

Using IT to Improve Knowledge about Political-Economic Space

Maarten Jongmans
289471mj@student.eur.nl

Master Thesis

Economics & ICT
Informatics and Economics
Econometrics Institute
Erasmus School of Economics
Erasmus University Rotterdam

Supervisor: Dr. Ir. Flavius Frasincar
Co-supervisor: Dr. Maarten. Marx
Co-reader: Alexander Hogenboom MSc

20th October 2010

Abstract

Nowadays the Internet plays an important role when trying to find information about a wide range of topics. We believe it is important to have an information source on the Internet where you can access information produced in parliament in an easy manner. This can help 'the people' in controlling the government, but also researchers when investigating political-economical topics. IT can help in bridging the gap between the public and the government, fostering the well-functioning of the democracy, using less resources than before.

We propose two IT solutions for the previously identified problem. The first solution is restructuring the current data set (in this case the amendments) to an information centric format making the information inside this data set useable. After restructuring, the set is made available via one channel which eliminates a lot of irrelevant data from other data sets improving the level of easiness to access the information. The second solution is the introduction of a new way of ranking search results (of political publications). We believe these publications (of debates) contain interesting metadata tags (like the attendees) and include text attributes which represent the sentiment of the publication. In this thesis we show how we calculate the importance of these publications by using these tags and attributes. We then use the importance value to introduce a new way of ranking search result; a ranking based on the importance.

Keywords: search, ranking, politics, ETL, publication importance.

Acknowledgements

I would like to thank my thesis supervisor Flavius Frasincar. He supported me during the writing of this thesis. Sometimes sharp comments on my thesis were made, while at the same time having good conversations about politics. I enjoyed our appointments.

I also want to thank Maarten Marx from the University of Amsterdam (UvA) for his ideas, suggestions, the data he provided and his network I could use. His contributions were of paramount importance for this thesis.

– Maarten Jongmans, 20th October 2010

Contents

1	Introduction	1
1.1	Background	1
1.2	Problem	5
1.3	Goal	6
1.4	Methodology	7
1.5	Structure	7
2	Amendments	8
2.1	Introduction	8
2.1.1	The Information Centric approach	10
2.2	Our Framework	10
2.2.1	Extract	11
2.2.2	Transform	12
2.2.3	Load	16
2.3	Evaluation	16
2.3.1	Which members of parliament are the most busy petitioning amendments?	16
2.3.2	Which members of parliament petitioned the most together (per year)?	18
2.3.3	Which month is the most popular to petition an amendment?	19
2.3.4	How many times did each block occur in the amendments?	20

2.3.5	How many amendments were petitioned every (parliamentary) year?	21
2.3.6	What is the highest number of petitioners for 1 amendments?	22
2.3.7	Which member of parlement uses the most blocks per amendment (at least 50 amendments petitioned)?	23
2.3.8	Which amendment has the most blocks?	24
2.3.9	Which petitioner(s) petitioned the amendment with the most words?	24
2.4	The Business Case	26
2.5	Conclusion	27
3	The Importance of a Debate	29
3.1	Introduction	29
3.2	Aspects of importance	30
3.3	Operationalization of importance	31
3.3.1	Key Players category	34
3.3.2	Debate Length category	34
3.3.3	Intensity Category	35
3.3.4	Normalization of attribute values	36
3.4	Different Models	38
3.5	Experiments and Evaluation	38
3.6	Conclusion	47
4	Debates Performance System	49
4.1	DPS Basics	49
4.2	Searching	50
4.3	Ranking	52
5	Conclusion	54
5.1	Conclusion	54
5.2	Future work	56

A Example of an amendment	58
B Amendment Structure	59
C Relax NG validation scheme debates	60
D XQuery Importance Attributes	62

Version 3.0

List of Figures

2.1	Overview of relations between debates, motions and amendments.	9
2.2	Diagram representing the structure of the new amendment format.	14
2.3	Members of parliament petitioning the most amendments	17
2.4	Members of parliament petitioning the most together	19
2.5	Overview of the most popular month to petition an amendment .	20
2.6	Overview of the occurrence of blocks per blocktype	21
2.7	Overview of the amount of amendments per year	22
2.8	Overview of the amount of amendments per year	24
2.9	Overview of petitioners petitioning amendments with the most words	25
3.1	ER diagram of a debate.	33
3.2	ER diagram of the metadata component of a ‘Handelingen’ document (which contains debates).	33
3.3	A visual overview of how important debates are	40
3.4	Overview of the survey for the parliamentary Journalists and experts	41
4.1	DPS Search Form.	50
4.2	DPS Search Result.	51
4.3	DPS Debate Page.	52
4.4	DPS Ranking with timeframe 2009-2010 and parliamentarian Atsma.	53
4.5	DPS Ranking with timeframe 2009-2010.	53
A.1	An amendment	58

B.1 Diagram representing the metadata part of the structure of the
new amendment format. 59

Version 3.0

List of Tables

3.1	Weight Distribution	39
3.2	Distribution of points over debates per returned survey	42
3.3	Model Rankings compared with Survey	43
3.4	Updated Weight Distribution	46
3.5	Updated Model Rankings compared with Survey	47

Version 3.0

Listings

2.1	Excerpt of metadata kamerstuk	11
2.2	Amendments Evaluation - XQuery 1	17
2.5	Showing how many times a block occurred in an amendment. . .	20
2.6	The amount of amendments per year	21
2.7	Maximum amount of petitioners per amendment	22
2.8	Most blocks per amendment	23
2.9	XQuery showing the amendments with the most blocks	24
2.10	XQuery showing overview of petitioners petitioning amendments with the most words	25
C.1	Relax NG validation scheme of a debate	60
D.1	XQuery calculating Importance Attributes	62

Chapter 1

Introduction

Section 1.1 of this chapter introduces how important IT is in order to get an overview of the economic and political playing field and its publications. We outline what technologies are currently used and what the applications are that make use of them in this fields. In section 1.2 we describe what the problem is that we investigated. Section 1.3 outlines our goal, including our research question. The methodology used for this work is discussed in section 1.4. Section 1.5 contains a description of the structure of this thesis.

1.1 Background

*“I disapprove of what you say, but I will defend
to the death your right to say it.” [1]*

The city council elections in the Netherlands on march the 3rd 2010 caused a new wave of media attention for politics. Every media outlet was used to promote the viewpoint of the candidates: newspapers, television, or the Internet. Independent websites popped up following candidates on social networks. For example <http://www.kamertweets.nl> [2] followed politicians on twitter or the 2009 startup polifeeds <http://www.polifeeds.nl> [3] which aggregated most of the political news feeds to one website. Research shows that since the end of the 2000 election campaign the proportions of Americans going online for news related to the politics more than doubled [4]. While in 2000 18% of all adults

consumed political news online this increased to 44% in 2008 [5]. As in the Netherlands the percentage of the population who have access to Internet (in 2009) is 93% [4] we suspect that the role of Internet in the political space can have at least the same (important) role as in the US.

A problem of having such an important role for the Internet in the political space can be that everyone, as in citizens, politicians, political (pressure) groups or qualified journalists can publish anything immediately available for a huge audience. This can result in an increasing amount of online news representing opinions as facts. With such a huge audience politicians will also be more tempted to increase the amount of negative campaigning on their websites (or of their related pressure groups). Research by Klotz shows that negative campaigning has become an accepted part of campaign websites. Extensive negative campaigning (devoting more than 1000 words placing opponents in a bad light) quadrupled between 1996 and 2002 [6]. The problem of representing opinions as facts, or displaying wrong information about politics on websites is not a issue just surrounding the few months in a year that there are elections. We believe that everyone interested in a certain political topic or politician should be able to access official, trustworthy information at any time and at any place. As a country in modern society, respecting its democratic values, we aim for a true e-democracy, especially when the political participation of “the people” is increasing, like in the case of the Netherlands [7].

To tackle the problem of (in)availability of information in the political space we believe it is very important to have an information source on the Internet where you can access information produced in parliament (e.g. meeting notes, voting records etc.) in an easy manner. An information source like this can help in bridging the gap between the public and the government, being the concrete foundation of democracy and a start of e-democracy. Not only can this help in answering questions like ‘How did parliament vote on ‘mortgage interest deduction’?’ but can also be useful in letting the people control the government. This

can be compared with Bentham's panopticon, in which the observer is allowed to observe the observed without the observed being able to tell whether they are being observed [8] [9]. Translated to the situation discussed in this proposal this means that the public can check what politicians did in their current term before they decide who to vote on for the next period. Even if the public does not use this information, politicians should still feel obliged to do what is right and act honest because they can be controlled everyday by everyone.

An example of a website for the disclosure of (British) parliamentary publications is TheyWorkForYou.com. TheyWorkForYou is an initiative by the British organization mySociety [10] which goal is to "give people simple, tangible benefits in the civic and community aspects of their lives". In Britain (1995) the public fought for free electronic access of parliamentary publications [11]. The Campaign For Freedom of Information (CFOI) got a positive answer from the government body regulating access to parliamentary publications stating "Parliament needs to ensure that those subject to its laws have easy access to them and the law making process" [11]. Research shows that the biggest group accessing the parliamentary information is the public (44%), followed by businesses (24%), and the media (10%), which is in contrast with the idea that only journalist or politicians access such information [12]. The combination of video streams, meeting notes, and voting records at TheyWorkForYou give, in our opinion, the user a true feeling of control over parliament.

An example of a system similar like TheyWorkForYou is Polidocs [13]. Polidocs is a "Web Information System for the disclosure of Dutch parliamentary publications" [13]. The problem with other available systems in the Netherlands like Parlando [14] or Staten-Generaal Digitaal [15] is that every system contains its own dataset (they do not have access to all databases containing parliamentary publications) and they lack some basic search (using metadata like people or parties) and representation features (an example is the direct link option). A list of requirements was put forward in 2005, by the political oriented weblog

Sargasso [16]. The list specified 15 requirements that a new system should meet. When the first version of Polidocs was released it almost met all of the requirements of the Sargasso list [17] similar to TheyWorkForYou with the exception of showing the voting records per member of parliament .

Polidocs consists out of several steps starting with the aggregation of data and concluding with the proper presentation of the information. The first step, aggregating data using eager aggregation, is pulling data from the three different sources (SGD, tweede-kamer.nl and Parlando) that publish parliamentary information in The Netherlands. These sources contain debates in parliament from 1900 to now, amendments and motions all in the pdf format. They do not contain any metadata describing these files. The next step Polidocs performs is processing the raw data into an XML file using XSLT. This format contains annotations [18] so that it is easy to access the information inside the files. Finally the data is stored in a new (XML) repository. One of the criticized aspects of current available systems was the way a user could search in the available data. Polidocs used various ways to solve this problem. The first improvement was better data structuring so that more information was available in the data. An example of this is that in the Polidocs files you can (easily) find the structure of a debate. Who is the first speaker, who reacts on the first speaker and so on. This is all done by using the XML format. The second improvement was the use of faceted search. Faceted search is a combination of navigational and direct search to “leverage the best of both approaches” [19]. Using faceted search a user can reformulate the used search query. This is especially useful when a user has limited knowledge of the search domain and want to broaden his/her understanding of the topic [20]. Polidocs is using a taxonomy [21] to provide faceted search. This taxonomy is used to link documents. A different method is the auto-discovery of facets using hyponyms . In both methods different tricks can be used to broaden the search space. Examples are the use of spelling variation (parliament/parliament), coverage of unknown WordNet terms (if they do not exist they can be new and important) or compound splitting (football can

be foot split up as foot and ball) [22]. An interesting field of research related to the two systems of TheyWorkForYou and Polidocs is using user ratings (users can rate a search result as more important and by that increase the value of that result) to design a personalized faceted search interface [23].

Although systems like Polidocs and TheyForWorkYou are better compared to the systems made available by the government, there are still ways to improve these systems. Several repositories of data are undiscovered making them almost unavailable for the public. Amendments in the Dutch parliament are for example not easily accessible. Apart from the unavailability, the relevance of search results can be improved by improving search (make use of data structures inside search objects) and ranking methods. We think this can result in a business case which shows that huge cost savings can be made, and share some ideas related to this business case. Costs are not the only benefit which can be achieved. The improvement of information availability in parliament is something which can be of importance when doing studies about certain (economical) topics.

In this thesis we propose two improvements based on current technologies. The first improvement is increasing the number of amendments available to search in. This will require us to transform pdf based documents containing no structure to well formed structured xml documents. The second improvement is related to ranking debates in search engines. While the current search system Parlendo is using a ‘word and time based relevancy’ ranking we propose a search ranking based on the importance of a debate. The details about this improvement can be found in chapter 3 while the improvements related to amendments will be discussed in chapter 2.

1.2 Problem

Recordings of parliamentary debates exist since ages. For example the “Handelingen van de Staten-Generaal”, which are the recordings for the Dutch House (Tweede-Kamer) and Senate (Eerste-Kamer) exist since 1815. This is because

in the 1815 constitution [24] a paragraph was included stating that from then on all meetings from the house and senate were public. These recordings and the documents representing amendments and motions petitioned in the house and senate are hard to search in. That is why since 1995 these documents are created digitally and that older documents are digitalized [15]. The recordings of the debates in house and senate alone will consume 30TB in data.

The 30TB consist of PDF files representing the documents. The problem with these PDF formatted documents is that they are not easy to search in and often do not contain metadata like attendees or speakers. In these huge amounts of data you can experience troubles to find what you need, especially when you don't have a document number but just a vague idea about your interest.

