Focused retrieval and result aggregation with political data

Kaptein, R.; Marx, M.

DOI
10.1007/s10791-010-9130-z
Publication date
2010
Document Version
Final published version
Published in
Information Retrieval

Inf Retrieval (2010) 13:412–433
DOI 10.1007/s10791-010-9130-z

Focused retrieval and result aggregation with political data

Rianne Kaptein · Maarten Marx

Received: 1 May 2009 / Accepted: 3 March 2010 / Published online: 20 March 2010

Abstract This paper presents a case-study in which we use a large semi-structured data set consisting of official transcripts of meetings of the Dutch parliament for focused retrieval and result aggregation. Transcripts of meetings are a document genre characterized by a complex narrative structure. The essence is not only what is said, but also by who and to whom. We have notes of more than 40 years of Dutch parliamentary debates where this structure is exploited to automatically make semantic annotations. These annotations yield numerous new ways of searching, browsing, mining and summarizing these documents. Concerning result aggregation, we summarise and visualise the structure of meetings into tables of content and interruption graphs. The contents of meetings or parts of meetings are condensed into word clouds that are created using a parsimonious language model. Furthermore, we have developed a search engine that exploits the structure and annotations of our data making it possible to provide entry points, to group search results, and to use faceted search techniques for data-exploration. Evaluation shows that our content and structure summarization tools provide a good first impression of a debate. Users reported that, compared to a standard document retrieval system, our search engine gives a better overview of the data. Search tasks are performed faster and the users felt more certain of their answers.

Keywords Political data · Word clouds · Focused retrieval · Result aggregation

1 Introduction

This paper is a case-study. We present a large real-life example of an IR problem whose solution naturally combines semi-structured data, focused retrieval and result aggregation.

R. Kaptein (✉)
Archives and Information Studies, University of Amsterdam, Amsterdam, The Netherlands
e-mail: kaptein@uva.nl

M. Marx
ISLA, University of Amsterdam, Amsterdam, The Netherlands
e-mail: maartenmarx@uva.nl

We describe our experience with building a dedicated search engine for the Dutch parliamentary proceedings. Parliamentary proceedings in general are a very interesting set of documents for IR research, because of the following characteristics:

- large historical corpora; For example, in Holland all data from 1814 will be available in 2010, at the time of writing it is available since 1917; for the Flemish parliament all data since 1971 is available in PDF; the British Hansard archives have all parliamentary minutes since 1803 available in XML.
- documents contain a lot of consistently applied structure which is rather easy to extract and make explicit;
- transcripts of meetings might be accompanied by audio and video recordings, creating interconnected multimedia data (Seaton 2005);
- challenging data integration issues and opportunities (Halevy et al. 2006; Lenzerini 2002; Levy et al. 1996) both within one country (collections from different periods, in different formats, styles, language,...), and across countries (Cross-lingual IR);
- natural corpus for content and structure queries, combining keyword search with XPath navigation and selection (Kamps et al. 2006; O'Keefe et al. 2004);
- natural corpus for search tasks in which the answers do not consist of documents: *expert* or *people search* (Balog 2008), video search (As done in the TRECVID workshop: http://www.-nlpir.nist.gov/projects/trecvid/) and *entry point retrieval* (Sigurbjörnsson 2006).

The collection of Dutch parliamentary proceedings has certain characteristics which make it both possible and necessary to apply focused retrieval and result aggregation techniques in our search engine. These characteristics are general and hold for many collections. While some characteristics pose problems (or challenges), others provide us with opportunities. The problems are closely related to specific IR research areas; the opportunities point to possible solutions. In this paper we describe the concrete implementations of these solutions in our case-study and evaluate them. We start with listing these characteristics and related IR fields, generalizing from our case study.

1.1 Characteristics: large collections with long documents

This case study is based on one type of document, the verbatim proceedings of meetings of parliament. We study proceedings where one document is a written copy of a complete meeting. In principle, every word spoken is in the proceedings, but the texts are edited to make them easy to read. Other common meeting-notes features like lists of present members, outcomes of votes, incoming post, etc. are also part of these proceedings. This format is now used in most countries to document parliamentary meetings (Kirschner et al. 2003). The British Hansard is a prototypical example of these notes, which is copied by many parliaments (http://www.hansard.millbanksystems.com).

Our document collection has two characteristics that are problematic and ask for special purpose IR techniques:

P1: Natural unit of retrieval is smaller than the document.
In several countries (The Netherlands, Belgium, Germany), parliamentary proceedings come as PDF files containing the notes of one whole day. These are long documents, typically between 50 and 80 pages two column PDF. For instance, the Dutch meeting of September 18, 2008 took the whole day and consisted of 624 speeches with a total of 74.068 words. Speeches seem to be natural units of retrieval, they are used for example in

the British parliamentary search engine http://www.theyworkforyou.com. We will also use speeches as the unit of retrieval. Special-purpose IR techniques that are applicable here are focused and entry point retrieval, and document summarization.

P2: Information needs beyond separate documents.

The documents in the collection have many interdependencies. Natural information needs are concerned with named entities like politicians, parties and ministries, whose data is spread over many documents. An example is expert search on politicians (Nusselder et al. 2009). Keyword search alone often gives too many results, these may become comprehensible by grouping and/or summarization.

Related IR aspects are faceted search, result aggregation, and result grouping or clustering.

A system that satisfies the information needs of users searching in this collection needs to be far more than a document retrieval system. Fortunately, there are some characteristics of the parliamentary proceedings that provide opportunities to be fruitfully exploited in IR systems:

O1: Hierarchically modelled documents. The meeting notes have a natural hierarchical organization with a complex but familiar data model. Meeting notes are partitioned into topics, which consist of speeches by persons, possibly interrupted by others.

O2: Containment relation has clear semantic meaning. Many documents satisfy this characteristic, e.g. journal articles and Wikipedia pages as used in INEX (Lalmas et al. 2007). In the parliamentary proceedings the containment relations have a semantical rather than a purely layout interpretation. For instance, interruptions of a speech are modelled in XML as sub-elements of speech-elements.

O3: Elements in the hierarchy have appealing metadata. Every word in the corpus is spoken by a person, who is member of a party, who has a certain function or role while speaking (e.g. chairman, minister of X, spokesman of party X for topic Y, etc). Every word has a time- or date-stamp. Thus the elements in the hierarchy have attributes whose values are (familiar) named entities or temporal references.

1.1.1 How the opportunities help solve the challenges

The first challenge P1 asked for focused and entry point retrieval. The natural hierarchy given by O1 means that text extraction techniques can be applied with high precision and that the collection can be viewed as an XML corpus (Baumgartner et al. 2001; Gielissen et al. 2009). The XML structure provides good entry points to the documents and we can use XML retrieval techniques to implement focused retrieval (Sigurbjörnsson 2006). P1 also asked for document summaries. We can use the XML structure for these summaries, and the fact that the XML hierarchy is natural (O2) makes these summaries useful. We present examples of document summaries using content and structure in Sect. 4.

The second challenge P2 asked for faceted search and result grouping and aggregation. Again the XML structure makes this possible together with the rich metadata available at all element levels (O3).

1.1.2 Use cases: who uses this collection and how?

We have done a user study among a variety of users of the collection of Dutch parliamentary proceedings, and extracted a number of use-cases. We present them here, because they are general and show how users exploit the characteristics of the data.

U1: Precise referencing with hyperlinks. This use-case came from political bloggers who want permanent URLs (permalinks) to very precisely point to interesting parts of official sources. They blog about the parliamentary meetings, and want to provide hyperlinks to the sources (Bennet et al. 2009).

U2: Known item search. Users want to retrieve notes in which a certain specific topic was discussed. This can be the notes of one meeting, or the information need can consist of tracking a topic, e.g. a complete law-creation process (Allan 2000).

U3: Mashups. To get a better overview of the search results, group search results by person or party; make top N lists ordered by some criterion, e.g. year or party.

