Calculating the women-friendliness of parliament

Marx, M.

Publication date
2010
Document Version
Final published version
Published in
Driven by data: exploring the research horizon

Calculating the women-friendliness of parliament

Maarten Marx

More and more data are becoming available to social scientists in the form of raw text as material for text analysis. These data are usually easy to find on the internet. However, it is often still a big step to turn all sorts of formats into a nice input file for the popular analytical package SPSS. But suitable tools are available.

A recent article in *Science*, entitled 'Computational Social Science', contains a plea for curricula in which students in the social sciences and humanities learn to use the tools for handling vast amounts of text. We can demonstrate what these tools do by using an example, and showing that using these tools is relatively straightforward. Computers have become powerful and simple enough to enable even information-challenged persons to carry out quantitative research based on huge amounts of text. The required knowledge can be taught to any student in a moderate-size course on text analysis for the social sciences.

A simple research question...

We use an example to illustrate this. Let's address the following research question. In the years 2006-2010 the Dutch parliament's so-called *Tweede Kamer* contained a record-high percentage of women, namely over forty per cent. Are these women merely there for the record or do they get speaking time, proportionally? This question yields various subsequent questions. For example, does this vary by subject? And are there any differences in women(un)friendliness among the parties? How were things in the past?

All the data needed to answer this research question are present. *The Handelingen der Staten Generaal* (the Acts) are public; they are available on the internet for the years since 1917, as PDF files. They contain the exact transcriptions of what everyone has said in the *Tweede Kamer*. In addition, the website *parlement.com* contains extensive biographies of anyone who has ever been part of this Chamber. We could therefore make a tally for each member of the *Tweede Kamer* how many times he or she contributed to a debate, how often a member interrupted, how long a member spoke, and how much speaking time that member used up. Although times are not listed in the Acts, we can approximate them by counting the number of spoken words.

...that cannot be easily answered

So far, nothing about this is difficult or special. And yet, it is not an easy thing to do because the data are not available in the right format. There are three problems. In the first place, it concerns a large amount of data:

3560 biographies and more than one hundred million words, spoken in the *Tweede Kamer* since 1995. Secondly, coupling the two data sets is difficult as the members of parliament are not consistently referred to by the same name. This problem is worse for data from before 1995, which were scanned and still contain errors as a result of optical character recognition (OCR). And finally, the Acts consist of text files in PDF format, with scant metadata. Of every word in the Acts, we do know on which day it was spoken and on which page it is, but not by whom it was said and in what capacity, as Member of Parliament or member of the government, as argument, interruption or response to an interruption.

The technique: Conversion to xml

Once the Acts are rendered machine-readable, techniques for text analysis can solve the problem of the recognizability and professional capacity of the speakers. With the aid of named entity recognition and reconciliation, we can recognize speakers and normalize their names and thus eliminate the thresholds for combining the two data sets. That is also the moment from which we can deploy computers to tackle the large data quantity. We no longer need to work with expensive coders and samples, and can carry out the analysis on the entire data set.

Next, we can take a look at how this conversion works in practice. These days, the Acts are already placed online as an html page the following day (www.tweedekamer.nl). Our example is based on that format. It works similarly for PDF input, but technically, it is a bit more complicated.

Here is an example from the debate of 13 January 2010 concerning the Davids Report (about the decision-making process surrounding the Dutch decision to back the war in Iraq):

```
<p>Minister <strong>Balkenende</
strong>:</p>
<p>Well, what am I doing here?</p>
<p>The <strong>chair</strong>:</p>
<p>Yes, the Prime Minister is still
standing too.</p>
<p>Minister <strong>Balkenende</
strong>:</p>
<p>And it is already a quarter to
three.</p>
<p>Mrs. <strong>Kant</strong>
(SP):</p>
<p> A few hours ago, I would not
have expected you to be still standing
there.</p>
```

Below is the text in a format that is legible to a computer.

```
<speaker name="Balkenende"
MPid="02207" type="Minister">
  <p>Well, what am I doing here?</p>
</speaker>
<speaker name="Ten Hoopen"
MPid="02573" type="Chair">
  <p> Yes, the Prime Minister is still
standing too.</p>
</speaker>
<speaker name="Balkenende"
MPid="02207" type="Minister">
  <p> And it is already a quarter to
three.</p>
</speaker>
<speaker name="Kant"
MPid="02226" party="SP"
type="Member of Parliament">
  <p>A few hours ago, I would not
have expected you to still be standing
there.</p>
</speaker>
```

We briefly explain four differences:

1. The structure now clarifies who says what. All sections with text spoken by a person are nested in a <speaker> element.
2. Only what was actually said is in <p> elements now. This allows us to distinguish between the "Kant" who is speaking and the "Kant" who is mentioned.
3. The names of the speakers are rec-

ognized and linked to their unique number at the Parliamentary Documentation Center. For example, the biography of Balkenende (number 02207) can be seen at http://www.parlement.com/9291000/biof/02207. Also pay attention to the "Chair". In the first version, it is not clear who this is at that time. Although changes of Chair are indicated in the Acts, this may have occurred pages ago.

