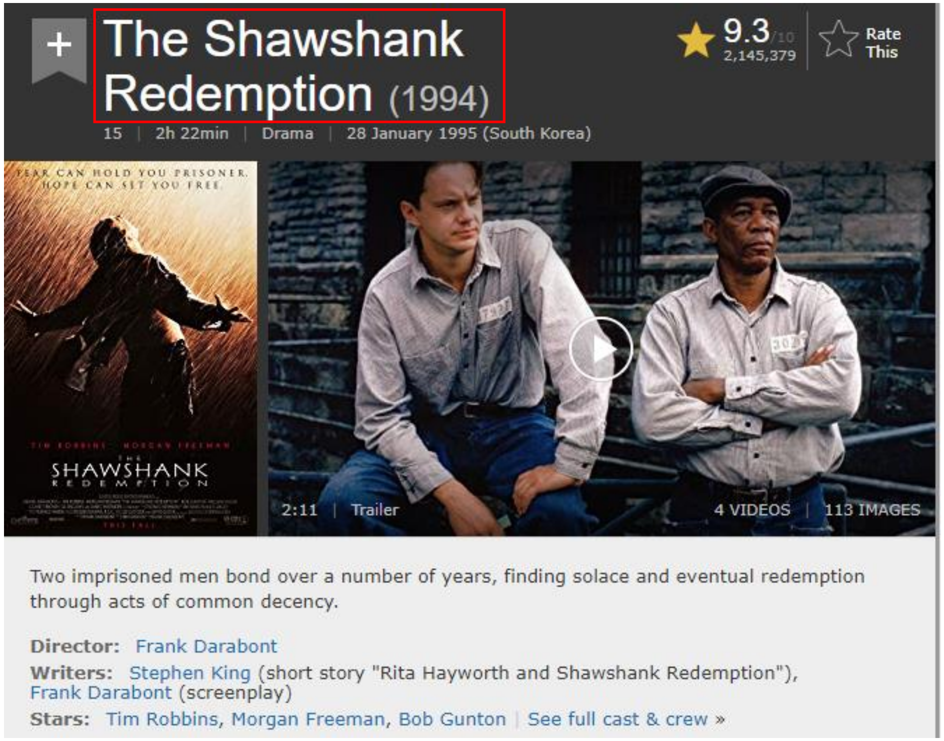
Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont
Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)
Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »

Web Scraping: IMDB Top 50 Movie Reviews

- Title and Year

```
# Extract title and year
title_year <- tmp_content %>% html_nodes('div.title_wrapper > h1') %>% html_text %>% str_trim
tmp_title <- substr(title_year, 1, nchar(title_year)-7)
tmp_year <- substr(title_year, nchar(title_year)-4, nchar(title_year)-1)
tmp_year <- as.numeric(tmp_year)
```



The Shawshank Redemption (1994)

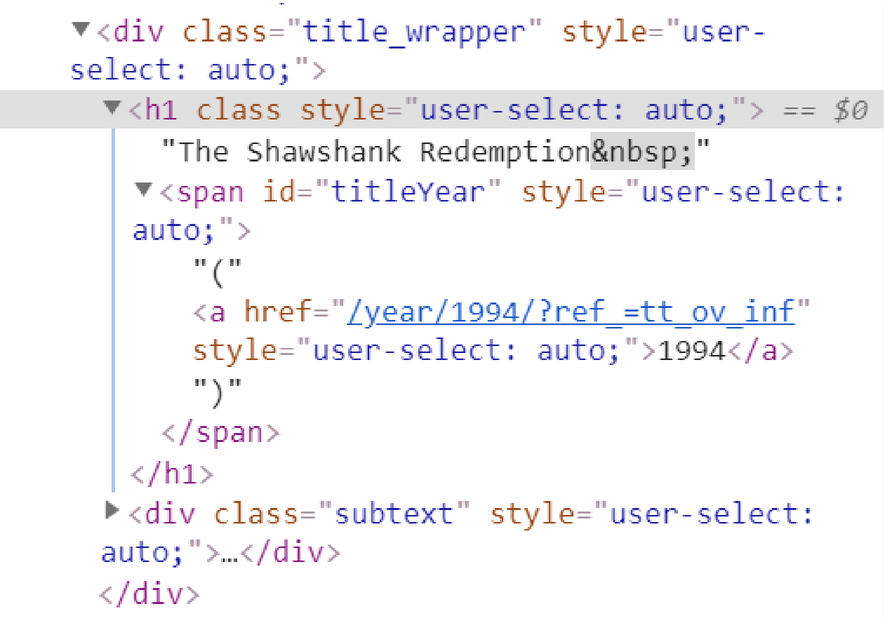
9.3/10 2,145,379

Rate This

15 | 2h 22min | Drama | 28 January 1995 (South Korea)

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont
Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)
Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »

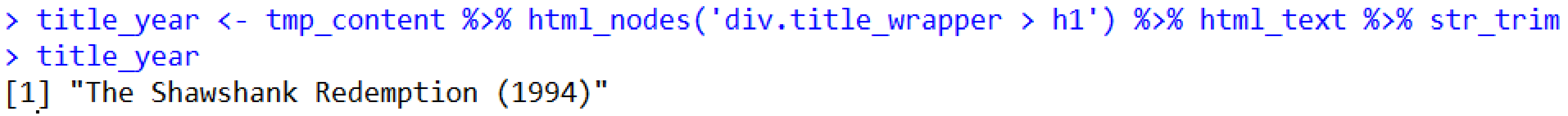


```
▼<div class="title_wrapper" style="user-
select: auto;">
  ▼<h1 class style="user-select: auto;"> == $0
     "The Shawshank Redemption&nbsp;"
    ▼<span id="titleYear" style="user-select:
    auto;">
       "("
       <a href="/year/1994/?ref_=tt_ov_inf"
       style="user-select: auto;">1994</a>
       ")"
     </span>
    </h1>
  ►<div class="subtext" style="user-select:
  auto;">…</div>
  </div>
```

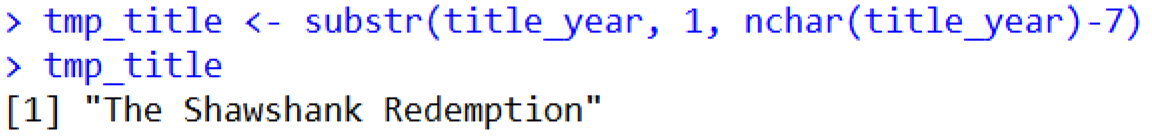
Web Scraping: IMDB Top 50 Movie Reviews

- Title and Year

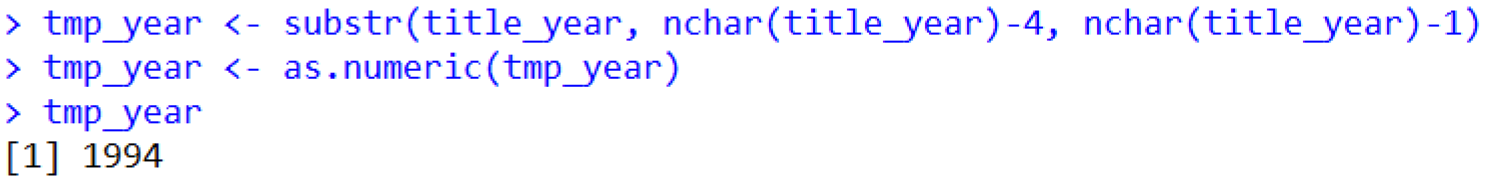
```
# Extract title and year
title_year <- tmp_content %>% html_nodes('div.title_wrapper > h1') %>% html_text %>% str_trim
tmp_title <- substr(title_year, 1, nchar(title_year)-7)
tmp_year <- substr(title_year, nchar(title_year)-4, nchar(title_year)-1)
tmp_year <- as.numeric(tmp_year)
```



```
> title_year <- tmp_content %>% html_nodes('div.title_wrapper > h1') %>% html_text %>% str_trim
> title_year
[1] "The Shawshank Redemption (1994)"
```



```
> tmp_title <- substr(title_year, 1, nchar(title_year)-7)
> tmp_title
[1] "The Shawshank Redemption"
```

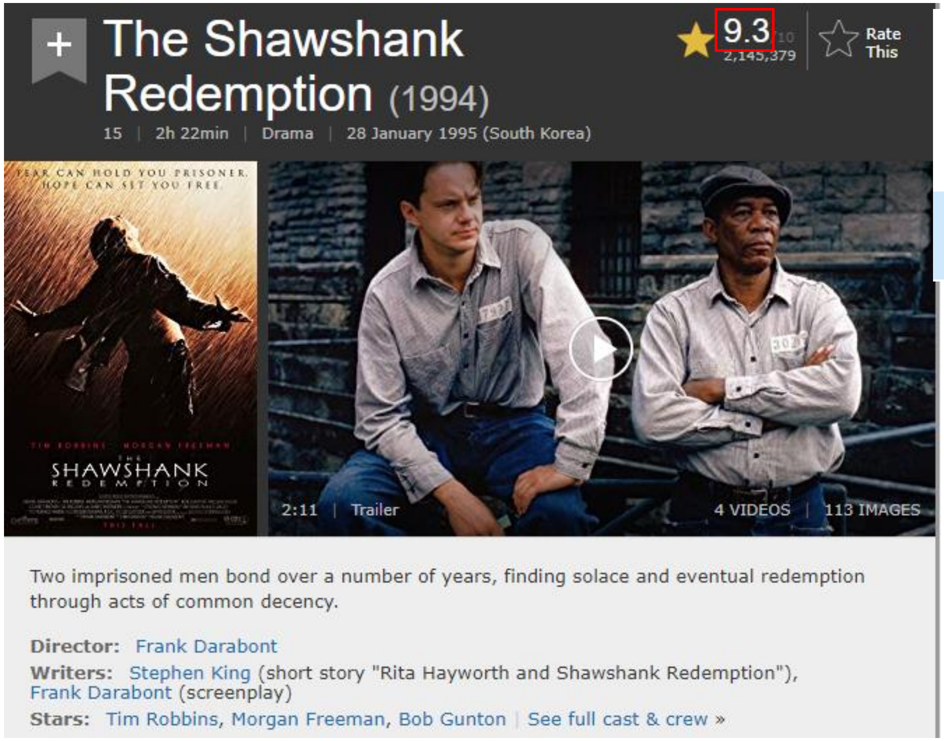


```
> tmp_year <- substr(title_year, nchar(title_year)-4, nchar(title_year)-1)
> tmp_year <- as.numeric(tmp_year)
> tmp_year
[1] 1994
```

