Volume 0 (1981), Number 0 pp. 1–13 COMPUTER GRAPHICS *forum*

Visual Analytics of Text Conversation Sentiment and Semantics

C. G. Healey[† 1,2], G. Dinakaran[2], K. Padia[2], S. Nie[2], J. R. Benson[3], D. Caira[3], D. Shaw[3], G. Catalfu[3], and R. Devarajan[3]

[1]Institute for Advanced Analytics, NC State University, Raleigh, NC, 27606
[2]Department of Computer Science, NC State University, Raleigh, NC 27695-8206
[3]SAS Institute, Cary, NC, 27513

Abstract

This paper describes the design and implementation of a web-based system to visualize large collections of text conversations integrated into a hierarchical four-level-of-detail design. Viewers can visualize conversations: (1) in a streamgraph topic overview for a user-specified time period; (2) as emotion patterns for a topic chosen from the streamgraph; (3) as semantic sequences for a user-selected emotion pattern, and (4) as an emotion-driven conversation graph for a single conversation. We collaborated with the Live Chat customer service group at SAS Institute to design and evaluate our system's strengths and limitations.

CCS Concepts

• ***Visualization application domains*** → *Visual analytics;* • ***Sentiment analysis*** → *sentiment;* • ***Applied computing*** → *Document management and text processing;* • ***Decision support systems*** → *Data analytics;*

1. Introduction

This paper proposes text analytic methods and an associated web-based system for visualizing large collections of text conversations between two participating individuals. Our problem is situated in the broad area of text analytics and visualization.

Text analysis has a long history, particularly in the area of natural language processing (NLP) problems like text representations, syntactic parsing, and document summarization [JM18, MS99]. Visualizing text is more recent, but is now a well-studied problem [CC16, KK15] with numerous techniques to visually represent text, documents, and document collections.

Despite impressive progress in both the NLP and visualization areas, unsolved research issues remain. For example, deep learning methods are now being used to model abstractive summaries that produce more human-like results [KAS19, SHR19]. Text narratives like those found in movies or novels are being visualized to provide overviews of plot, character interaction, and complex non-linear narrative patterns (*e.g.*, time travel) [LWW*13, PKH19, THM15]. Scale is a critical issue both in terms of computation: ($O(mn^2)$ for LSA on an $m \times n$ matrix) and semantics: How many sentences properly summarize a large document collection? What if there are more documents than pixels available in a visualization?

Our interest in this paper lies mainly in investigating text visualization and semantic scale. The high-level goals are to highlight and summarize patterns in large text conversations using: (1) *sentiment*, the emotional affect embodied in a text block, and: (2) *semantics*, a short summary of the meaning of a text block. The intended audience is analysts exploring large amounts of text, but without the time to read the text in its entirety. Semantic summaries provide access to an overall meaning of the text. Sentiment captures the emotional affect of the text. Both properties are commonly required during exploration of large text collections [KAS19, KKM*19, LZ12, Moh16, PL04, SHR19]. To address this need, we focus on four subgoals:

- $\mathbf{G}_{\text{LOD}}$. Develop methods to summarize conversations at different levels of detail, including topics, emotion[†] (sentiment) patterns, and semantic sequences. Although emotion includes dimensions like anger, surprise, and excitement. Our paper uses only a pleasure emotion.
- $\mathbf{G}_{\text{PROP}}$. Visualize both the raw text of a conversation and derived properties: sets of conversations forming topics, estimated emotion for both individual conversations and topic sets, and semantic summaries of emotion blocks within a conversation.
- $\mathbf{G}_{\text{SCALE}}$ Scale to support large collections of conversations.
- $\mathbf{G}_{\text{AUTO}}$. Automate the system to minimize the required user interventions.

The four goals were driven in part by ongoing research challenge of scale, and in part by requests from our collaborators

† Email: healey@ncsu.edu; Funded by SAS internal grant

† Due to the similarity between the words *sentiment* and *semantics*, we substitute emotion for sentiment throughout the remainder of the paper.

[KAS19, Moh16, NJM19]. All four goals address different aspects of scale: level-of-detail of information presented, properties to enhance and summarize text, methods to scale to massive text datasets, and minimal user intervention to allow analysis on large document collections.

We focus on online customer service chats as a practical testbed. Online chat services are an important customer interaction channel [Kla13]. We partnered with SAS Institute (`www.sas.com`), a well-known business intelligence software provider, to collaborate with SAS's Live Chat group. This allowed us access to real chat conversations and interaction with chat agents and managers to construct, refine, and evaluate our visualization designs.

Visualizing conversations is useful in domains where exploration of the conversations' meaning and emotional affect provides insight over raw text alone. Examples include transcripts, social media, or customer service sessions. We want to visualize conversations to expose both the raw text and derived properties at the level of detail best suited to a user's analysis needs. Our system uses a four-layer detail-on-demand design ($\mathbf{G}_{\text{LoD}}$, Fig. 1.) The four-layer structure was constructed in collaboration with SAS researchers, in part to address their chat analysis needs, and in part to explore potential solutions to our original four goals.

1. Layer 1 presents an overview of conversations for a user-chosen time period, subdivided by topic and colored by emotion.
2. Layer 2 focuses on conversations for a user-selected topic from Layer 1, grouped into emotion blocks built from emotional transition patterns (for example, *UP* where *P* is pleasant and *U* is unpleasant emotion.)
3. Layer 3 lists conversations with a specific emotion pattern from Layer 2, with each emotion block's text, summarized using semantic keywords.
4. Layer 4 visualizes a single conversation from Layer 3 as an emotion line graph of participants' utterances, together with a linked table of the raw text within the conversation.

SAS Live Chat managers identified numerous ways our system is useful for real-time analysis, posthoc exploration, training, identifying successful and unsuccessful chat patterns, and determining common issues raised by customers, particularly when new products are introduced. SAS is currently discussing partnering with LiveChat Incorporated (`www.livechatinc.com`), which provides SAS's chat infrastructure, to integrate the visualization tools into LiveChat's suite of chat analytics offerings. Our proposed visualization system offers the following novel contributions.

1. A four-layer, detail-on-demand visualization to explore and analyze large, text-based conversation collections ($\mathbf{G}_{\text{LoD}}$, $\mathbf{G}_{\text{SCALE}}$.)
2. Emotion estimation to build emotion patterns for a conversation ($\mathbf{G}_{\text{PROP}}$, $\mathbf{G}_{\text{AUTO}}$.)
3. Semantic keyword assignment to blocks within an emotion pattern ($\mathbf{G}_{\text{PROP}}$, $\mathbf{G}_{\text{AUTO}}$.)
4. Real-world evaluation using customer service chats ($\mathbf{G}_{\text{LoD}}$, $\mathbf{G}_{\text{SCALE}}$, $\mathbf{G}_{\text{AUTO}}$.)

2. Background

Text visualization techniques have been proposed for various types of text data and analytic tasks. We developed, combined, and extend new and existing text visualization and analytic algorithms. Here, we focus on the methods most relevant to our system. More general surveys on text visualization and text analytics are available for interested readers [AZ12, DL16, KK15].

2.1. Text Conversation Visualization

Different methods have been proposed to visualize text narratives as they unfold. Alharbi and Laramee provide a "survey of surveys" on existing text visualization methods [AL19]. For example, one approach is a standard line graph or scatterplot with time on the *x*-axis and nodes at different *y*-positions representing different participants. Hovering over a node reveals the text at the corresponding point in the chat. Nodes can also act as visual glyphs that present derived text attributes like topics or emotions.

Streamgraphs show the volume of a property over time as a stacked collection of *streams* that grow and shrink to visualize volume [HHWN02, SWL*10]. History Flow is designed to track changes in a text document over time [VWD04]. Parallel coordinates can be employed to represent conversations [Ins09], where axes track the sequence of text blocks, *y*-position on the axes map text to individuals, and each polyline represents a conversation. Another text narrative visualization technique is StoryFlow, first proposed by Tanahashi and Ma [TM12]. StoryFlow is based on xkcd's movie narrative charts (`xkcd.com/657/`.) Individuals are represented by horizontal lines running left-to-right over time. Lines converge during events at a common location, then separate as the event ends. Improvements have been proposed to optimize computational speed [LWW*13], and to handle streaming data rather than post-processing a completed narrative [THM15].