U4: (Scientific) analysis. Both qualitative (mostly browsing) and quantitative analysis is done. The latter typically involves positive XQueries with additional full-text search support producing output that will be fed into a program for statistical analysis like SPSS. An example is agenda-setting research in political science with research questions like: Who puts an issue on the political agenda, using which media? And then, how does an issue evolve over time? Who takes the lead: do the media follow debate in parliament or is it the other way around? Such questions are being researched in Roggeband et al. (2007) for a period of around 20 years for a number of hot issues like *"Islamic threat and terrorism"*. Roggeband et al. (2007) uses a search engine to get yearly counts of the number of documents about a specific issue. These are then temporally aligned with counts of newspaper articles on the same issue. With the proceedings in richly marked up XML more fine-grained measurements become possible, e.g. list all speeches about topic X; for each speech list the date, the speaker and her party. Afanasiev et al. (2009) operationalizes the agenda-setting research as XQueries on the XML-representations of the proceedings.

1.2 Main contributions

It is hard to discuss focused retrieval and aggregated search in an abstract setting. So we worked out an elaborate large real-life example and described that as a use-case. The example is general enough so that our insights have a wide application.

The paper has two main contributions. First, we provide an example of a true semi-structured corpus that can rather easily be brought into XML form, parliamentary proceedings. Unlike the collections used at INEX, like IEEE or the Wikipedia, the XML elements and attributes have a complex semantics, which can be readily exploited by IR systems. Our second contribution is the combination of structure diagrams and parsimonious language models to create high-level document summaries.

This paper is organized as follows. In Sect. 2 we describe related work. The data in our collection is described in Sect. 3. Sections 4 and 5 are on aggregated search and focused retrieval respectively. In Sect. 6 we evaluate our proposed solutions. Finally, in Sect. 7 we draw our conclusions and present future work.

2 Related work

Our work is rooted in information extraction (Doan et al. 2006) because we first turn unstructured text documents in PDF format into highly structured XML using handcrafted rules consisting of regular expressions (Rahm et al. 2000). The idea of using XML retrieval to implement focused information access comes from Sigurbjörnsson (2006).

The search aspects of our work are based on XML retrieval as done within INEX. For retrieval, we use the language modelling approach implemented in PF/Tijah (Hiemstra et al. 2006). PF/Tijah is a text search system that is integrated with an XML/XQuery database management system and provides support for all needed query types.

Our work on faceted search and the form of the interfaces we describe are influenced by the work of Hearst (Hearst 2006; Hearst et al. 2002, 1996). Different visualizations of debates and arguments are presented in Kirschner et al. (2003).

The first widespread use of tag clouds was on the photo-sharing site Flickr. Other sites that contributed to the popularisation of tag clouds were Del.ici.ous and Technorati. Nowadays tag clouds are often considered as one of the typical design elements in Web 2.0. Tag clouds generated by automatically analysing document contents are referred to as 'word clouds'. Word clouds have for example been generated for the inaugural speeches of American presidents.[1]

Word clouds are a relatively new phenomenon and have been studied scarcely in scientific literature. PubCloud uses clouds for the summarization of results from queries over the PubMed database of biomedical literature (Kuo et al. 2007). Recently, Koutrika et al. (2009) described the use of word clouds for summarising search results into key words to guide query refinement when searching over structured databases. In Dredze et al. (2008) summary keywords are extracted from emails. Common stopwords and e-mail specific stopwords such as 'cc', 'to' and 'http' are removed. Latent semantic analysis and latent Dirichlet allocation outperform a baseline of TF-IDF on an automated foldering and a recipient prediction task. Rayson et al. (2000) proposes a method to compare different corpora using frequency profiling, which could also be used to generate terms for word clouds. Their goal is to discover keywords that differentiate one corpus from another. The algorithm compares two corpora and ranks highly the words that have the most significant relative frequency difference between the two corpora. Words that appear with roughly similar relative frequencies in the two corpora will not be ranked high.

Making governmental and/or political data easily accessible through the internet is a major research area with a lot of ongoing activity. The W3C has a special interest group on eGovernment (http://www.w3.org/2007/eGov/) which encourages governments to publish their data in reusable, linkable, human- and machine-readable formats using open standards such as XML, RDF and Dublin Core (Alonso et al. 2009; Bennet et al. 2009). Independent non-profit organisations scrape governmental websites and create vertical search engines, mashups or appealing visualizations, e.g. http://www.theyworkforyou.com and http://www.capitolwords.or.

3 Description of the data

Notes of a formal meeting with an agenda (e.g., business meeting, council meeting, meeting of the members of a club, etc.) are full of implicit structure and contain many common elements. The notes of meetings with a large historical tradition, like parliamentary debates, are in a uniform format, which fluctuates very little in time. This makes these notes very well suited for semantic annotation efforts. To our knowledge there is at the time of writing no DTD or markup language for meeting notes available.[2]

[1] http://www.readwriteweb.com/archives/tag_clouds_of_obamas_inaugural_speech_compared_to_bushs.php

[2] The British Hansard employs a rather crude XML Schema for the debates in XML format published in 2008. This schema seems under development. Schemas from late 2009 show the use of over 12 different versions.

Transcripts of a meeting contain three main structural elements:

1. The topics: discussed in the meeting (the agenda);
2. The speeches: made at the meeting: every word that is being said is recorded together with (1) the name of the speaker, (2) her affiliation and (3) in which role or function the person was speaking;
3. Non verbal content or actions: These can be:
   - list of present and absent members;
   - description of actions like *applause by members of the Green Party*;
   - description of the outcome of a vote;
   - the attribution of reference numbers to actions or topics;
   - and much more.

The analogy with the structural elements in theatrical drama is striking: scenes, speeches and stage-directions are the theatrical counterparts of the three elements just listed. These are prominent elements in the XML version of Shakespeare's work.[3] The close relation between politics and drama is an emerging theme in political science, see e.g., (Hajer 2005; Hariman 1995).

These structural elements are related as follows:

$$\begin{aligned} \texttt{meeting} &\longrightarrow (\texttt{topic})+ \\ \texttt{topic} &\longrightarrow (\texttt{speech}|\texttt{stage-direction})+ \end{aligned}$$

where `speech` and `stage-direction` contain textual content and metadata in the form of attributes.[4] The first rule states that every meeting consists of one or more topics. The second rule states that every topic consists of at least one speech or stage-direction. The British digitized debates from 1803 till 2004 are available in XML[5] and basically have this structure.

Within the Dutch proceedings however there is an intermediate structural element—the block—, which distinguishes the theatre drama from the political debate. In Dutch parliament, the debate on each topic is organized as follows: each party may hold a speech given by a member standing at the central lectern; other members may interrupt this speech; the chairman can always interrupt everyone. Most often, when all parties had their say at the central lectern, a member of government answers all raised concerns while speaking from the government table and again he or she can be interrupted. In most cases this concludes a topic, but variations are possible and occur (e.g., several members of government speaking or a second round of the whole process).

The *block* is an important debate-structural element because it indicates who is being interrupted. Thus for the Dutch situation the main structure of the DTD becomes

$$\begin{aligned} \texttt{meeting} &\longrightarrow (\texttt{topic})+ \\ \texttt{topic} &\longrightarrow (\texttt{block})+ \\ \texttt{block} &\longrightarrow (\texttt{speech})+ \end{aligned}$$

[3] http://www.metalab.unc.edu/bosak/xml/eg/shaks200.zip

[4] More PRECISELY, the `speech` elements contain mixed content consisting of `stage-direction` and text.

[5] http://www.hansard-archive.parliament.uk/

Here we omitted the stage-directions, which may occur everywhere. If this block structure is not present in meeting notes, then each topic consists of exactly one block.

3.1 Impression of the dutch data

The research described in this paper is done on the proceedings of plenary meetings of the Dutch Parliament, on data from 1965 until early 2009. On average one document contains 51,000 words, is 50 pages long and has a file size of 16.5 Megabyte. Each document represents the meeting notes of one complete day, so on an average day in Dutch parliament some 50,000 words are officially spoken. The daily output in Germany and Belgium is comparable to these numbers. In the Netherlands, on average 140 documents are published in each parliamentary year.

The largest document is 49 MB (151,000 words) and the smallest is just 1 page, with 382 words. At the meeting of this one page document less than half of the Dutch MP's were present and then by law the meeting cannot start. Figure 1 shows the facsimile of a typical page.

3.2 Conversion of the dutch data

A technical description of the transformation from unstructured text to the XML representation for the Dutch data is given in Gielissen et al. (2009). Due to OCR-errors in the input data, this transformation is not always correct. Table 1 from Gielissen et al. (2009) shows the percentage of correctly marked elements for 7 typographical features.

4 Result aggregation

We have chosen to use speeches as the unit of retrieval. One speech without the context of the debate, can be difficult to interpret however. In this section, we describe some result aggregation techniques, which go up from the level of the speech to the level of the block and to the topic level. For example, we summarise all speeches of one person in a meeting.