Web Scraping: IMDB Top 50 Movie Reviews

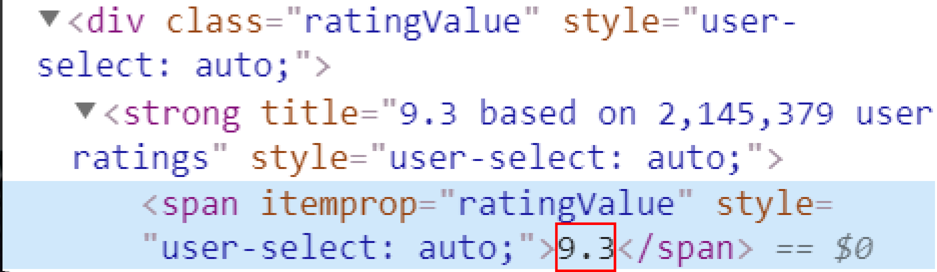
- Average Rating

```r
# Average rating
tmp_rating <- tmp_content %>% html_nodes('div.ratingValue > strong > span') %>% html_text
tmp_rating <- as.numeric(tmp_rating)
```


+
The Shawshank Redemption (1994)
9.3/10
2,145,379
Rate This
15 | 2h 22min | Drama | 28 January 1995 (South Korea)
2:11 | Trailer
4 VIDEOS | 113 IMAGES

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont
Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)
Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »

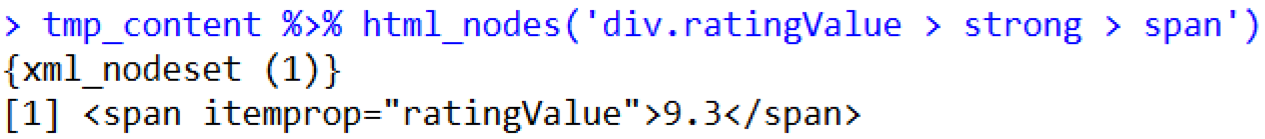


```html
▼<div class="ratingValue" style="user-select: auto;">
  ▼<strong title="9.3 based on 2,145,379 user ratings" style="user-select: auto;">
      <span itemprop="ratingValue" style="user-select: auto;">9.3</span> == $0
```

Web Scraping: IMDB Top 50 Movie Reviews

- Average Rating

```r
# Average rating
tmp_rating <- tmp_content %>% html_nodes('div.ratingValue > strong > span') %>% html_text
tmp_rating <- as.numeric(tmp_rating)
```

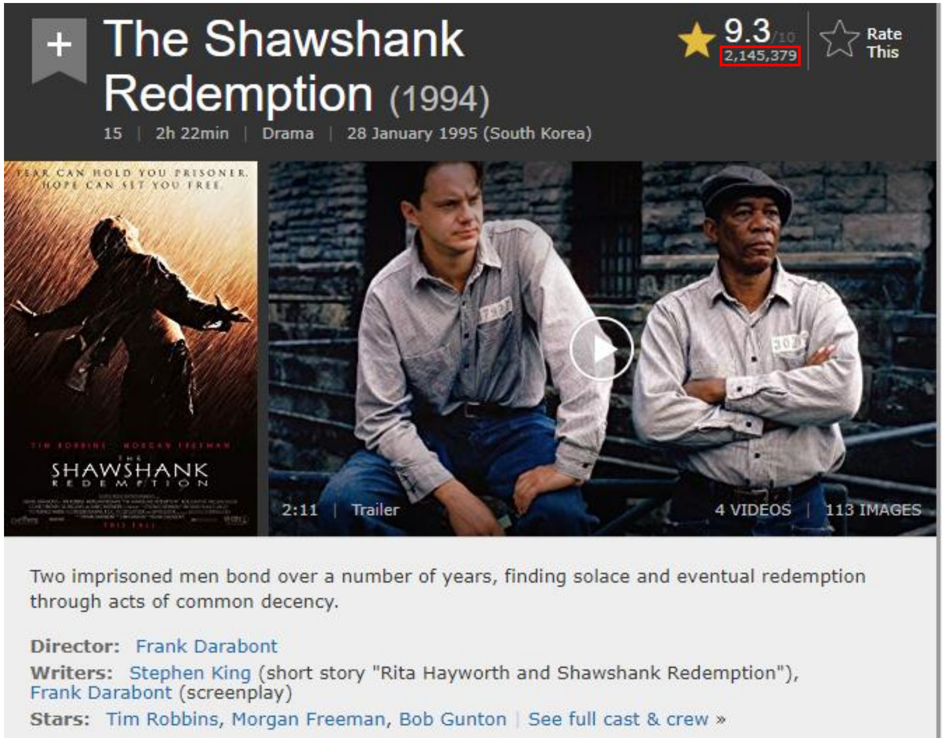


```
> tmp_content %>% html_nodes('div.ratingValue > strong > span')
{xml_nodeset (1)}
[1] <span itemprop="ratingValue">9.3</span>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Rating Counts

```
# Rating counts
tmp_count <- tmp_content %>% html_nodes('span.small') %>% html_text
tmp_count <- gsub(",", "", tmp_count)
tmp_count <- as.numeric(tmp_count)
```



The Shawshank Redemption (1994)

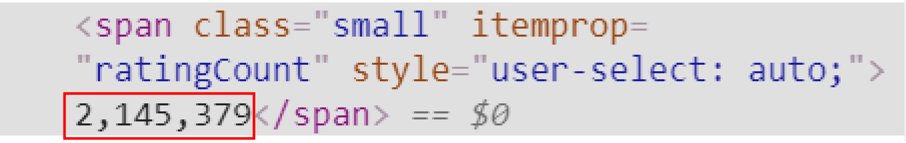
9.3/10 2,145,379 Rate This

15 | 2h 22min | Drama | 28 January 1995 (South Korea)

FEAR CAN HOLD YOU PRISONER. HOPE CAN SET YOU FREE.
TIM ROBBINS MORGAN FREEMAN
THE SHAWSHANK REDEMPTION
2:11 | Trailer
4 VIDEOS | 113 IMAGES

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont
Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)
Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »

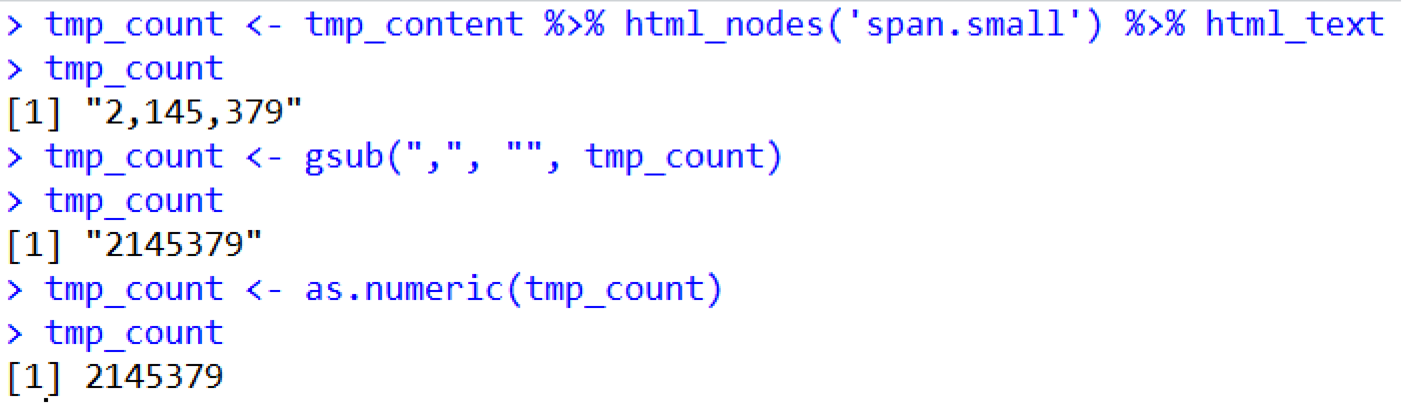


```
<span class="small" itemprop=
"ratingCount" style="user-select: auto;">
2,145,379</span> == $0
```

Web Scraping: IMDB Top 50 Movie Reviews

- Rating Counts