Kucher et al. provide an overview of recent sentiment visualization techniques [KPK17]. Cao et al. developed Whisper to monitor spatiotemporal diffusion of social media information. Sentiment polarity was visualized using a sunflower metaphor to identify influencers and geolocated groups receiving and spreading information [CLS*12]. SocialHelix followed, visualizing and tracking social media topics as they form and their sentiment diverges using a DNA-like representation [CLLW14]. Wu et al. presented opinion propagation in Twitter using a combination of streamgraphs and Sankey graphs [WLY*14]. Liu et al. linked primary and secondary text using semantic lexical matching. Results are presented in a dashboard containing topic keywords, concept clusters, and a causality timeline [LWLM17]. El-Assady et al. extracted conversation threads from large online conversation spaces using a combination of supervised and unsupervised machine learning algorithms [EASKC18]. Mohammad et al. extracted stance and sentiment in tweets using a labeled database, with results visualized using treemaps, bar graphs, and heatmaps [MSK17]. Kucher et al. identified stance and sentiment polarity in social media text, then used similarity over these properties to visualize analysis of collections of topic–data source streams [KMPK20]. Wei et al. proposed TIARA, a system to extract topics that are visualized in an annotated streamgraph [WSY*10]. Dörk et al. use a construct called "Topic Streams," a streamgraph approach to monitoring topics in a large online conversation environment over time [DGWC10].

Lu et al. summarize the use of "predictive visual analytics," the

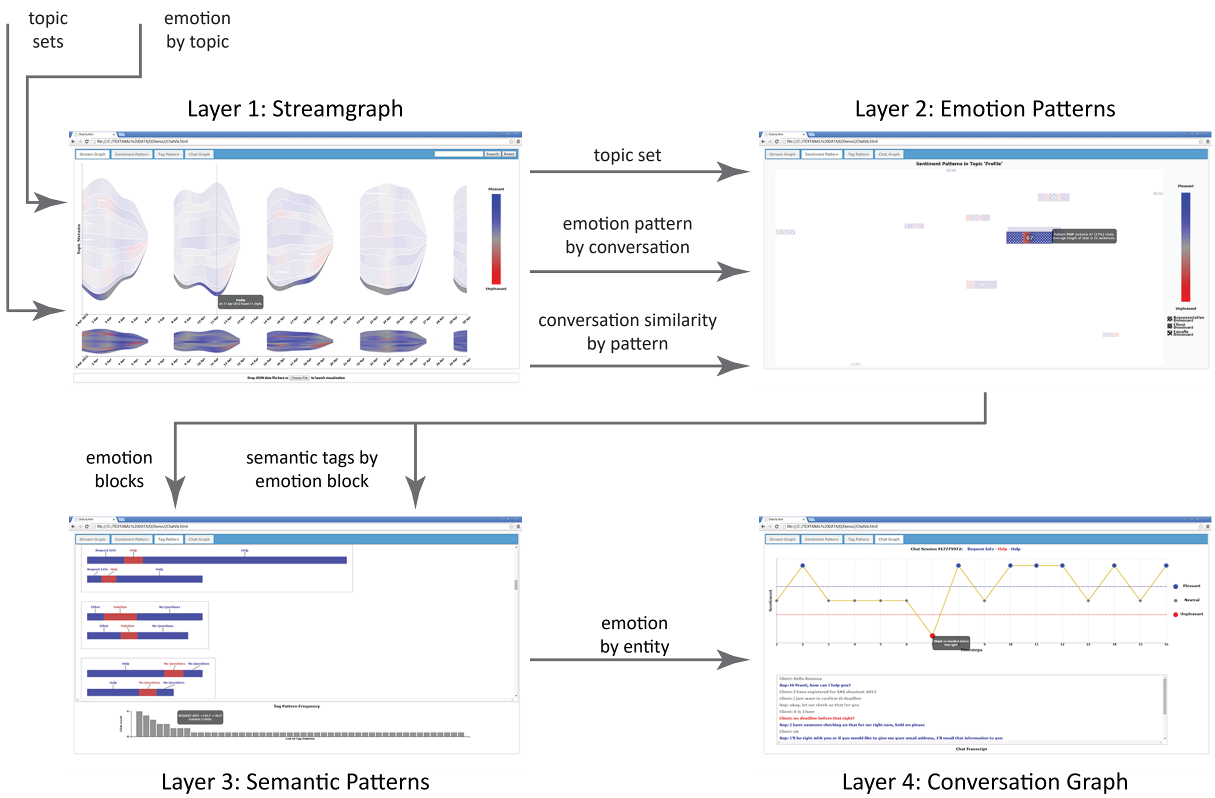


Figure 1. Pictorial description of the four layers in our conversation visualization, showing derived data (Fig. 2) flowing between layers

integration of interactive machine learning and user engagement with the modeling process to generate interactive visual analytics [LGH*17]. El-Assady et al. addressed a problem similar to ours: visualizing multi-party conversation behavior at the topic level with their ConToVi system [EAGA*16]. Speakers are attached to topics, focusing on speaker utterances and behavior patterns within conversations. Hoque and Carenini implement ConVis and MultiConVis, an ML, NLP, and visual analytic system to explore blog conversations [HC14,HC16]. A topic graph is built using individual conversation topics as sets to a graph-based topic clustering algorithm. Sentiment polarity is calculated using the Sentiment Orientation Calculator, then visualized as a dashboard with filtering and individual blog selection capabilities.

We considered both ConToVi and MultiConVis as solutions to our domain, but felt they lacked some of the novel semantic and emotion properties we provide: a focus+context level-of-detail interface, emotion estimates over individual documents and document collections, semantic tag sequences, and a web-based dashboard. Unique properties of ConToVi (conversation behavior patterns) and MultiConVis (complex filtering) could be integrated into our system to provide users with a more flexible interface and conversations visualizations from different perspectives.

2.2. Conversation Analysis

To visualize conversations, we derive supporting properties using text analytic algorithms. We focus on three issues directly related to our research goals: topic identification, text summarization, and sentiment analysis.

Topic Identification.

Topic identification clusters a collection of text documents D into sets of related documents based on pairwise document similarity, forming topic clusters. We use a mixed-model approach, where each document is assumed to contain some amount of each topic. Alternative methods (*e.g.*, Gibbs sampling Dirichlet mixture model [YW14]) assume one topic per document. Mixed models allow documents to belong to more than one topic, but may be less effective for short text like tweets or online comments. Standard preprocessing is performed: bag-of-words term vector creation for each document, stop word removal, stemming, TF-IDF term weight calculations on each term vector, vector normalization, and finally calculation of cosine similarity $\mathrm{sim}_{i,j} = \cos\theta = v_i \cdot v_j$ between pairs of document term vectors v_i and v_j.

$\mathrm{sim}_{i,j}$ are combined into a $|D| \times |D|$ similarity matrix SIM, used as input to a clustering algorithm. Alternatively, a TF-IDF-weighted document–term vector matrix X can be further processed to try to improve its representative accuracy. Two well-known approaches are latent semantic analysis (LSA) and latent Dirichlet allocation (LDA) [BNJ03, DDL*90]. Both attempt to form semantic concepts and adjust document representations to use concepts rather than terms alone. LSA uses SVD to separate X into $X = U\Sigma V^{\mathrm{T}}$, building concepts as linear combinations of terms in D.

Document d_j is defined by the amount of each concept it contains. The top k concepts based on variance in D (defined by eigenvalues in $\Sigma\Sigma^T$) are retained, significantly reducing the size of X. Similarly, LDA uses X to reinterpret each document as a mixture of inferred topics using Gibbs sampling. Once reduced term vectors are produced, cosine similarity or a similar approach like nearest neighbor in the document-topic space generates topic clusters.

Text Summarization.

Text summaries provide high-level overviews to highlight subsets of documents matching viewers' particular interests. Text summarization guides viewers to a small set of relevant documents, allowing them to maximize the return on their reading effort. Methods for text summarization are studied in natural language processing (NLP), information retrieval (IR), and more recently, in deep learning [ZWL18]. Approaches include summaries built from high-weight TF-IDF terms, short summaries made up of sentences extracted from a document—extracts [AKK04]—or generated from topics embedded in a document—abstracts [BM05]. Generative adversarial deep neural networks (GAN DNNs) have made significant progress in the abstract summarization area [KKM*19, NJM19].

Visualization techniques have also been proposed to present text summaries. One well-known method is a tag or word cloud that summarizes a document by generating a set of term–frequency pairs (t_i, f_i), then visualizing the k most frequent terms t_i sized by f_i (larger for more frequent) and using a space-filling algorithm to pack them into a "cloud" [VWF09].

Sentiment Analysis.

Sentiment analysis is an active research area in NLP, IR, and machine learning. Prior to analysis, text may be preprocessed, for example, by removing objective, fact-based text irrelevant to estimating emotion [PL04]. The two most common analysis methods are: (1) supervised, using a training set to build emotion estimation models, and (2) unsupervised, where raw text is converted directly into scores along emotional dimensions. Interested readers are directed to recent surveys on the topic [LZ12,Moh15,PL08,ZWL18].

Analysis is often built on psychological models of emotion that use orthogonal dimensions to describe emotional affect. For example, Russell defined three dimensions pleasure (or valence), arousal, and dominance—the PAD model—to represent emotion [Rus80, RFB99]. Plutchik's four-dimensional model of joy–sadness, anger–fear, trust–disgust, and surprise–anticipation uses a color wheel to represent basic emotions: hue for dimension endpoints (eight hues) and saturation for emotional intensity (weak saturation for low intensity to strong for high) [Plu80].

