A Web-based Tool for Communication Flow Analysis of Online Chats

H. Ulrich Hoppe Tilman Göhnert Christopher Charles

Laura Steinert

University of Duisburg-Essen
Lotharstr. 63/65
47048 Duisburg, Germany
{hoppe, goehnert, charles,
steinert}@collide.info

GOALS AND PREMISES

Chat as a communication medium has its own characteristics that need to be considered, especially regarding turn taking and interactional coherence. Following suggestions by Suthers et al. [8; 9], operational rules are used as a basis to detect general dependencies or "contingencies". Indicators may use lexical-semantic features but also time lapses between contributions play a crucial role in determining which utterances are to be linked with each other. This paper introduces a web-based system that automatically analyzes text chat logs for dependencies between posts and constructs a contingency graph from it. This graph is then used to further analyze the communication flow in the underlying text chat.

Our approach reconstructs and extends the above mentioned approach of "contingency analysis": First, the approach is refined by incorporating the concept of dialogue act tagging [6; 11] to enrich the basic set of indicators and to exploit existing techniques of linguistic processing. Second, in order to analyze the information flow in the graph, methods such as main path analysis (MPA) [3] enriched by information gathered from the web search algorithms PageRank [7] and HITS [4] are applied. While the latter two have been used frequently outside of their original domain, MPA has not been applied to chat networks before.

IMPLEMENTATION

The implementation uses a network analytics workbench that combines a web-interface for easily defining analysis workflows using a visual language with a multi-agent system as the computational backend [1]. The communication platform is based on SQLSpaces [10] and implements a blackboard architecture, mediating between the user interface, the computational backend, and the analysis agents. The underlying communication protocol is based on exchanging information through tuples placed on the blackboard (i.e., an SQLSpace). Each analysis step is performed by an individual agent. This architecture allows for an easy extension of the workbench by adding further processing agents that can be programmed in several different languages, including Java, R, Python, and Prolog.

In previous applications the workbench had already been used to analyze the evolution of knowledge in wiki environments [2] by incorporating "main path analysis" [3] as an analytic method. In order to analyze chat logs, a number of additional agents have been added. Among these are the *Chat-ECGBuilder* agent, which constructs the extended contingency graph (ECG) based on a chat log, the *Chat-PageRank* agent, which applies a page rank calculation to an ECG, the *Chat-MainPathAnalysis* agent for performing a main path analysis on an ECG, and the *Chat-Visualization* agent, which gives a visual representation of an ECG and of analysis results connected to that ECG. As a programming language for these agents Python was used together with the *NLTK* library[1], which allows natural language processing. Furthermore, the *igraph* network analysis library[2] was used for analyzing and visualizing graphs. Figure 1 shows an example workflow that is based on the modules described above.

EMPIRICAL RESULTS

So far, the automatically generated ECGs were compared to manually constructed graphs using results reported in [8] as a reference. This comparison yielded an F-score based similarity of 83 percent compared to a 97 percent F-score similarity between two manually generated graphs. Although this leaves room for improvement, the similarity values show that the automatically generated ECGs agree to a reasonable degree with contingencies detected by humans. This is further backed up by the inter-network comparison, where the majority of metrics show highly positive correlations for the different graphs based on the same chat log. Looking at the inter-network correlations between the individual metrics, it becomes clear that the rankings have different informative values based on their concepts of unidirectional influence (PageRank and input domain), bidirectional centrality (main paths) and mutual enforcement between two classes of nodes (hubs and authorities).

THE VISUAL REPRESENTATION OF WORKFLOWS

Our workbench facilitates the interactive construction of analysis workflows in a kind of visual programming approach: The "analyst" users may pull together data sources, processing units ("filters"), and export modules for visual rendering or download to form a workflow. Workflows can be shared between analysts and can be re-used with different data sets and/or modified. We believe that the level of visual representation of these workflows also provides an adequate reference for discussing the underlying processing schemes without entering into too much technical detail.