```r
# Rating counts
tmp_count <- tmp_content %>% html_nodes('span.small') %>% html_text
tmp_count <- gsub(",", "", tmp_count)
tmp_count <- as.numeric(tmp_count)
```

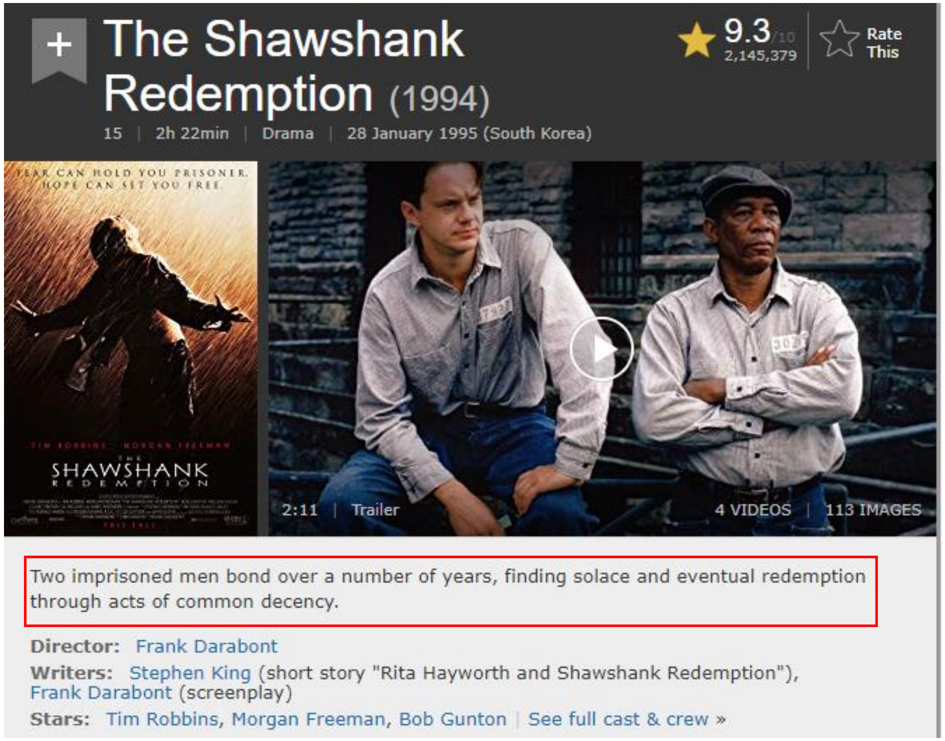


```
> tmp_count <- tmp_content %>% html_nodes('span.small') %>% html_text
> tmp_count
[1] "2,145,379"
> tmp_count <- gsub(",", "", tmp_count)
> tmp_count
[1] "2145379"
> tmp_count <- as.numeric(tmp_count)
> tmp_count
[1] 2145379
```

Web Scraping: IMDB Top 50 Movie Reviews

- Summary

```
# Summary tmp_summary <- tmp_content %>% html_nodes('div.summary_text') %>%
                                    html_text %>% str_trim
```



The Shawshank Redemption (1994)

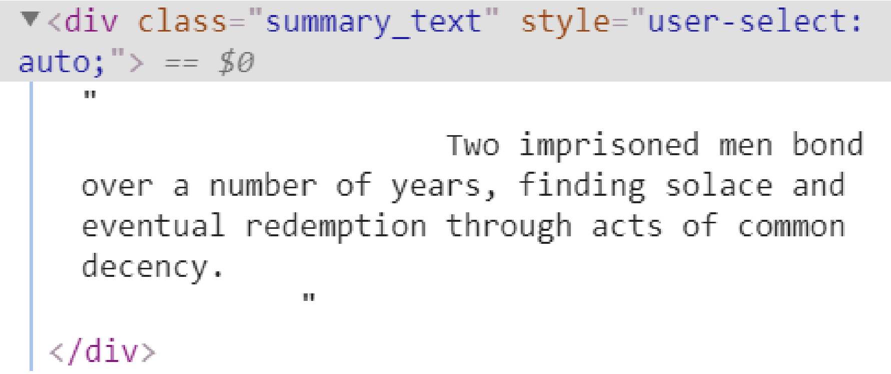
15 | 2h 22min | Drama | 28 January 1995 (South Korea)

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

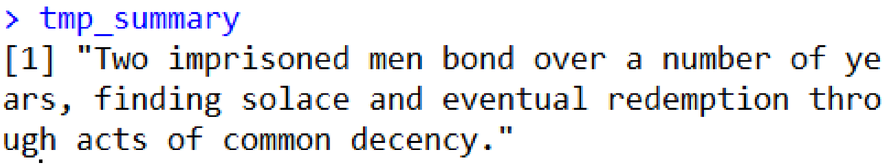
Director: Frank Darabont

Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)

Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »



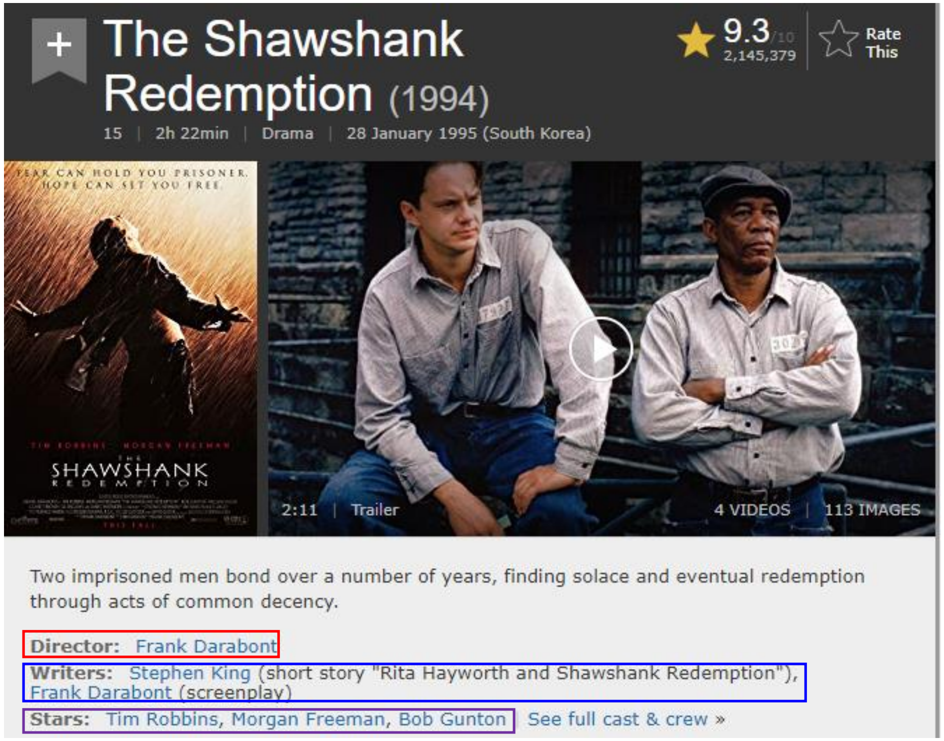
```
▼<div class="summary_text" style="user-select:
auto;"> == $0
    "
                        Two imprisoned men bond
    over a number of years, finding solace and
    eventual redemption through acts of common
    decency.
                    "
  </div>
```



```
> tmp_summary
[1] "Two imprisoned men bond over a number of ye
ars, finding solace and eventual redemption thro
ugh acts of common decency."
```

Web Scraping: IMDB Top 50 Movie Reviews

- Director, Writers, and Stars



The Shawshank Redemption (1994)

9.3/10 2,145,379

Rate This

15 | 2h 22min | Drama | 28 January 1995 (South Korea)

FEAR CAN HOLD YOU PRISONER. HOPE CAN SET YOU FREE.
TIM ROBBINS MORGAN FREEMAN
THE SHAWSHANK REDEMPTION

2:11 | Trailer

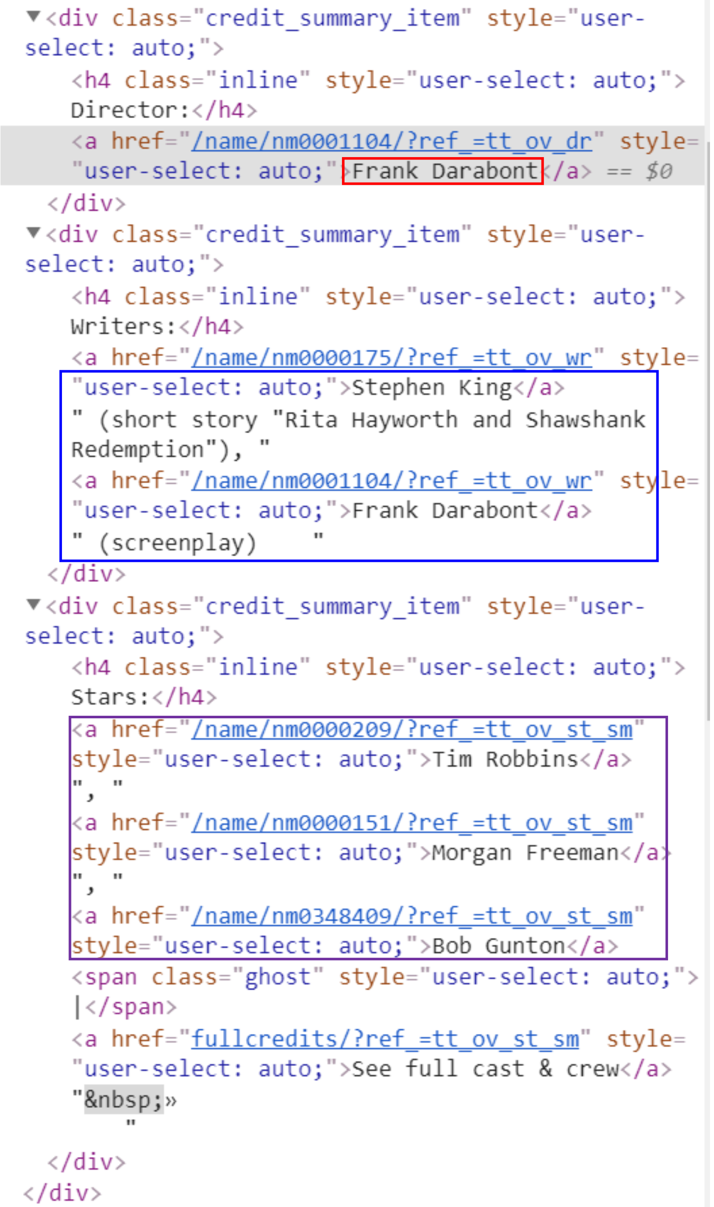
4 VIDEOS | 113 IMAGES

Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Director: Frank Darabont

Writers: Stephen King (short story "Rita Hayworth and Shawshank Redemption"), Frank Darabont (screenplay)

Stars: Tim Robbins, Morgan Freeman, Bob Gunton | See full cast & crew »

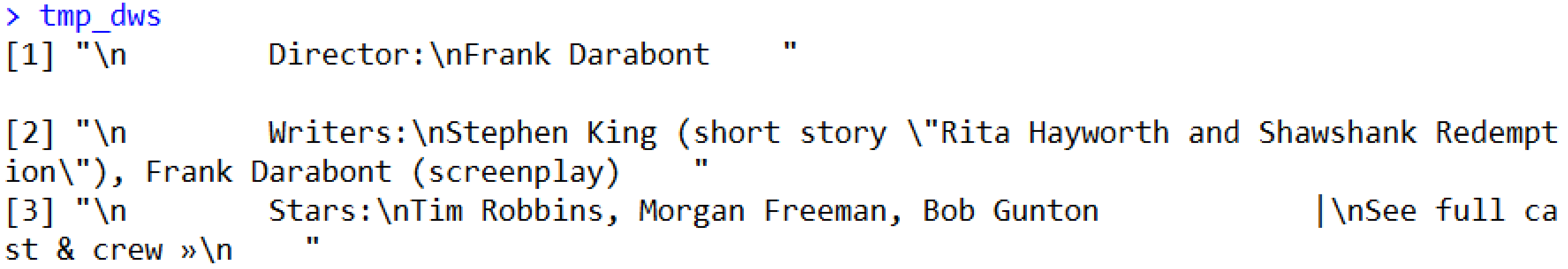