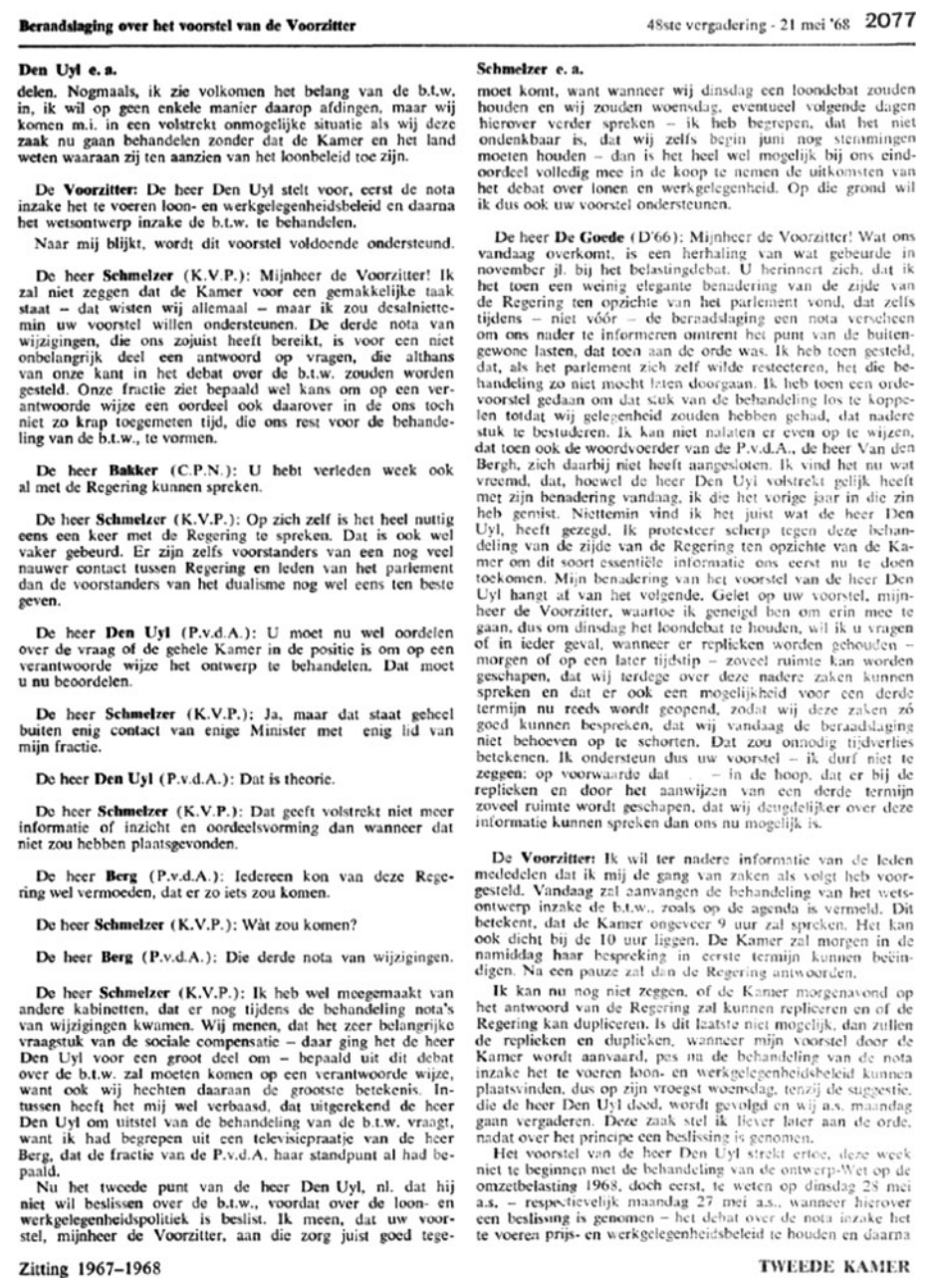
We describe two document structure summarisation techniques and a content summarisation technique. These techniques use the inherent structure of the data to present a meaningful overview of one document. Throughout this section we use an example document that contains the notes of the meeting of the Dutch Parliament of September 18, 2008. This particular meeting took the whole day (from 10.15 till 19.15), consisted of one topic, 11 blocks and 624 speeches with a total of 74.068 words. The notes take up 79 pages 2-column PDF. This is a typical length for a 1 day (8 h) meeting. The block structure described in Sect. 3 is used to break up this large chunk of text.

4.1 Structure summarisation

We have developed two techniques that can visually summarise the structure of meetings. These visualisations give information on the speakers and the relations between the speakers.

4.1.1 Tables of content from XML structure

The hierarchical meeting-topic-block-speech structure can be summarised in a table of contents. This table of contents can be created automatically from the XML structure of the



Berandslaging over het voorstel van de Voorzitter 48ste vergadering - 21 mei '68 2077

Den Uyl e. a.

delen. Nogmaals, ik zie volkomen het belang van de b.t.w. in, ik wil op geen enkele manier daarop afdingen, maar wij komen m.i. in een volstrekt onmogelijke situatie als wij deze zaak nu gaan behandelen zonder dat de Kamer en het land weten waaraan zij ten aanzien van het loonbeleid toe zijn.

De **Voorzitter:** De heer Den Uyl stelt voor, eerst de nota inzake het te voeren loon- en werkgelegenheidsbeleid en daarna het wetsontwerp inzake de b.t.w. te behandelen.

Naar mij blijkt, wordt dit voorstel voldoende ondersteund.

De heer **Schmelzer** (K.V.P.): Mijnheer de Voorzitter! Ik zal niet zeggen dat de Kamer voor een gemakkelijke taak staat – dat wisten wij allemaal – maar ik zou desalniettemin uw voorstel willen ondersteunen. De derde nota van wijzigingen, die ons zojuist heeft bereikt, is voor een niet onbelangrijk deel een antwoord op vragen, die althans van onze kant in het debat over de b.t.w. zouden worden gesteld. Onze fractie ziet bepaald wel kans om op een verantwoorde wijze een oordeel ook daarover in de ons toch niet zo krap toegemeten tijd, die ons rest voor de behandeling van de b.t.w., te vormen.

De heer **Bakker** (C.P.N.): U hebt verleden week ook al met de Regering kunnen spreken.

De heer **Schmelzer** (K.V.P.): Op zich zelf is het heel nuttig eens een keer met de Regering te spreken. Dat is ook wel vaker gebeurd. Er zijn zelfs voorstanders van een nog veel nauwer contact tussen Regering en leden van het parlement dan de voorstanders van het dualisme nog wel eens ten beste geven.

De heer **Den Uyl** (P.v.d.A.): U moet nu wel oordelen over de vraag of de gehele Kamer in de positie is om op een verantwoorde wijze het ontwerp te behandelen. Dat moet u nu beoordelen.

De heer **Schmelzer** (K.V.P.): Ja, maar dat staat geheel buiten enig contact van enige Minister met enig lid van mijn fractie.

De heer **Den Uyl** (P.v.d.A.): Dat is theorie.

De heer **Schmelzer** (K.V.P.): Dat geeft volstrekt niet meer informatie of inzicht en oordeelsvorming dan wanneer dat niet zou hebben plaatsgevonden.

De heer **Berg** (P.v.d.A.): Iedereen kon van deze Regering wel vermoeden, dat er zo iets zou komen.

De heer **Schmelzer** (K.V.P.): Wàt zou komen?

De heer **Berg** (P.v.d.A.): Die derde nota van wijzigingen.

De heer **Schmelzer** (K.V.P.): Ik heb wel meegemaakt van andere kabinetten, dat er nog tijdens de behandeling nota's van wijzigingen kwamen. Wij menen, dat het zeer belangrijke vraagstuk van de sociale compensatie – daar ging het de heer Den Uyl voor een groot deel om – bepaald uit dit debat over de b.t.w. zal moeten komen op een verantwoorde wijze, want ook wij hechten daaraan de grootste betekenis. Intussen heeft het mij wel verbaasd, dat uitgerekend de heer Den Uyl om uitstel van de behandeling van de b.t.w. vraagt, want ik had begrepen uit een televisiepraatje van de heer Berg, dat de fractie van de P.v.d.A. haar standpunt al had bepaald.

