This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

VisInReport: Complementing Visual Discourse Analytics through Personalized Insight Reports

Rita Sevastjanova, Mennatallah El-Assady, Adam Bradley, Christopher Collins, Miriam Butt, Daniel Keim

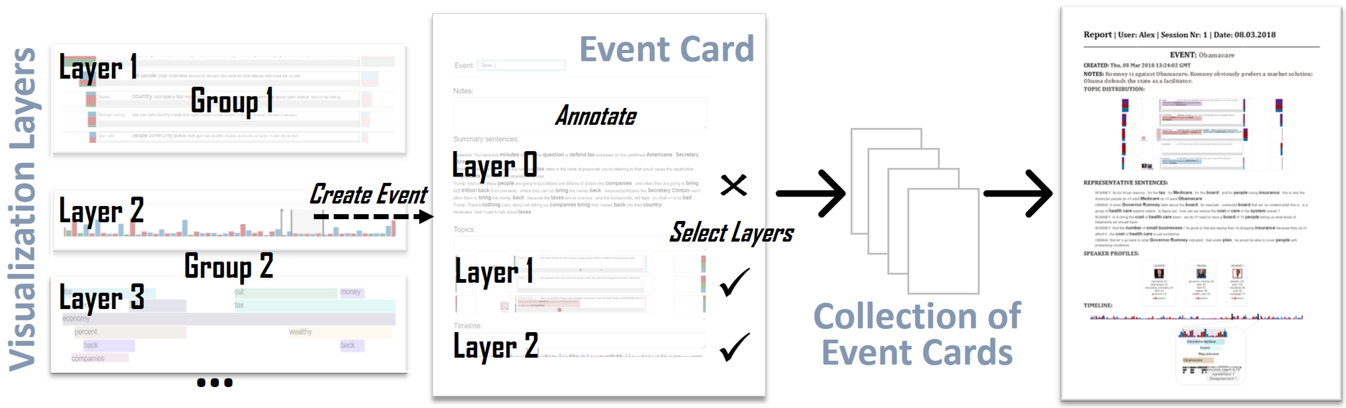


Fig. 1: To create a report, the user first explores the different visualization layers, extracts events, and specifies the relevant information for each of them to be included in the report. The collection of all event cards created in an analysis session is combined in a single insight report and can be exported as a PDF document.

Abstract—We present VisInReport, a visual analytics tool that supports the manual analysis of discourse transcripts and generates reports based on user interaction. As an integral part of scholarly work in the social sciences and humanities, discourse analysis involves an aggregation of characteristics identified in the text, which, in turn, involves a prior identification of regions of particular interest. Manual data evaluation requires extensive effort, which can be a barrier to effective analysis. Our system addresses this challenge by augmenting the users' analysis with a set of automatically generated visualization layers. These layers enable the detection and exploration of relevant parts of the discussion supporting several tasks, such as topic modeling or question categorization. The system summarizes the extracted events visually and verbally, generating a content-rich insight into the data and the analysis process. During each analysis session, VisInReport builds a shareable report containing a curated selection of interactions and annotations generated by the analyst. We evaluate our approach on real-world datasets through a qualitative study with domain experts from political science, computer science, and linguistics. The results highlight the benefit of integrating the analysis and reporting processes through a visual analytics system, which supports the communication of results among collaborating researchers.

Index Terms—Visual Analytics, Text Analysis, Report Generation, Visualization, Verbalization.

1 Introduction

One default type of verbal discourse is characterized by turn-taking among two or more speakers. This type of discourse can include rapid exchanges of ideas and opinions, leading to frequent shifts in the topics of the discussion. In our work, we have been concerned with discourse taking place in contexts of political argumentation. Researchers within political science working with such discourses tend to be interested in some of the following aspects: which topics are discussed, what vocabulary is used [1], how the argumentation evolves, to what degree it is deliberative, are speakers using the indirectness and other strategies of description [2]. Linguists tend to focus more on understanding the structure and use of language and thus concentrate on particular features of the discourse such as the type and amount of turn-taking, the precise morphosyntactic realization of the propositions that have been uttered, and the overall rhetorical structure of the discourse. While such discipline particular analysis differs, there is a cross-disciplinary need for tools that help investigate the context of specific linguistic features and the individual use of language. Visual analytics has the potential to help in analyzing such data by applying and combining computational and visualization methods [3].

Besides the computational and visualization methods for data examination, an effective exchange of the gained insights with colleagues as well as collaborators is essential for an effective analysis outcome. The exchange of insights might not only evoke new discussions, but also enable verification of the findings in situations when there is high data uncertainty. According to Mathisen et al. [4], often, if the analyst cannot share the findings with stakeholders or other analysts, the analysis process can be meaningless. Besides academic blogs or academic papers, reports are a

- *R. Sevastjanova, M. El-Assady, M. Butt, and D. Keim are with the University of Konstanz, E-Mail: firstname.lastname@uni-konstanz.de*
- *A. Bradley and C. Collins are with Ontario Tech University, E-Mail: firstname.lastname@ontariotechu.ca*

form of 'output' used for such a knowledge exchange [5].

The report generation is a time-consuming process, especially if done manually, and requires (1) the search for interesting regions within a discourse and (2) the extraction of relevant information for the research question at hand. Before reporting the findings, (3) analysts might take notes, summarizing more specific observations obtained during the analysis and integrating their domain knowledge in the final report (Figure 2). By collecting the user's findings during the analysis in an automatic manner, the reporting process would become more efficient. To support the analysis and report generation through a visual analytics system, one needs to consider that although the analysis process for different domains (e.g., political science, linguistics) is similar, the generated reports have to be user-targeted and task-specific due to the various research questions at hand. More specifically, a system that automates this process will need to solve three main requirements: (1) be granular in its interactions to allow for multiple disciplines to use it effectively; (2) to generate custom reports that mimic the work of the analysts; (3) to add efficiency to the processes.

With all of these requirements in mind, we present *VisInReport*, a visual analytics tool for interactive data analysis with the focus on report generation of discourse transcripts. We introduce the concept of *visualization layers*, which we define as a visual representation of one or more discourse components (utterances, speakers, time) at varying granularities. Each visualization layer is relevant for a specific task (e.g., topic analysis, feature correlation analysis), displaying one particular aspect of the data. These layers help in identifying regions of interest in the discourse by showing the distribution of linguistic features and give clues to the contextual development of the discussion. VisInReport not only allows analysts to find the relevant aspects of the discourse they are investigating, but easily combines the annotations into an automatically generated insight report. The report summarizes and organizes the analysts' work in a way that allows researchers to easily save and share the gathered insights. The system adds to existing processes by utilizing two types of information: data- and user-driven. The data-driven events are generated automatically and rely on a series of linguistic features. For discourse analysis, we include information about the topical changes within a discussion (e.g., topic-shift, topic recurrence), as well as argumentation features (e.g., consensus, denial, reason-giving) and rhetorical features such as projection of a common ground, leading questions or rhetorical questions. User-driven events are created by the analysts and can be used to verify a generated hypothesis. Due to a modular representation of the data, it is simple to adjust the report to the users' needs and research questions. All the events created in an analysis session are combined into a PDF file, which can be used for further research and knowledge dissemination.

To summarize, this paper contributes VisInReport, a tool for interactive text analysis that supports data and user-driven event creation and generates reports outlining the explored events in an automated manner. To evaluate our tool, we conducted a qualitative study using real-world datasets with domain experts from political science, computer science, and linguistics. The results highlight the benefit and efficiency of tightly integrating the analysis and reporting processes through visual analytics, which supports the communication of results among collaborating researchers.

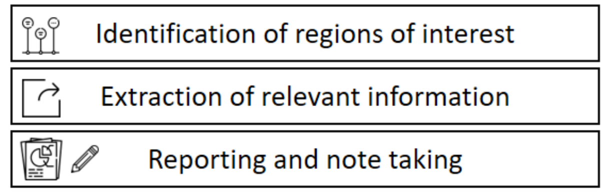


Fig. 2: Three main requirements gathered from stakeholders are the identification of relevant regions of the discussion, information extraction, and report generation.

2 Related Work

In this section we describe existing methods for visualizing discourse transcripts by identifying three main groups: content, time-aligned, and speaker interaction visualizations. Additionally, we look at techniques to extract representative sentences from a text document, and methods to generate reports automatically.

2.1 Revealing Text Content

Exploration and discovery in a large text document requires investigation at multiple levels of abstraction, from meta information about the document to individual utterances and words. LDA [6] topic modeling is the most frequently used algorithm to extract abstract *topics* in a text document. It decomposes a collection of documents into topic distributions where each document is represented by a weighted subset of topics.

FacetAtlas [7] and TopicPanorama [8] use a graph structure to visually display topics and their descriptive keywords. The graph structure shows the main content of the data. FacetAtlas employs a multifaceted graph visualization to show local keyword relations and a density map to portray global cross-document relationships. TopicPanorama supports analysis of relevant topics discussed in multiple sources, such as news, blogs, or micro-blogs. It shows common topics covered by multiple sources, as well as distinctive topics from each source.