```
▼<div class="credit_summary_item" style="user-
select: auto;">
    <h4 class="inline" style="user-select: auto;">
    Director:</h4>
    <a href="/name/nm0001104/?ref_=tt_ov_dr" style=
    "user-select: auto;">Frank Darabont</a> == $0
  </div>
▼<div class="credit_summary_item" style="user-
select: auto;">
    <h4 class="inline" style="user-select: auto;">
    Writers:</h4>
    <a href="/name/nm0000175/?ref_=tt_ov_wr" style=
    "user-select: auto;">Stephen King</a>
    " (short story "Rita Hayworth and Shawshank
    Redemption"), "
    <a href="/name/nm0001104/?ref_=tt_ov_wr" style=
    "user-select: auto;">Frank Darabont</a>
    " (screenplay)    "
  </div>
▼<div class="credit_summary_item" style="user-
select: auto;">
    <h4 class="inline" style="user-select: auto;">
    Stars:</h4>
    <a href="/name/nm0000209/?ref_=tt_ov_st_sm"
    style="user-select: auto;">Tim Robbins</a>
    ", "
    <a href="/name/nm0000151/?ref_=tt_ov_st_sm"
    style="user-select: auto;">Morgan Freeman</a>
    ", "
    <a href="/name/nm0348409/?ref_=tt_ov_st_sm"
    style="user-select: auto;">Bob Gunton</a>
    <span class="ghost" style="user-select: auto;">
    |</span>
    <a href="fullcredits/?ref_=tt_ov_st_sm" style=
    "user-select: auto;">See full cast & crew</a>
    "&nbsp;»
        "
  </div>
</div>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Director, Writers, and Stars

```r
tmp_dws <- tmp_content %>% html_nodes('div.credit_summary_item') %>% html_text
tmp_director <- tmp_dws[1] %>% str_trim
tmp_director <- sub("Director:\n", "", tmp_director)

tmp_writer <- tmp_dws[2] %>% str_trim
tmp_writer <- sub("Writers:\n", "", tmp_writer)

tmp_stars <- tmp_dws[3] %>% str_trim
tmp_stars <- strsplit(tmp_stars, "\nSee")[[1]][1]
tmp_stars <- sub("Stars:\n", "", tmp_stars)
tmp_stars <- substr(tmp_stars, 1, nchar(tmp_stars)-1) %>% str_trim
```

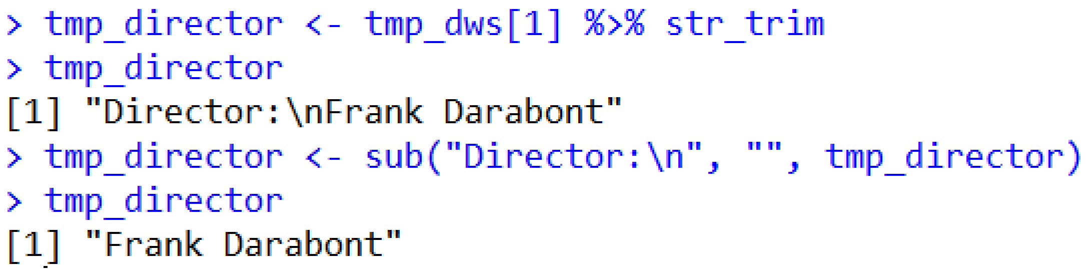


```
> tmp_dws
[1] "\n        Director:\nFrank Darabont    "

[2] "\n        Writers:\nStephen King (short story \"Rita Hayworth and Shawshank Redempt
ion\"), Frank Darabont (screenplay)    "
[3] "\n        Stars:\nTim Robbins, Morgan Freeman, Bob Gunton           |\nSee full ca
st & crew »\n    "
```

Web Scraping: IMDB Top 50 Movie Reviews

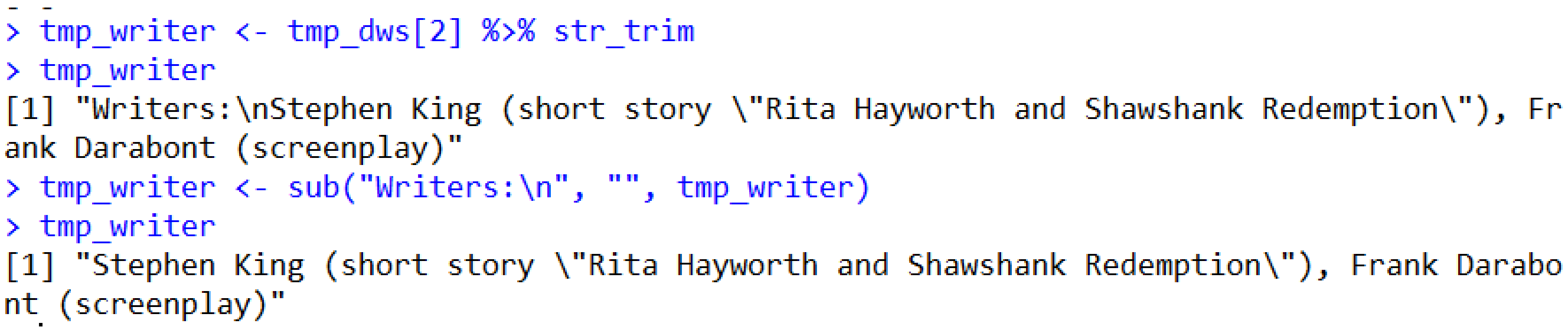
- Director, Writers, and Stars

✓ Director



```
> tmp_director <- tmp_dws[1] %>% str_trim
> tmp_director
[1] "Director:\nFrank Darabont"
> tmp_director <- sub("Director:\n", "", tmp_director)
> tmp_director
[1] "Frank Darabont"
```

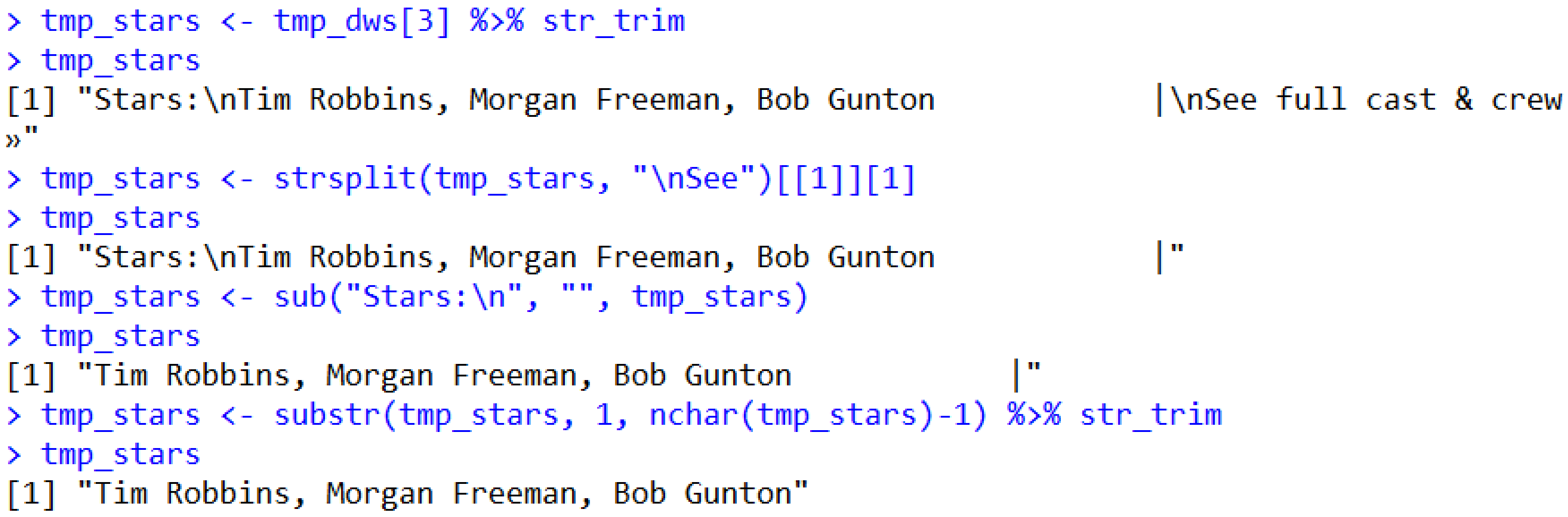
✓ Writers



```
> tmp_writer <- tmp_dws[2] %>% str_trim
> tmp_writer
[1] "Writers:\nStephen King (short story \"Rita Hayworth and Shawshank Redemption\"), Fr
ank Darabont (screenplay)"
> tmp_writer <- sub("Writers:\n", "", tmp_writer)
> tmp_writer
[1] "Stephen King (short story \"Rita Hayworth and Shawshank Redemption\"), Frank Darabo
nt (screenplay)"
```