---

[1] http://nltk.org/

[2] http://igraph.sourceforge.net/

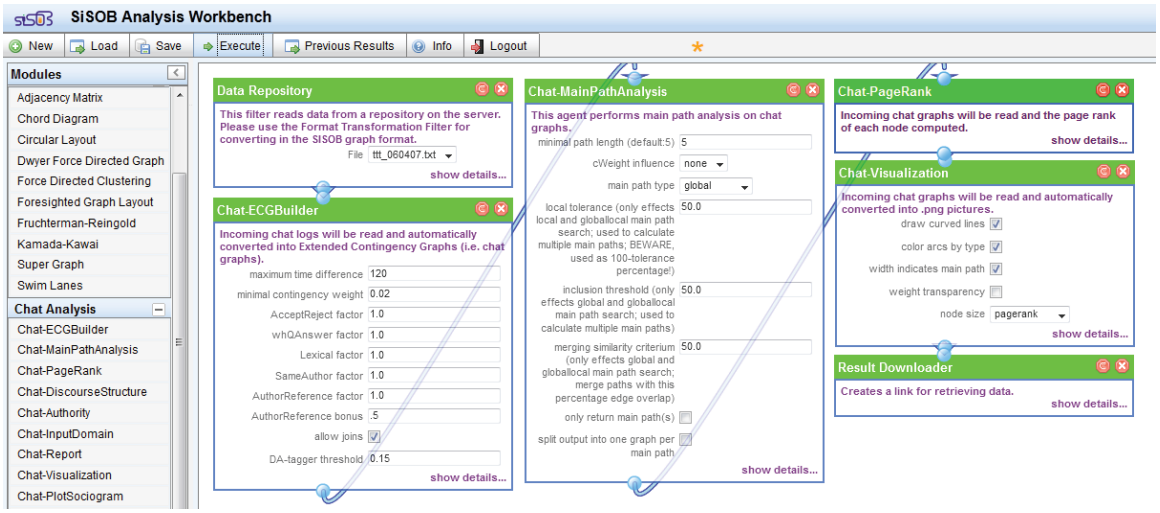

Figure 1. Workbench with ECG filters

Figure 1 shows a workflow in which six different agents are used. In this example the first agent loads a chat log and relays it to a second agent. Here the log is transformed into an ECG. The third component uses that graph to perform a main path analysis. In the next step the page rank of each node is calculated. The final results are then visualized in another component and made available to the user.

SUMMARY AND OUTLOOK

The current version's rules for detecting contingencies try to form a balance between sophistication and simplicity. Typing mistakes are quite common in text chats, yet they are not corrected. Additionally, in order to measure similarity between posts only a removal of stop words and a stemmer are applied. Further lemmatization, e.g. by WordNet [5], might improve the detection of semantic cohesion, yet could also increase the risk of erroneously detected contingencies. In order to avoid such false contingencies, simplicity was chosen over sophistication in this case.

As workflows in the analysis workbench are based on a modular concept and the technical platform supports adding additional features to it easily, variations of workflows are encouraged. The workflow presented here could be modified by adding new input components (e.g. for newsgroup dumps), analysis components (e.g. for pattern detection), or new output components (e.g. alternative visualizations, reports or graph formats).

In future works linguistic features such as coreference resolution could help detecting and filtering existing contingencies. So far text chats without any explicit threading information have been analyzed. However, it could be interesting to incorporate user generated threading information such as it is given in forums or newsgroups. In these systems it is often only allowed to explicitly reference one other message, but by using lexical coherence, author name referencing and syntactical patterns, further dependencies might be detected.

REFERENCES

[1] Göhnert, T., Harrer A., Hecking T., and Hoppe H. U. 2013. *A Workbench to Construct and Re-use Network Analysis Workflows - Concept, Implementation, and Example Case. The 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013).*

[2] Halatchliyski, I., Hecking, T., Göhnert. T., and Hoppe, H. U. 2013. *Analyzing the flow of ideas and profiles of contributors in an open learning community*. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge (LAK '13).* ACM, New York, NY, 66-74.

[3] Hummon, N. P. and Doreian, P. 1989. *Connectivity in a citation network: The development of DNA theory*. Social Networks, 11:39-63.

[4] Kleinberg, J. M. 1999. *Authoritative sources in a hyperlinked environment*. *Journal of the ACM*, 46(5):604-632.

[5] Miller, G. A. 1995. *Wordnet: a lexical database for English*. *Communications of the ACM*, 38(11):39-41.

[6] Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V., and Meteer, M. 2000. *Dialogue act modeling for automatic tagging and recognition of conversational speech. Computational Linguistics*, 26:339-373.

[7] Page, L., Brin, S., Motwani, R., and Winograd, T. 1998. *The PageRank citation ranking: Bringing order to the web.* Technical report, Stanford Digital Library Technologies Project, Stanford Unversity, Stanford, CA, USA.

[8] Suthers, D. D. and Desiato, C. 2012. *Exposing chat features through analysis of uptake between contributions.* In *Proceedings of HICSS 2012.* IEEE Computer Society, 3368-3377.

[9] Suthers, D. D., Dwyer, N., Medina, R., and Vatraou, R. 2010. *A framework for conceptualizing, representing, and analyzing distributed interaction. Int. Journal of Computer-Supported Collaborative Learning*, 5(1): 5-42.

[10] Weinbrenner, S. 2012. *SQLSpaces - a platform for flexible language-heterogeneous multi-agent systems.* Dr. Hut.

[11] Wu, T., M. Khan, F., A. Fisher, T., A. Shuler, L., and M. Pottenger, W. 2005. *Posting act tagging using transformation-based learning. Foundations of Data Mining and Knowledge Discovery*, 6:319-331