Many other types of visualizations have been used to display topic distribution within a document corpus. Serendip [9] addresses the challenges of scale and multiple information sources; it uses probabilistic topic models to structure exploration through multiple levels of inquiry (e.g., corpus level, passage level, keyword level). Conversation Clusters [10] present detected topics as clusters of words grouped together creating a visual summary of a discussion. Chaney et al. [11] display topics using bar representations. They "rank the topics by their relative presence in the corpus and display each in a bar with width proportional to the topic's presence score." [11]

Named entities in a conversation have also been used to describe the content of a transcription. For instance, NEREx [12] is an interactive visual analytics approach for the exploratory analysis of discourse transcripts based on named entity relations.

2.2 Visualizations of Time-Aligned Text

Multiple visualization techniques exist to show text data in a time-aligned manner. Time-aligned representation is important when visually displaying the contents of a discourse, as it can present topic changes over time and guide the user to interesting temporal regions of the discussion. Dou et al. [13] write that "visualizing topic trends is one of the many benefits of combining interactive visualization with topic models." These visualizations support tasks such as the evolution of topic trends over time and the emergence of individual topics [13].

Several time-aligned text visualizations use topic modeling results by applying a stream graph visualization introduced in the ThemeRiver system [14] to represent topic development over time. This concept has been used in several text analysis implementations. TIARA [15] derives time-sensitive keywords to depict the content evolution of a topic over time. TextFlow [16] expands this method and visualizes not only topic evolution trends, but also the critical events, and the keyword correlations. TopicFlow [17] visualizes the results of topic alignment between sets of tweets over time. In addition to topic clusters, Conversation Clusters [10] uses a thread history visualization to show a historical overview of salient topics in discourse over time. HierarchicalTopics [18] uses the Topic Rose Tree [19] algorithm to generate a hierarchy of topics and show their temporal evolution using the Hierarchical ThemeRiver technique.

Another example of time-aligned visualizations are Lexical Episode Plots [20]. Lexical Episodes are word sequences where a single word has a higher density than expected in the whole text document.

Recurrence Plots [21] can reveal trends and features in complex time series data. This information visualization technique works by measuring the similarity of points in a time series to all other points in the same time series and plotting the results in two dimensions. Angus et al. [21] apply this technique to plot the conceptual similarity between pairs of text utterances using a matrix representation. Another visualization method to plot recurrences is the arc diagram introduced by Wattenberg [22]. This technique is capable of representing complex patterns of repetition in string data. Multiple authors have used this technique for representing relationships between text elements in documents. For example, Kerr [23] used arc diagrams to visualize message relations (reply-relations) within a thread.

Heat maps are frequently used to show values of high-dimensional data, including textual patterns [24]. Multiple works use heat maps to illustrate occurrence and frequency of text patterns [25], [26]. The former example uses a heat map to show the distribution of normalized tag counts per text represented as bands of color. The latter uses a heat-map representation to show differences between opinions of customer feedback data.

Kucher et al. [27] in their tool uVSAT plot time-series data fetched from blogs and forums as line charts to display stances (e.g., sentiment terms and their frequencies).

2.3 Speaker Interactions

Beyond the development of topics, each turn taken by a speaker, which leads to changes in topics is relevant for many analyses. For example, ConToVi [28] supports the exploration of the dynamics of a conversation over time. The authors use Topic-Space Views to track the movement of speakers across the thematic landscape of a conversation. South et al. [29] summarize speaker behavior in a political debate, visualizing discussed topics and speaker interaction.

Frequently, several aspects of data (e.g., content, temporal feature changes, speaker activity) need to be analyzed in order to answer a single question (solve a single task). Thus, several visualizations need to be observed in parallel. The interface we present here, VisInReport, addresses this need by combining multiple visualization techniques into a single window enabling an efficient and flexible interface suitable for different analysis tasks. Soto et al. [30] present a tool ViTA-SSD, which is composed of multiple panels each aggregating a set of related functionalities. Although the general idea to represent the data in multiple panels is similar to ours, they all are statically integrated into the main view; thus, in contrast to our tool, the user is not able to flexibly arrange her workspace. Furthermore, this tool does not support discourse analysis.

2.4 Summary Generation

Allahyari et al. [31] have provided a brief survey on text summarization techniques. One well-known approach to generate a text-summary is the extractive method, which selects some linguistic units (e.g., words, sentences) from given documents [32]. The second approach to generating an automatic text summary is the abstraction method; its aim is to interpret and examine a document using NLP techniques in order to generate a new shorter version that conveys the most critical information from the original text [33], [34]. Summary generation can be performed on the single document or at the corpora level. Many existing approaches for single document summarization use statistical methods such as TF-IDF [35], linguistic approaches (e.g., n-grams, noun phrases), or a combination of both types. Also, machine learning techniques like Naive-Bayes [36], SVM [37], and HMM [38] are suggested to improve extractive summarization [31].

2.5 Report Generation

Commercial data-analysis tools such as Tableau, Looker, Leximancer, and Zoho privilege data summarization and reporting, since they both are integral to the business analytics process. Few papers in the data visualization domain talk about report generation. One example is Avocado [39], an interest-driven adaptive approach to provenance visualization which allows users to communicate complex multi-step analyses. Mathisen et al. [4] present a conceptual design for integrating reporting of data insights into visual analytics processes and enabling collaborators to interact with the report components at different levels of detail.

Interactive visual analytic activities often make it difficult for the user to capture the steps of the analytic process and understand how a particular insight was discovered or why a decision was made. Analytic provenance captures the interactive data exploration and human reasoning process, to support sensemaking [40]. Provenance can be seen as "reflection-in-action" during analysis which helps trace data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

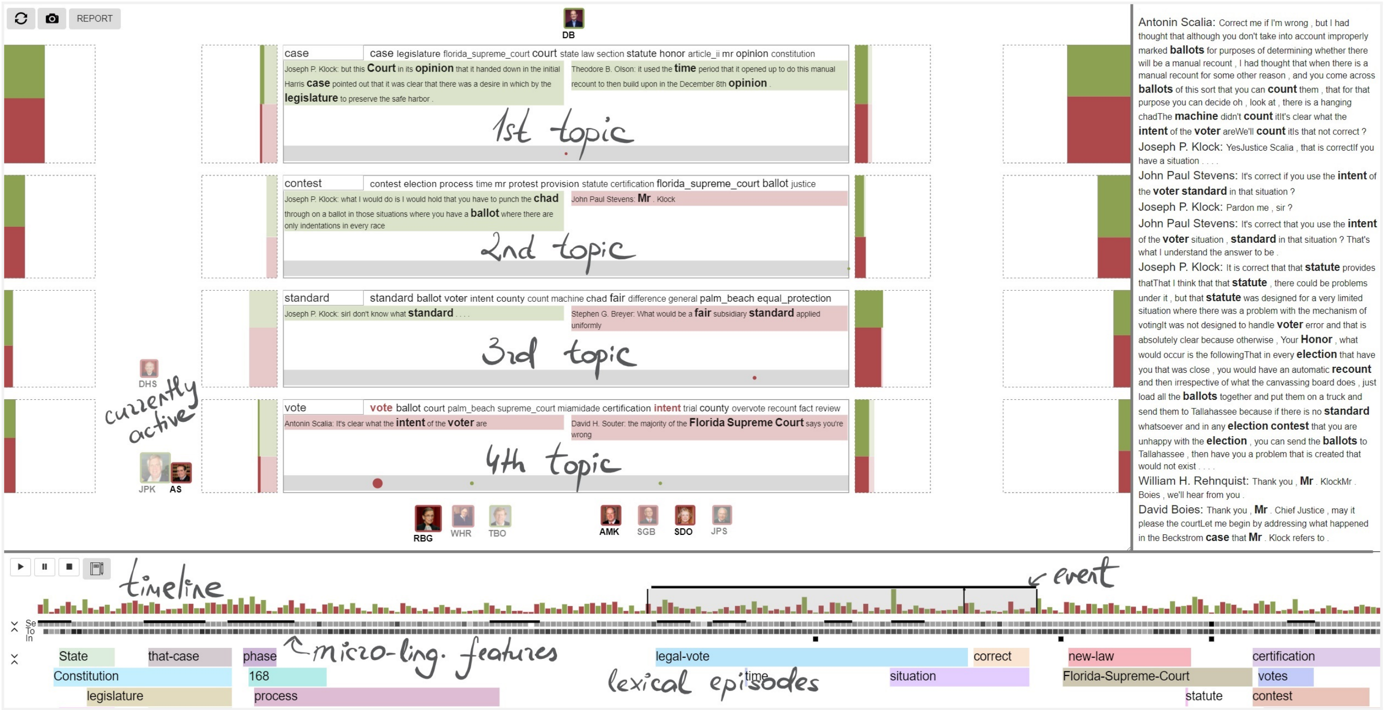


Fig. 3: The *VisInReport* interface (annotated for clarification) showing three re-sizable views containing a selection of *visualization layers*. The layout was arranged by a user to utilize most of the available space for the exploration of the (animated) speaker activity visualization. The analyzed US-supreme-court dataset of the case *Bush vs. Gore* consists of two speaker groups; **Advocates** and **Justices**. In the selected region of the discussion (also marked as an event), speakers discuss the *legal-vote topic*. The animation shows three active speakers and, in particular, Justice Antonin Scalia commenting on how to deal with the improperly marked ballots.

quality and uncertainty throughout the process. Our tool can be used for analyzing provenance, allowing the user to specify which information is relevant and, thus, should be included in the report.