Nu het tweede punt van de heer Den Uyl, nl. dat hij niet wil beslissen over de b.t.w., voordat over de loon- en werkgelegenheidspolitiek is beslist. Ik meen, dat uw voorstel, mijnheer de Voorzitter, aan die zorg juist goed tege-

Schmelzer e. a.

moet komt, want wanneer wij dinsdag een loondebat zouden houden en wij zouden woensdag, eventueel volgende dagen hierover verder spreken – ik heb begrepen, dat het niet ondenkbaar is, dat wij zelfs begin juni nog stemmingen moeten houden – dan is het heel wel mogelijk bij ons eindoordeel volledig mee in de koop te nemen de uitkomsten van het debat over lonen en werkgelegenheid. Op die grond wil ik dus ook uw voorstel ondersteunen.

De heer **De Goede** (D'66): Mijnheer de Voorzitter! Wat ons vandaag overkomt, is een herhaling van wat gebeurde in november jl. bij het belastingdebat. U herinnert zich, dat ik het toen een weinig elegante benadering van de zijde van de Regering ten opzichte van het parlement vond, dat zelfs tijdens – niet vóór – de beraadslaging een nota verscheen om ons nader te informeren omtrent het punt van de buitengewone lasten, dat toen aan de orde was. Ik heb toen gesteld, dat, als het parlement zich zelf wilde respecteren, het die behandeling zo niet mocht laten doorgaan. Ik heb toen een ordevoorstel gedaan om dat stuk van de behandeling los te koppelen totdat wij gelegenheid zouden hebben gehad, dat nadere stuk te bestuderen. Ik kan niet nalaten er even op te wijzen, dat toen ook de woordvoerder van de P.v.d.A., de heer Van den Bergh, zich daarbij niet heeft aangesloten. Ik vind het nu wat vreemd, dat, hoewel de heer Den Uyl volstrekt gelijk heeft met zijn benadering vandaag, ik die het vorige jaar in die zin heb gemist. Niettemin vind ik het juist wat de heer Den Uyl, heeft gezegd. Ik protesteer scherp tegen deze behandeling van de zijde van de Regering ten opzichte van de Kamer om dit soort essentiële informatie ons eerst nu te doen toekomen. Mijn benadering van het voorstel van de heer Den Uyl hangt af van het volgende. Gelet op uw voorstel, mijnheer de Voorzitter, waartoe ik geneigd ben om erin mee te gaan, dus om dinsdag het loondebat te houden, wil ik u vragen of in ieder geval, wanneer er replieken worden gehouden – morgen of op een later tijdstip – zoveel ruimte kan worden geschapen, dat wij terdege over deze nadere zaken kunnen spreken en dat er ook een mogelijkheid voor een derde termijn nu reeds wordt geopend, zodat wij deze zaken zó goed kunnen bespreken, dat wij vandaag de beraadslaging niet behoeven op te schorten. Dat zou onnodig tijdverlies betekenen. Ik ondersteun dus uw voorstel – ik durf niet te zeggen: op voorwaarde dat – in de hoop, dat er bij de replieken en door het aanwijzen van een derde termijn zoveel ruimte wordt geschapen, dat wij deugdelijker over deze informatie kunnen spreken dan ons nu mogelijk is.

De **Voorzitter:** Ik wil ter nadere informatie van de leden mededelen dat ik mij de gang van zaken als volgt heb voorgesteld. Vandaag zal aanvangen de behandeling van het wetsontwerp inzake de b.t.w., zoals op de agenda is vermeld. Dit betekent, dat de Kamer ongeveer 9 uur zal spreken. Het kan ook dicht bij de 10 uur liggen. De Kamer zal morgen in de namiddag haar bespreking in eerste termijn kunnen beëindigen. Na een pauze zal dan de Regering antwoorden.

Ik kan nu nog niet zeggen, of de Kamer morgenavond op het antwoord van de Regering zal kunnen repliceren en of de Regering kan dupliceren. Is dit laatste niet mogelijk, dan zullen de replieken en duplieken, wanneer mijn voorstel door de Kamer wordt aanvaard, pas na de behandeling van de nota inzake het te voeren loon- en werkgelegenheidsbeleid kunnen plaatsvinden, dus op zijn vroegst woensdag, tenzij de suggestie, die de heer Den Uyl deed, wordt gevolgd en wij a.s. maandag gaan vergaderen. Deze zaak stel ik liever later aan de orde, nadat over het principe een beslissing is genomen.

Het voorstel van de heer Den Uyl strekt ertoe, deze week niet te beginnen met de behandeling van de ontwerp-Wet op de omzetbelasting 1968, doch eerst, te weten op dinsdag 28 mei a.s. – respectievelijk maandag 27 mei a.s., wanneer hierover een beslissing is genomen – het debat over de nota inzake het te voeren prijs- en werkgelegenheidsbeleid te houden en daarna

Zitting 1967–1968 TWEEDE KAMER

Fig. 1 A typical page of the Dutch parliamentary proceedings. Page 2077 of the meeting of May 21, 1968. Available at http://resourcessgd.kb.nl/SGD/19671968/PDF/SGD_19671968_0000410.pdf (22Mb)

document. For the depth of the table, we choose the block level, because on the speaker level there are too many speeches (e.g., 11 vs. 624 in our running example). This table of contents is handy as a navigation device, but it does not provide much meaning or summarisation.

Table 1 Percentage correctly extracted structural elements

Feature	Score	Comments
Topics	77.8%	All recognized, but in 22.2% included too much text
Blocks	100%	
Speakers	88.7%	Caused by OCR errors
Paragraphs	93.5%	
Header	91.5%	Caused by OCR errors
Footer	92.5%	Caused by OCR errors
Stage directions	73.5%	

Evaluation done on two complete days of proceedings (50 pages)

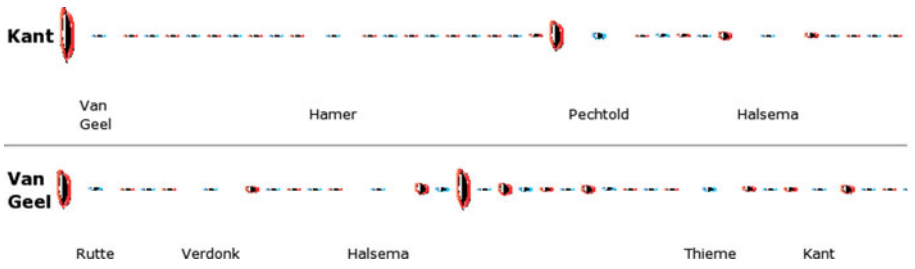


Fig. 2 High-level visualisation of two blocks of a debate

A meaningful structural relation in a debate is given by looking at who interrupts whom. This information is implicitly available in the proceedings because we know who is speaking and who is interrupting the speaker. In the Dutch context these interruptions are usually attacks. A structural summarisation of a debate is given by knowing who interrupts whom, in what order and for how long (Buckingham 2003). We decided to visualise this for every block in the way advocated in Kirschner et al. (2003).

Figure 2 shows the visualisation of two blocks of a debate on a topic. Each row in the figure summarises one block and each mouth stands for one speech. The size of the mouth is proportional to the length of the speech measured in number of words. The speaker on the central lectern (called *Kant* in the first row) has the red mouth, the persons interrupting have a blue mouth. The names of the persons interrupting are printed below the mouths, the first time they interrupt. Interruptions by the chairman are not shown.

For example, we see in the first block that *Kant* starts with a long speech (the first large mouth) and then is first interrupted by *Van Geel*. He interrupts her 5 times in a row, and she answers every time. Then the interruptions are taken over by *Hamer*. Then we see a larger red mouth in the middle. This is an indication that *Kant* has picked up her main argument. The block of *Van Geel* has a similar structure.

We call these structures *debate timelines*. These debate timelines are navigation menus: each mouth contains a hyperlink to exactly that part of the proceedings that records the speech represented by the mouth.

This visualisation can also be used in conjunction with an XML or entry-point retrieval search engine. In such systems users do not want to get the usual linear list of results ordered by relevance, because it distributes entry points to one document over the list (Tombros et al. 2005). Other systems like XMLfind group the entry points per document (Sigurbjörnsson 2006), but then the clear linear ranking is lost. An alternative is to project the results on the visual table of contents using heat-map visualisation techniques.

