Hitotsubashi Journal of Social Studies 49 (2018), pp.29-36. Ⓒ Hitotsubashi University

PRODUCING STRUCTURED PARLIAMENTARY DEBATE RECORDS: THE CASE OF NIGERIA*

JONATHAN LEWIS**

I. *Introduction*

This research note arises from a joint research project that focuses on the way the Nigerian Senate debates incidents of political violence. The project aims to use automatic text analysis to identify conflict-related statements, and also to group statements into clusters that can then be matched with attributes of violent events and individual senators to test hypotheses regarding Senate debates and the frequency and content of violent incidents.

The starting point for such an analysis is a database of individual statements in the National Assembly that can be linked with other data sources such as those about individual politicians and violent incidents. However, the context within which statements are made is also important to their meaning and significance.

This research note outlines the technical issues involved in producing a structured and accurate machine-readable version of the debates from scans of the paper-based originals. In particular, it discusses the pros and cons of using a standard document format, Akoma Ntoso, for the Nigerian transcript data.

II. *Technical Background*

Standard Document Formats

In many spheres of social and economic activity, there is a need to distribute electronic text content in a variety of formats (on paper, to websites, to mobile phones, as audio through text-to-speech applications, etc.). There is also a potential to gain insights and add commercial or other value by combining data from various sources. These pressures have prompted the development of document format standards that allow the separation of document structure from content, formatting and metadata. Standard document formats tend to be written in Extensible Markup Language (XML), and use a Document Type Definition (DTD) to define rules about the structure and elements of the document. Using these standard formats when

* This work was supported by the Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (C) No. 15K03271 (April 2015 - March 2018), and also by the Research Grant for Young Scholars, Hitotsubashi University (June 2014 – March 2015). Thanks are due to my fellow project members, Kazuhiro Obayashi (PI) and Rentaro Iida.

** Institute for the Study of Global Issues, Graduate School of Social Sciences, Hitotsubashi University, Tokyo. jonathan_lewis@mac.com

producing documents avoids the duplication of effort in devising formats, allows for interchange and the use of common publishing and analytic tools, and potentially both extends data longevity and expands the number of users and applications.

The possible downsides to using standard formats are the difficulty of customizing them for particular use-cases, which might require more effort than devising a format from scratch; and the learning curve in mastering the standard format, which may be much more complex and abstract than that required in a given application.

Structured Debate Transcripts

Akoma Ntoso

The publication of debate transcripts, which has its origins in the efforts of 18th century British reporters, has become a standard feature of national and regional legislatures around the world. Many countries now publish their legislative debate transcripts online, reducing production and distribution costs and allowing a wider audience easier and faster access. These transcripts allow the media and ordinary citizens to monitor lawmakers' activities; they are also a rich source of data for political scientists.

Electronic file formats and content management systems have been developed in the government and legal domains to store and distribute documents including debate transcripts. The UK Parliament Hansards, for example, are available in a simple XML format that renders the structure of debates and allows speeches by particular members to be identified and extracted.[1]

The notable initiative to establish international standards in this area is Akoma Ntoso, which defines a set of simple technology-neutral electronic representations in XML format of parliamentary, legislative and judiciary documents.[2] Akoma Ntoso has been adopted by the Organization for the Advancement of Structured Information Standards (OASIS) for its LegalDocumentML standard.[3] The standard is still evolving, and documentation for prospective users remains incomplete, but Akoma Ntoso has been used in the production of debate transcripts in Novia Scotia[4] and Chile.[5] In 2013 the US Library of Congress held a competition for the marking up of selected US bills in the format.[6]

Do-it-yourself markup

From the point of view of the researcher, it would be ideal if the producers of the original transcripts i.e. the parliaments, were to produce a structured version of their debate transcripts using a standard format such as Akoma Ntoso, or at least a generic markup such as the UK Hansards. However, in many cases, including that of Nigeria, such versions are not available. In these cases, the researcher has a human-readable digital text scanned from paper or generated from a word processor but not a file containing the debate structure and content in a

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

[2] http://www.akomantoso.org

[3] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legaldocml

[4] https://github.com/SpringtideCollectiveOrg/openhousens.ca/tree/master/akoma_ntoso

[5] http://library.ifla.org/1048/1/121-cifuentes-es.pdf

[6] https://akoma-ntoso-markup.devpost.com

machine-readable form.