Our reports are based on events detected in the discourse transcript. According to Hogenboom et al. [41], event extraction combines knowledge and experience from a number of domains, including computer science, linguistics, data mining, artificial intelligence, and knowledge modeling. It can be seen as the extraction of complex combinations of relations between actors (entities). For instance, Dou et al. [42] propose an interactive visual analytics system, LeadLine, to automatically identify events in the news and social media data. Our system includes the results of topic modeling in similar ways adding more micro-linguistic features covering a broader spectrum of analysis. TimelineCurator [43] uses a similar representation of events to ours, where the events are arranged on the x-axis. However, similar to LeadLine tool, the analysis is based on news article data. Thus it lacks a speaker analysis component.

3 System Design

Our tool aims at supporting scholars in exploring multiple aspects of discourse data. For instance, the users should be able to analyze the evolution of arguments and topics as well as interactions of different micro-linguistic feature. The found insights should be summarized in a report. Currently, the identification of relevant parts of a discourse and in-depth exploration of the content of these regions tend to require extensive and time-consuming manual effort. The main goal of our tool is to aid users in this process. During our long-term collaboration with political scientists and computational linguists, we identified several requirements for a visual analytics solution supporting discourse analysis and report generation. The requirement analysis included several informal interviews with two PhD students (one from each discipline) concerning their typical workflow of discourse data analysis. We support the following requirements gathered from stakeholders: (1) relevant region identification, (2) exploratory analysis, and (3) report generation. Due to the differing research goals across disciplines, there was a need to create a flexible interface that can be arranged based on user needs. Hence, the created views are abstract visual representations, each encoding a specific granularity of the data. They allow us to show the content of the discourse, speaker activity, various micro-linguistic features and other characteristics of a discourse independently from one another. It is a simple task for the analyst to arrange their work space to make it more appropriate for their specific task. To this end, we present these modular views to the user calling them visualization layers.

Some of the designed visualization layers share similar characteristics; hence, we divide them into three main groups: (1) *content visualizations*, (2) *speaker profile and activity visualizations*, (3) *time-aligned visualizations* (shown in Figure 4). As all of the components of these groups are generated as an independent entity, each group can be extended easily by adding a new visualization layer to the existing ones. The size of each view is variable and fully controlled

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

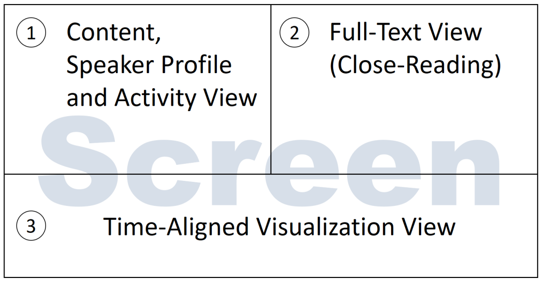


Fig. 4: The three groups of visualization layers that share similar characteristics are displayed separately, in user-adjustable views.

by the user. The **content**, **speaker profile**, and **activity view** shows visualization layers presenting the overview of the topics detected using a topic modeling algorithm (by default LDA) and arranges images of speakers in relation to their utterances across time. An animated visualization is used to present the dynamics of a discourse and the speaker interactions across its thematic landscape. To allow for comparisons over time, we show the past, present, and future of the topic distribution for distinct regions of a text. Additionally, we present speaker profiles containing a summary of their frequently used topic keywords and the average sentiment for the selected discourse region. The **full-text view** gives access to the complete discourse transcript, supporting manual analysis methods and hypothesis testing. The **time-aligned visualization view** includes visualization layers which show different perspectives of utterances displayed in relation to time.

4 VISUALIZATION LAYERS

To create a broad and flexible tool for report generation which supports different analysis tasks of researchers from multiple disciplines, we introduce the concept of visualization layers. A visualization layer is an abstract representation of a discourse which shows one or multiple discourse components (e.g., utterances, speakers, time) at a time. Our tool contains various visualization layers (a selection of these layers is shown in Figure 3); some of the layers are already familiar from other work, e.g., the Lexical Episode Plots [20] or Conceptual Recurrence [21] visualization. In the following, we list components available within our tool.

4.1 Components of a Discourse

A discourse transcript differs from traditional text documents (e.g., news articles) in multiple aspects. Not only does a discourse contain a high level of noise (e.g., errors occurring in the transcription process), making it hard to process, it also contains multiple components (e.g., utterances, speakers, and time) which all may be relevant to an analysis. Together these components enable a view of the underlying patterns from different linguistic perspectives.

Utterances One defining feature of a discourse transcript is that it involves multiple participants who engage in speaker turns. For the purposes of analysis, it is important to observe each speaker's turn (utterance) separately to expose patterns over time and to provide information on specific features such as interruptions. Utterances can also be analyzed as a single document collection and be used as input for a topic modeling algorithm.

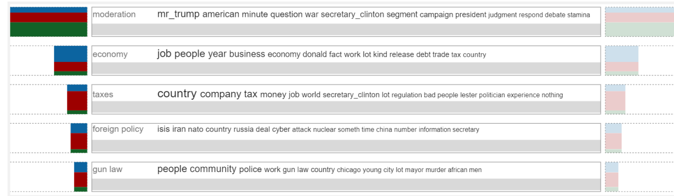


Fig. 5: *Topic Bars* show five topics from the first US presidential debate between Trump and Clinton in 2016; topics are extracted using the LDA algorithm. The user has labeled the topics manually. Colored bars show the topic distribution over the whole discussion. Each colored stacked bar shows the distribution between utterances said by **Clinton**, **Trump**, **Moderator** for the respective topic.

Speakers Another important component for analysis is information about the individual speakers. The analysis of utterances in relation to the speakers helps demonstrate how each of them influences the discussion's flow and topical changes. Furthermore, it is important to see not only what speakers are saying, but also what their overall opinion (sentiment) is about the topic at hand.

Time The third main component in a discourse is time. The position of an utterance in time plays an important role for the analysis, i.e., it is important not only to find out which topics have been addressed in a discourse, but also when they occur the first time and how they develop over time. For these reasons time is included as a controllable function.

4.2 Combination of Components into Layers

The tool has one central visualization layer – a *timeline* of utterances. On this layer we use a brushing function controlled by clicking and dragging the mouse to enable the selection of a distinct discourse region; the selected part of the discourse is displayed in the content visualizations.

Content Visualizations By applying a topic modeling algorithm, we are able to explore the content of the analyzed discourse. The user can choose between multiple topic modeling algorithms (e.g., LDA, Incremental Hierarchical Topic Modeling [44], and Biterm Topic Modeling [45]) for topic extraction. For a discourse transcript, we handle a single utterance as a separate document. The user can specify which features (e.g., distinct POS tags, named entities, stop-words) should be excluded from the keyword vectors. The output of the topic modeling algorithm is a topic distribution for each utterance, where each topic is represented by a list of descriptive keywords and their probability scores.

We use a similar idea to Chaney et al. [11] and show the extracted topics in a horizontal *topic bar visualization* except we apply a slightly different mapping to encode the size of the topic within the bar. A single topic representation consists of three components: a rectangle on which we place the fifteen most significant topic keywords, and two rectangles in each horizontal direction, where the width of the colored bar encodes the topic's size (shown in Figure 5). Each of the colored bars can be seen as a separate stacked bar, showing the distribution of speakers or speaker parties (as defined by the user) for the particular topic. The topics are displayed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

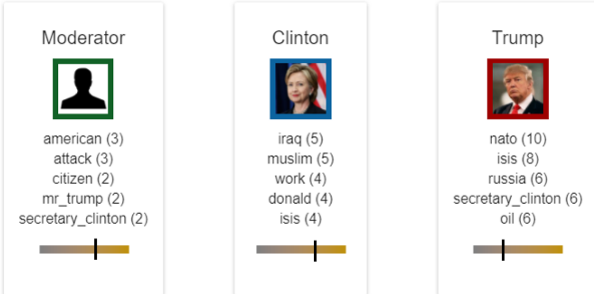


Fig. 6: *Speaker profiles* show the average sentiment and the most frequent keywords for the selected discussion's region. Here, **Trump** has a relatively negative sentiment while discussing the *ISIS* topic.

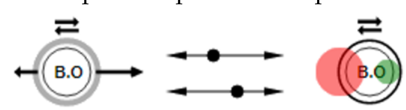
according to descending size. We scale the keywords to their relative frequency in the utterances of the particular topic. Depending on the visualization's size, the first x fully visible keywords are displayed in the visualization. Before we decided to use this visual representation, we tried out multiple other keyword layouts like the WordCloud [46] representation. We faced two problems though: (1) a WordCloud requires space which is limited in the used visualization especially if multiple topics are extracted; (2) the position of the words in a WordCloud don't have a meaning. The second limitation is crucial, as it influences how the user perceives the topic. Thus, we placed keywords in a single row sorted to their frequencies. To simplify topic recognition, we allow the user to manually label the topic.