4.1.2 *Interruption graphs*

The previous subsection contained a summarisation on the block level. Now we present a visualisation that summarises one level higher, a topic. Again we are interested in who is interrupting who, and how often. This information is visualised in an interruption graph, as shown in Fig. 3. The graph gives a high level summary of the structure of the debate.

In the graph, speakers are depicted as nodes, and interruptions are depicted as arrows from the person who is interrupting to the person who is speaking. Each person has a colour, which is used for the node and all interruptions made by that person. Persons with many incoming arrows are interrupted by many other people. Persons with many outgoing arrows interrupt many other people. The size of the arrow is representative for the number of interruptions. So, for example *Kant* interrupts *Hamer* and *Van Geel* most often in this debate, 24 and 18 times respectively.

The graph is constructed using a radial layout, where nodes are placed on concentric circles. The persons in the center and the innermost circle, are the persons at the centre of the debate, who are interrupted by and interrupt the most people. The persons on the outermost circle are the persons who do not participate much in the debate, in this case the smaller parties and the one-issue parties.

4.2 Content summarisation

Besides knowing who is interrupting whom, we also want to summarise the content of the interruptions, as well as speeches and topics. We summarise the content into word clouds. The content itself is not annotated, so we have to extract the most informative and meaningful words from the transcripts. We create word clouds for the following elements in the interruption graph: the complete debate (all text within a topic); for each person, all speeches of that person, all interruptions *by* that person, and all interruptions *of* that person.

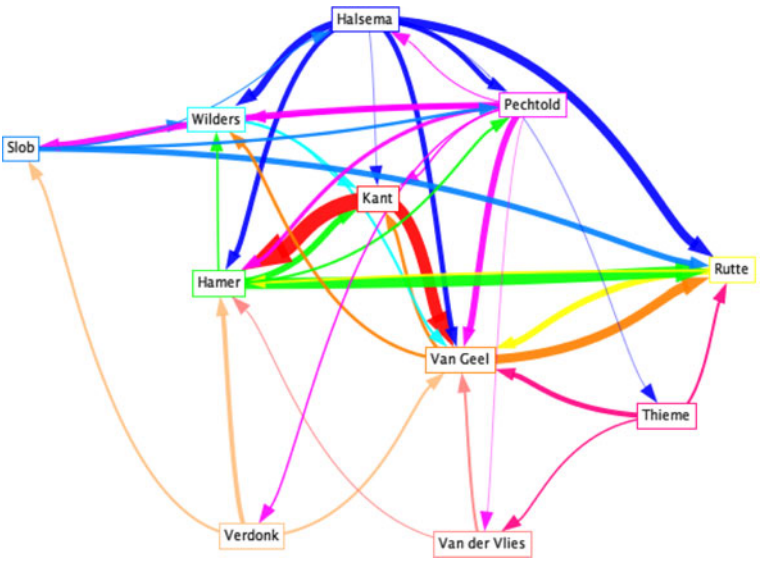


Fig. 3 Interruption graph of a topic consisting of 11 blocks

To create meaningful word clouds, we have to remove the usual stopwords, but we also have to exclude corpus specific stopwords, such as *parliament* and *president*. Furthermore, there are words that will be common and not informative in all interruptions on a certain person, e.g. the name of that person. To filter out all these non-informative words, we use a parsimonious language model (Hiemstra et al. 2004). The parsimonious language model concentrates the probability mass on fewer words than a standard language model. Only terms that occur relatively more frequent in the document than in the background language model will be assigned a non-zero probability. The model automatically removes common stopwords, corpus specific stopwords, and words that are mentioned only occasionally in the document.

Usually the complete test collection is used to estimate background probabilities. In addition here we also experiment with smaller and more focused background collections such as the topic, or all interruptions made by one person. In this way, we will be able to identify words that are used relatively more frequent in a speech or interruption than in the complete debate on a topic. Thereby we can create word clouds that can highlight differences between blocks in one debate.

4.2.1 Unigram word clouds

Unigram word clouds are generated as follows. We use a language model to estimate probabilities and generate word clouds, where we assume that the most probable words are the most informative. First, we collect the text that we want to use for generating the word cloud. For example, to make a word cloud of all speeches of one person in a debate, we concatenate the text of all these speeches. Maximum likelihood estimation is used to make an initial estimate of the probabilities of words occurring in the document.

$$P_{mle}(t_i|S) = \frac{tf(t_i, S)}{\sum_t tf(t, S)} \tag{1}$$

where S is either a speech or a set of interruptions, and $tf(t, S)$ is the text frequency, i.e. the number of occurrences of term t in S. Subsequently, parsimonious probabilities are estimated using *Expectation-Maximisation*:

$$\begin{aligned} \text{E-step: } e_t &= tf(t, S) \cdot \frac{(1-\lambda)P(t|S)}{(1-\lambda)P(t|S) + \lambda P(t|D)} \\ \text{M-step: } P_{pars}(t|S) &= \frac{e_t}{\sum_t e_t}, \text{ i.e. normalize the model} \end{aligned} \tag{2}$$

where D is the background model. In the initial E-step, maximum likelihood estimates are used for $P(t|S)$. Common values for λ are 0.9 or 0.99. We see that when $\lambda = 0.9$, the word clouds contain a lot of very general words and many stopwords. When $\lambda = 0.99$ the word clouds contain more informative words, and therefore in the rest of this work we set λ to 0.99. In the M-step the words that receive a probability below our threshold of 0.0001 are removed from the model. This threshold parameter determines how many words are kept in the model and does not affect the most probable words, which are used for the word clouds. In the next iteration the probabilities of the remaining words are again normalized. The iteration process stops after a fixed number of iterations.

Within a debate on one topic we distinguish the following pieces of text Wikipedia can be used as a background collection:

Fig. 4 Word cloud of the speech by the Animal Rights party leader

- All text
- All speeches made by a person
- All interruptions made by a person on everyone
- All interruptions on a person by everyone

Besides these topic and debate specific pieces of text, we can use all words from all debates in the collection to obtain a general background collection.

More than one background collection can be used to generate word clouds. The most general background collection will remove both common stop words and corpus specific stop words. But to distinguish between the speeches of different persons on the same topic a more focused background collection is needed. We estimate a mixed model with parsimonious probabilities of a word given two background collections as follows:

$$\begin{aligned} \text{E-step: } e_t &= tf(t,S) \cdot \frac{(1-\lambda-\mu)P(t|S)}{(1-\lambda-\mu)P(t|S) + \lambda P(t|D_1) + \mu P(t|D_2)} \\ \text{M-step: } P_{pars}(t|S) &= \frac{e_t}{\sum_t e_t}, \text{ i.e. normalize the model} \end{aligned} \tag{3}$$

There are two background models: D_1 and D_2. D_1 is the model based on the complete corpus. D_2 is the topic specific model. The weight of the background models is determined by two parameters, λ and μ. We want to keep the total weight of the background models equal, so we choose for λ and μ a value of 0.495. Using background models on different levels of generality helps to exclude non-informative words.

An example of a word cloud is shown in Fig. 4. This word cloud is created from a speech of the party leader of the *Animal Rights Party* (represented with 2 out of 150 seats in the Dutch Parliament). As the background collection we use the complete debate. Originally the speech was in Dutch, but we translated the word cloud to English. The translation introduced some bigrams, but in the Dutch original all words are unigrams.

Several tools to create attractive visualisations of word clouds are publicly available. In this example we use the Wordle[6] layout algorithm provided in a visualisation tool of IBM.[7] In this paper, we focus on creating the input to the visualisation, i.e. select words and estimate their probabilities. In the rest of this paper we will visualise word clouds simply as a ranked list of words.

[6] http://www.wordle.net

[7] http://www.manyeyes.alphaworks.ibm.com/manyeyes/

4.2.2 *Bigram word clouds*

In addition to single words, bigrams can be considered for inclusion in the word clouds. Bigrams are often easier to interpret than single words, because a little more context is provided. To create bigram word clouds, the method to create unigram word clouds can be used with some adjustments. A term t now consists of two words. The probabilities of bigrams occurring are estimated using the parsimonious model. To exclude stopwords from the bigrams, we add the restriction that bigrams can only contain words that are present in the unigram parsimonious model. This restriction can be applied either before estimating the bigram probabilities, we call this model the anterior filter, or posterior, after estimating the bigram probabilities, the posterior filter. Since the probabilities of the unigrams and the bigrams are estimated using the same approach, the resulting probabilities are comparable. So, besides creating word clouds consisting of only bigrams, we can create word clouds that are a mix of unigrams and bigrams. As an additional feature, we exclude from the mixed word clouds unigrams that also occur in a bigram.