The researcher in such a situation can choose to produce her own structured version of the text. The section below discusses the challenges involved in this process, but for now let us note that considerable resources are required to produce a structured version of debate transcripts. The three main requirements are knowledge of the structure of debates, understanding of structured text markup, and above all time for careful text processing and checking.

Copyright restrictions on the debate transcripts may restrict the researcher's freedom to redistribute marked-up versions, reducing the return on the investment of resources.

III. *Producing Structured Versions of Nigerian National Assembly Debates*

As in many former British colonies, the debate records of the Nigerian National Assembly are called Hansards. Hansards for the Senate dating back to 2000 and for the House of Representatives dating back to 2012 can be downloaded as PDF files from the National Assembly website[7]. For our project, we focused on the Hansards of the Seventh Senate from 6 June 2011 to 21 May 2015; 132 days' transcripts are available online. Each file has on average 18.1 pages of text.

It is clear that even for recent debates, paper remains the primary medium for production of the Nigerian Hansards: the PDF files consist of images scanned from records that have been printed on paper, with the text in two columns on each page.

In our project, we first extracted the text data from the downloaded PDF files, split the text into individual statements, and stored the statements in a PostGreSQL database. We tried to automate the process as much as possible in order to save time and also to reduce the scope for human error.

The resulting database of individual statements permitted us to do some analysis with statements and Senators as the units of analysis, but we lost the contextual information about the particular debate in which a statement was made. This could be important; for example, a simple statement agreeing or disagreeing with another Senator's statement might not introduce any new information, but it could be significant for us if it is made during a debate related to political violence.

We therefore needed to store the statements in a way that preserves the structure of the debates. If we can identify the specific part of each day's proceedings in which a particular statement was made, we can carry out various analyses very efficiently. For example, we might choose to restrict the parts of the debate we include in our analysis.

We decided to use the Akoma Ntoso format to mark up the Nigerian Senate Hansards. By using it we could benefit from the work by its designers, a team of legal and data format experts, and from the several years of feedback from users. Using Akoma Ntoso might also make it easier to do comparative research in the future and to use tools developed in other projects using the same format.

[7] http://www.nass.gov.ng/document/hansards

Challenges in Producing Machine-readable Transcripts

Producing machine-readable transcripts from the Nigerian PDF Hansards involved six tasks: (i) deciding how to render the structure and elements of the debates in XML; (ii) dealing with errors in the original text; (iii) dealing with errors in the PDF files; (iv) generating the structure of the machine-readable file using typographical and other signals in the PDF file; (v) dealing with cases where the original transcript does not have a nested structure; and (vi) storing the data in a format that allows us to analyze it, add links to other data and store our own metadata on it.

The following sections describe the decisions that had to be made and the issues experienced in producing XML versions of the Hansards.

(i) Deciding how to render the structure and elements of the debates in XML

The Akoma Ntoso standard, in keeping with its aim, allows all kinds of metadata, intra-textual information, and links to other texts to be included in documents. This allows those processing the documents to minimize the amount of information that has to be sacrificed when migrating from a paper-based to markup-based format. However, it also requires those performing the migration to decide how much to encode. There is no clear limit to how much markup could or should be done: for example, the Akoma Ntoso standard allows for references to Bills (or versions of Bills) to be added to debates, and Points of Order could be linked to the Senate regulation under which they are raised.

(ii) Dealing with errors in the original text

The original transcripts contain many errors, including spell checking and incorrect paragraph breaks. We face something of a dilemma here. On the one hand, it is important to retain the integrity of the canonical (paper/PDF) version as much as possible, and the dividing line between correction and modification is not completely clear. On the other hand, leaving obvious errors uncorrected could impact the results of automatic text analysis, and seems simply to be willfully perpetuating errors made not by the Senators themselves but by the transcript producers.

Members' names are also written inconsistently in the original, which makes it difficult to match them automatically with a list of members; the Akoma Ntoso standard requires all speakers to be entered as unique entities in the metadata at the beginning of each debate file, so in this case staying with the original unsystematic way of writing names is not an option.

(iii) Dealing with errors in the PDF files

Occasionally the quality of the scan of the original paper document is poor, which in turn produces OCR errors and requires manual re-entry of text. More frequently, the order of words obtained after OCR is wrong, necessitating time-consuming manual correction.

(iv) Generating the structure of the machine-readable file

There are four levels of structure in the Senate debate transcripts; level 1 sections e.g. Prayers, Votes and Proceedings, Announcements, Presentation of Bills, Orders of the Day and Adjournment; level 2 sections, for example the section Orders of the Day might have subsections Consideration of Bills or Motion; level 3 sections such as individual bills or reports being considered, and level 4 sections, such as “Findings”, “Recommendations” etc. in a report.