Web Scraping: IMDB Top 50 Movie Reviews

- Director, Writers, and Stars
  - ✓ Stars

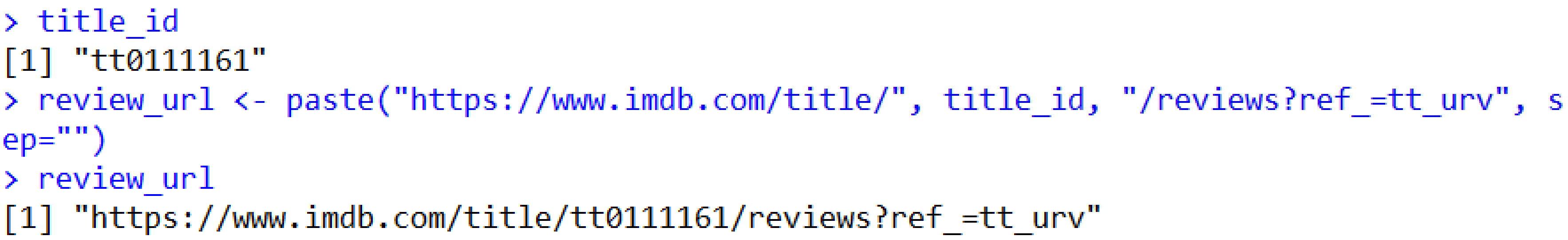


```
> tmp_stars <- tmp_dws[3] %>% str_trim
> tmp_stars
[1] "Stars:\nTim Robbins, Morgan Freeman, Bob Gunton            |\nSee full cast & crew
»"
> tmp_stars <- strsplit(tmp_stars, "\nSee")[[1]][1]
> tmp_stars
[1] "Stars:\nTim Robbins, Morgan Freeman, Bob Gunton            |"
> tmp_stars <- sub("Stars:\n", "", tmp_stars)
> tmp_stars
[1] "Tim Robbins, Morgan Freeman, Bob Gunton            |"
> tmp_stars <- substr(tmp_stars, 1, nchar(tmp_stars)-1) %>% str_trim
> tmp_stars
[1] "Tim Robbins, Morgan Freeman, Bob Gunton"
```

Web Scraping: IMDB Top 50 Movie Reviews

- Scrap the First 25 Reviews for Each Movie

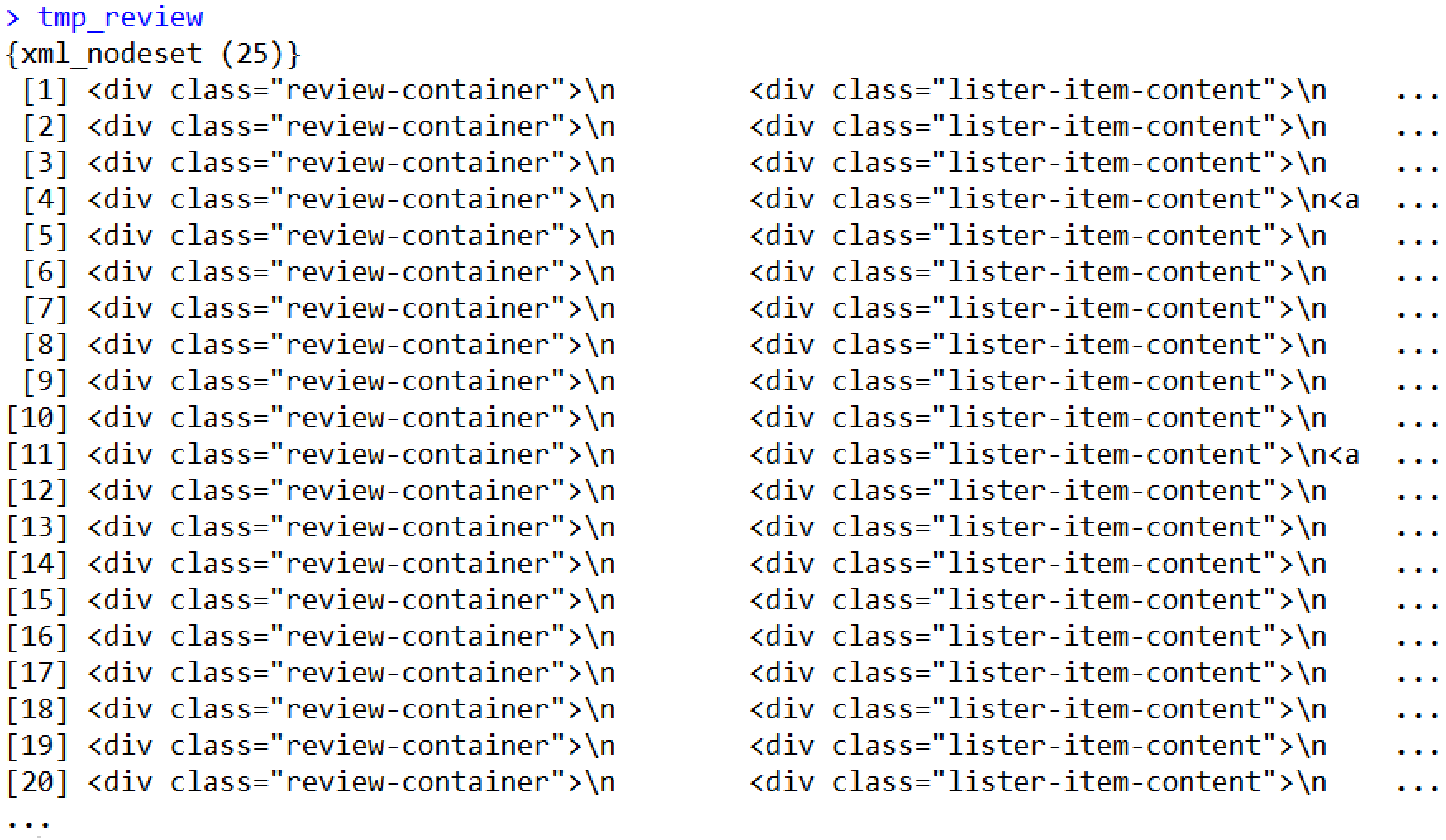
```r
# Extract the first 25 reviews
title_id <- strsplit(tmp_list[i], "/")[[1]][3]
review_url <- paste("https://www.imdb.com/title/", title_id, "/reviews?ref_=tt_urv", sep="")

tmp_review <- read_html(review_url) %>% html_nodes('div.review-container')
```



```
> title_id
[1] "tt0111161"
> review_url <- paste("https://www.imdb.com/title/", title_id, "/reviews?ref_=tt_urv", s
ep="")
> review_url
[1] "https://www.imdb.com/title/tt0111161/reviews?ref_=tt_urv"
```

Web Scraping: IMDB Top 50 Movie Reviews

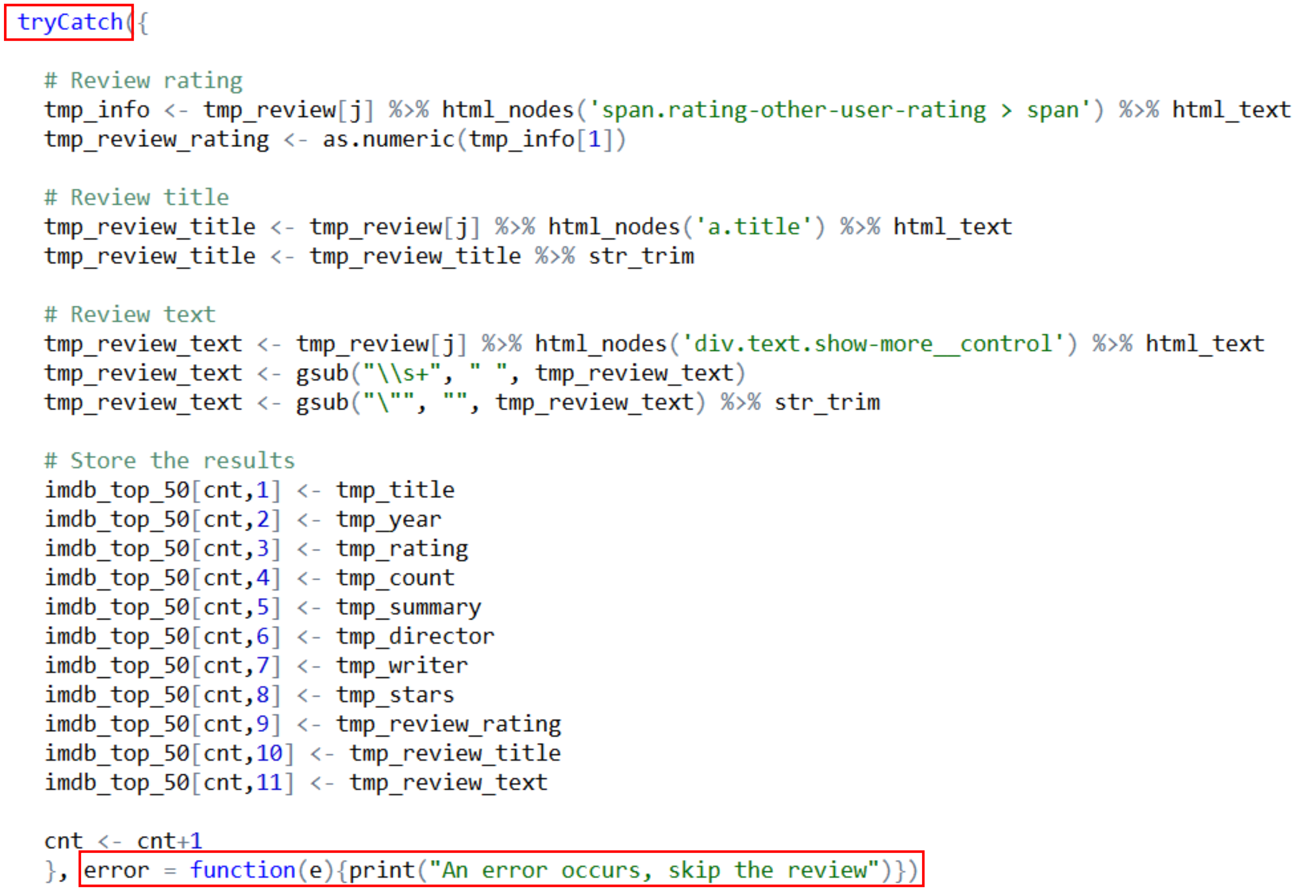
- Scrap the First 25 Reviews for Each Movie



```
> tmp_review
{xml_nodeset (25)}
 [1] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [2] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [3] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [4] <div class="review-container">\n        <div class="lister-item-content">\n<a  ...
 [5] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [6] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [7] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [8] <div class="review-container">\n        <div class="lister-item-content">\n    ...
 [9] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[10] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[11] <div class="review-container">\n        <div class="lister-item-content">\n<a  ...
[12] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[13] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[14] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[15] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[16] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[17] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[18] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[19] <div class="review-container">\n        <div class="lister-item-content">\n    ...
[20] <div class="review-container">\n        <div class="lister-item-content">\n    ...
...
```