Another visualization layer to show content information is an automatically extracted *summary of representative sentences*. This provides insights into the context of the topic keywords. The five most representative sentences are extracted for the observed discussion's region. We score sentences according to the topic keywords they contain and their significance scores as retrieved from the topic modeling algorithm. The sentence with the highest sum of significance scores is seen as the most valid representative sentence for the particular utterance.

Jänicke et al. [24] state that close reading is a fundamental method in text analysis applications and Cheema et al. [47] write that even in projects that use a distant reading technique it is important for analysts to actually see the underlying text sources in order to verify hypotheses and to build trust in the research approach. Thus, the last visualization layer for content representation is the *full-text view*, which supports the close reading method.

Speaker Profile and Activity Visualizations Another group of visualization layers which has relevance for analysis are speaker profile and activity visualizations. We provide one visualization layer showing the speaker profile information. In this layer, each speaker is represented by a *speaker's profile card* (Figure 6). These cards show general information like a speaker's name and his profile image with a border colored to indicate his party. We also display relevant topic keywords with their frequencies, sorted according to descending order. Additionally, we display the average sentiment (extracted using OpenNLP Sentiment Analysis algorithm[1]) of the observed utterances, using a slider visualization and a bipolar color scale.

For a more detailed representation of topic changes and speaker interactions, we use an *(animated) speaker activity visualization* (Figure 7). This is a complementary layer to the previously described topic bar visualization. Before the animation is started, we update speaker positions, topic bar width, and keyword sizes with respect to the selected utterance region. If the user has selected a distinct discussion's region (not the whole discussion), we display two additional bars for each topic. The bar on the left-hand side shows how frequently the topic has been discussed in the discourse before the selected region. The bar on the right-hand side shows how frequently the topic has been discussed in the discourse after the selected region. The mirrored bars in the middle show the topic distribution for the selected region. This concept has been recently presented for visualizing focus and context in time-based charts [48]. Speakers who have not talked in the discourse region before the selection are placed on top of the topic bar visualization. Speakers who have already spoken are placed underneath the topic bar visualization. This provides the first overview of speaker activity in the discussion until the first selected utterance. When the animation is started, we simulate the discussion's flow by highlighting active speakers and displaying their said utterances one by one. In order to show speaker activity, we use a "stage" metaphor. Speakers who actively participate in the conversation, spend more time on the "stage" than those who are less active. The currently active speaker  "walks" to the front of the stage while talking and takes a position which is next to the topic bar having the highest probability for the particular utterance; other speakers are faded out  and they "move" backward. In the meantime, a glyph representing the current utterance is created and placed on a gray "track" on top of the particular topic bar.



The final version of the utterance-glyph design is a result of an iterative process. The intermediate results were glyph representations, having multiple attributes encoded in different visual elements (e.g., arrows or circles were used to encode topic recurrence features). We collected qualitative feedback from three PhD students in computer science and figured out that due to the rapid movement of the elements, it was hard to capture the encoded information. Thus, we decided to show the utterance as a simple circle, by mapping the word-count to its radius and the color to its speaker or its speaker party's color. To highlight interruptions in the text, we display a vertical line for utterances which contain only one or two words. In addition to the glyphs, we highlight the topic keywords for each utterance and update their font-size to show their relative frequency in the currently observed utterances. To provide more context, we extract representative phrases by splitting the previously described representative sentences into lexical components separated by a comma. For each phrase, a new significance score is calculated using the same method as for the representative sentences. When the animation is played, we update representative phrases which are displayed on the topic bar, underneath the topic

1. https://opennlp.apache.org/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

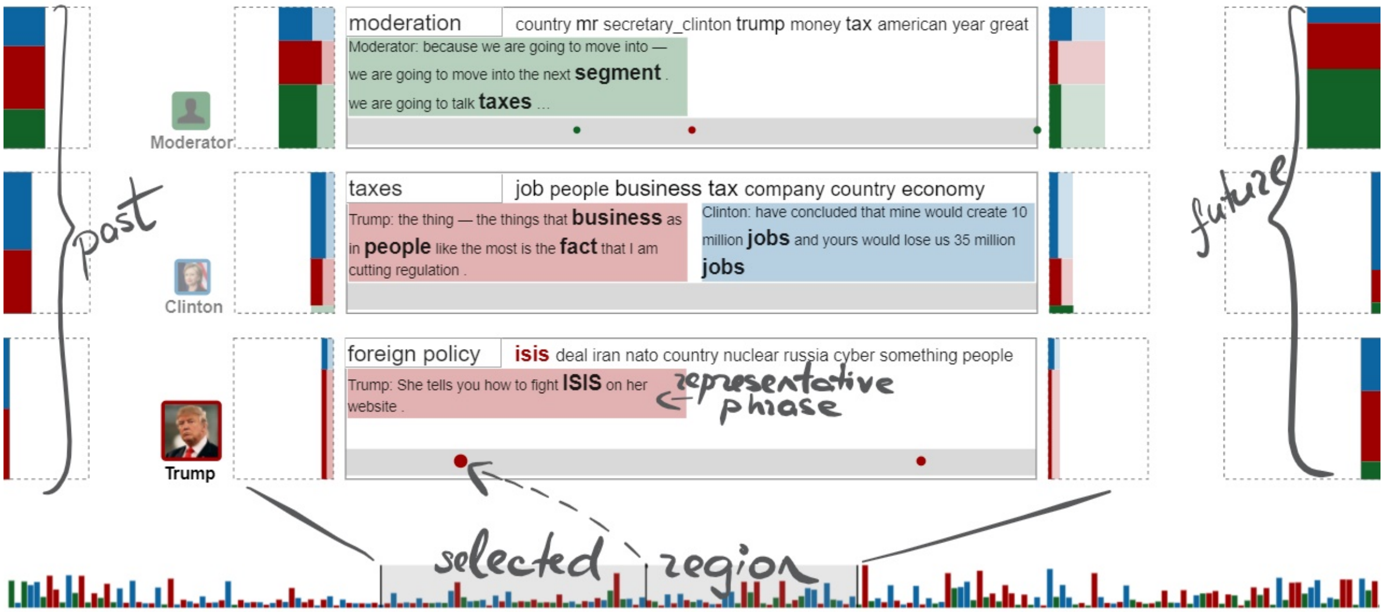


Fig. 7: *Animated speaker activity view* (annotated for clarification) showing an excerpt from the first US presidential debate between Trump and Clinton in 2016. Although the presidential candidates are discussing *tax-policy*, **Trump** suddenly brings up the *ISIS* topic, saying that "She [**Hillary Clinton**] tells you how to fight ISIS on her website."

keywords. At most one phrase with the highest significance score for a single utterance is shown and two phrases from one topic are displayed next to each other at a time.

Additionally, we update the mirrored bars placed on both sides of the topic bar which represent the topic summary for the selected discussion's region as well as displaying the discussion's flow. For each said utterance, the width of the mirrored bar of the speaker/party on the left-hand side is reduced. When the utterance glyph reaches the end of the gray "track", the width of the bar of the speaker/party on the right-hand side is increased appropriately.

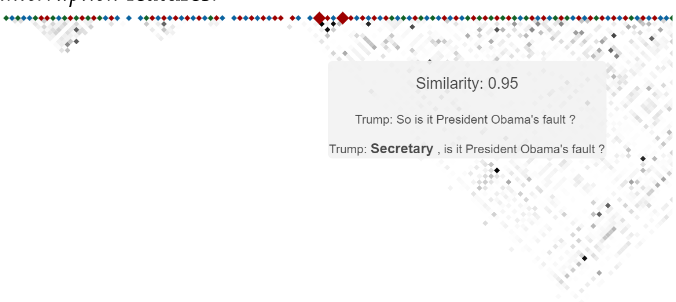
Time-Aligned Visualizations Time-aligned visualizations are another relevant group of visualization layers. In these, each utterance is treated as a single instance of the discourse. We arrange the time-aligned visualizations horizontally and stack them on top of each other. First, we create a visualization layer using a bar chart representation which has the functionality of a *timeline* (Figure 8a). We display each utterance as a bar of the same width, with the utterance's word count mapped to the bar's height. The user can define whether the bar's color should represent the speaker, or his party. A black pointer shows the relevant part of the discourse in the full-text view via mouse over. The bar chart visualization is displayed on top of all time-aligned visualizations. The order of other visualization layers from this group can be changed by the user, specifying their point of interest. We use the timeline visualization to allow the user to select a specific region of the discourse. The selection is performed by a brush function. The content visualizations are updated, depending on the selected utterances.