In terms of supervised NLP approaches, Pang et al. compared naive Bayes, maximum entropy, and support vector machines (SVMs) for classifying movie reviews as positive or negative [PLV02]. Basic unigrams (independent terms) performed best using SVM. Intuitive extensions like bigrams (pairs of co-occurring terms), term frequencies, part of speech tagging, and document position information did not improve performance. Similarly, Turney rated online reviews as positive or negative using pointwise mutual information to generate statistical dependence between review phrases and the anchor words "excellent" and "poor" [Tur02].

A common unsupervised approach employs sentiment dictionaries. Terms appear as keys, but each term is associated with one or more emotional dimension scores rather than definitions. POMS-ex (Profile of Mood States) is a 793-term dictionary designed to measure emotion on six dimensions: tension–anxiety, depression-dejection, anger–hostility, fatigue–inertia, vigor–activity, and confusion–bewilderment [PB08]. Affective Norms for English Words (ANEW) used the PAD model to score 1,033 emotion-carrying terms along each dimension using a nine-point scale [MLA*10]. Other dictionaries also exist: SentiStrength, built from MySpace comments [TBP*10]; Linguistic Inquiry and Word Count (LWIC), a dictionary that classifies terms as positive, negative, or neutral [TP10], and SentiWordNet, built from the well know WordNet synset dictionary [BES10]. More recently, researchers have applied Amazon MTurk to assign scores for emotional dimensions to large dictionaries. Mohammad and Turney created EmoLex, a dictionary of 14,182 terms that use Plutchik's four emotional dimensions [MT13]. Warriner extended the original ANEW dictionary to approximately 13,000 terms [WKB13], again using MTurk to obtain PAD scores and comparing results to the original ANEW scores for validation.

Numerous challenges in sentiment estimation continue to exist: more subtle text cues (*e.g.*, sarcasm, irony, humor, or metaphors), a writer's emotion versus what they write (*e.g.*, an author evoking a particular emotional affect), emotion towards different aspects of an entity, stance (*i.e.*, the opinion on a topic), or cross-cultural and domain differences (*e.g.*, "alcohol" can be evaluated differently depending on the underlying culture) [Moh15,Moh16,PL08]. We use a curated sentiment dictionary, so we highlight limitations specific to dictionary-based emotion estimation.

1. **Term independence.** We cannot derive context from neighboring terms, *e.g.* for "I am **happy**" versus "I am **not happy**."
2. **Missing terms.** We cannot score terms a dictionary does not contain.
3. **Term ambiguity.** We cannot differentiate between homonyms, *e.g.*, "I lie down" versus "I lie often"
4. **Dictionary size.** Increasing term count does not necessarily improve performance, since new terms are often neutral (*e.g.* "table.") This pushes the aggregate emotion towards neutral, producing similar estimates for text that is clearly different.

3. System Design

Given our goals: $\mathbf{G}_{\text{LOD}}$, support for multiple levels of detail; $\mathbf{G}_{\text{PROP}}$, visualizing both raw and derived data; $\mathbf{G}_{\text{SCALE}}$, scalability, and $\mathbf{G}_{\text{AUTO}}$, and minimal manual user intervention, we concluded no existing technique satisfies all of these requirements. New methods are needed, or existing methods must be extended.

Frameworks exist for real-world design studies, for example, investigations by Tory and Möeller on whether expert reviews are valuable, or Sedlmair et al.'s guidelines on designing evaluations that include domain experts [SMM12, TM05]. Design requirements were initially defined during twice-monthly meetings with SAS, followed by refinement based on our analytic and visualization knowledge. Using Sedlmair's nine-stage design framework, we completed: learn (NCSU), winnow (NCSU/SAS), cast (SAS), discover (NCSU/SAS), design (NCSU), implement (NCSU), deploy (NCSU/SAS), reflect (NCSU/SAS), write (NCSU). Individuals most expert for each stage participated, and significant collaboration occurred in multiple stages.

In summary, SAS identified four goals that were difficult or impossible to extract with their existing tools: (1) identifying semantic and emotion patterns in chats related to a topic of interest, (2) determining whether certain subtopics had significant activity, (3) identifying chats with similar semantic patterns, and (4) locating chats that ended poorly to try to determine what caused the failed interaction. Our design was driven in large part by these needs.

For text analytics, we applied text clustering to subdivide a conversation set into topics, sentiment analysis to assign emotional estimates to individual conversations and conversation sets, and topic summarization to assign semantic tags to blocks of text with common emotions. For visualization, we integrated streamgraphs and line graphs with two novel glyph-based approaches to form a level-of-detail visualization system. User interface operations tie the levels together to allow viewers to identify areas of interest, explore them in detail, then return to higher-level overviews to continue their investigations. Our system is implemented as a web-based application using JavaScript and the jQuery and d3 interface and visualization libraries [Bos21, Ope16].

3.1. Terminology

Before describing our visualization system, we provide definitions for the different types of derived data, their purposes, and their uses within the system. These are either existing attributes in SAS's chat datasets, or properties we derived to meet their analysis and visualization requests.

- **Entity.** A single contiguous text block, *e.g.*, a sentence or paragraph.
- **Chat.** An ordered sequence of entities that form a conversation.
- **Chat set.** A set of chats related in some way, *e.g.,* by topic.
- **Topic set.** A set of chats whose text discusses a common topic.
- **Chat set similarity.** The similarity between the text from two chat sets.
- **Emotion.** An estimation of pleasure based on a collection of text.
- **Entity-level emotion.** The emotion associated with an entity.
- **Emotion block.** A block formed from a collection of adjacent entities, each with identical entity-level emotion.
- **Emotion pattern.** The sequence of emotion blocks that occurs over a chat.
- **Topic emotion.** The overall emotion for chats in a topic set.
- **Semantic tag.** A term summary of an entity's semantics.
- **Emotion block semantic tag.** A semantic tag for the text in an emotion block.
- **Semantic tag sequence.** The sequence of tags for emotion blocks in a chat.

3.2. System Overview

The flow diagram in Fig. 2 outlines the data derivation stages, which include importing a raw chat dataset D and generation of derived results: topics, emotion, emotion patterns, chat set similarity, and semantic tags. *Topic sets* are defined over D. Emotion is computed both for sets of chats belonging to a common topic (*topic emotion*) and as *emotion patterns* for individual chats. *Chat set similarity* is computed between sets of chats with common emotion patterns. Semantic tags are generated for individual emotion blocks (*emotion block semantic tags*.) Stages highlighted in orange represent novel contributions to the text analytics, sentiment analytics, and visualization fields:

- **Entity-level emotion.** Weighting individual entities to aggregate independent term emotion for an entity emotion estimation ($\mathbf{G}_{\text{PROP}}$, $\mathbf{G}_{\text{SCALE}}$.)
- **Emotion pattern.** Chat summarization based on entity emotion sequences ($\mathbf{G}_{\text{SCALE}}$, $\mathbf{G}_{\text{AUTO}}$.)
- **Emotion pattern semantics.** Machine learning and dictionary-based approaches to assign descriptive tags to emotion blocks within an emotion pattern ($\mathbf{G}_{\text{PROP}}$, $\mathbf{G}_{\text{SCALE}}$, $\mathbf{G}_{\text{AUTO}}$.)
- **Chat set similarity.** Similarity calculation for pairs of chat sets based on a combination of text similarity and emotion ($\mathbf{G}_{\text{PROP}}$.)
- **Chat emotion.** Hierarchical emotion estimation at entity, chat, emotion pattern, and topic levels of detail ($\mathbf{G}_{\text{LOD}}$, $\mathbf{G}_{\text{PROP}}$.)

Once raw data is imported, topic sets, chat similarity, emotion, and semantic tags are computed to act as input for the four-layer visualization (Figs. 1 and 2).

3.3. Text Analytics

We motivate the different data derivation stages in Fig. 2, then explain how the derived data is used to construct the individual visualization layers in Fig. 1.

Chat Topic Clustering.

During our collaboration with SAS Institute, we applied their text analytics tools where appropriate, although these can easily be replaced with publicly available libraries. SAS Text Miner (TM) [SAS17b] clusters individual chats into topic sets using LDA. TM applies k-means to cluster the documents into topics, choosing k using root mean squared error (RMSE) [KM13]. For small datasets, a single, manageable k will often suffice. For large datasets, we apply hierarchical level-of-detail decomposition to avoid a single, large k.

Chat Emotion.

We estimate chat emotion at four levels of detail: (1) aggregated emotion for a topic set (*topic emotion*); (2) visual patterns representing sequences of emotion blocks (*emotion pattern*); (3) blocks combining adjacent entities with identical emotion (*emotion blocks*), and (4) emotion for each chat entity (*entity-level emotion*.) This addresses our level-of-detail and visualization goals ($\mathbf{G}_{\text{LOD}}$.)