1.3 Goal

The goal of this master thesis is to propose solutions on how to use IT to improve the knowledge of the political-economic space. Our main research question is:

“How can IT improve the Knowledge about Politico-Economic Spaces?”

To answer this research question we defined the following sub-questions:

- How can IT help in discovering new information in parliament?
- How can amendments be aggregated and transformed to discover hidden information?
- Can the ‘importance’ of a debate help in creating better and more relevant search results?
- Can IT increase information availability by creating a better overview of information about (political) topics?

The individual sub-questions belong either to chapter 2 (about Amendments) or 3 (about ranking debates) and their answers will be discussed in these chapters.

1.4 Methodology

To help us answer the (sub)research question(s) presented in section 1.3 we will use various research methodologies. The first methodology that we use is a literature study in which we investigate current approaches on analyzing the political-economic space using IT. We will also research what the important aspects of the political-economic space are so that we can think of new ways of improving the knowledge about these spaces.

We will evaluate our ideas and research by implementing them in a new framework improving the knowledge of the political-economical space and perform a survey under several political experts in the Netherlands. We prefer journalists from a wide range of Dutch national media because of their knowledge of the political area.

1.5 Structure

The remainder of this thesis is organized as follows. In chapter 2 we will introduce the amendment: a change to a bill. We will describe what an amendment is and what the problem is with how the digital copies of amendments are currently organized and why it is so important to do this differently. We will introduce a new information centric structure for these digital copies improving the quality of the content and increasing their ‘findability’. Building on chapter 2, chapter 3 will introduce new ideas on how to find a debate especially when you do not know what you are exactly searching for. This relates to the ranking of search results. We will introduce a new ranking method, the importance ranking. Chapter 4 will include a description of the Web Information System which contains the description of the implementation of the ideas discussed in chapter 3. In chapter 5 our conclusions and suggestions for future research are presented.

Chapter 2

Amendments

The first section of this chapter will introduce amendments and the problems surrounding the availability of these amendments. We will discuss a document and information centric approach. Section 2.2 will show how we used IT to improve the knowledge on amendments which resulted in section 2.3 showing the results of the evaluation of the software. In section 2.5 we will draw conclusions based on the earlier sections in this chapter.

2.1 Introduction

In the Dutch parliament debates are held, motions are petitioned and amendments are brought in to vote on. We define an amendment as:

“Voorstel van één of meer Kamerleden om een wetsvoorstel te wijzigen.” (a statement of one or more members of parliament that is added to or revises or improves a (law) proposal) [25].

The amendment is published by the Dutch parliament (the ‘Tweede-Kamer’) and is part of the collection of ‘Kamerstukken’ (Parliamentary Recordings). The amendment is discussed and voted on in the Second (The House) or First Chambre (The Senate) of the Dutch Parliament. An overview of the relation between an amendment, a motion and a debate is shown in Figure 2.1. We find that amendments are not as easily available as they should be. As discussed in

chapter 1 we find it important to be able to see what politicians did in parliament, when they did it and with whom they did. Especially an amendment, which can change a *law proposal* should be publicly available in an easy manner (a motion for example can only evoke or call for new policy). An example of an amendment as published by the Dutch parliament in the PDF format can be found in Appendix A.

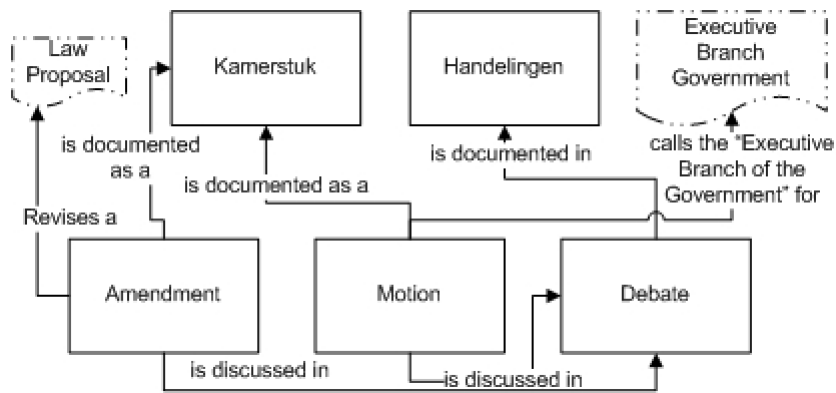


Figure 2.1: Overview of relations between debates, motions and amendments.

We assume that improving the availability of amendments will result in a better (more complete) overview when doing research on topics requiring documents created by politicians. We understand that a few decades ago when (electronic) processing power was very expensive, or even a century ago when there were no computers at all, it was (very) hard to deliver information about amendments especially when you did not know exactly what you were looking for. And this is exactly where we think that IT can help in discovering and/or displaying new information. We believe that displaying the amendments at one location and changing the way that they are handled from a document centric approach to a more information centric approach makes them better accessible and easier to research on.

2.1.1 The Information Centric approach

The amount of publications every year of the parliament is impressive. Even a small specific part of these publications, the amendments, get over hundreds to a thousand a year. Currently a document centric approach exist in organizing these parliamentary publications. This means that the documents do not contain a structure that you can use to extract information. In case of the amendments this means that you only have the text inside the file format currently used (which is PDF, since April 2010 also in XML, for more information see the next section).

An information centric approach on the other hand is an approach in which inside a document information is stored. This can sound a little bit theoretical, but what is meant with this is that inside the document a structure is applied to the data which makes it easier to retrieve information. An example of a question that is hard to answer in a document centric approach is for example “Which members of parliament petition amendments together which contain more than 3 different steps?” (see Appendix A for what a step is). This is currently impossible. Another question can be “How many amendments were petitioned before 2004 for every year?”. This is impossible or costs at least a lot of time. That is why we introduce an information centric approach which can help in answering these questions in just a few seconds. Apart from being able to answer just these two questions you can also save huge amounts of time because you can use any query on the repository of documents and get an answer. You don not need any people looking through folders. In the next section we propose a framework for amendments in the Dutch parliament applying an information centric approach.

2.2 Our Framework

To transform amendments petitioned in the Dutch parliament from a document centric approach to a better information centric approach we developed a framework to do this. This framework is based on the Extract, Transform,

Load (ETL) mechanism [26].

2.2.1 Extract

Currently there is no database available which can supply all amendments between 1995 and for example 2000. This kind of database does exist for some other publications. That is why we first need to extract the amendments from the current system(s) in order to process these amendments.

We used the mirror of the Parlando website [14] created by PoliticalMashup project at the University of Amsterdam (UvA). This consists of all PDF files available at Parlando together with a connected set of metadata: each PDF contains metadata. For an example, see Listing 2.1.

Listing 2.1: Excerpt of metadata kamerstuk

```
1 <meta>
  <dc:contributor>http://www.politicalmashup.nl</dc:contributor>
3 <dc:coverage> <country dcterms:ISO3166="NL">Netherlands</
  country>
  </dc:coverage>
5 <dc:creator>http://www.politicalmashup.nl</dc:creator>
  <dc:date>1994/1995</dc:date> <dc:description>Kamerstuk
  1994-1995,
7 23646, nr. 38, Tweede Kamer</dc:description> <dc:source>
  <pm:textsource>http://cdn.ikregeer.nl/pdf/KST7417.pdf</
  pm:textsource>
9 <pm:metasource>http://polidocs.nl/meta/KST7417.meta.xml</
  pm:metasource>
  </dc:source> <dc:title>Bepalingen inzake de arbeids- en
  rusttijden
11 (Arbeidstijdenwet); Gewijzigd amendement over vrijstelling van
  nachtdienst voor zwangeren</dc:title> <dc:type>Parliamentary
13 Documents</dc:type> </meta>
```

Amendments are published as Kamerstukken. There are 96287 PDF documents of this type in our corpus and the corpus contains data about the years 1995 to 2010. The reason that we did not use older documents is because these documents are structured in a different way. This is because before 1995 documents were never saved digitally. We do not see any problem in applying the method proposed in this paper to these earlier years when adapting the transformation phase to a format which can process scanned documents.

We created a classifier to extract the amendments out of the ‘Kamerstukken’

corpus. This classifier was created in the Java programming language. We used the SAXON XSLT and XQuery Processor [27] which is available as a Java library to access the XML metadata files. The classifier loops over all of the 96287 metadata files (which are in xml) and checks if the title contains the word ‘amendment’. If the metadata file contained the word ‘amendment’ in the title, the complete amendment was downloaded in the PDF file format using the tool ‘GNU Wget’ [28]. Out of 96287 metadata files 9680 files contained the string ‘amendment’ in their title. 9532 files were downloaded (149 download errors) and saved on a hard-disk. This is a success rate of 98,47%. A manual inspection of a random sample of 0,5% of the downloaded amendments indicate that there were no files not being an amendment. This indicates that checking if the title of a metadata file contains the word ‘amendment’ can be used to check if the ‘Kamerstuk’ is of the type ‘amendment’ (100% success rate).

2.2.2 Transform

The PDF files were converted into an XML format containing a structure which is primarily focussed towards the layout. This was done using a combination of Java and the tool ‘PDFTOHTML’ [29]. Every line out of the original PDF is now presented in the XML file containing the x and y position of that line and the height and width as in the original PDF document.

After we started processing pdf’s to xml documents the Dutch parliament came up with an XML format on their own. An example of this format can be found on the website of the Dutch Parliament (new amendments are available in this format) [30]. Although some metadata exists in this format and petitioners can be easily extracted the problem still exists that the content structure is primarily focussed towards layout purposes and not towards information purposes like we propose. The XML files made available by the Dutch parliament are also only available for amendments petitioned after 1995. This is because since then all files are stored digitally. We show that we can convert *almost any* pdf amendment using tools like PDFTOHTML and an XML processor like

SAXON to an XML format focussed towards information processing (an information centric approach).

After the first transformation we restructured the amendment (now in XML) by using XSLT [31] as the transformation mechanism. This is in order to add information into the structure of the XML so that you can use a query mechanism like XQuery [32] to retrieve information to answer specific questions. The XSLT was processed by using the Java programming language and the SAXON library. This resulted in a complete new format compared to the PDFTOHTML output/the original PDF document.

What were the criteria for the conversion?

To create a file structure which could be used to retrieve information easily we specified the following criteria:

1. The list of petitioners should be available in an easy manner;
2. The document identifier should be available in an easy manner;
3. The text of the amendment representing the changes to the law proposals should be structured so that you can use it to extract information;
4. The petition date should be available in an easy manner.

The first conversion step was to check if a document really was an amendment. If this was the case the real transformation started. We now briefly describe how the new format is set-up (Figure 2.2 can also be used to understand how the amendment structure is set-up).

The **root** element of each document contains two elements. One *metaData* element and one *content* element. The *metaData* (you can find a diagram representing the metadata structure in Figure B.1) element contains attributes containing the date, the document identifier, the title, the publisher and also the list of petitioners. The *content* element on the other hand contains the

actual text of the element. The text is structured by using blocks. Each block is based on a roman numeral in the original amendment. The roman numerals in the amendments each represent a new statement in the text. The blocks are linked to the numerals by using the *@triggeredBy* attribute for the block elements. Sometimes lines can exist without being part of a block. Additionally, the ‘amendmentTitle’ is include in the content structure.

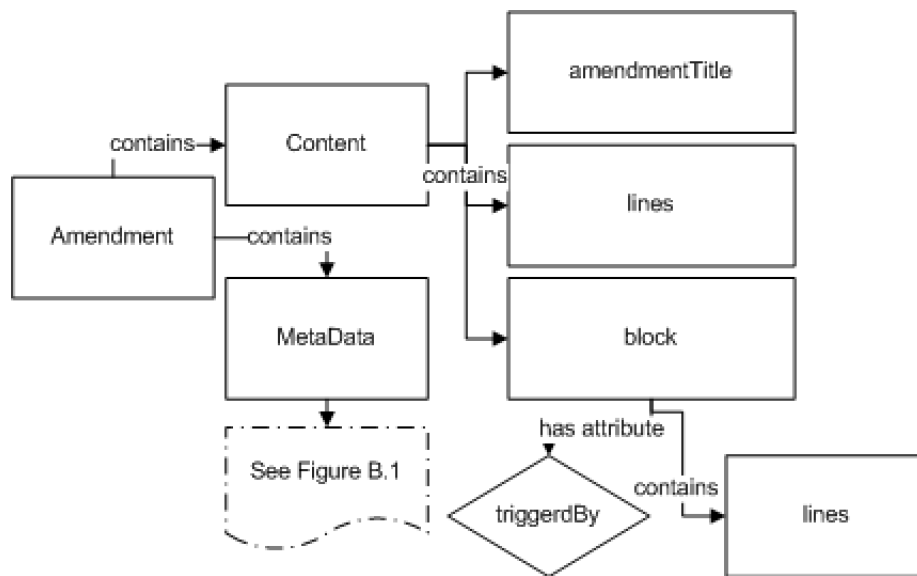


Figure 2.2: Diagram representing the structure of the new amendment format.

To create an information centric document like described above the XSLT transforms the original XML document. Our XSLT script started by finding the title by browsing the elements in an PDFTOHTML output document till the line where the document number was located. At this point the title is known. After defining the title the ‘infoLine’ is processed which contains metaData identifying the document. When the processing of the infoLine is finished the process is split up into two parts: one for amendments with 1 page and one for amendments with more than 1 page. The one page method starts by processing the page from the last line and then loop backwards to the top of the page. The

2 page continues by immediately processing the petitioners.