The typographical signals for Section 1 and Section 2 headers are the same (upper case, centered). This makes it necessary to go through each file and manually mark up the structure of the debate, adding <debateSection > elements as necessary.

The markup also needs to make the structure more explicit than in the PDF version. For example, in the Senate Hansard for 19 July 2012, two reports are presented and considered. The two reports are included in the level 2 Presentation and Consideration of Reports, and the start of the debate on each report is demarcated with a level 3 header (bold text, centered). We need to make the start and end of the debate on each report explicit, resulting in markup like this (NB we have not yet added eid attributes to the elements):

```
<debateSection name="Orders of the Day">
  <debateSection name="Presentation and Consideration of Reports">
    <debateSection name="Presentation and Consideration of Report">
      <heading>Report of the Committee on Judiciary, Human Rights and Legal Matters on
the Screening of...
      </heading>
    </debateSection >
    <debateSection name="Presentation and Consideration of Report">
      <heading>Report of the Joint Committee on Employment, Labour and Productivity...
      </heading
    </debateSection >
  </debateSection >
</debateSection >
```

Not infrequently two or more bills are considered together. A comprehensive implementation of Akoma Ntoso with links to bills would need to work out how to have a single <debateSection> contain the consideration of multiple bills.

(v) Dealing with cases where the original transcript does not have a nested structure

In the process of marking up debate transcripts in XML structure, we sometimes face the problem that the debate cannot be marked up with a nested structure. This is particularly the case with Points of Order. Points of Order pose a number of problems. First, they are not signaled typographically, so they must be identified by reading through the text. Second, while most Points of Order start with a Senator saying “Point of Order, Mr. President”, sometimes that text is omitted and only the President’s response to an intervention makes it clear that we are dealing with a Point of Order (for an example, see 3 October 2012, p.20). In other cases, it is not clear where the Point of Order finishes and the debate is resumed (e.g. 3 October 2012, p.17).

Points of Order can also cause the debate to move from one item to another, and they can be raised before a previous Point of Order has been dealt with. An especially thorny example of this can be found in the Senate Hansard for 1 November 2011. The Senate is discussing a bill to prohibit the illegal distribution of government papers such as laws passed by the National Assembly:

> “**Senator Bukar Abba Ibrahim** (Yobe East):
> ...I also support the idea of leaving only the Government Printer to do what the amendment has asked to do, which is that no person other than the government printers shall print or produce, distribute or sell any Act of the National Assembly.

Senator James Ebiowou Manager (Delta South): Point of Order, Mr. President!

The President: What is your Point of Order, Senator James Manager?

Senator James E. Manager: Mr. President, my Point of Order is predicated on Order 70 (1) (c) of the Standing Orders of this Senate. (...)
This Bill, first and foremost, came by way of a Motion and we debated it exhaustively in this Chamber only few months ago and before that one there was yet another Motion on this same Bill by Senator Victor Ndoma-Egba in the last Senate.
Again, Senator Chukwumerije brought another Motion on this particular Bill. Therefore, since we have exhaustively debated this Bill by way of Motion earlier on and from the three earlier speakers, everybody seemed to have said there is no need debating this Bill exhaustively again.
This is a Bill that is straightforward and we all know what is at stake. That is the reason I am coming by way of this Point of Order. If you would not mind, please rule it in my favour.

The President: I suspend the ruling. I will allow Senator Bukar Abba Ibrahim finish with his contribution.

Senator Bukar Abba Ibrahim: So far, I am in total support of this Bill and the issues raised by my earlier Colleagues. I urge Mr. President to put the question and make progress because the issue is very straightforward.

Senator Datti Baba-Ahmed (Kaduna North): Point of Order, Mr. President!

The President: Point of Order, Senator Datti Baba-Ahmed. But let me rule on Senator James Manger's Order. Your Point of Order is sustained.

Having upheld Senator James Manager's Point of Order let me put the question after which I will take your Point of Order.