Chats are subdivided into entities using a simple approach based on speaker transition. Regardless of the method, each entity is meant to contain a single emotion to avoid multiple emotions interfering with one another during estimation. We decided to compare and contrast three methods: dictionary-based, NLP-based, and a combination of the two.

We automatically build a chat training set using a sentiment dictionary. This training set acts as input to a supervised machine learning algorithm, in our case support vector machine (SVM). The

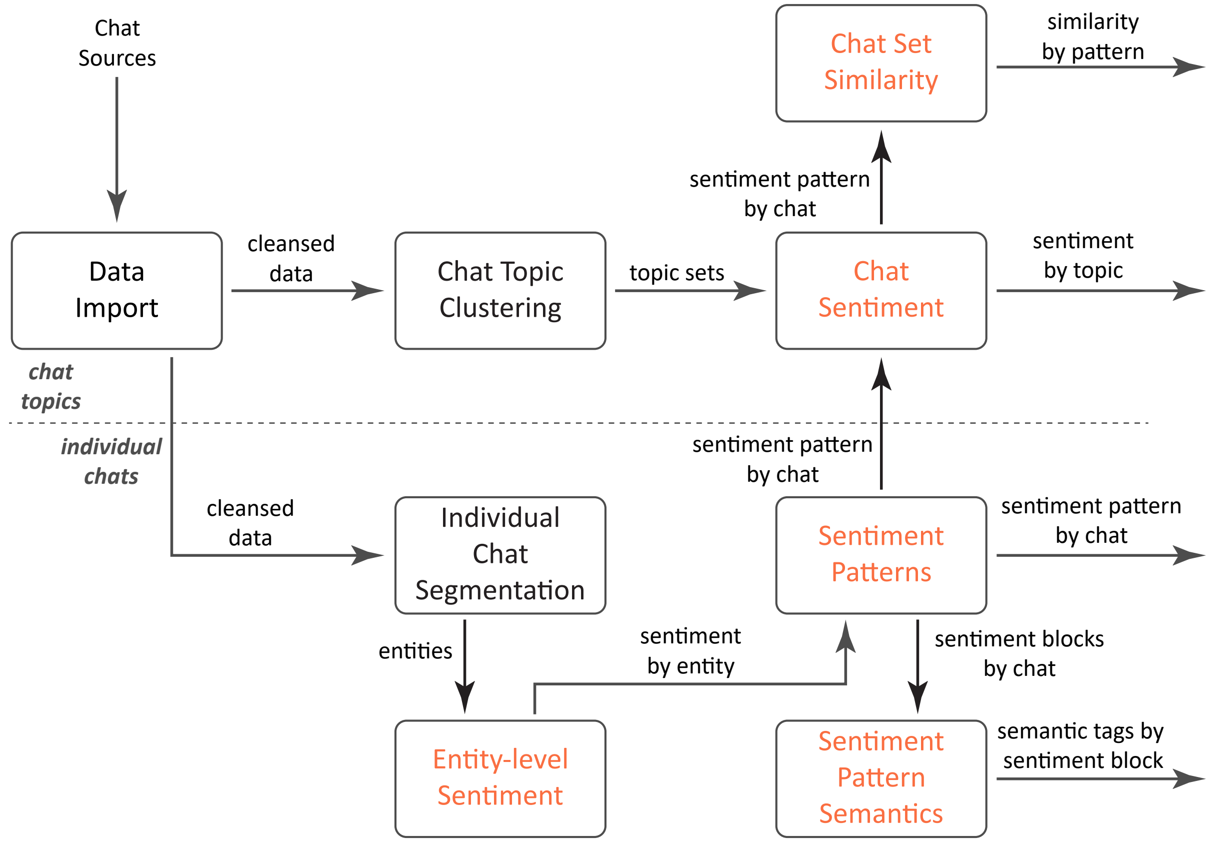


Figure 2. Data derivation to convert raw text chats using two separate stages: a chat topic stage to derive similarity and emotion by topic, and an individual chat stage to derive emotion and semantics by segmented blocks within each chat; orange stages represent stages with novel contributions

dictionary-based training set uses a sentiment dictionary in the standard way: recognized terms in a text block (or entity) are scored along a pleasure dimension with the dictionary, results are aggregated, and the entity is assigned a positive, neutral, or negative valence. This is used to build an emotion estimation model M. M can then assign emotions and confidences to untagged entities.

Our dictionary is curated from multiple sources: the original ANEW [MLA*10], Warriner's extended ANEW [WKB13], and a happiness dictionary that we extended to include arousal scores [DHK*10]. This produced 11,528 terms for the two emotional dimensions pleasure and arousal. A mean and standard deviation based on participant responses exists for each term–dimension pair. Higher standard deviations indicate disagreement among observers on the "correct" term score. Emotion estimates for each entity use a weighted average of the dictionary's pleasure scores for an entity's terms. Weights are derived from normal curves based on term standard deviation: higher weights for terms with a lower standard deviation.

A sentiment dictionary offers two practical advantages. First, it fully automates the process of emotion estimation, allowing minimal manual user intervention ($\mathbf{G}_{\text{AUTO}}$.) Second, it can generalize to a range of text sources and analysis tasks. Sentiment dictionaries also suffer from the limitations listed in Section 2.2, however, which can skew emotion estimates in various ways.

To investigate improving the emotion estimates, we constructed a rule-based emotion estimation model using SAS Sentiment Analysis Studio (SA) [SAS17a]. SA uses a supervised learning approach to model text-to-emotion rules. The training set is split into ten samples $S_1, \ldots, S_{10}$ to perform ten-fold cross-validation. We generate a final rule-based model as follows:

1. Load $S_2 \ldots S_{10}$ into SA to build a statistical training model.
2. Import learned features from the training model to create a rule-based model.
3. Test the rule-based model against S_1 for accuracy.
4. Save the model as M_1. Repeat steps 1 through 3 for $S_2, \ldots, S_{10}$.
5. Merge $M_1, \ldots, M_{10}$ weighted by their validation scores to create a final M.

M is used to re-estimate emotion and confidence scores for each chat. SA generates categorical emotions of unpleasant (U), neutral (N), or pleasant (P). We combine adjacent entities with identical categories to produce emotion blocks (Fig. 4b.) For example, a chat with SA-estimated entities scored pleasant, pleasant, unpleasant, pleasant, unpleasant, and unpleasant is represented by four emotion blocks in an emotion pattern *PUPU*. Follow-on processing with emotion blocks should reduce both computation time and visual complexity since the number of blocks in a chat is normally smaller than the number of entities. To produce aggregated emotion for a chat set, we calculate a weighted average of chat emotions in the set, where weight corresponds to SA-estimated confidence.

Validation against manual scoring showed that dictionary+SA estimates were more accurate than either dictionary or SA estimates alone. Four hundred chat entities were sampled from the chat dataset. Two volunteers hand-scored each entity as positive, neutral, or negative. In twenty-four cases (6%) the volunteers disagreed on U-N or N-P scores (*i.e.*, neutral versus non-neutral). In six cases (1.5%) the volunteers disagreed on U-P scores. Discussions were held to resolve which emotion to assign. The chats were then automatically scored for emotion with our dictionary method, SAS SA's default model, and our SA model trained on dictionary-scored entities (dictionary+SA.) Accuracy for the three methods dictionary, SA, and dictionary+SA were 45%, 64%, and 70%, respectively. F_1, precision, and recall results showed that in all three cases, dictionary+SA outperformed both dictionary and default SA.

No technique outperformed the others in all cases. For example, "I appreciate that!" was scored *N* by our dictionary, but *P* by the dictionary+SA model. "Our SAS EM won't start up properly" was scored *U* by our dictionary, but *N* by the dictionary+SA model.

Pattern Set Similarity.

Pattern set similarity represents the text similarity between two sets of chats with different emotion patterns. For example, to compute chat similarity between the pleasant–unpleasant–pleasant pattern set (*PUP*) and another set (say *PU*):

1. For each topic set T_i produced during chat topic clustering, identify all pattern sets $PS_j \in T_i$.
2. For every pair of pattern sets PS_j and PS_k, calculate a normalized similarity $\text{sim}_{j,k}$ between the text in PS_j and PS_k
3. Repeat this process for all n topics $T_i \in T_1, \ldots, T_n$.

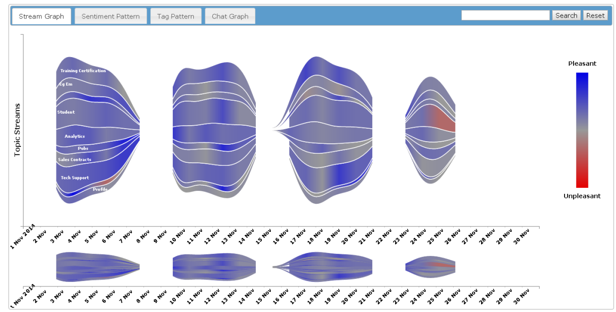
We compute pattern set similarity by topic since pattern sets are visualized after a user selects a topic from the streamgraph.