What is important to understand is that is very important to define till what point content exists. This is done by looping from the bottom of the page till we extracted all the petitioners. At that moment there is more than a 25 (pixel) margin between the *@top* attributes of the text tag. We saved the line and page number of this location so that in the next step, by looping from the top to the bottom for the content, this is the location to stop the processing.

Certain parts of the document indicate a new section. These parts are the lines with 'Ontvangen' of 'Voorgesteld' (defined by the pm:DateQuestion property) or the position of the document number, the place where the list of petitioners start and for the separation of blocks in the content and the 'Toelichting' are important.

During the transformation process we had some problems with amendments containing a different structure. An example of these differences is that some amendments included roman numeral in the content of an amendment on a single line. This created a problem because also the structure of an amendment is displayed by using these roman numerals (like I, II, III, IV). Another problem was the use of tables or images which resulted in a different PDFTOHTML output resulting in problems with the XSLT transformation.

After finishing the transformation the documents were validated using the Relax NG schema which was described earlier in this section [33]. See Appendix B for the full version of this schema.

Of the initial 9680 files tagged as an amendment, 9532 files were downloaded. The output by PDFTOHTML was 9500 amendments. Transforming these files using XSLT resulted in 9242 being left (258 amendments less than off the PDFTOHTML output). Of these 9242 files 14 files were not validated using the Relax NG schema.

This resulted in 9228 files going into the next phase (LOAD) of the ETL pro-

cess. 9228 files is 95.33% of the 9680 potential amendments, 4.67% of the files did not survive the conversion extraction and transformation phase.

2.2.3 Load

The transformed amendments in the XML format were initially separately saved on a hard-disk. During the process we imported the files into the XML database eXist [34].

2.3 Evaluation

To evaluate the surplus value of the information centric representation we performed a number of queries on our XML corpus. We did this to show that a range of complex analytical queries can be easily expressed in XQuery, yielding output which can be piped into visualization tools. The XML database is well suited for search and lookup and for heavy analytical querying.

Some of the queries give an answer to questions which could not be answered before. This is based on information from Monique Brouwers who is an employee of the Dutch Parliament. Some examples of questions which they were unable to answer:

1. How many amendments were petitioned by member X?
2. How many amendments were petitioned by party Y?
3. How many amendments were petitioned in a specific period?

In the next 9 (sub)sections we will display the queries we created including their result and analysis.

2.3.1 Which members of parliament are the most busy petitioning amendments?

Each example started with a question we came up with. These questions were based on ideas of our own or by reading publications online describing limitations

of the current systems. In this case we were wondering which members of parliament petitioned the most amendments and ideally on a per-year overview. The next step was to convert this question to an XQuery, which is the technology [32] we used to query the database. This resulted in the XQuery in Listing 2.2.

Listing 2.2: Amendments Evaluation - XQuery 1

```

1 pm="http://www.politicalmashup.nl"; declare namespace
  dc="http://purl.org/dc/elements/1.1/"; declare namespace
  js="http://www.jongmansolutions.nl";

5 <results> { let $documents :=
  collection("file:/c:/output?select=*.xml") for $petitioner
  in
7 distinct-values($documents//petitioner) let $totcount :=
  count($documents/document/metaData[petitioners/petitioner =
  $petitioner]) where $totcount > 200 order by $totcount
  descending,
  $petitioner return <result petitioner="{ $petitioner}"
11 totcount="{ $totcount}"> { for $year in
  distinct-values($documents/document/metaData[petitioners/
  petitioner=$petitioner]/dc:date/js:startYear)
13 let $count :=
  count($documents/document/metaData[petitioners/petitioner
  =
15 $petitioner][dc:date/js:startYear=$year]) order by $year
  descending return <year nr="{ $year}" count="{ $count}" /> s
  }
17 </result> } </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the graph showed in Figure 2.3.

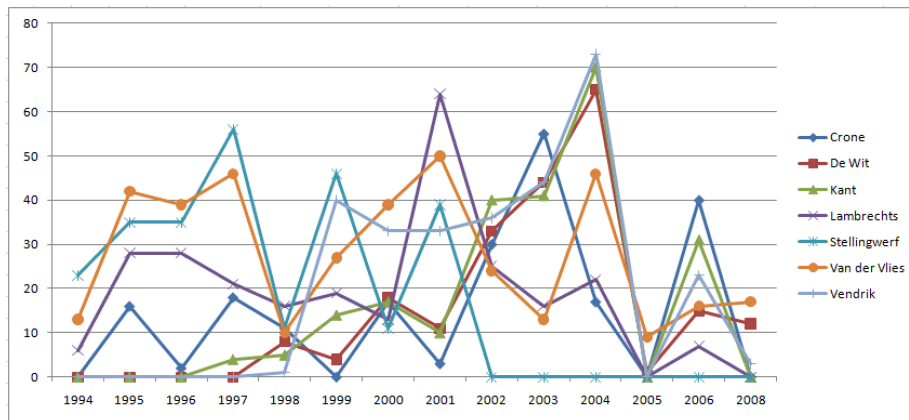


Figure 2.3: Members of parliament petitioning the most amendments

The graph shows that the top petitioners especially do that in 2001-2002 and 2004. 2005 and 2008 show a low amount of petitioned amendments which requires additional research to see why this is the case.

2.3.2 Which members of parliament petitioned the most together (per year)?

To get an answer to this question you first loop all the years available in the corpus, then loop the petitioners for every year and inside that petitioners loop, loop the petitioners again. You can then count how many times the petitioners worked together per year. This resulted in the XQuery in Listing 2.3.

Listing 2.3: Petitioning amendments together - XQuery 2

```

18 pm="http://www.politicalmashup.nl"; declare namespace
   dc="http://purl.org/dc/elements/1.1/"; declare namespace
20 js="http://www.jongmansolutions.nl";

22 <results> { let $documents :=
   collection("file:/c:/output?select=*.xml") for $year in
24   distinct-values($documents/document/metaData/dc:date/
     js:startYear)
   order by $year descending return <year nr="{ $year }"> { for
26   $petitioner in
     distinct-values($documents/document/metaData[dc:date/
       js:startYear=$year]/petitioners/petitioner)
28   for $petitionerb in
     distinct-values($documents/document/metaData[dc:date/
       js:startYear=$year][petitioners/petitioner=$petitioner
       ]/petitioners/petitioner/text())
30   let $count :=
     count($documents/document/metaData[dc:date/js:startYear=$
       year][petitioners/petitioner=$petitioner][petitioners/
       petitioner=$petitionerb])
32   where $petitionerb != $petitioner order by $count
     descending
     return <result>{concat($petitioner, ';', $petitionerb, ';'
       ,
34   $count, '\n')}}</result> </year> } </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the table showed in Figure 2.4.

The table shows that especially Vendrik en De Wit petition amendments together. We think that is not that strange because they are members of parties who are both positioned on the left side of the political spectrum. This tables immediately shows a feature which the system is currently lacking; members

Jaar	Indiener 1	Indiener 2	Aantal Keren Samen
2004	Vendrik	De Wit	49
2004	De Wit	Bussemaker	41
2004	Vendrik	Bussemaker	40
1996	Schimmel	Middel	37
1996	Van Heemskerck Pillis-Duvekot	De Koning	30
2008	Spekman	Ortega-Martijn	27
2003	Hessels	Crone	25
1996	Duivesteijn	Hofstra	25
2001	Hermann	Arib	24
2001	Eurlings	Hamer	24
2001	Lambrechts	Hamer	24

Figure 2.4: Members of parliament petitioning the most together

of parliament being connected to their parties so you can also use parties in a XQuery.

2.3.3 Which month is the most popular to petition an amendment?

To get an answer to this question we first loop over all the amendments and then count the amendments which are in the same month as the current amendment. This resulted in the XQuery in Listing 2.4.

Listing 2.4: XQuery showing the most popular petition month

```

36 pm="http://www.politicalmashup.nl"; declare namespace
dc="http://purl.org/dc/elements/1.1/"; declare namespace
js="http://www.jongmansolutions.nl";
38
40 <results> { let $documents :=
collection("file:/c:/output?select=*.xml") for $document
in
$documents/document let $month :=
42 month-from-date($document/metaData/pm:DateQuestion) let $
count
:=
44 count($documents/document/metaData[month-from-date(
pm:DateQuestion)=$month])
order by $month return concat($month,',';',$count) } </
results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the table showed in Figure 2.5.

Month	Amount of amendments
january	800
februari	659
march	920
april	715
may	498
june	1044
july	100
august	160
september	918
october	926
november	1551
december	937

Figure 2.5: Overview of the most popular month to petition an amendment

The table shows that especially the month november is very popular to petition an amendment. June and september, october and december are runner ups. We think that June is so popular because this is just before the summer holiday. The reason for the amendment explosion in and around november can be because this is the period around the budget meetings for the next year.

2.3.4 How many times did each block occur in the amendments?

A transformed amendment contains blocks in the content. Each block is triggered by a roman numeral (I, II, V, X) or by a "Toelichting". We think it is interesting to see how many times each block is used in the amendment repository. This resulted in the XQuery in Listing 2.5.

Listing 2.5: Showing how many times a block occurred in an amendment.

```

1 declare namespace pm="http://www.politicalmashup.nl";
  declare namespace dc="http://purl.org/dc/elements/1.1/";
3 declare namespace js="http://www.jongmansolutions.nl";

5 <results> {

```

```

let $documents := collection("file:/c:/output?select=*.xml")
7 for $value in distinct-values($documents//@triggeredBy)
  let $count := count($documents//block[@triggeredBy=$value])
9 order by $count descending,$value
  return
11 <triggeredBy value="{ $value}" count="{ $count}" />
} </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the graph showed in Figure 2.6.

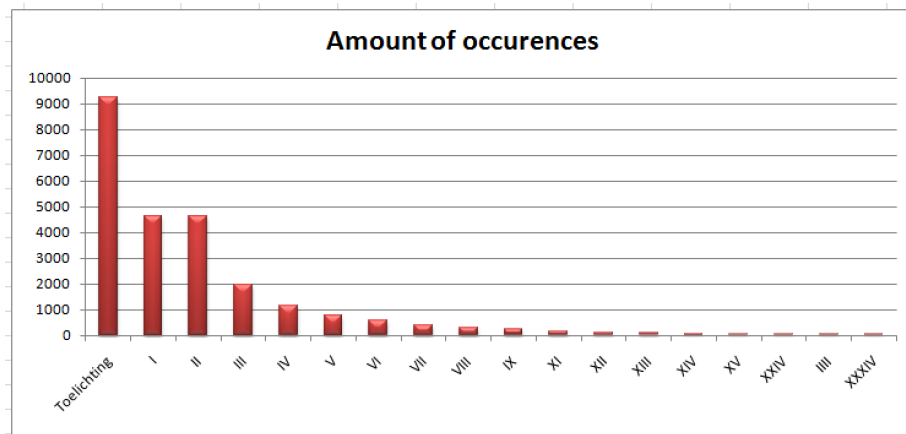


Figure 2.6: Overview of the occurrence of blocks per blocktype

The graphs shows that amendment with more than 10 blocks don't happen often.

2.3.5 How many amendments were petitioned every (parliamentary) year?

To answer this question we checked which years exist in the amendment repository and then for every year check how many amendments exist. This resulted in the XQuery in Listing 2.6.

Listing 2.6: The amount of amendments per year

```

declare namespace pm="http://www.politicalmashup.nl";
2 declare namespace dc="http://purl.org/dc/elements/1.1/";
  declare namespace js="http://www.jongmansolutions.nl";
4

```

```

<results> {
6 let $documents := collection("file:/c:/output?select=*.xml")
  for $year in distinct-values($documents/document/metaData/
    dc:date/js:startYear)
8 let $count := count($documents/document/metaData[dc:date/
    js:startYear=$year])
  order by $year descending
10 return
    <year nr="{ $year}" count="{ $count}" />
12 } </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the graph showed in Figure 2.7.

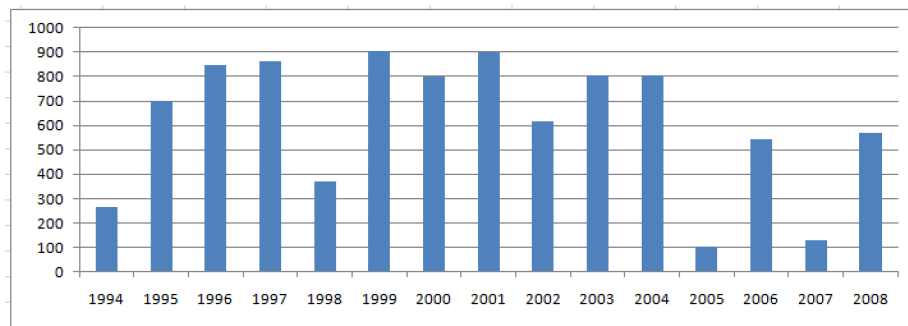


Figure 2.7: Overview of the amount of amendments per year

We suspect that for the years 2005 and 2007 we had some loading and transforming problems and that that is the reason for the low amount of amendments.

2.3.6 What is the highest number of petitioners for 1 amendments?

To answer this question you loop all the amendments and then count the petitioners per amendment. You then sort the result. This resulted in the XQuery in Listing 2.7.

Listing 2.7: Maximum amount of petitioners per amendment

```

declare namespace pm="http://www.politicalmashup.nl";
2 declare namespace dc="http://purl.org/dc/elements/1.1/";
declare namespace js="http://www.jongmansolutions.nl";
4
<results> {

```

```

6 let $documents := collection("file:/c:/output?select=*.xml")
  for $document in $documents/document
8 let $count := count($document//petitioner)
  order by $count descending
10 return
  $count
12 } </results>

```

The query turned out that in the available data there are 2 amendments which have 11! petitioners.

2.3.7 Which member of parlement uses the most blocks per amendment (at least 50 amendments petitioned)?

To answer this question you loop all the petitioners and count the amendments petitioned by the member of parlement and the amount of blocks inside the amendments. This resulted in the XQuery in Listing 2.8.

Listing 2.8: Most blocks per amendment

```

1 declare namespace pm="http://www.politicalmashup.nl";
  declare namespace dc="http://purl.org/dc/elements/1.1/";
3 declare namespace js="http://www.jongmansolutions.nl";

5 <results> {
  let $documents := collection("file:/c:/output?select=*.xml")
7 for $petitioner in distinct-values($documents//petitioner)
  let $blocks := count($documents/document[metaData/petitioners/
  petitioner=$petitioner]/content/blocks/block)
9 let $amendments := count($documents/document[metaData/
  petitioners/petitioner=$petitioner])
  let $avg := $blocks div $amendments
11 where $amendments > 50
  order by $avg descending
13 return
  <petitioner name="{ $petitioner}" blockCount="{ $blocks}"
  amendmentCount="{ $amendments}" avgBlocksAmendment="{ $avg
  }" />
15 } </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the graph showed in Figure 2.8.