Web Scraping: IMDB Top 50 Movie Reviews

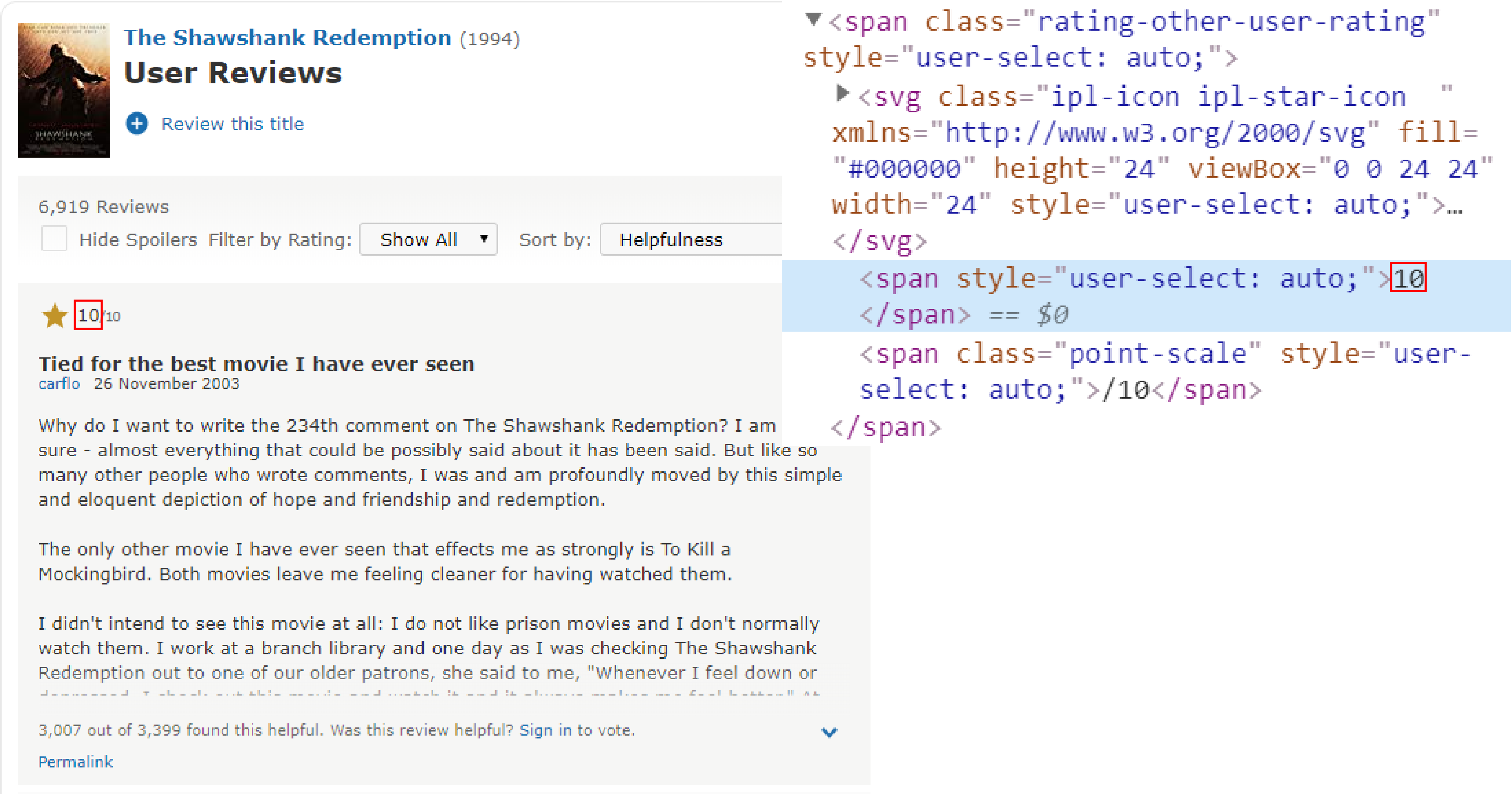
- (Note): To skip unexpected errors, use tryCatch function
  - ✓ Do the instruction inside the tryCatch
  - ✓ If there is an error, store NULL to the title



```
tryCatch({

  # Review rating
  tmp_info <- tmp_review[j] %>% html_nodes('span.rating-other-user-rating > span') %>% html_text
  tmp_review_rating <- as.numeric(tmp_info[1])

  # Review title
  tmp_review_title <- tmp_review[j] %>% html_nodes('a.title') %>% html_text
  tmp_review_title <- tmp_review_title %>% str_trim

  # Review text
  tmp_review_text <- tmp_review[j] %>% html_nodes('div.text.show-more__control') %>% html_text
  tmp_review_text <- gsub("\\s+", " ", tmp_review_text)
  tmp_review_text <- gsub("\"", "", tmp_review_text) %>% str_trim

  # Store the results
  imdb_top_50[cnt,1] <- tmp_title
  imdb_top_50[cnt,2] <- tmp_year
  imdb_top_50[cnt,3] <- tmp_rating
  imdb_top_50[cnt,4] <- tmp_count
  imdb_top_50[cnt,5] <- tmp_summary
  imdb_top_50[cnt,6] <- tmp_director
  imdb_top_50[cnt,7] <- tmp_writer
  imdb_top_50[cnt,8] <- tmp_stars
  imdb_top_50[cnt,9] <- tmp_review_rating
  imdb_top_50[cnt,10] <- tmp_review_title
  imdb_top_50[cnt,11] <- tmp_review_text

  cnt <- cnt+1
  }, error = function(e){print("An error occurs, skip the review")})
```

Web Scraping: IMDB Top 50 Movie Reviews

- Review Rating



The Shawshank Redemption (1994)

User Reviews

⊕ Review this title

6,919 Reviews

□ Hide Spoilers Filter by Rating: Show All ▾ Sort by: Helpfulness

★ 10/10

Tied for the best movie I have ever seen

carflo 26 November 2003

Why do I want to write the 234th comment on The Shawshank Redemption? I am sure - almost everything that could be possibly said about it has been said. But like so many other people who wrote comments, I was and am profoundly moved by this simple and eloquent depiction of hope and friendship and redemption.

The only other movie I have ever seen that effects me as strongly is To Kill a Mockingbird. Both movies leave me feeling cleaner for having watched them.

I didn't intend to see this movie at all: I do not like prison movies and I don't normally watch them. I work at a branch library and one day as I was checking The Shawshank Redemption out to one of our older patrons, she said to me, "Whenever I feel down or

3,007 out of 3,399 found this helpful. Was this review helpful? Sign in to vote.