4. The professional capacity in which speakers contribute is listed in the "type" attribute. The party is only listed for Members of Parliament.

The above consists of digital text and therefore does not contain any OCR errors. However, connecting a name with a unique person is not always simple. At the moment, there are two Members of Parliament with the name J. van Dijk, for example. Also, persons are not consistently indicated with the same name. See figure 1 for an example of a list we extracted from the Acts since 1995. Is that really only one person?

Figure 2 contains a number of party names from the scanned Acts between 1985 and 1995. Does this always refer to the *Partij van de Arbeid* (PvdA)?

Aadsted-Madsen
Aasted Madsen
Aasted-Madsen
Aasted Madsen-Stiphout
Aasted Madsen – van Stiphout
Aasted-Madsen van Stiphout
Aasted Madsen-van Stiphout
Aasted Madsen-Van Stiphout

Figure 1. The Acts contain multiple references to the same person.

Figure 2. OCR inaccuracies led to errors in the abbreviation PvdA.

One application: Summaries of speakers

In this new format, it is a piece of cake to use the Acts and produce all sections with text spoken by Balkenende in a debate. The following XPath query does that for the debate of 13 January 2010 which included the report of the Davids Commission:

```
doc('plenary_meeting_13_january_2010.xml')//
    speaker[@name="Balkenende"]//
    p
```

XPath is a very intuitive language, in which queries are set up as paths through the xml hierarchy. The above XPath query states: 'give the sections (p's) within the speaker element with the name Balkenende and extract them from said file'. We now have everything Balkenende said that day, on all topics covered that day. If we only want the text of what Balkenende said concerning the report of the Davids Commission in that meeting, the query becomes:

```
doc('plenary_meeting_13_january_2010.xml')//
    subject[@subject='Report Commission Davids']//
    speaker[@name="Balkenende"]//
    p
```

Figure 3. Tag cloud containing all the words used by Prime Minister Balkenende in his response to the findings of the Davids Committee on 13 January 2010. Tag cloud created with wordle.net.

It is not very interesting to read everything Balkenende said without having its context. But computers can also summarize texts, for example, as a word cloud as given in figure 3.

Another application: a political iPhone app

Students at the University of Amsterdam created a fun application: an iPhone app in which curious sentences spoken in Parliament fly across the iPhone screen. Having the debates in the new handy format, the developers of the app were able to solve the really difficult challenge: How do you pull interesting and curious sentences out of texts in an automated manner? Figure 4 shows a screenshot of their app. A photo of the speaker is displayed together with the quote. When you click on the quote, you are taken to the moment in the debate when this sentence was spoken.

Merely filling the quota?

Currently, the University of Amsterdam is cooperating with the National library of the Netherlands on making the Acts available in xml format. This

Figure 4. IPhone app showing quotes from parliament made by Felix Cornelissen and Thomas Kuipers. www.poliquote.nl

would facilitate finding the answer to the research question posed at the beginning.
Meanwhile, we have carried out some quick tallies to measure women-friendliness, as a test. If we exclude the Chair, women used 33% of the speaking time in the Tweede Kamer in the period 3 February 2009 up to and including 8 October 2009. That is almost 20% less than you would expect on the basis of the number of female members. Women occupy the speaker's position only 30% of the time. Of course, this can mean all sorts of things. Women may simply be less long-winded than men and more to the point when responding to interruptions.

Conclusion

Enormous investments have been made in language technology tools for the Dutch language. However, for 'simple' social scientists and the like, these tools still are often difficult to apply, certainly when they have to be used in combination with other tools. This is a shame as a plethora of raw data is freely available. The aforementioned Science article describes the real danger that the industry (Google, Amazon, etc.) will snatch the discipline of Computational Social Science out of the hands of science forever. Let's make sure that this won't happen.

Soon, everyone will be able to carry out this investigation on their own as we are placing all Acts in the EASY archive in DANS in a uniform xml format. And we are adding to it every day.

Reference

D. Lazer, A. Pentland, L. Adamic, S. Aral, A.-L. Barabsi, D. Brewer, N. Christakis, N. Contractor, J. Fowler, M. Gutmann, T. Jebara, G. King, M. Macy, D. Roy, and M. V. Alstyne. 'Computational social science.' *Science*, 323(5915):721–723, 2009.