Member of parlement Smits has the highest average of 4,01 block per amendment with 68 amendments petitioned. Member 'Heemskerck' has an average of 1,56 block per amendments with 54 amendments petitioned. This makes him the last one in the list (of the 94 members in the repository by the criteria).

blockCount	name	avgBlocksAmendment	amendmentCount
273	Smits	4,01	68
280	Jan de Vries	3,78	74
198	Ross-van Dorp	3,67	54
956	Lambrechts	3,61	265
191	Tichelaar	3,60	53
437	Schutte	3,55	123
230	De Krom	3,38	68
173	De Cloe	3,33	52
201	Van Vroonhoven-Kok	3,30	61
924	Vendrik	3,23	286

Figure 2.8: Overview of the amount of amendments per year

2.3.8 Which amendment has the most blocks?

To do this you loop all the amendments and count the blocks inside every amendment. Then you sort the results. This resulted in the XQuery in Listing 2.9.

Listing 2.9: XQuery showing the amendments with the most blocks

```

1 declare namespace pm="http://www.politicalmashup.nl";
2 declare namespace dc="http://purl.org/dc/elements/1.1/";
3 declare namespace js="http://www.jongmansolutions.nl";
4
5 <results> {
6   let $documents := collection("file:/c:/output?select=*.xml")
7   for $amendments in $documents/document
8   let $count := count($amendments/content/blocks/block)
9   order by $count descending
10  return
11    <amendment blockNr="{ $count }">{ $amendments/content/
12    amendmentTitle/text() }</amendment>
13 } </results>

```

5 amendments were returned by this query with a block count of 17 blocks per amendment.

2.3.9 Which petitioner(s) petitioned the amendment with the most words?

To achieve this you loop all the amendments and then count the words in every amendment. This resulted in the XQuery in Listing D.1.

Listing 2.10: XQuery showing overview of petitioners petitioning amendments with the most words

```

declare namespace pm="http://www.politicalmashup.nl";
2 declare namespace dc="http://purl.org/dc/elements/1.1/";
declare namespace js="http://www.jongmansolutions.nl";
4
declare function local:wordcount ($arg as xs:string) as
xs:integer
6 {
count(tokenize($arg, '\W+') [. != ''])
8 };

10 <results> {
let $documents := collection("file:/c:/output?select=*.xml")
12 for $amendments in $documents/document
let $bcount := count($amendments//block)
14 let $wcount :=
sum(
16 for $line in $amendments/content//line/text()
return local:wordcount($line)
18 )
order by $wcount descending
20 return
<amendment petitioner="{ $amendments/metaData/petitioners//
petitioner/text()}" bcount="{ $bcount}" wcount="{ $wcount}"
"/>
22 } </results>

```

We imported the output of the XQuery in Microsoft Excel. Excel was then used to create the graph showed in Figure 2.9.

petitioner	wcount
Van Middelkoop Dankers	15280
Van Middelkoop Dankers	15166
Stellingwerf Halsema Van den Berg	9140
Douma Van Egerschot	6218
De Wit Schimmel	4939
Van der Knaap Harrewijn De Wit Schimmel	4938
De Wit Vendrik Depla	4914
De Wit Vendrik Depla	4849
Wilders Schimmel Noorman-den Uyl	4696
De Krom Crone Hessels	3985

Figure 2.9: Overview of petitioners petitioning amendments with the most words

Van Middelkoop and Dankers petitioned the 2 amendments with the most words. In the table you can see the top 10 amendments. The graphs is made of the top 30 amendments.

The nine questions and their answers in the last few (sub) sections showed interesting information about amendments and gave some statistics about them. What is interesting to us is that answering these questions would be almost impossible when not using IT.

2.4 The Business Case

Digitizing old material can cost a lot resources. An example of this is the digitalization of old newspapers in the Netherlands, which costs about 12 million euro's [35]. When we think of all the historical publications in politics which are currently not accessible or at least not easy to find, like the documents published by the Dutch parliament, we know that digitalizing these documents costs a lot of resources. We think that, in this chapter, we showed some possibilities of how we can convert a document centric PDF corpus to an information centric xml database. We also showed that we can answer questions with our information centric approach which had no answers till this moment. That is why we think it is especially important to pay attention to the way document are saved in their digitalized format. We think this shows that there is a business case for converting all the amendments available to a rich information centric xml format. New possibilities which should be taken into account when creating a business case is:

- The ability to supply information about how many amendments were petitioned from 1995-1998. When an employee at the Dutch Parliament was asked about this kind of information he was not able to give this information. This would take to much time, or was impossible. With the enriched information centric structure this will only cost 5 minutes.
- Suppose you want to create an overview of parties working the most to-

gether in order to see which parties should be asked to form a new government after the elections. This overview can be created using a few queries on a rich information centric database as opposed to the current structure for which this will take hours.

The described situation are only two examples of how a business case can be set-up (especially related to the enriched content) using our thesis as a basis.

2.5 Conclusion

In this chapter we proposed a framework to improve the knowledge about one of the types of political documents, the amendment. This framework, by use of the ETL approach, can convert amendments from the current document centric structure to an information centric structure. What we showed is, that with a low error margin, older documents can be converted to a new structure and that the new corpus can be used to answer all kind of questions about its content. This kind of information can help journalists, student, researchers and above all the public when doing research on several (economical) topics. It also helps the public in controlling the government.

We overcame a lot of problems during the process of implementing the framework. The biggest problem was the transforming of the corpus from a document centric file store to an information centric xml database. That is why we recommend to save all documents in an information centric way, by starting today. By working with an information centric approach you will always be able to use the information in any way you want.

What we did not solve and what we would like to suggest as future research is how to link the amendments and their petitioners to their parties. Creating such a link creates all kind of opportunities. This is especially so when combining the link between petitioners and parties and the voting records of the amendments. This way you can think of answering questions like ‘What parties have the most success in getting an amendment approved when doing a combined petition?’. We think that when developing these suggestions for future research it is im-

portant to keep in mind the business case behind the ideas. When extending this topic one more step we think this can reduce costs that much that this project can result in a live environment.

In the next chapter about the importance of an debate we will go one level deeper into the research process and introduce some new ideas on how to use IT in improving the knowledge of the political-economical space. We will try to define what importance is and how you can calculate such an importance value.

Version 3.0

Chapter 3

The Importance of a Debate

The first section of this chapter will motivate why it is important and interesting to know what the importance of a debate is. In section 3.2 we will describe the aspects of importance followed by the operationalization of importance in section 3.3. Section 3.4 contains the description of the different models we have set-up for quantifying debate importance. The evaluation of the models is described in section 3.5. The chapter conclusion is drawn in section 3.6.

3.1 Introduction

Whenever you want to search for something on the world wide web you will need a good search engine. Today the standard for web search is Google [36], but what about searching through reports of the Dutch Parliament? In the Netherlands we have Parlando [14], but Parlando has its limitations (see chapter 1). Especially when you do not know what you are (exactly) searching for, when you do not have a document number to get the exact document that you need, you can get lost when using Parlando to search through debates.

To solve the problems which you can experience when using Parlando we want to introduce a new ranking method for the search results: the importance ranking. We researched the possibilities of using methods like PageRank [37] or HITS [38]. For these rankings you require some kind of graph of items connect-

ing through links. Although debating report documents have some references to documents like motions, these references are at the same time incomplete so that there is no possibility to create a graph containing debates which can be used for PageRank or HITS.

When creating a new ranking method, based on the importance of a debate, it is important to define what importance is. We define the importance of a debate as *the chance that the public finds a debate important*. If the importance is 0.8 then we define this as ‘that on average 8 out of 10 people will think that this debate is important’.

This importance value is based on the debates in the Dutch parliament and will not be related to the amendments discussed in chapter 2. The two chapters (2 and 3) do have the same interest in showing how IT can help in exploring the political-economical space saving time and money. Chapter 2 contained an enhancement of what currently existed, the ranking of a debate based on an importance calculation described in this chapter is something which has not been researched before.

With our newly introduced ranking method we created the possibility to explore economical publications in politics which would otherwise never come to our attention. Questions like “Are debates in specific topic more important than debates in other topic?” or “What were the most important debates having a relation with the economy?” or “What are the debates about the problems with the Dutch Pension Funds that Macro-Economists should take into consideration?”. In the next section we will describe the different aspects that play a role in our definition of importance.

3.2 Aspects of importance

To be able to show search results ranked by an importance value we first defined the aspects of importance. We created 3 categories (aspects) for the importance value. These categories are:

Key Players We believe that quantity and quality counts for this aspect. With quantity we mean that this category is based on *how many* people are at a debate. The quality aspect means that we also take into consideration *who* is present at a debate. An example is that if for every party the floor leader¹ is the speaker the debate can be more important when the speaker is just a “normal” member. On the side of the executive branch of the government, the debate will be more important when the prime-minister or the deputy prime minister is at the debate then for example when there is a minister of agriculture.

Debate Length The debate length category is based on how much time is required for a debate. A debate can take very long when parties do not agree with each other or with the executive branch. It can also be that the topic of a debate just requires some time to discuss because it is very important to discuss the topic properly.

Intensity The intensity category is based on how intense a debate is. When people argue a lot with each other (being intense) the debate often is important or receives a lot of attention from the media.

Each of these categories represent a part of the importance value and consist out of several attributes. These attributes are the features we measure in the debating reports and are discussed in the next section.

3.3 Operationalization of importance

We see the importance value as the chance that people find a debate important. To calculate the importance value we created an importance formula. We see the outcome of this formula as the chance that people find a debate important. If $I(\text{Importance}) = 0.80$ then we think that *there is a chance that 80% (8 out of 10) of the people find the debate being calculated important*. The formula

¹The floor leader is the member elected by his or her own parliamentary caucus to be their political leader in the house.

includes the aspects of importance as described in section 3.2. The formula can be found in equation 3.3, where i is a debate and p is the amount of attributes.

$$I_i = \beta_1 \times x_{i1} + \beta_2 \times x_{i2} + \dots + \beta_p \times x_{ip} \quad (3.1)$$

This *linear* formula is the sum of the values of all the attributes (x) which are multiplied by their corresponding weights (β). The attributes will be described below. The range of I (importance) is between 0 and 1. The 11 different weights (because we have 11 different attributes) sum up to 1 for the pre-defined models. To define the right importance value we came up with several methods on how to set-up the distribution of the weights. These weight distribution models will be described in the next section.

To calculate the importance we made use of documents available in Polidocs [13]. These documents are enriched recordings of debates in the Dutch Parliament [39] and are part of the ‘Handelingen’ (‘De Handelingen’ is the name for the recordings of all the meetings in the House and Senate of the Dutch Parliament). What we mean with enriched is that the documents contain tags in xml which bring meaning to parts of the document; we can say that the documents contain an information centric structure. The corpus contains 729 debates which were held between the 1st of January 2009 and the 31st of March 2010.

Each document contains all the debates of *one day*. Every debate in a document contains *blocks*. These blocks indicate that a new member of parliament starts his/her speech. In a block you can have *multiple speakers* interrupting other speakers. These speakers are each defined by the speaker element. The ER diagram describing the relations in this type of document can be found in Figure 3.1. We used a Relax-NG [33] schema to validate the structure of every debate.