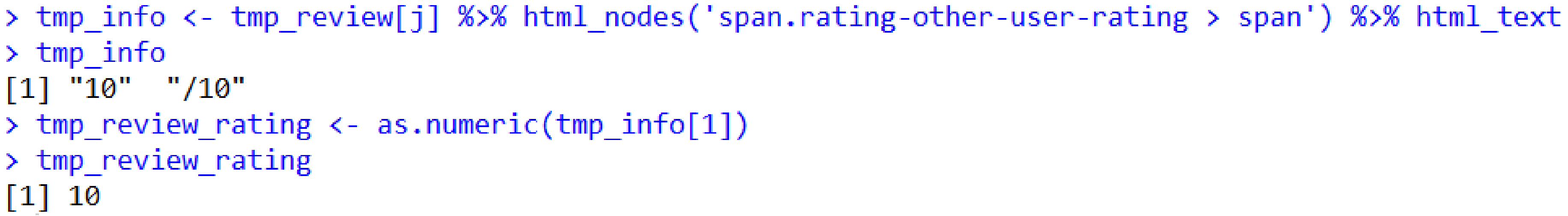
Permalink

```
▼<span class="rating-other-user-rating" style="user-select: auto;">
  ▶<svg class="ipl-icon ipl-star-icon  " xmlns="http://www.w3.org/2000/svg" fill="#000000" height="24" viewBox="0 0 24 24" width="24" style="user-select: auto;">…</svg>
    <span style="user-select: auto;">10</span> == $0
    <span class="point-scale" style="user-select: auto;">/10</span>
  </span>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Review Rating

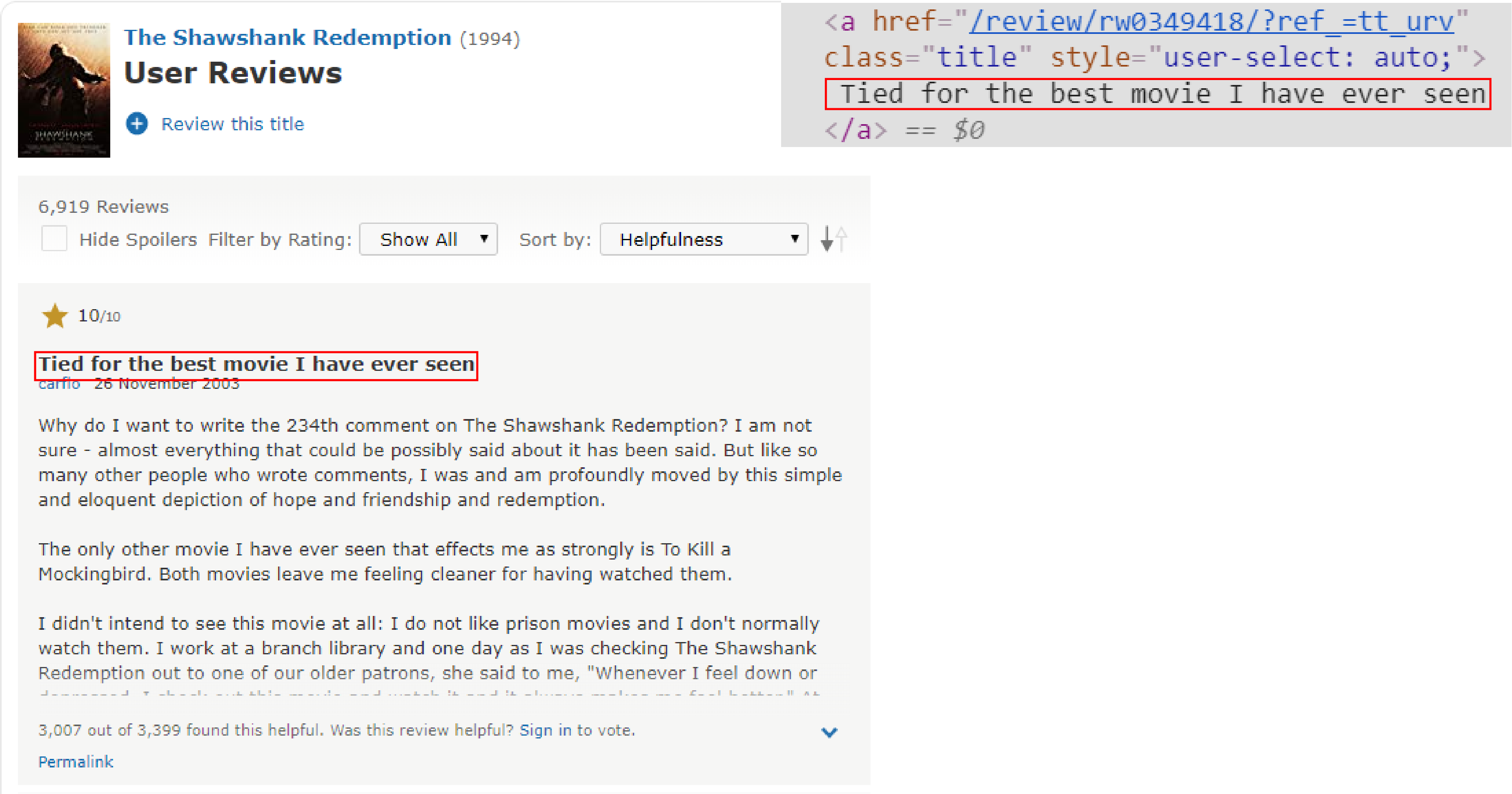
```
# Review rating
tmp_info <- tmp_review[j] %>% html_nodes('span.rating-other-user-rating > span') %>%
               html_text
tmp_review_rating <- as.numeric(tmp_info[1])
```



```
> tmp_info <- tmp_review[j] %>% html_nodes('span.rating-other-user-rating > span') %>% html_text
> tmp_info
[1] "10"  "/10"
> tmp_review_rating <- as.numeric(tmp_info[1])
> tmp_review_rating
[1] 10
```

Web Scraping: IMDB Top 50 Movie Reviews

- Review Title



```
<a href="/review/rw0349418/?ref_=tt_urv"
class="title" style="user-select: auto;">
 Tied for the best movie I have ever seen
</a> == $0
```

The Shawshank Redemption (1994)

User Reviews

6,919 Reviews

□ Hide Spoilers Filter by Rating: Show All ▾ Sort by: Helpfulness ▾

★ 10/10

Tied for the best movie I have ever seen

carflo 26 November 2003

Why do I want to write the 234th comment on The Shawshank Redemption? I am not sure - almost everything that could be possibly said about it has been said. But like so many other people who wrote comments, I was and am profoundly moved by this simple and eloquent depiction of hope and friendship and redemption.

The only other movie I have ever seen that effects me as strongly is To Kill a Mockingbird. Both movies leave me feeling cleaner for having watched them.

I didn't intend to see this movie at all: I do not like prison movies and I don't normally watch them. I work at a branch library and one day as I was checking The Shawshank Redemption out to one of our older patrons, she said to me, "Whenever I feel down or depressed, I check out this movie and watch it and it always makes me feel better." At

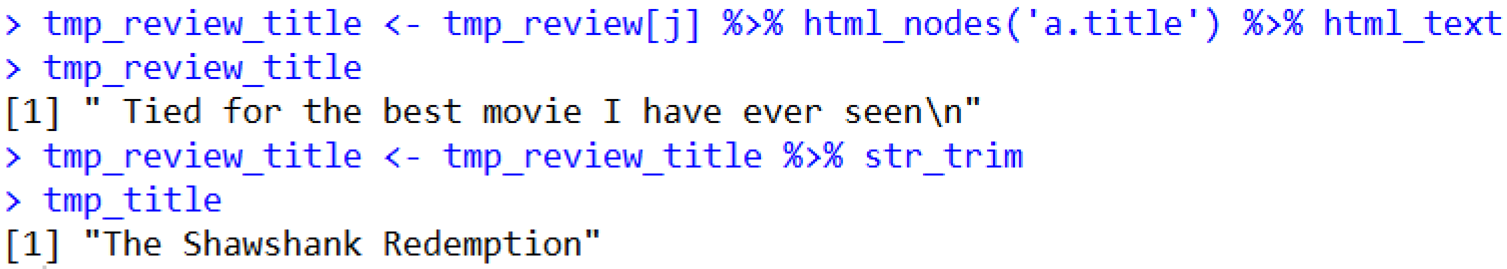
3,007 out of 3,399 found this helpful. Was this review helpful? Sign in to vote.

Permalink

Web Scraping: IMDB Top 50 Movie Reviews

- Review Title

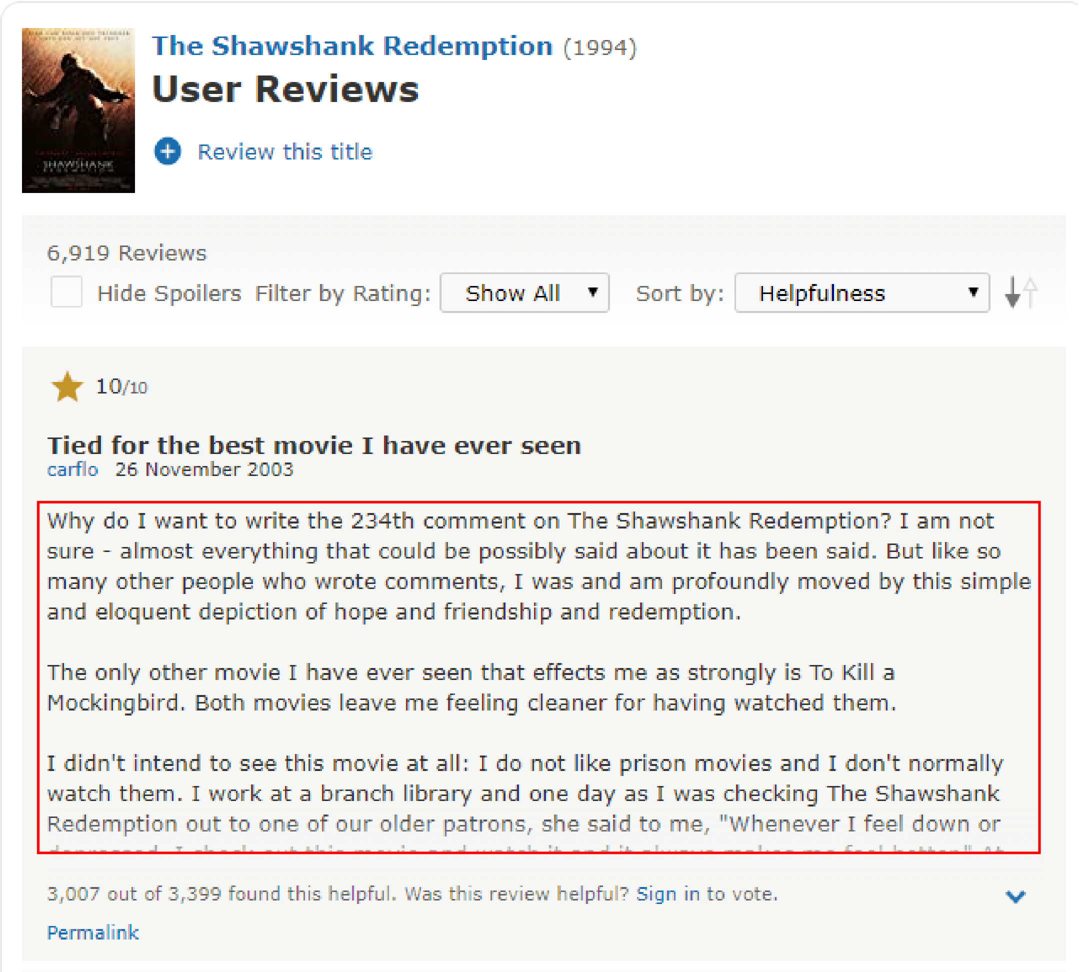
```r
# Review title
tmp_review_title <- tmp_review[j] %>% html_nodes('a.title') %>% html_text
tmp_review_title <- tmp_review_title %>% str_trim
```