The similarity matrix SIM is used in the Emotion Patterns visualization (Sec. 3.4) to force-project glyphs representing each pattern set. We chose multidimensional scaling (MDS) as one method to project pairs of pattern sets with a higher similarity closer to one another on a 2D plane. This allows analysts to identify whether pattern sets are similar or dissimilar based on their text.

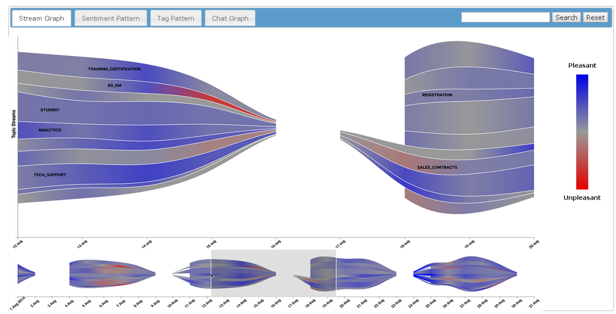
Emotion Block Semantic Tags.

We compute a semantic tag (*e.g.*, "greeting", "question", "feature") for each emotion block in a chat to summarize its text. This provides an overview of the chat without forcing viewers to read its raw text. To do this, we:

1. Sample each topic set to select a representative group of chats.
2. For each emotion block in the sample set, manually assign an appropriate semantic tag.
3. Combine manually tagged emotion blocks into a training set, then use TM to model semantic tags based on a block's text.
4. Use the model to assign semantic tags to every emotion block.



(a)



(b)

Figure 3. Streamgraph of topics in a chat dataset: (a) day on the *x*-axis, chat volume on the *y*-axis, a "stream" corresponds to a topic with color representing overall emotion for chats in the topic; the lower, miniaturized streamgraph shows all available chats, and allows users to select time intervals to visualize in more detail in the upper streamgraph; (b) using the mini-streamgraph at the bottom of the visualization to zoom in on a specific time period

This is the one situation where manual intervention is required. It is also possible to fully automate this process as follows.

1. Topics are defined as pairs of individuals participating in conversations, removing TM's topic clustering.
2. A rule-based emotion model is not constructed with SA. Instead, we use our dictionary to estimate emotion directly.
3. Semantic tags are chosen as an emotion block's highest-weight TF-IDF terms, removing TM's text-to-tag model.

3.4. Visualization

Topics, emotion, emotion patterns, chat similarity, and semantic tags are presented in a four-layer visualization. Each layer occupies a separate screen in an overview+detail manner. Live Chat users preferred this to a single screen focus+context approach.

Streamgraph.

A streamgraph provides an overview of topics, topic volume, and topic emotion over time for chats in *D*. Streamgraphs can visualize tens of thousands of chats without difficulty ($\mathbf{G}_{\text{SCALE}}$).

The Live Chat group asked to examine blocks of chats over time, separated by topic and annotated with topic emotion. They monitor chat volume by topic (*e.g.*, during a new product or update release), as well as emotion and how it changes within a topic over time. Based on several alternatives (stack bar charts, line charts, and streamgraphs) the Live Chat users were most positive about the streamgraph, requesting the ability to select specific time intervals (Fig. 3a). A miniaturized version of the streamgraph at the bottom of the visualization allows users to select a time interval for the main streamgraph while maintaining overall context (Fig. 3b).

Topic sets and emotion over time for each topic are input to the streamgraph, where they are visualized using height (chat volume) and a double-ended color scale (overall chat emotion): saturated blue for pleasant to saturated red for unpleasant [HE99]. Fig. 3 covers one month with ten streams representing ten topics. The gaps in the graph show weekends when chat agents are not available. Novel findings by the Live Chat users included: (1) the fact that overall call volume seems to increase at the beginning of the week, then taper off towards Fridays, and (2) the negative emotion in the Student topic at the end of the fourth week.

The colors, both here and in other layers in the visualization, are selected based on perceptual guidelines [HE99]. Red and blue are used to respect colorblind individuals. Hovering over a topic's stream will highlight the stream. A tooltip shows the number of chats in the topic for the currently selected time. Analysts can focus on chats that contain specific keywords using a Search field that filters on keywords of interest to only include chats that contain the specified keywords. As with time intervals, keyword filtering applies to all layers beneath the streamgraph visualization.

Emotion Patterns.

The emotion pattern layer visualizes emotion transition pattern sets for a user-chosen topic, the volume of chats in each pattern set, and the text similarity between sets. This layer highlights patterns of interest (*e.g.*, single emotion chats, or chats ending in negative emotion), and visually "groups" patterns with similar chat text.

Live Chat users often focus on a specific topic to examine. Emotion estimates are beneficial since they allow for evaluating how chats progress over time. This motivated the idea of emotion patterns. The Live Chat users requested removing neutral blocks since they are most interested in "outlier" emotions. We suggested two additional properties: a representation of the volume of chats for each emotion pattern and the ability to see which participant, customer or agent, contributed the majority of the text in an emotion block. This shows positive or negative emotions associated with customers and chat agents. Given this, we developed an emotion pattern glyph. Choosing a topic stream in the streamgraph generates a visualization for chats in the selected topic, presented as emotion pattern sets: collections of chats with identical positive–negative emotion transitions (Fig. 4a.) Input includes chats for the selected topic, emotion patterns for each chat, and similarity scores for all pairs of pattern sets.

An emotion pattern set is visualized as a rectangle made up of the pattern's emotion blocks (Fig. 4b.) The same colors from the streamgraph are used to represent a block's emotion (*e.g.*, blue-red-blue for a two-transition *PUP* emotion pattern). The size of a

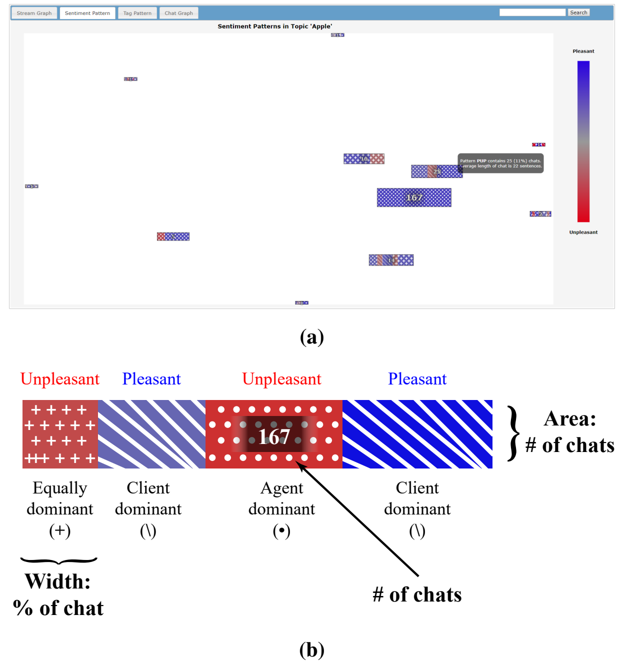


Figure 4. *Emotion patterns* in a user-chosen topic: (a) one rectangle per emotion pattern set, blue…red for pleasant…unpleasant blocks, distance between pattern rectangles represent text similarity; (b) emotion pattern rectangle, block color → emotion, block width → percentage of chat, block texture → dominant speaker, area → number of chats with emotion pattern set

rectangle visualizes the number of chats in the pattern set, with the exact number of chats overlaid on the pattern's rectangle. The emotion blocks' widths correspond to the percentage of the chats' text that occurs within the block. Emotion blocks are textured to show the participant dominating the discussion (Fig. 4b): diagonal lines for customer, dots for agent, or pluses for equal participation.

Rectangles are positioned based on text similarity between emotion pattern sets: closer for more similar. We use MDS to position rectangles so that Euclidean distance estimates similarity. Jittering was used to control rectangle overlap. This visually groups pattern sets with similar text. Hovering over an emotion rectangle generates a tooltip with the number, order, and emotion of its subblocks, the number of chats in the pattern, the percentage of chats for given topic, and the average number of entities in each chat (Fig. 4).

Novel findings by the Live Chat users included recognizing how many chats went well (*e.g.*, the large all-blue rectangle in Fig. 4a) and how many went poorly (*e.g.*, pattern sets endings in red, unpleasant emotion.) They also identified common patterns (large rectangles), more complex chats (rectangles with numerous blocks), or chats where the agent expressed negative emotion (red blocks with a dot pattern texture).

Semantic Sequences.