While Figure 3.1 is describing the debate in a document, Figure 3.2 pictures

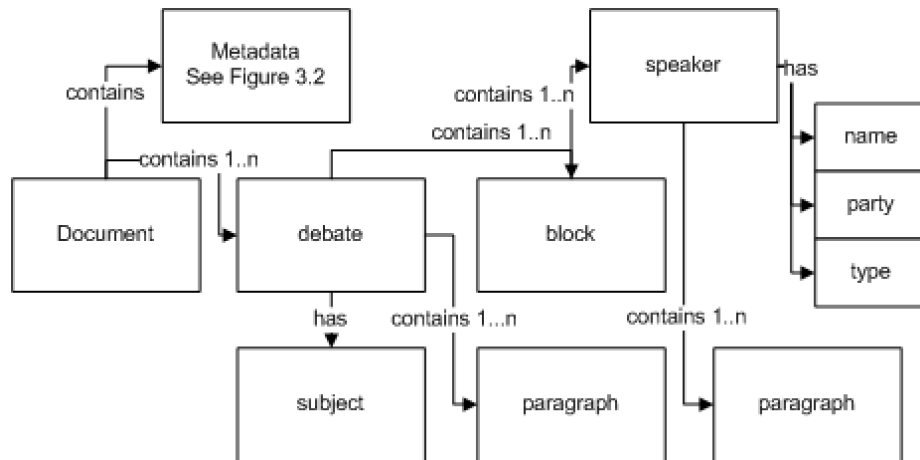


Figure 3.1: ER diagram of a debate.

the ER diagram of the attributes of a document. This document, as said before, contains the recordings of the debates of one day. The attributes described in the ER diagram are metadata for the debates. The attributes described are the start (time), end (time), number of present members (maximum is 150 for the House), the chamber (first (Senate) or second (House)), the source (where does the date originate from) and the date which can be used to check, for the documents described in the document, when they were held.

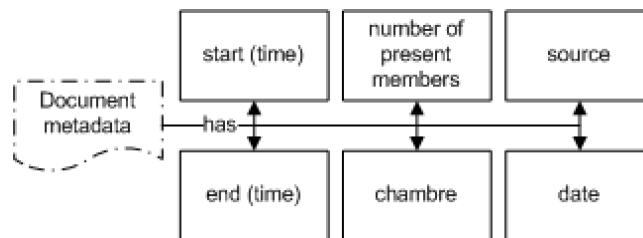


Figure 3.2: ER diagram of the metadata component of a 'Handelingen' document (which contains debates).

For every category discussed in section 3.2 we created attributes to make them measurable. We will now show which attribute is selected for every category and the reason behind these choices. We will also refer to the part of the

XQuery which calculates the value of the attribute for every debate (the line numbers refer to lines in Appendix D). Our hypothesis is that every attribute will have a positive correlation with the importance value.

3.3.1 Key Players category

The Key Players category contains attributes which are related to the people who are at the debate and to the people who participate in the debate, either on the side of the members of parliament or on the side of the executive branch.

Percentage Present The percentage of members of parliament who are at the debate. Lines 14 & 17 in Appendix D.

Deputy Prime Minister Bonus Are the two vice Prime Ministers at the debate? Lines 44, 45 & 64-69 in Appendix D.

Prime Minister Bonus Is the Prime Minister at the debate? Lines 43 & 58-63 in Appendix D.

Percentage of Floor Leaders The amount of speakers in the debate (on the side of the members of parliament) who are also the floor leaders of their party. Lines 46 & 89-96 in Appendix D.

Number of members speaking The headcount of the members of parliament speaking in the debate. Line 47 in Appendix D.

Number of members of the executive branch speaking The amount of members of the executive branch of government who are speaking. Line 19 in Appendix D.

3.3.2 Debate Length category

This category contains attributes which help in defining the debate length category. This is either on how long the debate takes or on what time the debate ends.

Wordcount This attribute is based on the the amount of words spoken in the debate. Lines 29-33 in Appendix D.

Closing time of a debate The time the debate closes. Line 16 in Appendix D.

Amount of blocks In a debate in the House of the Dutch Parliament every party can first let someone of their party defend the viewpoint of that party (which we will call a ‘speech’ from this point on). Every time during a debate a new party starts this speech a new block is created in the recordings. If there are 12 parties and 2 terms are available (a term is a round of speeches) you can have a maximum of 24 blocks (12*2).

We think that blocks are an indicator for the amount of speakers. We expect that if there a lot of blocks this often means a second term. A second term means that more time is required.

As apposed to the ‘speech’, which can only take place 1 time in each term for every party, interruptions on the speech of others can be made multiple times . The only limitations are the ones that are set by the chairman of the house. An example is that if the chairman thinks that the speaker is interrupted enough he can limit the amount of interruptions. These interruptions are measured by the attributes ‘Speaker Switches’ or ‘Debate Length’. Line 20 in Appendix D.

Second term The amount of times a member of the executive branch says ‘Tweede Termijn’ (second term). This can indicate that there will be a second term. Lines 31-41 in Appendix D.

3.3.3 Intensity Category

This category consists of only 1 attribute describing how many times participants interrupt each other in the debate making the debate more intense.

Speaker Switches The amount of switches between speakers in a debate. If this number is high the speakers get interrupted a lot. Line 22 in Appendix

D.

3.3.4 Normalization of attribute values

The attributes values which are calculated by the XQuery in Appendix D which we just described are not directly suitable to use in the importance formula as they have different ranges. To use these values for our importance calculation we normalize the values where needed. We either use a custom normalization function or min-max normalization [40]. The formula of min-max normalization is the following:

$$s'_k = \frac{s_k - \min(s_k)}{\max(s_k) - \min(s_k)} \quad (3.2)$$

The reason that we do not use this normalization function for every attribute is that it sometimes necessary to have a hard cut (non-continuity) in the value of an attribute instead of a continuous value. This is for example when for an attribute everything below 10 is worthless and 10+ is reasonable but 15+ is perfect. In this case a debate with 5 as the value for this attribute scores down the score of a debate with 15 as the value for the same attribute. We now describe what kind of normalization is used for each attribute of the importance value resulting in all attributes having a value between 0 and 1.

Percentage Present No normalization, value is already between 0 and 1.

Deputy Prime Minister Bonus, Prime Minister Bonus No normalization, value is already either 0 or 1.

Number of member speaking, Wordcount, Speaker Switches Min-Max normalization, value converted to a value between 0 and 1.

Percentage of Floor Leaders If the percentage of floor leaders speaking is higher than 0.9 the value is 1. If the percentage is between 0.6 and 0.9 the value is 0.3 otherwise it is 0. The value is multiplied is by 0.5 if the amount of speakers (not only the floor leaders) is lower than 9.

This is because for some smaller parties the floor leader will often be the

speaker while for big parties each member has its own area he or she is responsible for and will discuss in parliament. That is why the percentage of the time the floor leader speaks in smaller parties is higher than the average. This can increase the percentage of floor leaders in a wrong way (and devalue its value) if you want this attribute to add value to the importance index. One way of seeing this is that when all the parties are available and still the percentage of floor leader is high this deserves a bonus. The other way is that when, say we have 12 parties of which 5 are very small the floor leader is almost always the speaker for that party. In that case either if we have a debate with 8 or with 12 parties 5 are always speaking (or at least most of them). And this is not because the debate is important, but because no one else is available. This way the percentage of floor leaders is influenced by factors that we do not want to be part of the calculation. That is why we introduced a penalty for this attribute when most of the parties are not present (12 parties were in parliament for the data set used in this thesis).

Number of members of the executive branch speaking If the number of members speaking is 2 the value is 0.35. If it is higher than 2 the value is 1. Below 2 the value is 0.

Closing time of a debate For every debate the amount of minutes is calculated between midnight and the closing time of *the last debate that day*. This is because only the closing time of the last debate is known. If the debate ends before midnight this is zero. Suppose the last debate ends at 01:46 the amount of minutes is $60 + 46 = 106$. After calculating the minutes Min-Max normalization is applied. That way the value is converted to a value between 0 and 1, with a debating day ending very late close to 1 and debates closing before 12:00 a.m. having a value of 0.

Amount of blocks If the amount of blocks is higher than 10 then the value is 1, else it is 0.

Second term If the calculated value is higher than 1 the value is 1. Else it is 0.

At this point all the attributes are ready to be used for our importance calculation.

3.4 Different Models

We developed 5 different methods for the distribution of the attribute weights. The first method is that we give every attribute the same weight (0.091). In this way all attributes contribute the same amount to the importance value of a debate. The second method is that we give the attribute category ‘Key Players’ all the weight (every attribute in that category will get 0.166). For the third method we distribute all the weight to the ‘Debate Length’ category (the 4 attributes in this category each get 0.25). The fourth method distributes all the weight to the ‘Intensity’ category. Because this category only consists out of 1 attribute the attribute ‘Speaker Switches’ receives the full weight (1.0). The fifth and last method is based on our own experience in politics, the weights are distributed by what we think should be more (or less) important. In Table 3.1 we show all the weight distributions in one overview.

When we take a look at the ‘Own Weight’ model in Table 3.1 we see that we give ‘Percentage Floor Leaders’, ‘Wordcount’ and ‘Speaker Switches’ a lot of weight. We expect these 3 values to have the most predictive value of the importance of a debate. Summed up these attributes receive 56% of the weight that can be distributed.

3.5 Experiments and Evaluation

To evaluate the weights as set-up in Table 3.1 we have created an evaluation set-up. The first thing we did is calculating the importance value for every

Table 3.1: Weight Distribution

Attribute name	Weight Models				
	Equal Weight	Key Players	Debate Length	Intensity	Own Weight
<i>Percentage Present</i>	0.091	0.166	0.00	0.00	0.05
<i>Dep. Pri-Min Bonus</i>	0.091	0.166	0.00	0.00	0.05
<i>Pri-Min Bonus</i>	0.091	0.166	0.00	0.00	0.05
<i>Perc. Floor Leaders</i>	0.091	0.166	0.00	0.00	0.20
<i>Number mem. Speaking</i>	0.091	0.166	0.00	0.00	0.07
<i>Number exe. branch speaking</i>	0.091	0.166	0.00	0.00	0.05
<i>Wordcount</i>	0.091	0.00	0.25	0.00	0.20
<i>Closing time</i>	0.091	0.00	0.25	0.00	0.06
<i>Amount of blocks</i>	0.091	0.00	0.25	0.00	0.08
<i>Second term</i>	0.091	0.00	0.25	0.00	0.03
<i>Speaker Switches</i>	0.091	0.00	0.00	1.00	0.16

debate we have in our corpus. We did this for every weight model, so we will get 5 different importance values for every debate.

Our analysis is that we expect that only a small percentage of the debates will be found important by the public. This is supported by *all* of our models. In Figure 3.3, with the debates on the X-axis and the importance value on the Y-axis, you can see that only ± 50 debates have an importance value of over 0.4. This means that most of the debates (93%) are not found important by 6 out of 10 people according to our calculations.

We see a lot of similarity with the Pareto Distribution, which says that 20% of the population controls 80% of the wealth. In this graph you can see the way the importance is distributed over the debates with a small percentage of debates ($100/729=14(\%)$) being important (> 0.3) and the majority of the debates (86%) with a low importance value (≤ 0.3).

To evaluate the weights which are part of the importance formula (see Table 3.1) we did a survey under political experts in The Netherlands. We have sent surveys to political experts who work at national news media like BNR Newsradio, NOS, NRC Handelsblad, HP/De Tijd and Volkskrant. We have also sent

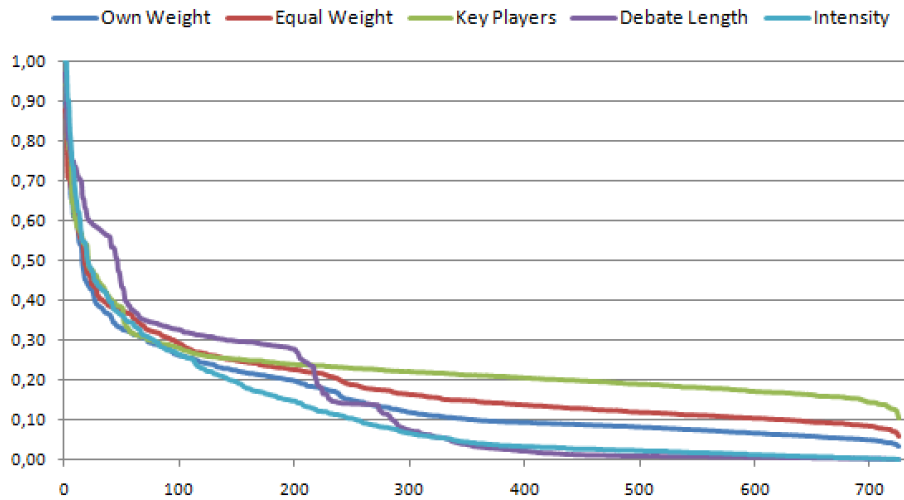


Figure 3.3: A visual overview of how important debates are

surveys to a political scientist at the VU University and a student who is the owner of pentapolica.nl.

Out of the 17 surveys we have sent we have got a response of 9 surveys. This is a response rate of 53%. The survey was set-up using a web page with javascript. This web-page contained a rated list (sorted A-Z) of the 100 most important debates according to their importance value. This importance value was based on the average of the top 30 of the 5 weight models. We first selected which debates were in the top 30 in every model, then we selected the debates which were in the top 30 in 4 models and so on. When the first 30 debates were selected the rest was added from the 'Own Weight' model. This was done because a lot of the debates were ranked in the top 100 in all of the models, with the difference that in one model a debate was on place 49 and in another model the debate was on 64. We reckon that especially debates close to the 100th position in the survey could have been different when using a different model, and see room for a improvement here when doing this survey on a bigger scale. We do think that this does not influence the results on a significant scale because only the top 30 is ranked in the survey, and we suspect that people will not rank a

debate which was in one of the 5 original models around the 100th position to be on the 25th or even higher position in a survey result.