```
> tmp_review_title <- tmp_review[j] %>% html_nodes('a.title') %>% html_text
> tmp_review_title
[1] " Tied for the best movie I have ever seen\n"
> tmp_review_title <- tmp_review_title %>% str_trim
> tmp_title
[1] "The Shawshank Redemption"
```

Web Scraping: IMDB Top 50 Movie Reviews

- Review Text



6,919 Reviews

□ Hide Spoilers Filter by Rating: Show All ▾ Sort by: Helpfulness ▾

★ 10/10

Tied for the best movie I have ever seen

carflo 26 November 2003

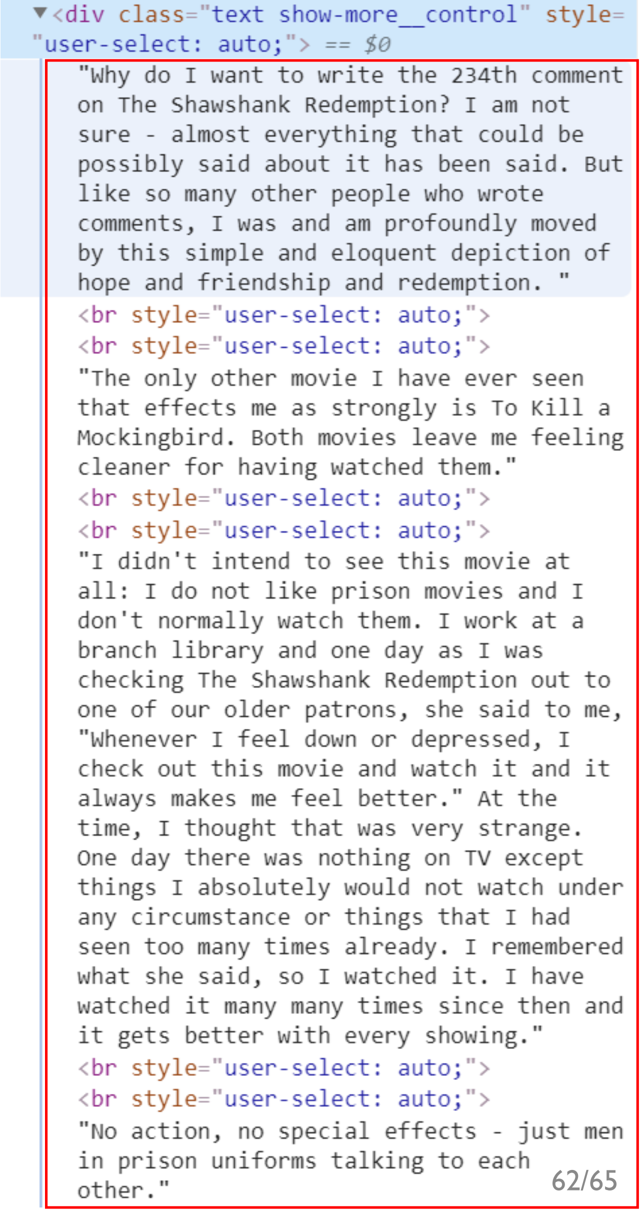
Why do I want to write the 234th comment on The Shawshank Redemption? I am not sure - almost everything that could be possibly said about it has been said. But like so many other people who wrote comments, I was and am profoundly moved by this simple and eloquent depiction of hope and friendship and redemption.

The only other movie I have ever seen that effects me as strongly is To Kill a Mockingbird. Both movies leave me feeling cleaner for having watched them.

I didn't intend to see this movie at all: I do not like prison movies and I don't normally watch them. I work at a branch library and one day as I was checking The Shawshank Redemption out to one of our older patrons, she said to me, "Whenever I feel down or depressed, I check out this movie and watch it and it always makes me feel better." At

3,007 out of 3,399 found this helpful. Was this review helpful? Sign in to vote.

Permalink

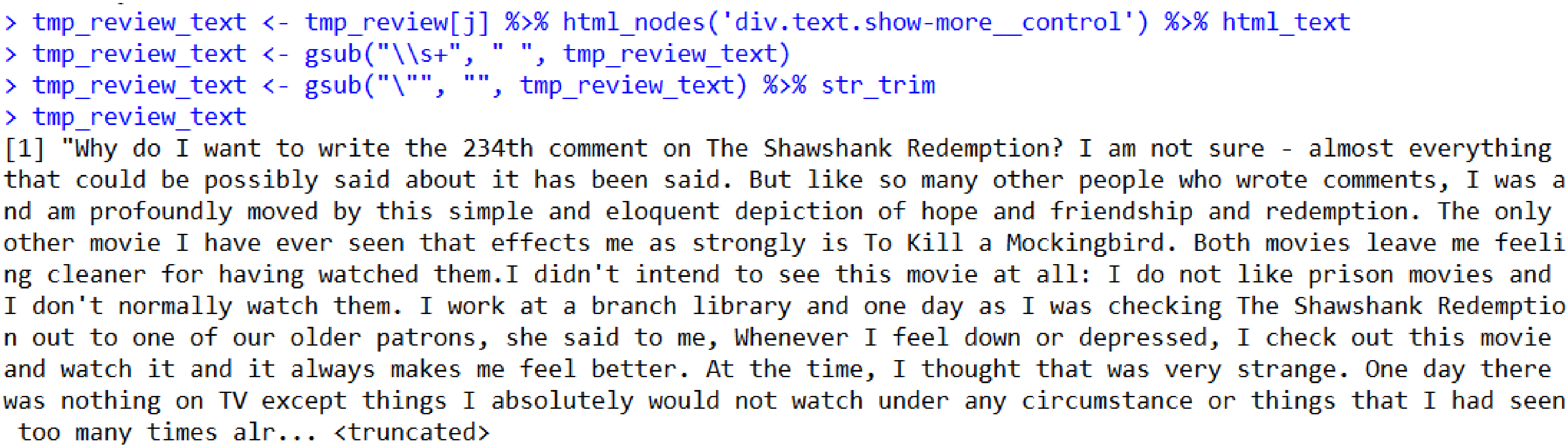


```
▼<div class="text show-more__control" style=
 "user-select: auto;"> == $0
    "Why do I want to write the 234th comment
    on The Shawshank Redemption? I am not
    sure - almost everything that could be
    possibly said about it has been said. But
    like so many other people who wrote
    comments, I was and am profoundly moved
    by this simple and eloquent depiction of
    hope and friendship and redemption. "
    <br style="user-select: auto;">
    <br style="user-select: auto;">
    "The only other movie I have ever seen
    that effects me as strongly is To Kill a
    Mockingbird. Both movies leave me feeling
    cleaner for having watched them."
    <br style="user-select: auto;">
    <br style="user-select: auto;">
    "I didn't intend to see this movie at
    all: I do not like prison movies and I
    don't normally watch them. I work at a
    branch library and one day as I was
    checking The Shawshank Redemption out to
    one of our older patrons, she said to me,
    "Whenever I feel down or depressed, I
    check out this movie and watch it and it
    always makes me feel better." At the
    time, I thought that was very strange.
    One day there was nothing on TV except
    things I absolutely would not watch under
    any circumstance or things that I had
    seen too many times already. I remembered
    what she said, so I watched it. I have
    watched it many many times since then and
    it gets better with every showing."
    <br style="user-select: auto;">
    <br style="user-select: auto;">
    "No action, no special effects - just men
    in prison uniforms talking to each
    other."
```

Web Scraping: IMDB Top 50 Movie Reviews

- Review Text

```
# Review text
tmp_review_text <- tmp_review[j] %>% html_nodes('div.text.show-more__control') %>% html_text
tmp_review_text <- gsub("\\s+", " ", tmp_review_text)
tmp_review_text <- gsub("\"", "", tmp_review_text) %>% str_trim
```



```
> tmp_review_text <- tmp_review[j] %>% html_nodes('div.text.show-more__control') %>% html_text
> tmp_review_text <- gsub("\\s+", " ", tmp_review_text)
> tmp_review_text <- gsub("\"", "", tmp_review_text) %>% str_trim
> tmp_review_text
[1] "Why do I want to write the 234th comment on The Shawshank Redemption? I am not sure - almost everything
that could be possibly said about it has been said. But like so many other people who wrote comments, I was a
nd am profoundly moved by this simple and eloquent depiction of hope and friendship and redemption. The only
other movie I have ever seen that effects me as strongly is To Kill a Mockingbird. Both movies leave me feeli
ng cleaner for having watched them.I didn't intend to see this movie at all: I do not like prison movies and
I don't normally watch them. I work at a branch library and one day as I was checking The Shawshank Redemptio
n out to one of our older patrons, she said to me, Whenever I feel down or depressed, I check out this movie
and watch it and it always makes me feel better. At the time, I thought that was very strange. One day there
was nothing on TV except things I absolutely would not watch under any circumstance or things that I had seen
 too many times alr... <truncated>
```

Web Scraping: IMDB Top 50 Movie Reviews

- Store the Results

```r
# Store the results
imdb_top_50[cnt,1] <- tmp_title
imdb_top_50[cnt,2] <- tmp_year
imdb_top_50[cnt,3] <- tmp_rating
imdb_top_50[cnt,4] <- tmp_count
imdb_top_50[cnt,5] <- tmp_summary
imdb_top_50[cnt,6] <- tmp_director
imdb_top_50[cnt,7] <- tmp_writer
imdb_top_50[cnt,8] <- tmp_stars
imdb_top_50[cnt,9] <- tmp_review_rating
imdb_top_50[cnt,10] <- tmp_review_title
imdb_top_50[cnt,11] <- tmp_review_text
```

Web Scraping: IMDB Top 50 Movie Reviews

- Post-processing
  - ✓ Assign the column names
  - ✓ Store the result as a Rdata and csv file

```r
names(imdb_top_50) <- c("Title", "Year", "Avg.Rating", "RatingCounts", "Summary", "Director",
                    "Writer", "Stars", "Review.Rating", "Review.Title", "Review.Text")
...
# Export the result
save(imdb_top_50, file = "imdb_top_50.RData")
write.csv(imdb_top_50 , file = "imdb_top_50.csv")
```

ANY
questions ?