Heat maps (Figure 8b) are frequently used to visualize high-dimensional numerical data. We provide a list of micro-linguistic features [49] for the selection and enable the user to interactively generate a heat map of their interest. We extract the features with a combination of statistical, linguistic, and machine learning techniques; the feature values can be either boolean or numerical. The flexible and robust design aims to support easy additions of new features, each visualized in the appropriate time-aligned layer. Our tool supports several types of micro-linguistic features, such as **Argumentation Features**, **Question Types**, and **Topic Modeling Features**. For an effective analysis of the deliberative quality of communication, linguistic and statistical cues are important. Thus, we provide a list of argumentation features such as *agreement* and *disagreement*. Altogether, we include 53 linguistic features, grouped into four categories, which can present the dynamics of deliberation (see [50] for reference). These features are extracted via the parsing pipeline developed by Hautli-Janisz et al. [51]. Via rule-based methods we extract different types of questions [52]. Information seeking questions (ISQ) genuinely seek an answer and include constituent questions (e.g., *what, who, where*), yes/no questions and alternative questions (*or*). Non-information seeking questions (NISQs), on the other hand, are employed to express a range of different functions, many of them signaling particular attitudes or assumptions by the speaker. Furthermore, we extract multiple topic modeling features such as *topic shift* (someone changes the topic), *topic persistence* (someone keeps insisting on a topic), and *self recurrence* (someone stresses a topic) [53]. They are descriptive for topic modeling analysis, highlighting themes in the text. The extraction of these features is based on topic distribution over utterances/speakers.

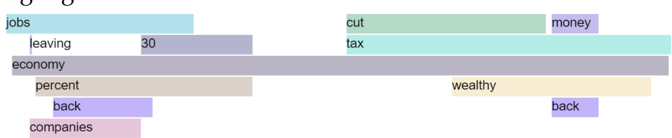
The user can select features for analysis; a row of squares is added to the existing heat map, showing the values of the selected feature for each utterance. The layout is similar to other time-aligned visualizations, enabling the analysis of correlations between different layers. For features having boolean values, we display a square in the particular row if the feature is present in the utterance. Due to the usage of color variable for multiple visualizations, we show the heat

(a) Selection in the *utterance timeline*.

(b) *Feature heat map*. Displayed are *sentiment*, *topic shift*, and *interruption* features.



(c) *Similarity matrix*, where two similar repeating questions are highlighted.



(d) *Lexical episode plots* highlight discussion's regions where word *jobs*, *cut*, *money* occur more dense than expected.

(e) Lexical episodes in a collapsed representation to reduce the *information overload* if this visualization is not relevant for the analysis.

Fig. 8: Time-aligned visualizations are arranged horizontally, underneath one another. This allows the detection of correlations between features and the content of different visualization layers.

map in a gray scale.

The *similarity matrix* (Figure 8c) is taken from Angus et al. [21], who used it to represent conceptual recurrence. In contrast to the specification by Angus et al., we use the matrix to show utterance similarity calculated using a cosine similarity function [54], which was a requirement from the domain experts. To make the visualization more compact, we extract groups of neighboring-utterances which occur between topic shifts with a minimum threshold 0.5 (this is a heuristic and could be changed) and display the similarity only for the neighboring utterances. The representation is a rotated matrix-like visualization where the similarity is mapped to the opacity of the entry. It enables the user to detect utterances which contain similar content and are repeated in the discussion multiple times. By hovering over a data item, the particular utterance pair is highlighted and the utterance text is displayed in a tooltip.

While we have used existing visualizations, we have adapted several parameters. For example, we use the idea of Lexical Episode Plots [20], but unlike the original representation, we align the visualization horizontally to fit it our timeline (Figure 8d). We also represent the lexical episode bars in distinct colors. As we already use the color variable to keep speakers or their parties separate, we reduce the level of opacity, distinguishing these colors from the primary ones. We place the lexical episode keywords on top of the particular bars. This, and other time-aligned visualizations may be collapsed in order to decrease their spatial and perceptional influence. The user can collapse visualizations which are not of importance in their analysis; these layers are automatically excluded from the event described in the following section. An example of a collapsed lexical episode visualization is shown in Figure 8e.

5 Insight Reports and Event Generation

Within the VisInReport tool, insight reports are compiled from individual annotated *events*. Due to the frequent and rapid changes of topics, and the spontaneous reactions of speakers, all utterances of a discourse do not have the same relevance for an analysis. Thus, it is important to enable the user to define events of interest and to analyze these regions separately. This is done using the slider selection on top of the utterance timeline, described in the previous section. For each selected region, the user can create an event where all relevant findings and observations can be stored. Each user-selected event is automatically stored in the final insight report. Additionally, the system automatically extracts event suggestions based on the selected micro-linguistic features which can be added to the final report as well.

For each event, the analyst can specify which information and which visualization layers the event should contain. The flexible creation of events enables the user to individualize the report depending on her tasks and interests. Users can analyze data on different levels using different visualization layers; they can update already created events or delete them if they are not of interest anymore.

The event generation is done in two steps: the user selects a subset of utterances by using a brush function over the utterance timeline presented in Figure 8a. An event for the selected region is created by a click on an event icon . Then, an *event card* is displayed in which the automatically verbalized summary for the selected utterance region, an overview of the speaker' profiles, screen shots showing the topic distribution, and time-aligned visualization layers are displayed as a single event's summary. In addition to the automatically generated components, the user can extend the event by note-taking, which is an important building block in the report generation. Users can name the event and add notes describing the findings; they can specify which visualizations are relevant for the particular event; others can be removed. After the particular event is saved, an *event line* (black line) is displayed on top of the utterance timeline highlighting the particular region. This helps to recognize which regions of the discourse have been analyzed.

Our tool supports the generation of two types of events: data-driven and user-driven events. The data-driven events are based on micro-linguistic features. Patterns of the feature values point out interesting regions of the discussion. The user-driven events are created based on users' hypotheses.

A report is generated as follows: For each detected event, the system stores the information of the time it has been generated, the title of the event, and textual descriptions (notes and automatically created summary) which the user finds relevant for the final report. We use the dom-to-image[2]

2. https://github.com/tsayen/dom-to-image

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

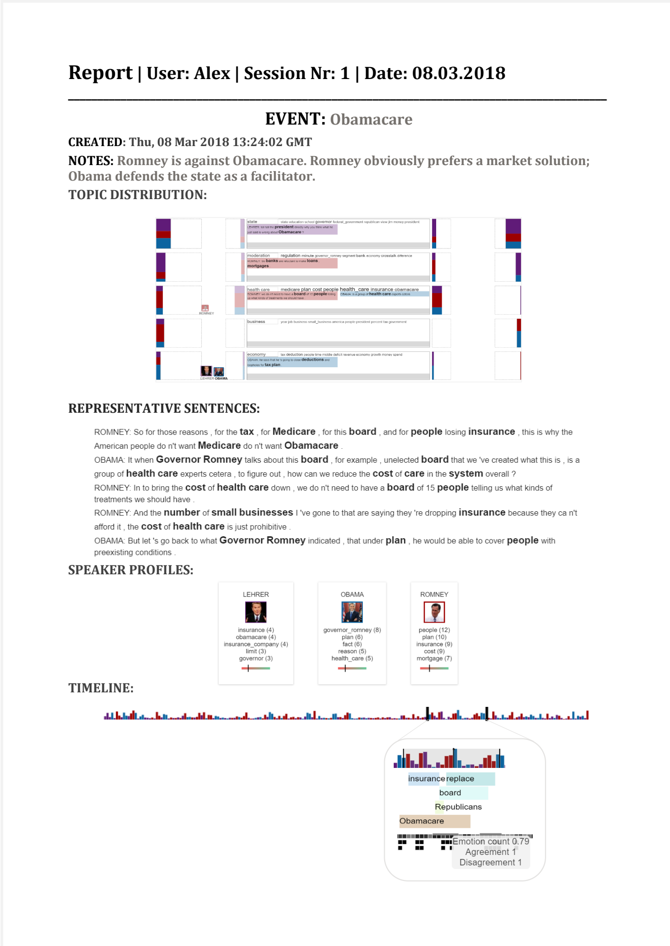


Fig. 9: An example of an event created by the political scientist during the user study, based on the detected lexical episode *Obamacare* in the second US presidential debate between **Obama** and **Romney** in 2012. The user has named the event and added observations as notes. Additionally, all relevant textual (e.g., representative sentences) and visual (e.g., speaker profiles, animated speaker interaction visualization, utterance timeline, lexical episode plots, and feature heat map) visualization layers were added to the event card. The event was part of the final insight report.

library to create images of the visualization components, and arrange them vertically. As previously described, each event is associated with a specific time frame (i.e., a subset of utterances). To help the users recall specific characteristics of the generated events, the system creates a screen shot of each time-aligned visualization for the particular time frame and adds them to the event card. In particular, the system obtains the coordinates from the selected time frame in the *timeline* visualization and uses these coordinates to *cut* the screenshots of the remaining time-aligned visualizations to create callouts showing the specific feature values (shown in Figure 9). Furthermore, a snapshot of the animated speaker activity visualization representing the selected time frame is also added to the event card. The visual components are added to a predefined HTML template; afterward, the angular-save-html-to-pdf[3] library is used to generate a PDF file which combines all created events.

3. https://github.com/hearsid/ng-html-to-pdf-save

Data-Driven Events Using micro-linguistic features, we automatically detect and highlight interesting patterns in the data. Currently, a simple heuristic is used, where we check if at least five subsequent utterances have a low or a high value of the observed feature. If the evaluation is positive, this region of the heat map visualization is highlighted; we display a black line on top of the particular region.

User-Driven Events User-driven events are manually created based on the user's previous, revised, or new hypothesis. Different visualization layers (e.g., lexical episode plots, topic bars, feature heat map and similarity matrix) guide the user to the interesting discourse regions.