The semantic sequence layer visualizes semantic tags for blocks in an emotion pattern set. Chats in the set are grouped by their seman-

Figure 5. Semantic sequence visualization of chats in a set, color represents emotion (blue → pleasant, red → unpleasant), tags above emotion blocks summarize the block's text, the histogram visualizes x-axis → semantic sequence, y-axis → sequence frequency

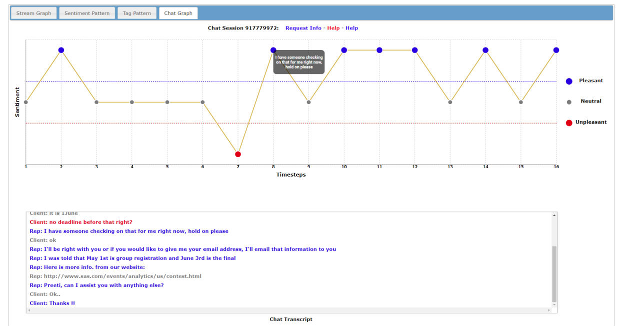


Figure 6. Visualizing a user-chosen chat, x-axis → time, y-axis → pleasure, blue, gray, and red nodes (top, center, bottom y-position) for pleasant, neutral, and unpleasant entities, respectively; text window shows raw text for the chat entities

tic tag sequences. Tags indicate an emotion block's content without needing to read the block's text. They also identify semantic tag sequences that occur frequently within the pattern set. NCSU and SAS researchers examined individual text blocks to produce fourteen representative semantic tags.

Live Chat users use a summary of a chat set's text content to investigate chats with specific emotion patterns. One need is to identify frequent chats with particular text content in a specific order. The semantic sequence layer visualizes the chats in a user-chosen emotion pattern set. Chats' blocks are tagged using a descriptive keyword to present the "gist" of the chats' conversations. Semantic tag sequences group chats to identify common sequences. A histogram shows each semantic group in descending order of chat count to further support analysis (Fig. 5).

Choosing a rectangle in the emotion pattern layer generates the semantic sequence visualization for chats in the emotion pattern: a list of chats' tagged semantic sequences and a histogram of semantic sequence frequencies. Input includes chats for the selected pattern, emotion blocks for each chat, and semantic tags for each emotion block.

The list of emotion patterns represents individual chats. Because all chats are from a common emotion pattern, they all have the same color sequence. The length of each rectangle represents the length of the chat in entities. Every block is annotated with a semantic tag to summarize the text in the block's entities. Chats are combined into semantic tag sequence groups. The first three chats in Fig. 5 have a "Solution–No Questions" semantic sequence. The next two have a "Problem–Unresolved" sequence. Semantic tag groups are sorted in descending order of chat count, placing sequences that are most common at the top of the list. A frequency histogram of semantic sequences is used to represent semantic patterns (*e.g.*, a bar for Solution–No Questions.) Hovering over a bar generates a tooltip with the bar's semantic sequence and chat count (Fig. 5.) Selecting a bar will scroll the list of chats to the start of the semantic tag group that contains the bar's semantic sequence.

The Live Chat group used the semantic sequence visualizations to: (1) identify frequent chats with a common order of content; (2) summarize chat content through semantic tags, and (3) present the distribution of chats by semantic tag sequence. This highlighted important semantic patterns that were part of selected emotion groups. Novel findings included a large number of chats that started negatively and ended positively. Although suspected, this confirmed that agents led numerous chats to a successful conclusion.

Chat Graph.

A chat graph visualizes one chat's entities: its raw text, emotion, and the emotion transitions as the chat unfolds. Selecting a chat rectangle in the semantic sequence visualization presents a chat graph (Fig. 6.) Input includes the text of the selected chat, the chat entities, and their emotion.

Live Chat users understand that reading chats is sometimes unavoidable. Indeed, this is the current method for exploring a chat dataset. The Live Chat users employ our summaries and derived properties to direct exploration and reveal insights in large chat sets. A chat graph summarizes a single chat with a line graph: nodes represent alternating client and agent utterances. y-position and color dual-encode the overall emotion of a node's text. The raw text is listed below the graph, again colored to highlight estimated emotion.

Blue, gray, and red nodes at the top, center, and bottom of the graph represent pleasant, neutral, and unpleasant entities, respectively. Hovering over a node generates a tooltip that contains the text of the node's entity (Fig. 6.) Below the graph are entities that occurred during the chat, colored blue, gray, and red. Selecting an entity highlights the corresponding node in the chat graph by generating its tooltip (Fig. 6.) Similarly, selecting a node scrolls the list of entities to the selected node's entity.

The Live Chat group uses the chat graph to visualize the length of a chat and how emotion flows within it. An unexpected finding was that the graph–text linking was most useful since it allowed viewers to focus on specific entities, then scroll directly to the entity's raw text. This made it much faster to determine the chat's details and what was driving its success or failure.

Summary.

The novel contributions of our four-layer design include:

- **Level-of-Detail.** A level-of-detail text visualization based on emotion that allows users to move from large document collections through emotion and semantic groupings and finally to an individual document ($\mathbf{G}_{\mathrm{LOD}}$, $\mathbf{G}_{\mathrm{SCALE}}$.)
- **Emotion Glyphs.** A narrative subdivided at emotion transitions using a glyph to visualize emotion and text properties. Glyphs are positioned using MDS to visually encode text similarity across emotion pattern sets ($\mathbf{G}_{\mathrm{PROP}}$.)
- **Semantic Tag Sequences.** Entity and text document summarization through integration into a level-of-detail document visualization system ($\mathbf{G}_{\mathrm{LOD}}$, $\mathbf{G}_{\mathrm{PROP}}$, $\mathbf{G}_{\mathrm{AUTO}}$.)
- **Visualization.** Visual analytics for document sets, an item of interest to both researchers and practitioners, beyond what the research and commercial communities provide ($\mathbf{G}_{\mathrm{PROP}}$, $\mathbf{G}_{\mathrm{SCALE}}$.)

4. Real-World Validation

To test our visualization design with real-world data, we validated with real chat data from the SAS Live Chat group.

4.1. SAS Live Chat

Our preference was to conduct controlled experiments with Live Chat agents. Unfortunately, it was not possible to schedule this validation. Instead, we worked with two Live Chat managers who used the system over four months of actual chat data, then provided feedback comparing the tools they are currently using, their understanding of how agents manage chats, and the types of problems encountered. The Live Chat managers estimate received 5,675 chats per month or 22,700 chats over our four-month test period.

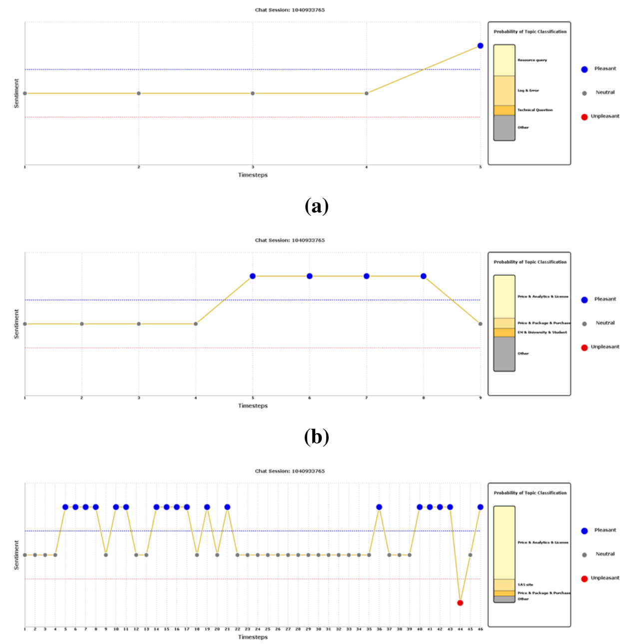
Monthly Posthoc Analysis.

Live Chat managers conduct monthly posthoc analysis to compile summary statistics, search for situations where chats went well or poorly, and monitor chats related to specific topics to communicate issues back to the product teams. To do this, the Live Chat team had already built analysis tools in SAS to explore their data.

Managers reported that the visualizations significantly improved the ability to locate items of interest and gain better insights into the data. For example, they would select topics in the streamgraph, then search the emotion patterns for chats that ended in unpleasant emotion, or for entirely unpleasant chats. They would drill down into these emotion patterns to see if there were a few high-frequency semantic sequences or a broad range of different issues causing the unsuccessful resolution. The managers would further probe individual chats to view the exact interactions that pushed emotion in different directions. This ability improved their situation awareness of the month's chats and reduced the time needed to understand specific types of problems.

Training.

A second novel idea the Live Chat team discussed was using the visualization tool as a training aid. Chat agents are trained to handle different customers and interactions, which may be infrequent but are important and challenging to manage. Visualizations allow Live Chat managers to locate chats representing these situations and present successful and unsuccessful resolutions. This can be done in terms of the full chat between customer and agent or based on emotion transitions that occurred during the chat. Comparing successful and unsuccessful chats highlights responses that are more likely to lead to successful resolutions.