In the survey the user was asked to select 30 debates (out of 100) and put these 30 debates into a category. The 100 debates were by default in the top 30-100 category (this means the debate was not part of the 30 most important debates). Apart from this category, 3 categories existed. One category was the ‘top 10’ (the 10 debates you find the most important), another category was the ‘top 10-20’ category and the last category was the ‘top 20-30’ category. At the end of the survey the goal was to have 10 debates in every ‘top’ category and 70 debates in the ‘unordered (30-100)’ category. 7 out of 9 surveys were completed for 100%, 1 survey only had a top 10 selected and 1 survey had a top 10 and a top 10-20 selected. An overview of the survey can be found in Figure 3.4.

Naam debat	Buiten top 30	Top 10	Top 10-20	Top 20-30
Aandelen viroloog Osterhaus (29 oktober 2009)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Aanpassen veiligheidsregio's (27 januari 2010)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

100 debates

Wet dieren (7 oktober 2009)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Winkelshuiving (19 november 2009)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Totaal geselecteerd	100	0	0	0
Opmerkingen (mist u een debat, wilt u iets anders opmerken?):	<input type="text"/>			
Verzenden				

Figure 3.4: Overview of the survey for the parliamentary Journalists and experts

To use the ranking created by the experts for the evaluation of the weight models, the ranking in categories per debate had to be converted to a value between 0 and 1. This was done by giving each returned survey 100 points. We distributed these 100 points over the debates based on the categorization in the survey. How this was done depends if every ‘top category’ was used. If you only selected a top 10, 90 debates were unranked and 20 positions were not used (10

Table 3.2: Distribution of points over debates per returned survey

Returned Survey Options	Points per debate in category			
	Top 10	Top 10-20	Top 20-30	Uncategorized
Top 10+20+30	4	3	2	$\frac{10}{70}$
Top 10+20	4	3	0	$\frac{30}{80}$
Top 10	4	0	0	$\frac{60}{90}$

position for the top 10-20 category and 10 position in the top 20-30 category). In Table 3.2 an overview is given of the possible distribution models.

If only a top 10 was selected a debate in the top 10 category received 4 points ($4 * 10 = 40$ points in total for this category) and an uncategorized debate received $\frac{60}{90}$ points. With 90 debates uncategorized the total amount of points is $\frac{60}{90} * 90 = 60$. In the case that 30 of the 100 debates were ranked, only 70 debates were unranked and these unranked debates each receive $\frac{10}{70}$ points.

Per debate the scores received out of every survey was summed up. Suppose a debate was ranked in the top 10 in 5 returned surveys, and in the top 20-30 for the other 4 surveys. Then this debate got $4 \text{ points} * 5 \text{ surveys} = 20 \text{ points}$ plus an additional $3 \text{ points} * 4 \text{ surveys} = 12 \text{ points}$ making this a total of 32 points. The next step is to divide this value by 9 because of the 9 surveys and you want to create an average value; 32 divided by $9 \approx 3.56$. To convert this value to a value between 0 and 1 the last step is to divide the value by 4. This way you will get a value between 0 and 1 ($3.56/4 \approx 0.89$).

From this ranking the 30 highest ranked debates were selected. This selection was compared with the the top 30 of each of the models using excel. We compared if a debate from one top 30 ranking also existed in the top 30 ranking from the survey. The result of the comparison can be found in Table 3.3. The result shows that especially the debate length model is not performing well, while the other models score either 53% or 57% at P@30. The difference of 4%

Table 3.3: Model Rankings compared with Survey

Model Name					
	Equal Weight	Key Players	Debate Length	Intensity	Own Weight
P@10	80%	90%	40%	90%	80%
P@20	55%	60%	25%	60%	55%
P@30	53%	57%	33%	57%	53%

(53-57) is only 1 debate, which we see as a small margin.

Apart from the evaluation of the models by comparing them with the survey results, we used another method to evaluate the attributes and weights of the models. This method is called Multivariate Linear Regression (MLR). With this method, given the depending variable y and multiple independent variables one aims to find a linear model. We applied this method to the situation described in this chapter. The formula used for calculating the importance of a debate, which includes the strength between the attribute weights and attribute values is the following (i is a debate, p is the amount of attributes):

$$I_i = \beta_1 \times x_{i1} + \beta_2 \times x_{i2} + \dots + \beta_p \times x_{ip} \quad (3.3)$$

The attributes $x_{i1} \dots x_{ip}$ for each debate i are calculated by our XQuery (see Appendix D), while the parameters $\beta_1 \dots \beta_p$ (which are the same for all debates) are unknown. That is why we require a method like MLR which calculates the linear correlations between (in our case) 11 independent (predictor) variables and a single dependent (response) variable [41] [42].

For this method we have a matrix Y which includes the response variable (which is in this case the importance based on the observations in (the training part of) our survey, with the dimension $n \times 1$):

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

We have a matrix called \mathbf{X} , which has the form $n \times p$, where n is the amount of debates and p is the amount of attributes.

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & & & \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

The transposed vector x_i^T , representing all the attribute values for one debate is:

$$x_i^T = [x_{i1} \quad x_{i2} \quad \cdots \quad x_{ip}]$$

The result which we require is a vector β (with the dimension $p \times 1$, where p is the amount of attributes).

$$\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}$$

This way the equation can be written in matrix terms as:

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{e} \quad (3.4)$$

The error vector \mathbf{e} is unobserved, it contains random noise of mean 0. The algorithm used by MLR for obtaining the parameters in vector β is called least squares. In this case this means the least square estimate $\hat{\beta}$ of β is chosen to minimize the residual sum of squares. The residual sum of squares is:

$$RSS(\beta) = \sum_{i=1}^n (y_i - \mathbf{x}_i^T \beta)^2 = (\mathbf{Y} - \mathbf{X}\beta)^T (\mathbf{Y} - \mathbf{X}\beta) \quad (3.5)$$

Simplifying the equation results in [43] [44]:

$$\begin{aligned} RSS &= (\mathbf{Y}^T - \beta^T \mathbf{X}^T)(\mathbf{Y} - \mathbf{X}\beta) \\ RSS &= \mathbf{Y}^T \mathbf{Y} - \mathbf{Y}^T \mathbf{X}\beta - \beta^T \mathbf{X}^T \mathbf{Y} + \beta^T \mathbf{X}^T \mathbf{X}\beta \\ &\text{we simplify using } \mathbf{Y}^T \mathbf{X}\beta = \beta^T \mathbf{X}^T \mathbf{Y} \\ RSS &= \mathbf{Y}^T \mathbf{Y} - 2\mathbf{Y}^T \mathbf{X}\beta + \beta^T \mathbf{X}^T \mathbf{X}\beta \end{aligned} \quad (3.6)$$

Taking derivatives with respect to β , and setting these to 0 will lead to the following equations:

$$\begin{aligned} \frac{\delta RSS(\beta)}{\delta \beta} &= \frac{\delta}{\delta \beta} \mathbf{Y}^T \mathbf{Y} - 2\mathbf{Y}^T \mathbf{X} \beta + \beta^T \mathbf{X}^T \mathbf{X} \beta \\ &\text{we compute the derivative using } \frac{\delta \mathbf{A} \beta}{\delta \beta} = \mathbf{A}^T \\ &\text{we compute the derivative using } \frac{\delta \beta^T \mathbf{A} \beta}{\delta \beta} = \mathbf{A} \beta + \mathbf{A}^T \beta \\ \frac{\delta RSS(\beta)}{\delta \beta} &= -2\mathbf{X}^T \mathbf{Y} + 2\mathbf{X}^T \mathbf{X} \beta \quad (3.7) \\ -2\mathbf{X}^T \mathbf{Y} + 2\mathbf{X}^T \mathbf{X} \beta &= 0 \\ -2\mathbf{X}^T \mathbf{X} \beta &= -2\mathbf{X}^T \mathbf{Y} \implies \text{multiply by } -\frac{1}{2} \\ \mathbf{X}^T \mathbf{X} \beta &= \mathbf{X}^T \mathbf{Y} \implies \text{apply the inverse of } \mathbf{X}^T \mathbf{X} \\ \hat{\beta} &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \quad (3.8) \end{aligned}$$

To perform these calculations we made use of the matlab [45] method *mvregress* [46] which has the earlier described theory implemented. We first created the test and training set. The training set contains a random sample of 60 of the 100 debates we used in our survey. The test set contains the 40 other debates. For the *mvregress* method we created 2 vectors. The first vector contained 1 column and 60 rows, every row represented one debate from the survey. For every debate the ranking for a debate based on the average score of the survey results was selected. The second vector contained 11 columns and 60 rows. The 11 columns represented the 11 attributes of the importance formula. Every row (debate) contained the attribute values for that debate calculated by the XQuery discussed in section 3.3 (after normalization). The return value of *mvregress* is a vector β of coefficient estimates for I . You can find the result of the *mvregress* method in the updated weights distribution Table 3.4.

We notice a few remarkable things when we compare the Matlab Weight Model with the other models. The first thing is that the value of the ‘word-count weight attribute’ is *very* negative. While we expected the wordcount attribute to have a positive correlation with the Importance value, we see that

Table 3.4: Updated Weight Distribution

Attribute name	Weight Models					
	Equal Weight	Key Players	Debate Length	Intensity	Own Weight	Matlab ¹
<i>Percentage Present</i>	0.091	0.166	0.00	0.00	0.05	0.156
<i>Dep. Pri-Min Bonus</i>	0.091	0.166	0.00	0.00	0.05	0.0133
<i>Pri-Min Bonus</i>	0.091	0.166	0.00	0.00	0.05	0.1342
<i>Perc. Floor Leaders</i>	0.091	0.166	0.00	0.00	0.20	0.0916
<i>Number mem. Speaking</i>	0.091	0.166	0.00	0.00	0.07	0.1885
<i>Number exe. branch speaking</i>	0.091	0.166	0.00	0.00	0.05	-0.0116
<i>Wordcount</i>	0.091	0.00	0.25	0.00	0.20	-0.6847
<i>Closing time</i>	0.091	0.00	0.25	0.00	0.06	0.1292
<i>Amount of blocks</i>	0.091	0.00	0.25	0.00	0.08	-0.06
<i>Second term</i>	0.091	0.00	0.25	0.00	0.03	-0.0264
<i>Speaker Switches</i>	0.091	0.00	0.00	1.00	0.16	0.7818

¹ The weights in this model were calculated with the use of training data from the survey. The training data consisted of 60% of the ranked debates in the survey.

the opposite is the case in the ‘Matlab’ model. In line with the wordcount attribute, 2 out of the 3 other attributes (Second term, Amount of blocks) of the Debate Length Attribute Category have a negative value in the Matlab Weight Model. This is not the case in the other models.

We can not think of a good reason why the wordcount attribute weight was predicted with a negative value and were surprised by this. An option could be that at a certain moment politicians take too much time for their speeches and forget what it is about so it loses its importance. We also see strong evidence against the assumption that long debates are not important: the weights for the attributes in the Intensity category (‘Speaker Switches’) show a positive value (0.7818). A lot of interruptions can lead to an increased amount of words spoken. That is why we strongly recommend to do future research why the wordcount attribute value is so negative for the matlab model.

For the Key Players category 5 out of 6 attributes score a positive correlation. We see for this category as the most remarkable difference with the other models that there is a negative value for the attribute ‘Number executive branch speaking’. The reason behind this can be that for some long debates multiple subjects are discussed and that most of them are not important but do require different secretaries.

Table 3.5: Updated Model Rankings compared with Survey

Model Name						
	Equal Weight	Key Players	Debate Length	Intensity	Own Weight	Matlab ¹
P@10	80%	90%	40%	90%	80%	80%
P@20	55%	60%	25%	60%	55%	85%
P@30	53%	57%	33%	57%	53%	77%

¹ This model was compared with the test set survey results and not with the complete set of survey results which was done for the other models.

To evaluate the Matlab model in the same way the other models were evaluated using the P@ test in combination with the survey, we used the weights supplied by the mvregress method for the importance function I . These weights were based on the 60 debates in the training set and were used to calculate the importance I for the 40 debates in the test set. The ranking of these 40 debates were then compared with the same 40 debates but then ranked by the importance based on the survey result. We computed the P@10, P@20 and P@30 for the two models. The results can be seen in the updated ‘Model Rankings compared with Survey’ Table, Table 3.5.

While the P@10 score for the Matlab model is about the same as for the other models, the P@20 and especially the P@30 value are significantly higher than they are in the other models. We advise for future work to increase the scale of the survey on the number of participants and also by letting the participants rank an increased number of debates instead of 100 which were ranked in our survey. We think this can increase the validity of the results.

3.6 Conclusion

In this chapter we proposed a new way of ranking search results: the importance ranking. This is a new way of looking at search results in the area of political documents. Based on the importance aspects ‘Key Players’, ‘Debate Length’ and ‘Intensity’ we defined aspect attributes. By evaluating these attributes using surveys and multivariate linear regression we found some attributes to be more important than others. Our assumption that all attributes had a positive

correlation with the importance value was contested by the regression method. These findings are in our opinion opportunities for future research. Tuning the weights and the amount and variety of the attributes is something which can increase the accuracy of the importance ranking, especially when combining these optimizations with a broader spectrum of data (several years). The importance ranking is in our opinion something which can help in bringing the public closer to politics and politicians. No matter if you are a journalist, a researcher or 'just' a citizen you will get attracted by the possibility of creating an overview of important debates over the last few years, per politician or topic with a mouse click.