Final Analysis Reports Once these events have been created they are automatically added to the final analysis report. Depending on the individual analysis, these reports could be a single page with one annotated event or could contain many event pages. The reports are compiled as PDFs and can be saved by the user. Before the compilation, users can specify if events should be concatenated in the order of occurrence or in the order of creation. Figure 9 shows an example of a report with an event generated by the political scientist during the user study.

6 Expert Study

To evaluate our tool, we conducted a qualitative user study. Our participants were three experts from different research fields: linguistics, political science, and computer science. Because each expert has a different focus for their research, they each had a different task in mind when using our tool for their analysis allowing us to evaluate the broad applicability of our tool. For each participant, we conducted a two-hour long session in order to explain the functionality of the tool and gather qualitative feedback. Each session was both screen captured and audio recorded for later analysis.

Dataset Each participant used one of two datasets for the analysis. These were chosen to demonstrate how the tool copes with different data. Furthermore, to show that our tool supports the analysis of multiple tasks on the same text, two participants explored the same dataset. We chose datasets with generally familiar content to our experts; (1) the first presidential debate between Obama and Romney in 2012, and (2) the first presidential debate between Trump and Clinton in 2016.

Procedure We began each session with an interview regarding their previous experience of discourse analysis and report generation. After we introduced the participant to the tool, we gathered a first round of feedback about the value of the visualizations and the benefit and intuitiveness of the interaction. The visualization layers were explained separately, without providing suggestions for a possible use case. The aim was to see if the participants intuitively created their own workflow, and how that differed depending on their task. We then gave the participants full control over the interface. Each expert had approximately thirty minutes to create a report of the given discussion. We used the think-aloud method to gather information regarding their choices for using specific visualization layers. We also made note of any challenges or issues the participants faced, if any. The session was ended with a semi-structured interview about the overall usability of the tool.

Tasks Each participant defined their own analysis task. The participant from linguistics analyzed question features in order to accumulate more information about the structure and use of different types of questions within the discourse. The political scientist explored argumentation features in order to extract content information about the topics which discussants (dis)agreed on (e.g., *Obamacare*, *Dodd-Frank banking reform*). The computer scientist was interested in finding out what each politician said about their *economic policies*.

In the expert studies, we wanted to answer the following questions: (Q1) Can the users fulfill the given tasks and generate knowledge specific for their task? (Q2) Is the analysis process and report generation intuitive? (Q3) Does the automatically generated report includes the relevant information for their analysis? In order to answer these questions, we evaluated the qualitative user feedback.

6.1 Task-Driven Workflow

Each participant performed a different task with the tool and created a report which included findings and observations. The participants were free to decide which visualization layers and workflow to use.

Question Types to Analyze ISQ and NISQ The participant from linguistics used a subset of micro-linguistic features to verify assumptions about the usage and context in which ISQ and NISQ occur. After being introduced to the visualizations, she saw the benefit of the different visualization layers. She stated that for other analysis tasks, the lexical episode plots and topic bars were important to get a general overview of the discussed topics without the need to read the whole discussion. But for her specific analysis task, the feature heat map, utterance timeline, and full-text view were relevant.

At the beginning of the analysis, she selected all question types in order to display them in the heat map visualization. After taking a look at the distribution of different question features, she began to select regions that contained questions in order to create events. For each created event, she used the full-text view to learn the context of the questions.

The participant created an event labeled "Money and why not" which she annotated writing that "In the particular example, questions with 'why' are NISQ. Otherwise, wh-questions are ISQ." Another event was named as "Consecutive ISQ" having multiple "Yes-No" questions, and she annotated it writing that "It is obvious that consecutive questions which appear in the same utterance are NISQ."

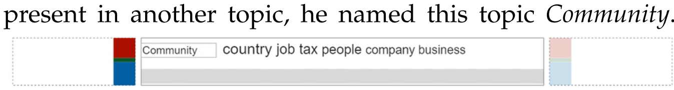
After creating multiple other events, she commented that the tool was very helpful to make judgments about ISQ and NISQ. For all judgments that she made using the tool, she had already formed a hypothesis. Using our tool, the participant could test her hypothesis and identify additional, previously unanticipated examples. The participant appreciated the possibility of annotating the events manually and the generated insight report was thought to have value for her project, where the goal is generating linguistic insights into the question classification problem [55].

Agreement and Disagreement The participant from political science analyzed agreement and disagreement between Obama and Romney in the first US presidential debate in 2012 on various topics. He used a workflow which was repeated throughout the session. First, he found a region of interest based on micro-linguistic feature values. He observed lexical episode plots to gain a first insight into the discussed context. Then, he selected the particular region in the utterance timeline. The participant played the (animated) speaker activity visualization to gain a deeper insight into the context. Afterward, he read some of the selected utterances in the full-text view to get more detailed information. Then, he created an event.

As first, the participant selected micro-linguistic features indicating agreement, disagreement, and emotion to be displayed in the heat map visualization. He observed their values in order to find regions of the discussion where multiple utterances in a row correlated with the agreement feature and a high emotion count. He created multiple events, like "Dodd-Frank banking reform," "Medicare," and "Obamacare." He annotated the first event saying that "There is an agreement between two presidential candidates regarding the need for banking regulation, but there is no agreement on how to do it." The participant observed a disagreement between the presidential candidates on the "Obamacare" topic. He noticed that similarly to the "Medicare" topic there is a separation between the state and private sector. The participant annotated the "Obamacare" event stating that "Romney is against Obamacare. Romney obviously prefers a market solution, Obama defends the state as a facilitator."

After the analysis session, the participant stated that the tool highlights interesting discussion regions with a high agreement and disagreement between the speakers and that the visualization layers were good for multiple purposes. He found the animated view and the extracted phrases very important, as they provide insight into whether a particular region is interesting for further analysis. When the representative phrases confirmed his suspicion that the regions were interesting, he read some relevant utterances in the full-text view in order to find out what exactly had been said. Based on this new knowledge, he could generate a new hypothesis.

Content Summarization Similar to the participant from linguistics, the computer scientist used the first US presidential debate transcript between Trump and Clinton in 2016. The participant created a report summarizing the information regarding the goals of the presidential candidates in the context of economic policy. In particular, he focused on promises for societal improvements by the candidate in the event of being elected. The participant found the topic shown in the in-line figure as the most relevant one for his analysis, and although the keyword "community" was present in another topic, he named this topic *Community*.



He stated that exactly for such situations it is important to be able to label the topic manually. During the analysis, he created multiple events. Even though the participant did not read the text of the discussion, he found out that Clinton supports the generation of energy by using solar panels, and that she sees it as a good chance to create millions of more new jobs. However, Trump was strictly against it. The participant found out that Trump blames foreign countries for stealing companies and jobs. He was unfamiliar with these statements before, but they confirmed his previous knowledge about the US politics. Additionally, he informed himself about the "cyber attacks." He stated that the report generation as a process is helpful to understand the discussion's content better, especially due to the time aspect. He felt it was less time consuming given that the most relevant information was highlighted, and that the user can explore only the part of the discussion that seems to be relevant. He emphasized the importance of note-taking by saying that "it is impossible to capture everything using automatic methods. Therefore, the note-taking is very important to capture exactly that what is relevant for the particular expert who works with the system."

6.2 Discussion: Opportunities and Lessons Learned

Overall, we received positive feedback regarding the usability and the functionality of the tool. All three experts could perform their tasks and gather interesting findings with VisInReport, satisfying Q1. The generated reports were evaluated as helpful for their research, satisfying Q3. All of them stated that due to the numerous versatile visualization layers, the tool is broadly applicable for different tasks and different use cases. The tool also proved to be helpful for the detection of interesting regions within a discussion. Although the user frequently needs access to the original text (which is provided) to understand the complete context, the overall analysis process is simpler using our tool.

The creation of events using annotations is the second main component in the analysis process. The generation of events was seen as a simple and intuitive process, satisfying Q2. The participants stated that the annotation of the events is highly important as it is impossible otherwise to include all aspects of the data into the visualizations.

The participant from computer science paid more attention to usability than the researchers from other disciplines. Overall, he was satisfied with the usability, but mentioned a few minor issues which might be improved. One suggestion was to automatically resize the views, possibly by a click on the separation line or a button, without requiring the user to manually resize the views using a dragging method.

During the user studies, we gathered multiple suggestions for additional visualization layers or micro-linguistic features. The expert from linguistics stated that her colleagues might use the tool to explore prosodic features such as the distribution of pitch values over the utterances if the original spoken version of the transcript were integrated in a time-aligned manner into the system. The prosodic features could then be added to the heat map visualization as an additional point of analysis. An additional visualization layer showing representative sentences from each topic separately could also be useful for the analysis. A summarization of the topic modeling results has been shown to be important [56]; the authors use a textual topic summary to provide a better understanding how topic modeling methods work.

While giving his final feedback, the participant from computer science commented that "[despite some minor concerns] the tool is really cool! It has a good design. It is too time-consuming to read the whole text; as I can reduce the work-load and concentrate my attention only on discussion's regions which I think are interesting, I can save a lot of time." He mentioned that the tool is a good example of Shneiderman's mantra: overview first, zoom and filter, then details-on-demand [57]. An overview of the whole discourse is shown in the topic bar visualization and lexical episode plots. Users can zoom in on an utterance to read the full text and filter a discourse region for a closer analysis; they can define which micro-linguistic features should be displayed in the heat map visualization. Speaker profiles are shown only on demand as a tooltip.