(a)

(b)

(c)

Figure 7. Predictive topic visualization with: (a) 10%; (b) 20%, and (c) 100% of the chat text available

Predictive Topic Identification.

One area of future interest is a chat graph that tracks emotion in real-time and tries to predict the topic of a chat based on its text. This is valuable because one common customer complaint is that it takes too long for an agent to "understand" the customer's issue.

We integrated a predictive topic model into the chat graph built on past chats and their associated topics. Predicted topics are shown as a rectangle to the right of the line graph (Fig. 7), split into the four topics with the highest probabilities. We experimented with one, two, four, and all predicted topics. The Live Chat agents felt two topics were too few, especially when chats were beginning, and all topics were too many. The height of each topic represents its relative probability. Fig. 7 shows an example of a chat when 10% (7a), 20% (7b), and 100% (7c) of the chat is complete. Initially, the model suggests *resource query* as the chat's topic, but at 20% it switches to *price & analytics & license*. The relative probability of *price & analytics & license* continues to climb, proving to be the correct choice when the chat concludes. Agents could combine domain knowledge with predictive topic probabilities to reach a resolution more quickly.

General Feedback.

The Live Chat managers offered general feedback on our system and how it compared to their existing analytic workflow.

- "*The visualizations are clear and simple, which is not typically the case with tools like this. We're able to simply look at the visualizations and 'get' what we're seeing, without the need for extensive training prior to using them.*"
- "*The streamgraph provides a quick high-level view of top topics along with emotion and is simple to navigate. It's easy to adjust the time range. The search capability enhances discovery of particular subtopics, which is very useful.*"
- "*Aggregation of similar intra-chat [emotion] patterns is useful.*"
- "*The histogram [in the semantic sequence visualization] is a good guide to explore and find specific semantic sequences.*"

Feedback sessions were conducted every two weeks with our SAS collaborators. Although formal accuracy experiments were not conducted due to time constraints with Live Chat users, we received no feedback on situations where results were considered incorrect. Feedback focused almost exclusively on novel insights the users derived from the system, and on extensions the users identified as potentially improving the system. Properties for each layer were computed in near real-time, and the visualizations themselves were designed to run interactively, allowing the Live Chat users to explore their data at will.

Live Chat personnel also suggested improvements. They proposed selecting semantic tags in the semantic sequence visualization to only show chats with blocks that contained the tags. Hovering over a block in the semantic sequence visualization could reveal the block's text, freeing them from drilling down to the chat graph. Exporting streamgraph data as a table would allow follow-on filtering, sorting, and other types of useful organization of the topic and emotion categories. These suggestions are planned as future work.

5. Conclusions and Future Work

This paper presents a four-layer level-of-detail system to visualize large collections of conversations between pairs of participants. Each layer filters conversations to provide more detail, from an overview of conversation topic, volume, and aggregated emotion at the top layer to emotion transitions, semantic tag summaries, and raw text of a conversation at the lowest layer.

We address our goals of $\mathbf{G}_{\text{LOD}}$, summarization at different levels of detail; $\mathbf{G}_{\text{PROP}}$, visualize both raw text and derived properties; $\mathbf{G}_{\text{SCALE}}$, scalability, and $\mathbf{G}_{\text{AUTO}}$, minimal manual user intervention.

1. **Level-of-detail.** Viewers can transition from high-level overviews of a conversation dataset, through topic conversations sorted by emotion pattern, to tagged conversations with a common emotion pattern, and to a graph of a conversation's emotion transitions, semantic tags, and raw text.
2. **Raw and Derived Data.** Each layer displays both original and derived data: emotion, volume, and topic in the streamgraph; emotion, participant, conversation count, and text similarity in the emotion pattern layer; emotion and text length in the semantic sequence layer; and emotion in the conversation graph layer.
3. **Scalability.** The streamgraph can visualize thousands of conversations and provides visual cues on topic, emotion, and volume to allow viewers to focus their explorations.
4. **Minimal user intervention.** A sentiment dictionary supports emotion estimation, and data analytics is used for topic clustering, force-directed layout, and semantic tagging of text.

Future Work

Several issues remain for future work. We want to extend our technique to conversations with more than two participants. We require an approach that extends to all four visualization layers. There is also the issue of scalability, that is, how many participants can we support, and is there a reasonable upper limit on the number of participants in a conversation-based visualization domain? We want to improve the derived data techniques. One candidate is semantic tagging. We can currently tag in a fully automatic manner. Still, we plan to re-implement a bootstrapping approach to see if an initial TF-IDF estimation followed by a more sophisticated modeling approach can yield improved results. We may allow users to update or extend the sentiment dictionary from within the visualization system. This would focus the default dictionary on their domain, potentially improving emotion estimates within that domain.

References

[AKK04] ALTHAUS E., KARMANIS N., KOLLER A.: Computing locally coherent discourses. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL '04)* (Barcelona, Spain, 2004), pp. 309–406. 4

[AL19] ALHARBI M., LARAMEE R. S.: SoS TextViz: An extended survey of surveys on text visualization. *Computers 8*, 1 (2019), http://doi.org/10.3390/computers8010017. 2

[AZ12] AGGARWAL C. C., ZHAI C. X.: *Mining Text Data*. Springer Publishing, New York, NY, 2012. 2

[BES10] BACCIANELLA S., ESULI A., SEBASTIANI F.: SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC '10)* (Valletta, Malta, 2010), pp. 2200–2204. 4

[BM05] BARZILAY R., MCKEOWN K. R.: Sentence fusion for multi-document news summarization. *Computational Linguistics 31*, 3 (2005), 297–328. 4

[BNJ03] BLEI D. M., NG A. Y., JORDAN M. I.: Latent Dirichlet allocation. *Journal of Machine Learning 3*, 4–5 (2003), 993–1022. 3

[Bos21] BOSTOCK M.: D3.js 6.7.0, 2021. URL: https://d3js.org. 5

[CC16] CAO N., CUI W.: *Introduction to Text Visualization*. Springer Publishing, New York, NY, 2016. 1

[CLLW14] CAO N., LU L., LIN Y.-R., WANG F.: SocialHelix: Visual analysis of sentiment divergence in social media. *Journal of Visualization 18*, 2 (2014), 221–235. 2

[CLS*12] CAO N., LIN Y.-R., SUN X., LAZER D., LIU S., HUAMIN Q.: Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Transactions on Visualization & Computer Graphics 18*, 12 (2012), 2649–2658. 2

[DDL*90] DEERWESTER S., DUMAIS S., LANDAUER T., FURNAS G., HARSHMAN R.: Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*, 6 (1990), 391–407. 3

[DGWC10] DÖRK M., GRUEN D., WILLIAMSON C., CARPENDALE S.: A visual backchannel for large-scale events. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1129–1138. 2

[DHK*10] DODDS P. S., HARRIS K. D., KLOUMANN I. M., BLISS C. A., DANFORTH C. M.: Temporal patterns of happiness and information in a global social network: Hedonometrics and twitter. *PLOS One* http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0026752 (2010). 6

[DL16] DOU W., LIU S.: Topic- and time-oriented visual text analysis. *IEEE Computer Graphics & Visualization 36*, 4 (2016), 8–13. 2

[EAGA*16] EL-ASSADY M., GOLD V., ACEVEDO C., COLLINS C., KEIM D.: ConToVi: Multi-party conversation exploration using topic-space views. *Computer Graphics Forum (Proceedings EuroVis 2016) 35*, 3 (2016), 431–440. 3

[EASKC18] EL-ASSADY M., SEVASTJANOVA R., KEIM D., COLLINS C.: ThreadReconstructor: Modeling reply-chains to untangle conversational text through visual analytics. *Computer Graphics Forum (Proceedings EuroVis 2018) 37*, 3 (2018), 351–365. 2

[HC14] HOQUE E., CARENINI G.: ConVis: A visual text analytic system for exploring blog conversations. *Computer Graphics Forum 33*, 3 (2014), 221–230. 3

[HC16] HOQUE E., CARENINI G.: MultiConVis: A visual text analysis system for exploring a collection of online conversations. In *Proceedings of the 21st International Conference on Intelligent User Interfaces (IUI '16)* (Sonoma, CA, 2016), pp. 96–107. 3

[HE99] HEALEY C. G., ENNS J. T.: Large datasets at a glance: Combining textures and colors in scientific visualization. *IEEE Transactions on Visualization and Computer Graphics 5*, 2 (1999), 145–167. 8

[HHWN02] HAVRE S., HETZLER E., WHITNEY P., NOWELL L.: ThemeRiver: Visualizing thematic changes in large documents. *IEEE Transactions on Visualization and Computer Graphics 8*, 1 (2002), 9–20. 2

[Ins09] INSELBERG A.: *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer Publishing, New York, NY, 2009. 2