Version 3.0

Chapter 4

Debates Performance System

In order to make the performance index available in an easy manner the Debates Performance System (DPS) was developed. This framework is based on the theory described in the last chapters. The DPS can be used to search through the debates in the Dutch parliament using certain criteria. In the next sections we explain these criteria as well as the data set used.

4.1 DPS Basics

The DPS has as its basis the eXist Database [34]. The package of this Open Source Native XML Database contains a Web Server and Apache Lucene. Apache Lucene is “a high-performance, full-featured text search engine library written entirely in Java” [47]. Both the Web Server and Apache Lucene are used for DPS.

The Web Server is used to deliver the pages to the user, which can be viewed by any ordinary web browser. The description of the functionality can be found in the next section. Apache Lucene is used to search through the debates. The reason for using Lucene is that it speeds up the search process by creating indexes.

In the database debates exist between 1995 and 2010. All the debates were processed using XQuery to create a file including the importance value for every debate. In this (XML) file there is an XML element for every debate which has as the attributes the debate id and 6 attributes representing the different importance models. The XQuery which created the importance file is also available on the database. It took 25 minutes to calculate the importance values for 9087 debates on an 8 core machine with 21GB of memory.

4.2 Searching

To show the DPS we did a search like we suppose an economist would do. This is a search about the ‘AOW’, which is a state installed pension guaranteed for all. Currently there is a discussion at what age you will get this pension. The line is 65 at the moment, but because of an increasing amount of elderly (not working) people the workforce has to be increased. That is why some parties are suggesting to increase the minimal age for receiving the ‘AOW’ to 66 or even 67.

← → ↻ ☆ http://localhost:8080/exist/rest/db/XQueries/search.html

DPI 1.0

Use this form to search through the debates.

Search Form

text search:	<input type="text"/>		
text said by:	<input type="text"/>		
text said in time frame:	from	January ▾	- 1 ▾ - 1995 ▾
	to	December ▾	- 31 ▾ - 2010 ▾
<input type="button" value="Search through Debates"/>			

Figure 4.1: DPS Search Form.

To do this you can set the following search parameters (see Figure 4.1 for the actual search form):

- search query including the topic to search for;
- the politician who said the terms in the query;
- the timeframe when the said terms were used.

As an example we have set the timeframe to 01-01-2009 to 07-01-2010, we did not set a politician and as query we used ‘AOW’. This resulted in the overview shown in Figure 4.2.



Figure 4.2: DPS Search Result.

The search result shows the first 10 debates matching the search criteria sorted by their importance value. For every debate a short snippet of text is showed and a link to the page of the debate itself. This page provides more details like all the topics of the day that the debate was held, a list of the speakers in the debate and a link to the original file. A screenshot of this page is shown in Figure 4.3. The software for this page is not included in the DPS,

it is software running at the University of Amsterdam (UVA)¹.



Debat naar aanleiding va... x

http://mashup1.science.uva.nl:8080/nl/data/HAN1995/h-tk-20072008-6088-6092.xml?view=nl/proceedings#h-tk-2007200

86e vergadering 2008-05-21
Jaargang: 2007-2008
Tweede Kamer
Bronbestanden: [HTML](#) - [PDE](#) - [XML](#)

DutchParl

ALGEMENE GEGEVENS

Titel: Debat naar aanleiding van een algemeen overleg op 13 mei 2008 over de financiële problematiek bij inburgeringscursussen

Datum: 2008-05-21

Categorieën: Cultuur en recreatie
Cultuur
Recht
Staatsrecht
Migratie en integratie
Immigratie

Spreekers in dit debat: [Karabulut \(SP\)](#), [Dijsselbloem \(PvdA\)](#), [Van Toorenburg \(CDA\)](#), [Kamp \(VVD\)](#), [Dibi \(GroenLinks\)](#), [Vogelaar](#), [Van Rijsterveldt-Vliegenthart](#)

Ingediende moties: [Karabulut_Nr_15 \(31143\)](#)
[Dibi_Nr_16 \(31143\)](#)

INHOUDSOPGAVE

[Onderwerp 1 - Financiële problematiek inburgeringscursussen](#)

- [Mevrouw Karabulut \(SP\)](#)

- [De heer Dibi \(GroenLinks\)](#)

- [Minister Vogelaar \(\)](#)

- [Staatssecretaris Van Rijsterveldt-Vliegenthart \(\)](#)

L = Permalink, copy link to your clipboard, /> Terug naar boven

> 1. Financiële problematiek inburgeringscursussen

Aan de orde is het debatnaar aanleiding van een algemeen overleg op 13 mei 2008 over de financiële problematiek bij inburgeringscursussen.

Aan het woord is: Mevrouw Karabulut (SP)

 **Mevrouw Karabulut SP**

Voorzitter: Tot vandaag aan toe vallen er ontslagen bij roc's en particuliere aanbieders. Dit is het gevolg van een domme, ingewikkelde en bureaucratische Wet inburgering. Dom omdat marktwerking in de publieke sector niet werkt. Dom ook omdat deze bureaucratische wet veel te haastig werd ingevoerd, zodat gemeenten geen tijd hadden om de wet goed uit te voeren. Daarvoor is gewaarschuwd, zowel binnen als buiten

Figure 4.3: DPS Debate Page.

4.3 Ranking

It is also possible to list the 20 most important debates in a timeframe or for a politician. This can be done by using a similar method as in the case of searching through debates. When we rank the most important debates in which the politician Atsma (politician from the CDA party) was speaking, in the timeframe january 2009 - december 2010. It is interesting to see that most of the topics in this list are about the area Atsma is the specialist of in his party: agriculture. This list can be found in Figure 4.4.

¹See <http://mashup1.science.uva.nl>



Figure 4.4: DPS Ranking with timeframe 2009-2010 and parliamentarian Atsma.

It is also possible not to select the politician option. When we keep the same timeframe, january 2009 - december 2010, the software will rank all the debates in this timeframe. This results in the list shown in Figure 4.5. Important to understand is that these queries (and especially the global query) are very heavy for the server running the software. On a server with a Core2Duo E6600 and 3GB of memory a global ranking is very hard to calculate and takes about 1 minute. A ranking of the debates within the timeframe 1995-2010 takes at least 10 minutes. Both rankings were ranked by the model which is based on the results of the surveys; the 'matlab' model.



Figure 4.5: DPS Ranking with timeframe 2009-2010.

Chapter 5

Conclusion

In this chapter we answer the research question(s) defined in chapter 1. We will also outline the future research that can be done and discuss the improvements that can be made to the proposed framework.

5.1 Conclusion

This thesis proposes several solutions that aim to improve the knowledge about the political-economic space. We currently experience problems when trying to get insight into this space. In the case of the amendments, which was described in chapter 2, the problem was that there was a huge amount of data, which had a document centric structure. The structure containing no metadata, limited the options you had for retrieving information hidden in the data. In the case of an enriched data corpus, you can use query languages to do research increasing the effectiveness and speed of the calculations. Additionally, retrieving all amendments through one portal without irrelevant data from other sources was very hard, if not impossible. This made the user check several sources making the research of the political-economic space a resource intensive task. In this thesis we suggested that amendments should be made available in an information centric format for the full corpus. We used the ETL (Extract, Transform, Load) mechanism which resulted in a database of XML documents containing an information centric structure. We showed that for an expert user it is rel-

atively easy to perform queries on this database which helps them in delivering information to an inexperienced user. This proves that IT can help in disclosing information in data in a relatively easy manner, which can lower project costs because less hours of manpower are required. This can not only help in solving current problems, but can also trigger users to ask new questions because they become aware of new possibilities. One such possibility is combining several enriched repositories. An example is that a user can make an overview of parties who petition the most amendments together and combine this with election polls to see if a coalition could have a reasonable chance to survive.

In chapter 3 the ranking of debates based on their importance was researched. The idea behind the research was that current search results, when searching using the parliamentary search engine, are sorted by the amount of occurrences of a word or by time. In this thesis we showed that sorting search results based on importance is something which can help the user in finding relevant information in a quick manner. To calculate the importance we used information inside a document, like the attendees, the length of the document, the amount of interruptions or the amount of speakers. We call these aspects attributes. These attributes were calculated for every debate and then used to calculate the importance value for these debates. To validate our ideas we held a survey under an expert panel of political journalists. This expert panel ranked a list of 100 debates. The ranking based on the survey was then compared with several importance models. Based on the survey ranking we have created an importance model using multivariate linear regression. This regression method was used to calculate the weights for the attributes in our importance model. 5 out of 6 importance models (including the multivariate regression model) had at least 50% of the same debates in their top 30 as the ranking based on the survey.

5.2 Future work

Related to the amendments in chapter 2 the most important aspect is to improve search and exploration of these documents. That is why we suggest to do a survey under the current user base of search engines containing publications related to politics to get an image of what kind of information these users miss in current data. A result of this survey might be that the users miss the relation between politicians and their parties. We think introducing this relation into the structure of the amendments can improve the system. The second recommendation for future work related to amendments (especially to the Extract, Transform, Load (ETL) mechanism) is that the way the information structure is currently inserted into the data using XSLT could be improved especially by lowering the time costs of the transformation part of the mechanism. One way to improve the transformation is by using XSLT indexing to improve the speed of accessing frequently accessed elements. Another way is to limit the use of recursive loops. In the current transformation process these loops are often used and we suspect that they are resource-intensive. We feel that the use of these loops could be optimized using XSLT or that a different programming language (like Java) could be implemented to limit the amount of recursive loops.

The importance ranking which was introduced in chapter 3 is one of the most novel aspects of our thesis. The proposed ranking could be improved in several ways. First, the survey which was held for validation and creation of one of the models in our thesis, should be done on a larger scale by finding more participants. Second, each participant should rank a larger dataset which costs more time but will improve the scale of the available information. Our third recommendation is to train the models on a larger dataset with a broader timeframe. This will let the models perform better when calculating global rankings for a large timeframe because the chance of overfitting on a specific political period is reduced in this case.

After improving the way the models were calculated, we think more attention should be spent at investigating the result of the calculations, especially in relation to the number of the attributes and the correlation between these attributes. We can imagine that new attributes are introduced and/or other attributes will be removed. This might be because some attributes have a very high correlation so that it would not matter if one would be removed. An attribute you could add the time period of the debate. We suspect that there are certain periods in a year when the most important debates are held (just before the recess of the parliament, october-november when the debates about the budget take place). A bonus could be given to debates which are held in these ‘important periods’. Both removing and adding attributes require additional research.

Our final recommendation is about the representation of the debates using the performance index. Currently the cost of searching through the debates and then ordering by the importance value is high because it costs a few minutes when using a timeframe of more than a few years. We suggest that the way the database indexes are built in the current set-up and how they are used by the queries for searching and/or ranking should be reviewed and ways of improvement should be looked after. Decreasing the size of the index, by removing unnecessary data can help in increasing the speed.

Appendix A

Example of an amendment

Figure A.1 shows a screenshot of an amendment. Amendments are set-up in a specific format. The actual content of an amendment consist of one or more blocks. We use these blocks to calculate statistics about the amendments.

Tweede Kamer der Staten-Generaal **2**

Vergaderjaar 1994-1995

22 485 **Wijziging van de Wet op de dierproeven**

Nr. 28 **NADER GEWIJZIGDE AMENDEMENTEN VAN HET LID CHERRIBI C.S. TER VERVANGING VAN DIE GEDRUKT ONDER NR. 24**
Ontvangen 7 juni 1995

De ondergetekenden stellen de volgende amendementen voor:

Step 1 I
In artikel I, onderdeel C, wordt na artikel 10c een artikel ingevoegd, luidende:
Artikel 10d
Het is verboden een dierproef te verrichten voor het ontwikkelen van nieuwe danwel het testen van bestaande cosmetica waarvoor regels zijn vastgesteld op grond van de Warenwet.

Step 2 II
In artikel I, onderdeel K, wordt in artikel 25, eerste lid na «10b, eerste lid,» ingevoegd: 10d,.

Step 3 III

Figure A.1: An amendment

Appendix B

Amendment Structure

Figure B.1 describes the metadata part of the new amendment format as proposed in this thesis. It includes a few different aspects like the data, the petitioners and the document number.

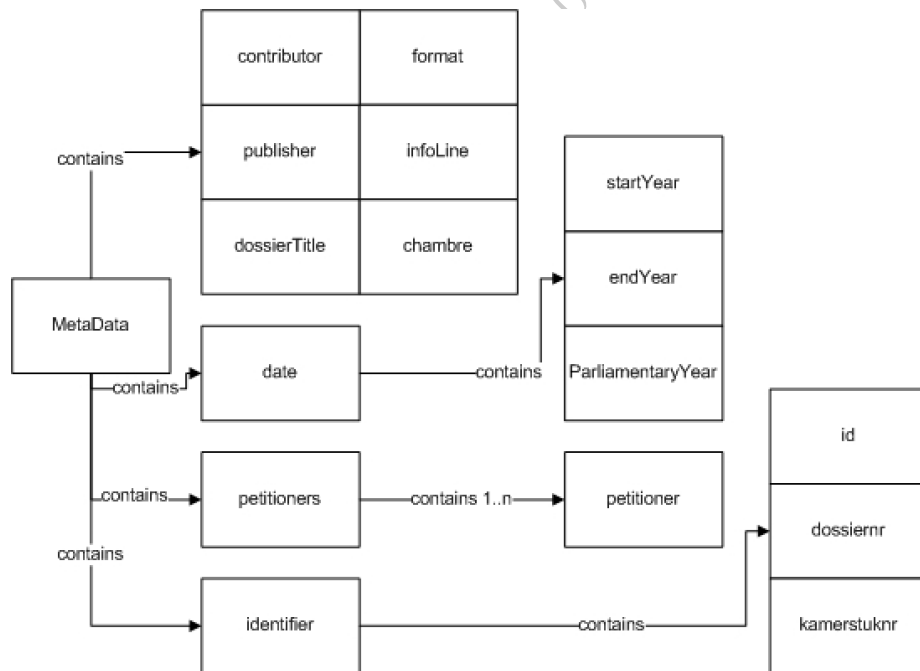


Figure B.1: Diagram representing the metadata part of the structure of the new amendment format.