In comparison to tools such as Leximancer and Tableau that also support reporting of analysis insights, VisInReport has two main advantages. First, VisInReport integrates a wide range of diverse content and micro-linguistic features. The diversity of the integrated features enables testing for a wide range of hypotheses. Leximancer, on the contrary, is mainly built to support the analysis of concept maps and integrates only a limited number of additional features (i.e., sentiment). Second, VisInReport integrates automated approaches and suggests events for analysis based on patterns detected in micro-linguistic features. None of the mentioned alternative tools guides users towards interesting regions in the analyzed data.

Limitations: Although we received positive feedback during the expert studies, the system has a few limitations. As we display several visualizations simultaneously, the space for each component is limited. Depending on which topic modeling algorithm is used, the content visualization showing extracted topics can require more space than available to be displayed on a single screen. The need to scroll in order to observe all visual representations can influence the analysis process negatively. The participant from political science was unsure if the visualization layers which are not relevant for the final report could be removed automatically, as he forgot to remove irrelevant layers manually. All participants were skeptical about the similarity matrix visualization. It is indeed a very specifically oriented visualization, which might be too complex. Highlighting of the micro-linguistic features in the full-text view was also desired.

Take Home Message: During the system's design period, the close collaboration with computational linguists and political scientists, and the conducted user studies, we have gained new insights that are worth sharing with the visualization community concerning the automated report generation process. **(1) Targeted automated event extraction.** As the user studies show, users commonly have different goals when analyzing discourse transcripts. Although it is crucial to support users in performing their tasks more effectively and efficiently through automated approaches, the amount of automatically created events might be overwhelming, especially if these events were not representative of the user's particular analysis focus. Hence, it might be crucial understand user interest, either through automatic interest modeling or by querying the user regarding their analysis focus (e.g., features of interest), to provide more targeted automated suggestions. **(2) More intelligible event extraction.** One of VisInReport's limitations is the automatic extraction of events, which is currently done using a naive assumption that the only regions of interest are those with several utterances in a row having similar values. To address this, we are currently testing a pattern mining algorithm to extract more interesting relationships between different micro-linguistic and content features. Nevertheless, there is a need for further meaningful, complex, task-specific event extraction methods. **(2) The quantity of created events.** In a time-intensive analysis session, the amount of both automatically and manually created events may be large. One interesting future research opportunity is a (hierarchical) event grouping. An automatically generated summary of similar events and their common characteristics could help to overcome the hurdle of manually processing the potentially vast amount of created events at the end of the analysis session.

7 Conclusion

In this paper, we presented VisInReport — a visual analytics tool for insight report generation from discourse transcripts. Researchers across different disciplines routinely analyze discourse transcripts as part of their research and currently the only way to generate reports is to do it manually requiring a lot of effort and time. We presented the idea of visualization layers and enable the detection of interesting regions within the discourse. We allow for different investigations of the content of these regions and provide a means for combining the gathered insights and findings into a single summary report. Due to the flexibility of the visualization layers, we can support different user groups with a single tool. The user can decide which visualization layers to use for the analysis.

There are a number of further visualization layers which can be added to the tool to support other analysis tasks. A named entity graph presented in NEREx [12] could be used as an alternative for the topic-bars visualization. The participants of the user study mentioned multiple micro-linguistic features which might be added to the heat map visualization like other question types. Furthermore, the system detects simple patterns for each selected micro-linguistic feature. We could improve this pattern detection technique, by applying the method presented by Jentner et al. [58]. This method allows for the detection of higher level patterns containing correlations between different features. The tool could also support event generation by providing more suggestions of possible interesting events based on a set of further indicators or a combination of existing criteria such as topic modeling results, keywords, or micro-linguistic features. In our future work, we plan to perform an additional evaluation by expanding the range of the analysis tasks and by examining the quality of the generated reports using quantitative properties. The system will be included into the lingvis.io framework [59].

Acknowledgments

We thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for funding within project KE 740/17-2 and BU 1806/10-2 of the FOR2111 "Questions at the Interfaces". We thank the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] T. A. Van Dijk, "What is political discourse analysis," *Belgian Journal of Linguistics*, vol. 11, no. 1, pp. 11–52, 1997.

[2] ——, "Discourse semantics and ideology," *Discourse & Society*, vol. 6, no. 2, pp. 243–289, 1995.

[3] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," in *Information Visualization*. Springer, 2008, pp. 154–175.

[4] A. Mathisen, T. Horak, C. N. Klokmose, K. Grønbæk, and N. Elmqvist, "Insideinsights: Integrating data-driven reporting in collaborative visual analytics," in *Computer Graphics Forum*, vol. 39, no. 3, 2019, pp. 649–661.

[5] J. P. Halsall and J. Powell, "Crafting knowledge exchange in the social science agenda," *Cogent Social Sciences*, vol. 2, no. 1, p. 1244145, 2016.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[7] N. Cao, J. Sun, Y. R. Lin, D. Gotz, S. Liu, and H. Qu, "FacetAtlas: Multifaceted visualization for rich text corpora," *IEEE Trans. on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1172–1181, 2010.

[8] X. Wang, S. Liu, J. Liu, J. Chen, J. Zhu, and B. Guo, "TopicPanorama: A full picture of relevant topics," *IEEE Trans. on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2508–2521, 2016.

[9] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher, "Serendip: Topic model-driven visual exploration of text corpora," in *Proc. IEEE Symp. on Visual Analytics Science and Technology (VAST)*, 2014, pp. 173–182.

[10] T. Bergstrom and K. Karahalios, "Conversation clusters: Grouping conversation topics through human-computer dialog," in *Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI)*, 2009, pp. 2349–2352.

[11] A. Chaney and D. Blei, "Visualizing topic models," in *Int. AAAI Conf. on Web and Social Media (ICWSM)*, 2012, pp. 419–422.

[12] M. El-Assady, R. Sevastjanova, B. Gipp, D. Keim, and C. Collins, "NEREx: Named-entity relationship exploration in multi-party conversations," *Computer Graphics Forum*, vol. 36, no. 3, pp. 213–225, 2017.

[13] W. Dou, X. Wang, R. Chang, and W. Ribarsky, "ParallelTopics: A probabilistic approach to exploring document collections," in *Proc. IEEE Symp. on Visual Analytics Science and Technology*, 2011, pp. 231–240.

[14] S. Havre, B. Hetzler, and L. Nowell, "ThemeRiver: Visualizing theme changes over time," in *Proc. IEEE Symp. on Information Visualization (InfoVis)*, 2000, pp. 115–123.

[15] F. Wei, S. Liu, Y. Song, S. Pan, M. Zhou X., W. Qian, L. Shi, L. Tan, and Q. Zhang, "TIARA: A visual exploratory text analytic system," in *Knowledge Discovery and Data Mining*, 2010, pp. 153–162.

[16] W. Cui, S. Liu, L. Tan, C. Shi, and Y. Song, "TextFlow: Towards better understanding of evolving topics in text," *IEEE Trans. on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2412–2421, 2011.

[17] S. Malik, A. Smith, T. Hawes, P. Papadatos, J. Li, C. Dunne, and B. Shneiderman, "TopicFlow: Visualizing topic alignment of Twitter data over time," in *Proc. IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining (ASONAM)*, 2013, pp. 720–726.

[18] W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky, "HierarchicalTopics: Visually exploring large text collections using topic hierarchies," *IEEE Trans. on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2002–2011, 2013.

[19] C. Blundell, Y. W. Teh, and K. A. Heller, "Bayesian rose trees," *arXiv e-prints*, vol. arXiv:1203, 2012.

[20] V. Gold, C. Rohrdantz, and M. El-Assady, "Exploratory text analysis using lexical episode plots," in *Eurographics Conf. on Visualization (EuroVis) - Short Papers*, 2015.

[21] D. Angus, A. Smith, and J. Wiles, "Conceptual recurrence plots: Revealing patterns in human discourse," *IEEE Trans. on Visualization and Computer Graphics*, vol. 18, no. 6, pp. 988–997, 2012.

[22] M. Wattenberg, "Arc diagrams: Visualizing structure in strings," in *Proc. IEEE Symp. on Information Visualization (InfoVis)*, 2002, pp. 110–116.

[23] B. Kerr, "Thread arcs: An email thread visualization," in *Proc. IEEE Symp. on Information Visualization (InfoVis)*, 2003, pp. 211–218.

[24] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann, "On close and distant reading in digital humanities: A survey and future challenges," in *Eurographics Conf. on Visualization (EuroVis)*, 2015, pp. 83–103.

[25] M. Correll, M. Witmore, and M. Gleicher, "Exploring collections of tagged text for literary scholarship," *Computer Graphics Forum*, vol. 30, no. 3, pp. 731–740, 2011.

[26] D. Oelke, M. Hao, C. Rohrdantz, D. A. Keim, U. Dayal, L.-e. Haug, and H. Janetzko, "Visual opinion analysis of customer feedback data," in *Proc. IEEE Symp. on Visual Analytics Science and Technology (VAST)*, 2009, pp. 187–194.