[JM18] JURAFSKY D., MARTIN J.: *Speech and Language Processing, 2nd Edition*. Prentice Hall, Upper Saddle River, NJ, 2018. 1

[KAS19] KOURIS P., ALEXANDRIDIS G., STAFYLOPATIS A.: Abstractive text summarization based on deep learning and semantic content generalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)* (Florance, Italy, 2019), pp. 5082–5092. 1, 2

[KK15] KUCHER K., KERREN A.: Text visualization techniques: Taxonomy, visual survey, and community insights. In *Proceedings IEEE Pacific Visualization Symposium (PacVis 2015)* (Hangzhou, China, 2015), pp. 14–17. 1, 2

[KKM*19] KRYSCINSKI W., KESKAR N. S., MCCANN B., XIONG C., SOCHER R.: Neural text summarization: A critical evaluation. In *Proceedings of the 1029 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)* (Hong Kong, China, 2019), pp. 540–551. 1, 4

[Kla13] KLAUS P.: The case of Amazon.com: Towards a conceptual framework of online customer service experience (OCSE) using the emerging consensus technique. *Journal of Service Marketing 27*, 6 (2013), 443–457. 2

[KM13] KODINARIYA T. M., MAKWANA P. R.: Review on determining number of clusters in *k*-means clustering. *International Journal of Advance Research in Computer Science and Management Studies 1*, 6 (2013). 5

[KMPK20] KUCHER K., MARTINS R. M., PARADIS C., KERREN A.: StanceVis Prime: Visual analysis of sentiment and stance in social media texts. *Journal of Visualization 23*, 6 (2020), 1015–1034. 2

[KPK17] KUCHER K., PARADIS C., KERREN A.: State of the art in sentiment visualization. *Computer Graphics Forum 37*, 1 (2017), 71–96. 2

[LGH*17] LU Y., GARCIA R., HANSEN B., GLEICHER M., MACIEJEWSKI R.: The state-of-the-art in predictive visual analytics. *Computer Graphics Forum 36*, 3 (2017), 539–562. 3

[LWLM17] LIU Y., WANG H., LANDIS S., MACJEJEWSKI R.: A visual analytics framework for identifying topic drivers in media events. *IEEE Transactions on Visualization and Computer Graphics 24*, 9 (2017), 2501–2515. 2

[LWW*13] LIU S., WU Y., WEI E., LIU M., LIU Y.: Storyflow: Tracking the evolution of stories. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2436–2445. 1, 2

[LZ12] LIU B., ZHANG L.: A survey of opinion mining and sentiment analysis. In *Mining Text Data*, Aggarwal C. C., Zhai C. X., (Eds.). Springer Publishing, New York, NY, 2012, pp. 415–463. 1, 4

[MLA*10] MISLOVE A., LEHMANN S., AHN Y.-Y., ONNELA J.-P., ROSENQUIST J. N.: Pulse of the nation: U.S. mood throughout the day inferred from Twitter. http://www.ccs.neu.edu/home/amislove/twittermood, 2010. 4, 6

[Moh15] MOHAMMAD S. M.: Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In *Emotional Measurement*, Meiselman H., (Ed.). Elsevier, Atlanta, GA, 2015, pp. 201–237. 4

[Moh16] MOHAMMAD S. M.: Challenges in sentiment analysis. In *A Practical Guide to Sentiment Analysis*, Das D., Cambria E., Bandyopadhyay S., (Eds.). Springer Publishing, New York, NY, 2016, pp. 61–83. 1, 2, 4

[MS99] MANNING C. D., SCHÜTZ H.: *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, 1999. 1

[MSK17] MOHAMMAD S. M., SOBHANI P., KIRITCHENKO S.: Stance and sentiment in tweets. *ACM Transactions on Internet Technology 17*, 3 (2017), article 26. 2

[MT13] MOHAMMAD S. M., TURNEY P. D.: Crowdsourcing a word–emotion association lexicon. *Computational Intelligence 29*, 3 (2013), 436–465. 4

[NJM19] NASAR Z., JAFFRY S. W., MALIK M. K.: Textual keyword extraction and summarization: State-of-the-art. *Information Processing & Management 56*, 6 (2019), article 102088. 2, 4

[Ope16] OPENJS FOUNDATION: jQuery 1.12, 2016. URL: https://jqueryui.com. 5

[PB08] PEPE A., BOLLEN J.: Between conjecture and memento: Shaping a collective emotional perception of the future. In *AAAI Spring Symposium on Emotion, Personality, and Social Behavior* (Stanford, CA, 2008), pp. 111–116. 4

[PKH19] PADIA K., KAVEEN H. B., HEALEY C. G.: Design considerations for optimizing storyline visualizations. *Computers & Graphics 78* (2019), 64–75. 1

[PL04] PANG B., LEE L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL '04)* (Barcelona, Spain, 2004), pp. 271–278. 1, 4

[PL08] PANG B., LEE L.: Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval 2*, 1–2 (2008), 1–135. 4

[Plu80] PLUTCHIK R.: A general psychoevolutionary theory of emotion. In *Theories of Emotion Volume 1: Emotion, Theory, Research, and Experience*, Plutchik R., Kellerman H., (Eds.). Academic Press, New York, NY, 1980, pp. 3–31. 4

[PLV02] PANG B., LEE L., VITHYANATHAN S.: Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)* (Philadelphia, PA, 2002), pp. 79–86. 4

[RFB99] RUSSELL J. A., FELDMAN BARRETT L.: The structure of current affect: Controversies and emerging consensus. *Current Directions in Psychological Science 8*, 1 (1999), 10–14. 4

[Rus80] RUSSELL J. A.: A circumplex model of affect. *Journal of Personality and Social Psychology 39*, 6 (1980), 1161–1178. 4

[SAS17a] SAS INCORPORATED: SAS Sentiment Analysis Studio. http://support.sas.com/software/products/san, 2017. Online; accessed 25-Mar-2017. 6

[SAS17b] SAS INCORPORATED: SAS Text Miner. http://support.sas.com/documentation/onlinedoc/txtminer/index.html, 2017. Online; accessed 25-Mar-2017. 5

[SHR19] SONG S., HUANG H., RUAN T.: Abstractive text summarization using LSTM-CNN based deep learning. *Multimedia Tools and Applications 78* (2019), 857–875. 1

[SMM12] SEDLMAIR M., MEYER M., MUNZNER T.: Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2431–2440. 4

[SWL*10] SHI L., WEI F., LIU S., TAN L., LIAN X., ZHOU M. X.: Understanding text corpora with multiple facets. In *Proceedings IEEE Symposium on Visual Analytics Science and Technology (VAST 2010)* (Salt Lake City, UT, 2010), pp. 99–106. 2

[TBP*10] THELWALL M., BUCKLEY K., PALTOGLOU G., CAI D., KAPPAS A.: Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology 61*, 12 (2010), 2544–2558. 4

[THM15] TANAHASHI Y., HSUEH C.-H., MA K.-L.: An efficient framework for generating storyline visualization from streaming data. *IEEE Transactions on Visualization and Computer Graphics 21*, 6 (2015), 730–742. 1, 2

[TM05] TORY M., MÖLLER T.: Evaluating visualizations: Do expert reviews work? *IEEE Computer Graphics & Applications 25*, 5 (2005), 8–11. 4

[TM12] TANAHASHI Y., MA K.-L.: Design considerations for optimizing storyline visualizations. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2679–2688. 2

[TP10] TAUSCZIK Y. R., PENNEBAKER J. W.: The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology 29*, 1 (2010), 25–54. 4

[Tur02] TURNEY P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL '02)* (Philadelphia, PA, 2002), pp. 417–424. 4

[VWD04] VIÉGAS F. B., WATTENBERG M., DAVE K.: Studying cooperation and conflict between authors with *History Flow* visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2004)* (Vienna, Austria, 2004), pp. 24–29. 2

[VWF09] VIÉGAS F. B., WATTENBERG M., FEINBERG J.: Participatory visualization with Wordle. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1137–1144. 4

[WKB13] WARRINER A. B., KUPERMAN V., BRYSBAERT M.: Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods 45*, 4 (2013), 1191–1207. 4, 6

[WLY*14] WU Y., LIU S., YAN K., LIU M., WU F.: OpinionFlow: Visual analysis of opinion diffusion on social media. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1763–1772. 2

[WSY*10] WEI F., SHIXIA L., YANGQIU S., SHIMEI P., ZHOU M. X., QIAN W., LEI S., LI T., QIANG Z.: TIARA: Interactive, topic-based visual text summarization and analysis. In *Proceedings of the 16th SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2010)* (Washington, DC, 2010), pp. 153–162. 2

[YW14] YIN J., WANG J.: A Dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2014)* (New York, NY, 2014), pp. 233–242. 3

[ZWL18] ZHANG L., WANG S., LIU B.: Deep learning for sentiment analysis: A survey. *WIREs Data Mining and Knowledge Discovery 8*, 4 (2018), 1–25. 4