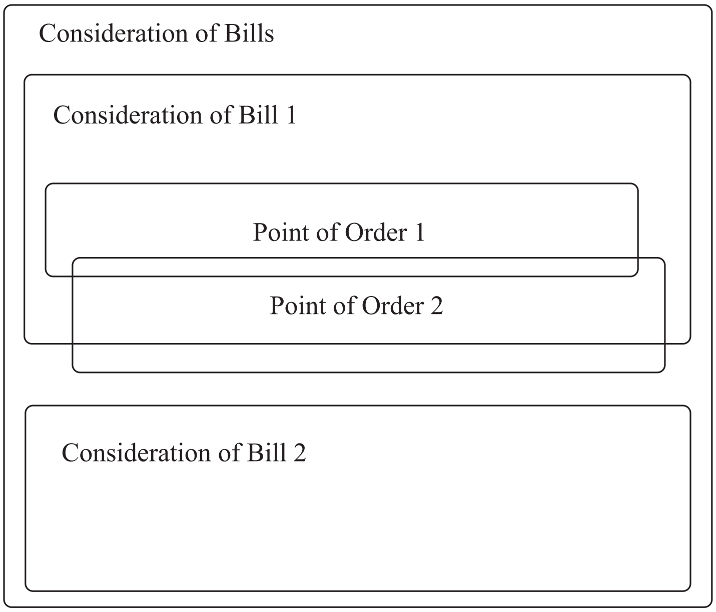
Question put and agreed to.
(Bill read the Second Time and referred to the Committee on Judiciary, to report in two weeks)

The President: Go ahead with your Point of Order, Senator Datti Baba-Ahmed.

Senator Datti Baba-Ahmed: Mr. President, my point of Order was meant to be Order 55 (2) but the last action of Mr. President has given a technical knock-out to the matter. ..."

It is impossible to mark this text up in a nested structure for two reasons. First, the second Point of Order is raised before the first one has been dealt with, but is only dealt with after the first Point of Order has been finalized. Second, the second Point of Order is raised during the debate on the bill, but is only dealt with after the debate has finished (see Figure 1). There are no very satisfactory solutions that remain faithful to the original transcript while satisfying the requirement for XML elements to be nested within each other. One option might be to split the second Point of Order into two, one contained within the first Point of Order and one after the bill debate has finished. That would result in valid XML, but at the cost of causing possible confusion, either by creating two <pointOfOrder> elements where only one was made, or by

FIGURE 1. EXAMPLE OF SENATE HANSARD TEXT NOT FITTING A NESTED STRUCTURE



allowing the same <pointOfOrder> element to exist in two places in a document. Another possibility would be to massage the statement order so that the second Point of Order occurs immediately after the end of the debate bill; but then the whole meaning of that Point of Order would be lost.

While these considerations may seem pedantic and trivial, they show that the task of converting paper-based debate transcripts to the international standard machine-readable format is not a straightforward mechanical operation. While Assembly debates generally conform to a nested structure, sometimes they are more free-flowing. When the marked-up transcripts are being used for research purposes and not redistributed, the researchers can make decisions to omit or move text. However, if the producers of official debate transcripts were to adopt Akoma Ntoso, they would have to decide how best to impose a strictly nested structure onto semi-structured but organically flowing exchanges.

(vi) Working with Data stored in Akoma Ntoso Format

Currently we are marking up our data in Akoma Ntoso format, producing an XML file for each debate. We still need to establish the best way to store this data for analysis purposes. The volume of text data is not so large, and the processing does not need to be very fast, so keeping the data in XML files rather than in an indexed database may be efficient. When we run an analysis, data can be extracted using an XML library such as BeautifulSoup[8]. However, we need to decide how best to store the metadata that we will generate for our own analysis. For example, we will want to label particular debates, and possibly particular statements and

[8] https://www.crummy.com/software/BeautifulSoup/

particular paragraphs, as relevant to political violence; the obvious place to store this metadata is as attributes on the <debateSection >, <speech> or <p> elements in question. If our original data was perfect and we were 100% confident in the quality of our markup, we could make working copies of the debate files and add metadata to them. However, it is possible we will need to perform more data cleaning and improve our markup, resulting in confusion if we have two versions of every file. Probably the best interim solution is to add our own metadata to our "master" versions, but to give our attributes names that make it clear they are related to our current project. When we are confident of the quality of the data and the markup, we can save "canonical" versions stripped of our project-related metadata, for future research and possible exchange with other researchers.

IV. *Conclusions*

This research note has outlined the benefits of using the Akoma Ntoso mark-up format for parliamentary debate transcripts from the point of view of the researcher. Based on experience of marking up transcripts of debates in the Senate of the Nigerian National Assembly, it has shown that while much of the structure of the debates can be generated using typographical hints in the downloadable paper-based versions, a large amount of manual coding needs to be performed. It has also shown that debates do not conform fully to the nested structure required in XML.