The word clouds described here can be used in conjunction with the interruption graph, described in the previous subsection, to summarise the content as well as the structure of a meeting. Each element in the graph, nodes and edges, refers to a word cloud that is a summarization of the context belonging to that element. Interacting with the interruption graph will give a quick first impression of a meeting.

5 Focused retrieval

So far, we have concentrated on representing a single debate. Now, we will focus on how to represent search results. The hierarchical structure of the data (characteristic **O1**) and the rich metadata (**O3**) make it possible to provide entry points, to group search results, and to use faceted search techniques for data-exploration.

5.1 Speeches as entry points to a debate

A natural answer unit in a retrieval system for parliamentary debates is the speech. Several parliamentary information systems implement this, e.g. in the UK on the site http://www.theyworkforyou.com and on the site of the European Parliament.

The entries (speeches) in the result list can be rich in information because they have natural metadata (opportunity O3) and the speeches are naturally contained in debates (O2). Figure 5 gives an example from our retrieval system. Each result snippet has as metadata the name of the speaker, his portrait and the logo of his party. Four pieces of

Fig. 5 Answer snippet from result list: photograph of the speaker linking to his bio, logo of his party, a link to the official PDF source, the first 100 characters of his speech and a link to the speech

information about the speech itself are given: a snippet from the speech, the date of the speech, a link to the original source (a PDF file), and the entry point—a link to the speech within the context of the debate.

Our entry-point retrieval system is implemented as a very restricted form of content-only XML retrieval as done in INEX. The user enters keywords and will always receive a list of entry points to a debate. The user is not even aware that she queries XML data.

In our implementation we employ the MonetDB/XQuery XML-database engine (Boncz et al. 2006) with the PF/Tijah IR extension (Hiemstra et al. 2006). Every keyword query *X* is translated into the NEXI (Trotman et al. 2005) query `collection('HAN')//speech[about(.,'X')]`, which returns a list of speech-elements ordered by relevance. From these elements we construct the ranked list of entry points, as given in Fig. 5. We use PF/Tijah with a full-text index covering our complete test collection and we use the Normalized Log Likelihood Ratio (NLLR; Kraaij 2004) with interpolation parameter $\lambda = 0.8$ for scoring and ranking the relevant XML elements. NLLR is a IR language model that measures the difference between the cross-entropy of the query and the collection and the cross-entropy of the query and the retrieval unit (here the speech)—the better a speech fits a query, the higher the NLLR score will be.

5.2 Faceted search

The availability of rich meta-data (opportunity O3) for each unit of retrieval can be exploited for a faceted search interface. For each retrieval unit, we know (1) the speaker, (2) her party and (3) the date. The availability of this information allows us to structure the search results on different dimensions besides relevance.

Figure 6 shows the results for the search term *knettergek* ("bonkers"), which was made popular recently by the Dutch populist politician Geert Wilders. The results are displayed in a standard "ranked list" search environment. Between the search box and the ranked list of answers, the page contains three top five lists. They list (from left to right) the five persons, parties and years, which contain most hits, ordered by number of hits. The layout is based on the Flamenco design for combining keyword search with facet hierarchy navigation (Hearst 2006).

Clicking on the value of a facet refines the search to that respective person, party or year. After clicking, the number of hits are recomputed, and the user can continue drilling down with the other facets. Hierarchical faceted metadata has been found to be a highly understandable data model for search interfaces (Hearst et al. 2002).

5.3 Aggregated search results

A keyword search often returns an overwhelming number of hits. We make an aggregation of search results by grouping these results on a temporal and a political (left–right) dimension, thereby yielding insight and the possibility to drill down the search (Hearst et al. 1996).

In particular when searching for a political theme it is interesting to see how that theme developed over time and which political parties "claimed" that theme. Agenda setting research within political science is exactly about this information need (Jones et al. 2005). The display in Fig. 7 shows the volume of the search results grouped by

– x-axis: the nine parties now in parliament ordered on a left–right scale, with two catch-all categories at both ends of the political spectrum;

Fig. 6 Result page after submitting the query *knettergek* ("bonkers") http://www.polidocs.nl/ResultsBasic.php?zoektermen=knettergek&order=relevance+DESC&limit=5.

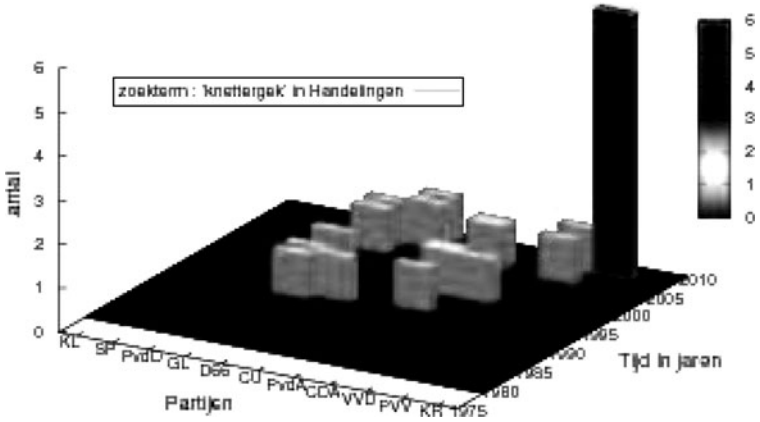


Fig. 7 Number of speeches (y-axis) in the Dutch Parliament about *knettergek* ("bonkers") grouped by political party (x-axis) and year (z-axis). Data used: all speeches in Dutch Parliament between 1980 and June 2008. http://www.zookst2.science.uva.nl/politicalmashup/index.php?query=knettergek&sel=ALL

– z-axis: years (from 1980 till 2008);
– y-axis: the number of results.

Figure 7 shows the results for the search term *knettergek* ("bonkers"). The graph clearly shows the exceptionally high part of the total number of hits attributed to the populist *PVV* party (the high dark bar in the right corner at the back). More interestingly, the graph shows that this particular term has been used already a long time in parliament and by almost all parties.

6 Evaluation

In this section, we evaluate the techniques described in the previous sections. We analyse the different word clouds that we can generate, and describe two small user studies that have been done to evaluate the content and structure summarization techniques and the search engine.

6.1 Analysis of word clouds

We have analyzed the word clouds that are produced by the various methods described in Sect. 4.2. A general problem with the word clouds is that some of the speeches or interruptions are very short, maybe only one sentence. For these short texts we cannot estimate reliable probabilities to create reasonable word clouds. Therefore, we set the restriction that only texts of 100 words or more, and words that occur at least twice will be used to generate word clouds.

6.1.1 Varying the term selection algorithm

First of all, we compare our unigram parsimonious word cloud with two alternative word cloud generation algorithms. The first algorithm is simply frequency counting combined with stopword removal. This technique is usually applied in online word cloud visualisation tools. Secondly, we use a log likelihood model as given in Eq. 4 (Rayson et al. 2000). This algorithm compares two corpora, in our case a specific piece of text and the background collection, and ranks highly the words Wikipedia have the most significant relative frequency difference between the two corpora.

$$\text{Log likelihood} = 2 * \sum_t P(t|D) * ln \frac{P(t|D)}{P(t|C)} \tag{4}$$

Table 2 illustrates the differences between these three ways of creating word clouds. All three clouds were created from the same speech. The parsimonious cloud is identical to the one in Fig. 4. In all our word clouds the Dutch words are translated into English, and originally in Dutch all words are unigrams. As the background collection we use the complete debate.

The frequency count word cloud does not contain many informative words. Although a standard stopword list is used to filter out stopwords, common words like 'how', 'more' and 'goes' are not removed. Words that can be regarded as corpus-specific stopwords like 'parliament' and 'Netherlands' occur very frequently, but are therefore also not informative. The log-likelihood model does retrieve some informative words like 'animals' and 'animal welfare', but also retrieves some stopwords. Our parsimonious model correctly

Table 2 Three word clouds created from the same speech and using the same background collection

Frequencies	Log-likelihood	Parsimonious
Parliament	Animals	Animals
Netherlands	That	Budget memorandum
People	Budget memorandum	Bio
Budget memorandum	Bio	Industry
Animals	I	Animal welfare
Mostly	Animal welfare	Purchasing power
How	Industry	Earth
More	The	Businesses
World	Of	Cattle feed
Goes	Purchasing power	Inv (a Ministery)

removes non-informative stopwords, which still remain in the log-likelihood cloud. Common stopwords can be removed using a standard stopword list as is done in the frequency count model, but these lists are usually not exhaustive. When the parsimonious model is used, no stopword list is needed, and also corpus specific stopwords are removed.