[27] K. Kucher, T. Schamp-Bjerede, A. Kerren, C. Paradis, and M. Sahlgren, "Visual analysis of online social media to open up the investigation of stance phenomena," *Information Visualization*, vol. 15, no. 2, pp. 93–116, 2016.

[28] M. El-Assady, V. Gold, C. Acevedo, C. Collins, and D. Keim, "ConToVi: Multi-party conversation exploration using topic-space views," *Computer Graphics Forum*, vol. 35, no. 3, pp. 431–440, 2016.

[29] L. South, M. Schwab, N. Beauchamp, L. Wang, J. Wihbey, and M. A. Borkin, "DebateVis: Visualizing political debates for non-expert users," in *IEEE Vis. Conf. (VIS) 2020*, 2020, pp. 241–245.

[30] A. J. Soto, R. Kiros, V. Kešelj, and E. Milios, "Exploratory visual analysis and interactive pattern extraction from semi-structured data," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 3, pp. 16:1–16:36, Sep. 2015.

[31] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut, "Text summarization techniques: A brief survey," *arXiv e-prints*, vol. arXiv:1707, 2017.

[32] H. Takamura and M. Okumura, "Text summarization model based on maximum coverage problem and its variant," *Proc. Conf. of the European Chapter of the Association for Computational Linguistics*, pp. 781–789, 2009.

[33] I. Mani, *Automatic Summarization*. Amsterdam: John Benjamins, 2001.

[34] S. Riezler, T. H. King, R. Crouch, and A. Zaenen, "Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for lexical-functional grammar," in *Proc. Conf. of the North American Chapter of the Assoc. for Comp. Linguistics on Human Language Technology*, vol. 1, 2003, pp. 118–125.

[35] J. Ramos, "Using TF-IDF to determine word relevance in document queries," *Proc. of the First Instructional Conf. on Machine Learning*, pp. 1–4, 2003.

[36] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document," *Proc. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 68–73, 1995.

[37] K.-F. Wong, M. Wu, and W. Li, "Extractive summarization using supervised and semi-supervised learning," in *Proc. Int. Conf. on Computational Linguistics*, 2008, pp. 985–992.

[38] L. Zhou and E. Hovy, "A web-trained extraction summarization system," in *Proc. of the 2003 Conf. of the North American Chapter of the Assoc. for Comp. Linguistics on Human Language Technology - Volume 1*, ser. NAACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 205–211.

[39] H. Stitz, S. Luger, M. Streit, and N. Gehlenborg, "AVOCADO: Visualization of workflow-derived data provenance for reproducible biomedical research," *Computer Graphics Forum*, vol. 35, no. 3, pp. 481–490, 2016.

[40] K. Xu, S. Attfield, T. J. Jankun Kelly, A. Wheat, P. H. Nguyen, and N. Selvaraj, "Analytic provenance for sensemaking: A research agenda," *IEEE Computer Graphics and Applications*, vol. 35, no. 3, pp. 56–64, 2015.

[41] F. Hogenboom, F. Frasincar, U. Kaymak, F. De Jong, and E. Caron, "A survey of event extraction methods from text for decision support systems," *Decision Support Systems*, vol. 85, pp. 12–22, 2016.

[42] W. Dou, X. Wang, D. Skau, W. Ribarsky, and M. X. Zhou, "LeadLine: Interactive visual analysis of text data through event identifcation and exploration," *Proc. IEEE Conf. on Visual Analytics Science and Technology*, pp. 93–102, 2012.

[43] J. Fulda, M. Brehmel, and T. Munzner, "TimeLineCurator: Interactive authoring of visual timelines from unstructured text," *IEEE Trans. on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 300–309, 2016.

[44] M. El-Assady, F. Sperrle, O. Deussen, D. A. Keim, and C. Collins, "Visual analytics for topic model optimization based on user-steerable speculative execution," *IEEE Trans. on Visualization and Computer Graphics*, vol. 25, pp. 374–384, 2019.

[45] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," *Proc. of the 22nd Int. Conf. on World Wide Web*, pp. 1445–1456, 2013.

[46] M. A. Hearst and D. Rosner, "Tag Clouds: Data analysis tool or social signaller?" *Proc. of the Annual Hawaii Int. Conf. on System Sciences*, pp. 160–160, 2008.

[47] M. F. Cheema, S. Jänicke, and G. Scheuermann, "AnnotateVis: Combining traditional close reading with visual text analysis," in *IEEE VIS 2016 Workshop on Visualization for the Digital Humanities*, 2016.

[48] B. Morrow, T. Manz, A. E. Chung, N. Gehlenborg, and D. Gotz, "Periphery plots for contextualizing heterogeneous time-based charts," *IEEE Vis. Conf. (VIS) 2019 (short papers)*, 2019.

[49] D. Biber, *Variation Across Speech and Writing*. Cambridge University Press, 1991.

[50] M. Butt, A. Hautli-Janisz, and V. Lyding, *LingVis: Visual Analytics for Linguistics*, ser. CSLI lecture notes. CSLI Publications/Center for the Study of Language & Information, 2020.

[51] A. Hautli-Janisz and M. El-Assady, "Rhetorical strategies in German argumentative dialogs," *Argument and Computation*, vol. 8, no. 2, pp. 153–174, 2017.

[52] I. Koshik, *Beyond Rhetorical Questions: Assertive Questions in Everyday Interaction*. Amsterdam: John Benjamins Publishing Company, 2005.

[53] D. Angus, A. E. Smith, and J. Wiles, "Human communication as coupled time series: Quantifying multi-participant recurrence," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1795–1807, 2012.

[54] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Engineering Bulletin*, vol. 24, no. 4, pp. 1–9, 2001.

[55] R. Sevastjanova, M. El-Assady, A. Hautli, A.-L. Kalouli, R. Kehlbeck, O. Deussen, D. A. Keim, and M. Butt, "Mixed-initiative active learning for generating linguistic insights in question classification," in *DSIA: Data Systems for Interactive Analysis*, 2018.

[56] M. El-Assady, R. Sevastjanova, F. Sperrle, D. Keim, and C. Collins, "Progressive learning of topic modeling parameters: A visual analytics framework," *IEEE Trans. on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 382–391, 2018.

[57] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proc. IEEE Symp. on Visual Languages*. IEEE, 1996, pp. 336–343.

[58] W. Jentner, M. El-Assady, B. Gipp, and D. Keim, "Feature alignment for the analysis of verbatim text transcripts," in *EuroVis Workshop on Visual Analytics (EuroVA)*, 2017, pp. 13–18.

[59] M. El-Assady, W. Jentner, F. Sperrle, R. Sevastjanova, A. Hautli-Janisz, M. Butt, and D. A. Keim, "lingvis.io - A linguistic visual analytics framework," in *Assoc. for Comp. Linguistics, ACL System Demonstrations*, 2019.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2021.3104026, IEEE Transactions on Visualization and Computer Graphics

Rita Sevastjanova is a Research Associate in the Data Analysis and Visualization Research Group at the University of Konstanz, Germany. She received her M.Sc. degree in Information Engineering from the University of Konstanz 2017. Her research focuses on Natural Language Processing, Language Modeling, and the integration of game elements into visual analytics processes to strengthen user motivation.

Mennatallah El-Assady is a Research Associate in the Data Analysis and Visualization Research Group at the University of Konstanz, Germany, and in the Visualization for Information Analysis lab at Ontario Tech University, Canada. She received her M.Sc. degree in Information Engineering from the University of Konstanz in 2015. Her research interests include visual text analytics, user-steerable topic modeling, and discourse/conversational text analysis.

Adam Bradley holds a PhD in English and Systems Design Engineering from the University of Waterloo. He is currently Principal Research Scientist at Uncharted Software in Toronto, Canada and is a Research Associate in the Visualization for Information Analysis lab at Ontario Tech University. His work is focused on human-centered data science, which is concerned with the overlaps between human generated data, natural language processing, and data visualization.

Christopher Collins received the PhD degree from University of Toronto in 2010. He is currently the Canada Research Chair in Linguistic Information Visualization and Associate Professor at Ontario Tech University. His research focuses on information visualization and human-computer interaction with natural language processing. He is a member of the IEEE, a past member of the executive of the IEEE VGTC and has served several roles on the IEEE VIS Conference Organizing Committee.

Miriam Butt is Professor of General and Computational Linguistics at the Department of Linguistics at the University of Konstanz. Butt earned her doctorate in linguistics in 1993 at Stanford University. She is the author or editor of 11 books and numerous publications in the areas of theoretical, historical and computational linguistics, where she has focused on developiing computational tools for linguistic analysis and visual analytics for linguistis (LingVis).

Daniel Keim is professor and head of the Data Analysis and Visualization Research Group in the Computer Science Department at the University of Konstanz, Germany. He has been actively involved in data analysis and information visualization research for more than 25 years and developed novel visual analysis techniques for large data sets. Dr. Keim received his Ph.D. degree in Computer Science from the University of Munich, Germany. Before joining the University of Konstanz, Dr. Keim was associate professor at the University of Halle, Germany, and Technology Consultant at AT&T Shannon Research Labs, NJ, USA.