Appendix C

Relax NG validation scheme debates

We use Relax NG to validate the xml representation of the debates. The files consist primarily out of text and metadata elements, which are accordingly described in the Relax NG file.

Listing C.1: Relax NG validation scheme of a debate

```
default namespace = ""
2
start =
4   element handling {
      element metadata {
6         element item {
            attribute attribuut { xsd:NCName },
8            (text
              | element leden {
10               attribute aantal { xsd:integer },
                  attribute status { xsd:NCName },
12               element lid {
                    attribute aanspreektitel { xsd:NCName }?,
14                    attribute departement { text }?,
                        attribute geslacht { text },
16                    attribute partij { text },
                            attribute soort { xsd:NCName }?,
18                    text
                }+
            }+
20        }+
    }+,
22   element text {
      element onderwerp {
24         attribute onderwerp { text },
                attribute pagina { text },
26         p*,
28         element blok {
```

```

30     element spreker {
31         attribute anker { xsd:integer },
32         attribute geslacht { text },
33         attribute naam { text },
34         attribute pagina { text },
35         attribute partij { text },
36         attribute soort { text },
37         p*,
38         element motie {
39             attribute nummer { text },
40             p+
41         }?
42     }+
43 }*
44 }
45 }
46 p = element p { text }

```

Version 3.0

Appendix D

XQuery Importance Attributes

The XQuery showed in this appendix calculates the attributes used for the importance score.

Listing D.1: XQuery calculating Importance Attributes

```
1 declare namespace pm="http://www.politicalmashup.nl";
2 declare namespace dc="http://purl.org/dc/elements/1.1/";
3 declare namespace js="http://www.jongmansolutions.nl";
4
5 declare function local:wordcount ($arg as xs:string) as
6   xs:integer
7 {
8   count(tokenize($arg, '\W+') [. != ''])
9 };
10 <results> {
11   let $personen := doc('file:///c:/output-intensity/xquery/
12     personen.xml')/personen
13   for $documents in collection("file:/c:/output-intensity?select
14     =*.xml")
15     let $file := substring-after($documents/handeling/metadata/item[
16       @attribuut='Vindplaats']/text(), 'verslagen/')
17     let $aanwezig := number($documents/handeling/metadata/item[
18       @attribuut='presentie']/leden/@aantal)
19     let $bwaanwezig := count($documents/handeling/metadata/item[
20       @attribuut='presentie']/leden/lid[@soort='minister-president
21       ' or @soort='viceminister-president' or @soort='minister' or
22       @soort='staatssecretaris'])
23     let $sluiting := string($documents/handeling/metadata/item[
24       @attribuut='sluiting']/text())
25     let $percaanwezig := $aanwezig div 150
26   for $subject in $documents/handeling/text/onderwerp[not(starts-
27     with(@onderwerp, 'Stemmingen')) and not(starts-with(
28     @onderwerp, 'Regeling'))]
```

```

let $bwcountspr := count(distinct-values($subject//blok/spreker[
    @soort='Minister' or @soort='Staatssecretaris']/@naam))
20 let $block := count($subject//blok)
    let $title := string($subject/@onderwerp)
22 let $scount := count($subject/blok/spreker[@soort!='Voorzitter'
    ])
    let $wcount :=
24     sum(
        for $line in $subject/blok/spreker[@soort!='Voorzitter']//p
        /text()
26     return local:wordcount($line)
    )
28 let $scountincvz := count($subject/blok/spreker)
    let $wcountincvz :=
30     sum(
        for $line in $subject/blok/spreker//p/text()
32     return local:wordcount($line)
    )
34 let $tweedeterminj :=
    sum(
36     for $line in $subject/blok/spreker[@soort='Minister']//
        p/text()
        return
38         if(contains($line, 'tweede termijn')) then
            1
40         else
            0
42     )
    let $premierbonus := count(distinct-values($subject/blok//
    spreker[@naam=$personen/premier/persoon/@naam]/@naam))
44 let $bosbonus := count(distinct-values($subject/blok//spreker[
    @naam='Bos']))
    let $rouvoetbonus := count(distinct-values($subject/blok//
    spreker[@naam='Rouvoet']))
46 let $fvcount := count(distinct-values($subject/blok//spreker[
    @soort='Kamerlid' and @naam=$personen/fractie//persoon/
    @naam]/@naam))
    let $klidcount := count(distinct-values($subject/blok//spreker[
    @soort='Kamerlid']/@naam))
48 return
    <onderwerp>
50     <file>{$file}</file>
        <titel>{$title}</titel>
52     <sluiting>{$sluiting}</sluiting>
        <blokken>{$block}</blokken>
54     <aanwezig>{$aanwezig}</aanwezig>
        <bewindsliedenpotentieel>{$bwaanwezig}</
        bewindsliedenpotentieel>
56     <bewindsliedenspreken>{$bwcountspr}</
        bewindsliedenspreken>
        <percaanwezig>{$percaanwezig}</percaanwezig>
58     <premierbonus> {
        if($premierbonus > 0) then
60         1
        else
62         0
    } </premierbonus>

```

```

64     <vpbonus> {
        if($bosbonus > 0 and $rouvoetbonus > 0) then
66         1
        else
68         0
    } </vpbonus>
70 <wordcountincvz>{$wcountincvz}</wordcountincvz>
    <scountincvz>{$scountincvz}</scountincvz>
72     <avgwordpersprekerincvz> {
        if($scountincvz > 0) then
74         $wcountincvz div $scountincvz
        else
76         $scountincvz
    }
78 </avgwordpersprekerincvz>
    <wordcount>{$wcount}</wordcount>
80     <scount>{$scount}</scount>
    <avgwordperspreker> {
82         if($scount > 0) then
            $wcount div $scount
84         else
            $scount
86     }
    </avgwordperspreker>
88     <kmrsprekers>{$klidcount}</kmrsprekers>
    <fractievoorzitters>{$fvcount}</fractievoorzitters>
90     {
    if($fvcount >= 1) then
92         <prcntfrvz>{$fvcount div $klidcount}</prcntfrvz>
    else
94         <prcntfrvz>0.00</prcntfrvz>
    }
96     <tweedeterminj>{$tweedeterminj}</tweedeterminj>
    </onderwerp>
98 } </results>

```

Bibliography

- [1] Evelyn Beatrice Hall. *The Friends of Voltaire*. Smith Elder & Co, 1906.
- [2] Kamertweets.nl, de plek om te twitteren met kamerleden. <http://www.kamertweets.nl>.
- [3] Polifeeds brengt nederlandse politiek in kaart. <http://www.bright.nl/polifeeds-brengt-nederlandse-politiek-kaart>, 1 2010.
- [4] CBS. ICT gebruik van personen naar persoonskenmerken. <http://statline.cbs.nl>, 2009.
- [5] Aaron Smith. The Internet's role in campaign 2008, April 2009.
- [6] Robert J Klotz. *The Politics of Internet Communication*. Rowman & Littlefield Publishers, Inc., December 2003.
- [7] CBS. Politieke participatie. <http://statline.cbs.nl>.
- [8] Jeremy Bentham. *Panopticon, or the Inspection House*. T. Payne, London, 1791.
- [9] M Foucault. *Discipline and punish: The birth of the prison*. Pantheon Books, 1977.
- [10] An organization running most of the best-known democracy and transparency websites in the uk. <http://www.mysociety.org>.
- [11] Andrew Chadwick and Philip N Howard. *Routledge handbook of Internet politics*. Routledge, August 2008.

- [12] Rita Marcella, Graeme Baxter, and Nick Moore. The effectiveness of parliamentary information services in the united kingdom. *Government Information Quarterly*, 20(1):29–46, 2003.
- [13] T. Gielissen and M.J. Marx. The design of PoliDocs: a Web Information System for the Disclosure of Dutch Parliamentary Publications. In *CEUR*, volume 461, 2009. <http://CEUR-WS.org/Vol-461/>.
- [14] Parlando. <http://parlando.sdu.nl/cgi/login/anonymous>.
- [15] Staten-Generaal Digitaal. <http://www.statengeneraaldigitaal.nl/>.
- [16] Politiek weblog sargasso. <http://www.sargasso.nl>.
- [17] Sargasso.nl. Actie open democratie 1.0 - de brief. <http://www.sargasso.nl>.
- [18] Tim Gielissen and Maarten Marx. Exemelification of parliamentary debates. In *DIR '09: Proceedings of the 9th Dutch-Belgian Workshop on Information Retrieval*, 2009.
- [19] Daniel Tunkelang. Dynamic category sets: An approach for faceted search. In *Workshop on Faceted Search (ACM SIGIR 2006)*, 2006.
- [20] Bill Kules, Robert Capra, Matthew Banta, and Tito Sierra. What do exploratory searchers look at in a faceted search interface? In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries (JCDL 2009)*, pages 313–322, New York, NY, USA, 2009. ACM.
- [21] Ori Ben-Yitzhak, Nadav Golbandi, Nadav Har'El, Ronny Lempel, Andreas Neumann, Shila Ofek-Koifman, Dafna Sheinwald, Eugene Shekita, Benjamin Sznajder, and Sivan Yogev. Beyond basic faceted search. In *Proceedings of the international conference on Web search and web data mining (WSDM 2008)*, pages 33–44, New York, NY, USA, 2008. ACM.
- [22] Emilia Stoica, Marti A. Hearst, and Megan Richardson. Automating creation of hierarchical faceted metadata structures. In *The Conference of the*

North American Chapter of the Association for Computational Linguistics (HLT 2007), pages 244–251. Association for Computational Linguistics, 2007.

- [23] Jonathan Koren, Yi Zhang, and Xue Liu. Personalized interactive faceted search. In *Proceeding of the 17th international conference on World Wide Web (WWW 2008)*, pages 477–486, New York, NY, USA, 2008. ACM.
- [24] Grondwet 1815. <http://www.wetboek-online.nl/wet/Grondwet.html>.
- [25] Amendment definition. <http://www.tweedekamer.nl/applicaties/begrippenlijst.jsp>.
- [26] E. Rahm and H.H Do. Data cleaning: Problems and current approaches. *IEEE Techn. Bulletin on Data Engineering*, 23(4):3–13, 2000.
- [27] Saxon: The xslt and xquery processor. <http://saxon.sourceforge.net/>.
- [28] Gnu wget. <http://www.gnu.org/software/wget/>.
- [29] Pdftohtml. <http://pdftohtml.sourceforge.net/>.
- [30] Example of amendment in xml format dutch parliament. <https://zoek.officielebekendmakingen.nl/kst-31316-10.xml>, indexed august 2010.
- [31] Xsl transformations (xslt) version 2.0. <http://www.w3.org/TR/xslt20/>, 2007.
- [32] Xquery 1.0: An xml query language. <http://www.w3.org/TR/xquery/>, 2007.
- [33] James Clark and MURATA Makoto. Relax ng. <http://www.relaxng.org/spec-20011203.html>, 2001.
- [34] exist-db open source native xml database. <http://exist.sourceforge.net/>.

- [35] Koninklijke Bibliotheek eind op dreef met krantendigitalisering. <http://www.automatiseringgids.nl/artikelen/2010/08/koninklijke-bibliotheek-eind-op-dreef-met-krantendigitalisering.aspx>, September 2010.
- [36] Top search engine - volume. <http://www.hitwise.com/us/datacenter/main/dashboard-10133.html>, June 2010.
- [37] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.
- [38] Jon M. Kleinberg, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew S. Tomkin. *The Web as a Graph: Measurements, Models, and Methods*. Springer Berlin / Heidelberg, 1999.
- [39] Tweede-kamer. <http://www.tweedekamer.nl>.
- [40] Anil Jain, Karthik Nandakumar, and Arun Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270 – 2285, 2005.
- [41] Xiao-Li Meng and Donald B. Rubin. Maximum likelihood estimation via the ecm algorithm: A general framework. *Biometrika Trust*, 80(2):267–278, Jun 1993.
- [42] Sanford Weisberg. *Applied Linear Regression*. John Wiley & Sons, second edition, 1985.
- [43] Christopher M. Federico. The mathematical derivation of least squares. http://www.psych.umn.edu/courses/spring06/federicoc/psy8815/lectures/derivation_ols.pdf.
- [44] Matrix calculus. <http://www.colorado.edu/engineering/cas/courses.d/IFEM.d/IFEM.AppD.d/IFEM.AppD.pdf>.
- [45] Matlab. <http://www.mathworks.de>.

- [46] Matlab - multivariate linear regression. <http://www.mathworks.de/access/helpdesk/help/toolbox/stats/mvregress.html>.
- [47] Apache lucene. <http://lucene.apache.org/>, 2010.

Version 3.0