6.1.2 Varying the background collection

In Table 3 we show three word clouds for the same speech but generated using different background collections. The word clouds in the first two columns with background collections 'Test collection' and 'Debate' are generated with a single background collection using Eq. 2. The third word cloud uses a mix of the 'Test collection' and 'Debate' background collections as formulated in Eq. 3. The word cloud of the mixed model contains a mixture of terms from the the first two models plus some new terms.

Which background collection is most appropriate depends on specific use-case of the word cloud. If the unit of study is one complete debate on a topic, and the goal is to discover the themes emphasized by the different speakers, the debate should be used as a background. When studying one specific speaker, it is better to use the complete test collection, or a mixture of the complete test collection and the debate in which a speech is held as background collection. The specific topic of the speeches by that speaker will then be better represented.

Table 3 Word clouds generated with different background collections

Test collection	Debate	Mixed
No	Queen	Sweet
Speech	Throne speech	Throne speech
Defend	Sweet	Defend
President	Tax increases	Care
Care	Sour	Sour
Claim guidelines	Congress	Tax increase
Billion	Strange	Economy
Nursing homes	Collection of poems	Queen
Separate	Defense	Collection of poems
Freedom	Present	Immigration law

6.1.3 Bigrams versus unigrams

We now consider word clouds consisting of unigrams and bigrams. We employ two methods of estimating the probability of a bigram in the mixed model. The methods differ in the moment that bigrams with words that do not occur in the unigram parsimonious model are filtered out. In the "Anterior filter" model, these bigrams are filtered out before the EM algorithm, in the "Posterior filter" model these bigams are filtered out after the EM algorithm. In the model that consists of only bigrams, we also filter out the bigrams with words that do not occur in the unigram parsimonious model. Here it doesn't matter if the filtering is done before or after the EM algorithm. The words that are filtered out are mostly stopwords. The resulting word clouds can be found in Table 4.

The bigram word clouds often contain less than 10 bigrams, because there are simply not enough bigrams in the speeches and interruptions, that occur at least twice, and do not include a word Wikipedia does not occur in the unigram parsimonious model. When bigrams with stopwords are removed using the anterior filter, only few bigrams remain in the model, and these will therefore all get high probabilities. In this example there are five bigrams, which all have higher probabilities than any unigram. On average around 6 bigrams out of 10 places are filled by bigrams. The deviation is large, anything from 0 to 10 bigrams can occur. When the stopwords are removed after the EM algorithm using the posterior filter, the probabilities of occurrence of bigrams are divided over many more bigrams, and therefore the probabilities are smaller. Here only one bigram makes it into the top 10, on average less than 1 bigram will be included in the word cloud. The mixed model with the anterior filter is to be preferred, because it leads to more bigrams being included in the word cloud. The bigrams provide users more context and are therefore good to have in the word cloud. By filtering out the words that do not occur in the unigram parsimonious language model, a basic quality of the bigrams is guaranteed.

6.2 Content and structure summarization evaluation

In addition to the analysis of the word clouds, we have executed a user study to evaluate the unigram word clouds and the interruption graph (Kaptein et al. 2009). The test persons are 20 experts familiar with the Dutch political landscape. First of all we asked our test persons what information they get from the graph that visualizes the structure of the debate

Table 4 Unigram and bigram word clouds

Unigrams	Bigrams	Mixed (ant. filter)	Mix (post. filter)
Claim discount	No claim discount	No claim discount	Turkey
Church	Catholic church	Catholic church	No claim discount
Turkish	Valuable ally	Valuable ally	Halsema
Appoint	Fundamentalist muslims	Fundamentalist muslims	Money
Defense	Chronically ill	Chronically ill	Turkish
Turkey		Turkey	Appoint
Separation		Halsema	Sympathetic
Canossa		Money	Chronically
Brussels		Turkish	Separation
Muslims		Appoint	Canossa

(Fig. 3). The most frequent answers are in general: Who interrupts who, and how often, and who is actively involved in the debate. Thus the users clearly understood the intention of the visualization. More specifically, test persons see that coalition partners do not interrupt each other often and that opposition parties interrupt governing parties. Thus the interruption graph conveyed high level strategical political information, which was caught by the respondents. On the question if the graph gives a good overview of the structure of the debate, we get a score of 3.7 on a 5-point Likert scale, where 1 means strongly disagree, and 5 means strongly agree.

Secondly, we take a look at the word clouds. We generated 12 word clouds of speeches and 17 word clouds of interruptions using the mixture model of Eq. 3 with as background collections the complete test collection and the debate. Each test person was given 3 word clouds of speeches, and 3–5 word clouds of interruptions using a rotation system over the generated word clouds. We asked the test persons whether they think the word clouds are useful summaries of the speeches and the interruptions. The interruptions received an average score of 3.1, the speeches a score of 3.0. This means the test persons do not agree or disagree with the statement. Furthermore, the test persons were asked to judge a number of word clouds of speeches as well as interruptions. For each word in the clouds, they mark whether they think the word is informative or not. We have defined informative as 'a word that gives a good impression of the contents of a speech or interrupt'. It should be both a word 'relevant' to the debate, as well as 'discriminative' for the speaker or part.

Averaged over all test persons and word clouds, 47% of the words in the word clouds of the speeches are considered informative. The standard deviation of average scores between test persons is 13.4, the minimum percentage of informative words per user is 27%, the maximum is 63%. The standard deviation of average scores between word clouds is lower, 8.6. This means that it depends more on the user than on the word cloud how many words are considered relevant. Of the interruptions, on average less words are considered informative, i.e. on average 41%. The standard deviation of average scores between test persons is 15.3, and between word clouds it is 14.0. Since the interruptions are build from smaller pieces of text than the speeches, it is more risky to generate the word cloud since the differences in term counts are small. Some word clouds do not contain any informative words according to our test persons.

Besides the (corpus specific) stopwords, there are many other words that are not considered informative. For example, the parsimonious word cloud in Table 2 does not contain any stopwords, but the test persons consider on average only 58% of the words in this cloud informative. Some of these words would be informative if placed in the right context, but it can be difficult to grasp the meaning of words without the context.

Our final question is whether looking at the graph and the word clouds gives a good first impression of the debate. Most of our test persons agree with this statement, the average score is 3.8. We can conclude that our tools adequately capture the structure and content of the debate at an aggregated level. There is still room for improvement, we have to take into account that the typical user might not have a lot of search experience, and that there is a certain learning curve associated with interpreting this type of information.

6.3 Search engine evaluation

We tested the faceted search engine yielding speeches described in Sect. 5.2 in informal user studies with professional users (political scientists, journalists, and civil servants working in Parliament). A prototype of the search engine can be used at http://www.polidocs.nl.

Concretely, we asked professional users to perform simulated work tasks in two environments: our faceted search engine and the existing search interface provided by the Dutch government (Borlund et al. 1997). These two systems differ in two respects. First, the existing system returns complete documents (in PDF), whereas our system gives entry points to documents (in XML). Second, the existing system combines keyword search with additional restrictions (like document type, year, etc), but does not provide faceted search.

An example of a task for the users was to find the first politician who used the term *knettergek* ('bonkers') in Parliament. Another task concerned topic-tracking: find all debates about the *Joint Strike Fighter* and list the dates.

Users reported they could perform the tasks up to 8 times faster using the entry point retrieval and faceted search interface. They reported that they felt more certain of their answers and that they had a better overview of the data. They appreciated the highly informative answer snippets (Fig. 5) and they found the faceted search interface with the three top-5 lists very useful for exploring the data.

Instead of a controlled user study with students we decided to do a study with experienced professional users. Due to time- and space-constraints on their part this study was more informal and qualitative than can be achieved in a controlled lab-situation. This informal user study may be supplemented with a controlled study measuring user productivity on simulated work tasks as in Wu et al. (2001).

7 Conclusions and future work

We have provided a worked out example of an information retrieval system for a corpus of truly semi-structured documents: the proceedings of the Dutch parliament. We invested in turning these documents into XML (Gielissen et al. 2009) and applied XML retrieval techniques. We have addressed the challenges and opportunities that this collection poses. The hierarchical XML structure together with the rich metadata makes it possible to efficiently implement entry point retrieval, faceted search and several forms of result aggregation and summarization. To address the first problem (**P1**) that the natural unit of retrieval is smaller than the document, we have examined document summarization techniques. Document structure can be summarised visually by tables of contents and interruption graphs. The content of documents or parts of documents can be condensed into word clouds. Together, these tools provide a quick first impression of a debate.

Furthermore, we have developed a search engine that effectively uses speeches as entry points for the usually long meeting notes. To address the second problem (**P2**), information needs beyond separate documents, we have exploited the availability of annotations that identify persons, parties and years to allow for effective faceted search, and aggregation of results on different dimensions. Our evaluations show that it is worth the effort to turn implicitly structured text documents into explicitly structured XML. The XML structure provides many opportunities for creative search engine interface design. These designs naturally blend focused search and result aggregation techniques.

Our results go beyond parliamentary proceedings and are applicable to any vertical search engine with a richly structured corpus. As future work we plan to extend our search engine prototype with full NEXI search functionality. This will create both challenges regarding performance and interface design. Furthermore, we plan to integrate the result aggregation techniques for complete documents into the search interface.

Acknowledgments Rianne Kaptein was supported by the Netherlands Organization for Scientific Research (NWO, grant # 612.066.513). Maarten Marx acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under the FET-Open grant agreement FOX, number FP7-ICT-233599.

References

Afanasiev, L., & Marx, M. (2009). Operationalization of policy framing questions on parliamentary data with XQuery. http://www.politicalmashup.nl/framing-questions-on-polidocs-data.

Allan, J. (Ed.). (2000). *Topic detection and tracking: Event based information organization*. New York: Kluwer.

Alonso, J., et al. (2009). Improving access to government through better use of the web. W3C Interest Group Note 12 May 2009. http://www.w3.org/TR/egov-improving/.

Balog, K. (2008). *People search in the enterprise*. PhD thesis, University of Amsterdam.

Baumgartner, R., Flesca, S., & Gottlob, G. (2001). Visual web information extraction with lixto. In *Proceedings VLDB '01* (pp. 119–128).

Bennet, D., & Harvey, A. (2009). Publishing open government data (W3C Working Draft 8 September 2009). http://www.w3.org/TR/gov-data.

Boncz, P. A., Grust, T., van Keulen, M., Manegold, S., Rittinger, J., & Teubner, J. (2006). MonetDB/XQuery: A fast XQuery processor powered by a relational engine. In *Proceedings SIGMOD* (pp. 479–490).

Borlund, P., & Ingwersen, P. (1997). The development of a method for the evaluation of interactive information retrieval systems. *Journal of Documentation, 53*(3), 225–250.

Doan, A., Ramakrishnan, R., & Vaithyanathan, S. (2006). Managing information extraction: State of the art and research directions. In *Proceedings SIGMOD '06* (pp. 799–800).

Dredze, M., Wallach, H., Puller, D., & Pereira, F. (2008). Generating summary keywords for emails using topics. In *Proceedings of the 2008 International Conference on Intelligent User Interfaces*.

Gielissen, T., & Marx, M. (2009). Digital weight watching: Recreation of scanned documents. In *Proceedings Third Workshop on Analytics for Noisy Unstructured Text Data (AND 2009)* (pp. 25–31).

Gielissen, T., & Marx, M. (2009). Exemelification of parliamentary debates. In *Proceedings of the 9th Dutch-Belgian Information Retrieval Workshop (DIR 2009), Twente, The Netherlands* (pp. 19–25).

Hajer, M. (2005). Setting the stage, a dramaturgy of policy deliberation. *Administration and Society 36*(6), 624–647.

Halevy, A., Rajaraman, A., & Ordille, J. (2006). Data integration: The teenage years. In *Proceedings VLDB '06* (pp. 9–16).

Hariman, R. (1995). *Political style. The artistry of power*. Chicago: University of Chicago Press.

Hearst, M. (2006). Design recommendations for hierarchical faceted search interfaces. In *ACM SIGIR Workshop on Faceted Search*.

Hearst, M., Elliott, A., English, J., Sinha, R., Swearingen, K., & Yee, K. (2002). Finding the flow in web site search. *Communications of ACM, 45*(9), 42–49.

Hearst M., & Pedersen, J. (1996). Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings SIGIR '96* (pp. 76–84).

Hiemstra, D., Robertson, S., & Zaragoza, H. (2004). Parsimonious language models for information retrieval. In *Proceedings SIGIR 2004* (pp. 178–185).

Hiemstra, D., Rode, H., van Os, R., & Flokstra, J. (2006). PF/Tijah: Text search in an XML database system. In *Proceedings of the 2nd International Workshop on Open Source Information Retrieval (OSIR)* (pp. 12–17).

Jones, B., & Baumgartner, F. (2005). *The politics of attention: How government prioritizes problems*. Chicago: University of Chicago Press.

Kamps, J., Marx, M., de Rijke, M., & Sigurbjörnsson, B. (2006). Articulating information needs in XML query languages. *ACM Transactions on Information Systems, 24*(4), 407–436.

Kaptein, R., Marx, M., Kamps, J. (2009). Who said what to whom? Capturing the structure of debates. In *Proceedings SIGIR '09* (pp. 831–832).

Kirschner, P., Shum, B. S., & Carr, C. (Eds.). (2003). *Visualizing argumentation: Software tools for collaborative and educational sense-making*. London: Springer.

Koutrika, G., Mohammadi Zadeh, Z., & Garcia-Molina, H. (2009). Data clouds: Summarizing keyword search results over structured data. In *Proceedings EDBT 2009* (pp. 391–402).

Kraaij, W. (2004). *Variations on language modeling for information retrieval*. PhD thesis, University of Twente.

Kuo, B., Hentrich, Th., Good, B. M., & Wilkinson, M. (2007). Tag clouds for summarizing web search results. In *Proceedings WWW '07* (pp. 1203–1204).

Lalmas, M., & Tombros, A. (2007). Evaluating XML retrieval effectiveness at INEX. *SIGIR Forum, 41*(1), 40–57.

Lenzerini, M. (2002). Data integration: A theoretical perspective. In *Proceedings PODS* (pp. 233–246).

Levy, A., Rajaraman, A., & Ordille, J. J. (1996). Querying heterogeneous information sources using source descriptions. In *Proceedings VLDB '96* (pp. 251–262).

Nusselder, A., & Marx, M. (2009). Expert finding of dutch politicians. In *Proceedings of the 9th Dutch-Belgian Information Retrieval Workshop (DIR 2009), Twente, The Netherlands* (pp. 103–105).

O'Keefe, R. A., & Trotman, A. (2004). The simplest query language that could possibly work. In *Proceedings of the 2nd INEX Workshop*.

Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current approaches. *IEEE Technical Bulletin on Data Engineering, 23*(4), 3–13.

Rayson, P., & Garside, R. (2000). Comparing corpora using frequency profiling. In *Proceedings of the workshop on Comparing Corpora*.

Roggeband, C., & Vliegenthart, R. (2007). Divergent framing: The public debate on migration in the Dutch parliament and media, 1995–2004. *West European Politics, 30*(3), 524–548.

Seaton, J. (2005). The Scottish Parliament and e-democracy. *Aslib Proceedings New Information Perspectives, 57*(4), 333–337.

Shum, S. B. (2003). The roots of computer supported argument visualization. In P. Kirschner, S. B. Shum, & C. Carr (Eds.), *Visualizing argumentation: Software tools for collaborative and educational sense-making* (pp. 3–24). London, UK: Springer-Verlag.

Sigurbjörnsson, B. (2006). *Focused information access using XML element retrieval*. PhD thesis, University of Amsterdam.

Tombros, A., Malik, S., & Larsen, B. (2005). Report on the INEX 2004 interactive track. *SIGIR Forum, 39*(1), 43–49.

Trotman, A., & Sigurbjörnsson, B. (2005). Narrowed Extended XPath I (NEXI). In *Advances in XML Information Retrieval* (pp. 16–40).

Wu, M., Fuller, M., & Wilkinson, R. (2001). Searcher performance in question answering. In *Proceedings SIGIR '01* (pp. 375–381).