

SINNET: Social Interaction Network Extractor from Text

Apoorv Agarwal

Computer Science, Columbia University
New York, NY, USA
apoorv@cs.columbia.edu

Anup Kotalwar

Microsoft, Inc.
Redmond, WA, USA
ankotalw@microsoft.com

Jiehan Zheng

Peddie School
Hightstown, NJ, USA
jzheng-14@peddie.org

Owen Rambow

CCLS, Columbia University
New York, NY, USA
rambow@ccls.columbia.edu

Abstract

In this paper we present a demo of our system: Social Interaction Network Extractor from Text (SINNET). SINNET is able to extract a social network from unstructured text. Nodes in the network are people and links are *social events*.

1 Introduction

Language is the primary tool that people use for establishing, maintaining and expressing social relations. This makes language the real carrier of social networks. In this paper, we present a demo of our system that automatically extracts a social network from raw texts such as literary texts, emails, blog comments and news articles.¹ We take a “social network” to be a network consisting of individual human beings and groups of human beings who are connected to each other through various relationships by the virtue of participating in *social events*. We define social events to be events that occur between people where at least one person is aware of the other and of the event taking place. For example, in the sentence *John talks to Mary*, entities John and Mary are aware of each other and of the talking event. In the sentence *John thinks Mary is great*, only John is aware of Mary and the event is the thinking event.

There has been recent work on extracting social networks from literary text (Elson et al., 2010; He et al., 2013). However, both these works focus on extracting only conversational links between people, signaled in text by quotation marks. They do not extract *social event* links from other parts

¹A web demo is available at <http://nlp.ldeo.columbia.edu/sinnet/>

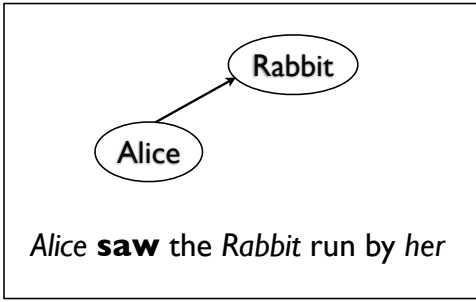
of text such as reported speech and other non-dialogue text. Our system overcomes this limitation.

The rest of the paper is structured as follows: In section 2, we briefly describe the research that has gone into building the system. In section ??, we present the technical details of SINNET and describe our web demo.

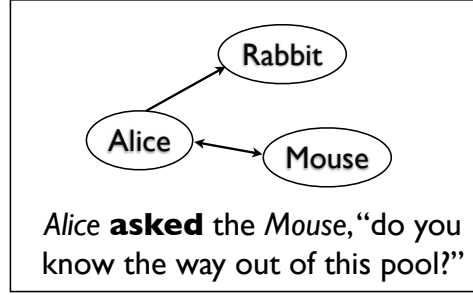
2 Research

The SINNET system is the result of several years of research (Agarwal et al., 2010; Agarwal and Rambow, 2010; Agarwal et al., 2012; Agarwal et al., 2013). In Agarwal et al. (2010), we introduced the notion of *social events*. A *social event* is a *happening* between two people, at least one of whom is cognizant of the other and of the event taking place. At a broad level, there are two types of social events: interaction (**INR**) and observation (**OBS**). INR is a bi-directional event in which both parties are mutually aware of each other. Examples of INR are a meeting or a dinner. OBS is a one-directional event in which only one party is aware of the other. Examples of OBS are thinking about someone, or missing someone.

In Agarwal and Rambow (2010), we presented a preliminary system that uses tree kernels and Support Vector Machines (SVMs) to extract social events from news articles. In Agarwal et al. (2012), we presented a case study on a manually extracted network from *Alice in Wonderland*, showing that analyzing networks based on these social events gives us insight into the roles of characters in the story. Also, static network analysis has limitations which become apparent from our analysis. We propose the use of dynamic network analysis to overcome these limitations. In Agarwal et al. (2013), we introduce two baselines for

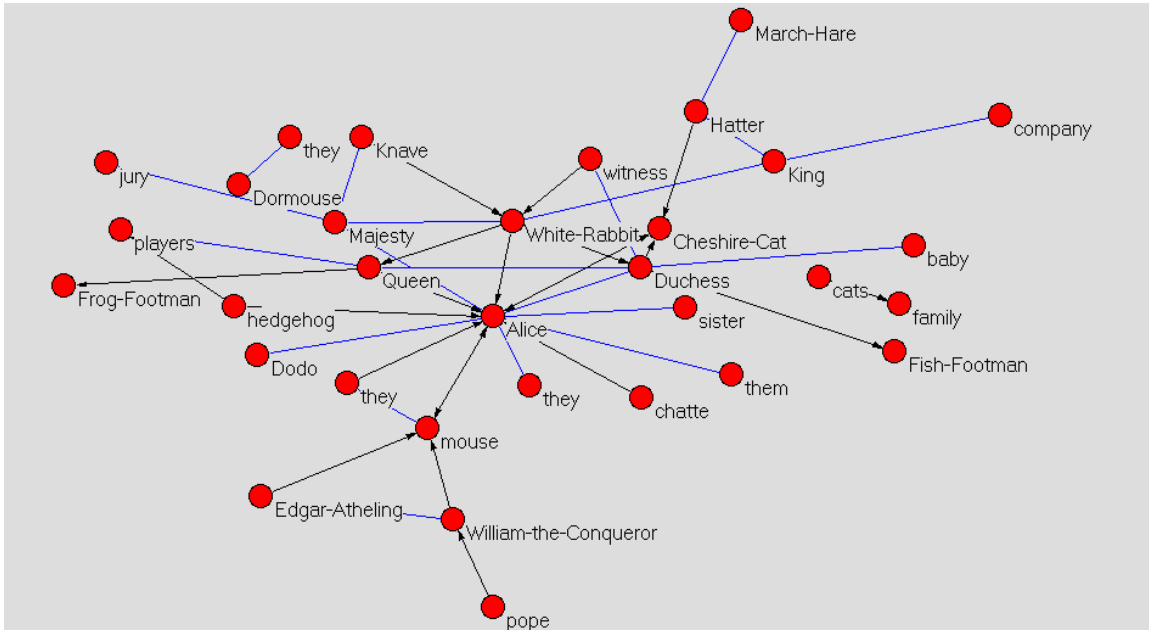


(a)



(b)

Figure 1: Two figures exemplifying the meaning of social events and social network.

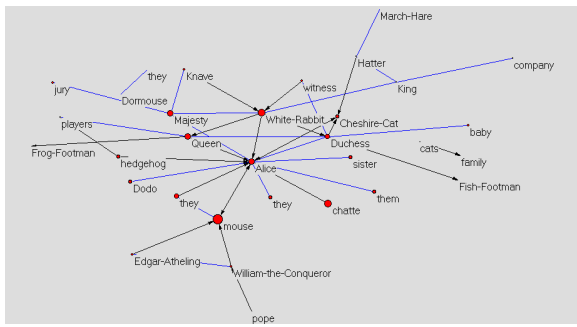
Figure 2: Social network of the entire *Alice in Wonderland*.

the social event extraction task and show that our system trained on a news corpus using tree kernels and support vector machines beats the baseline systems by a statistically significant margin. We also show that while the performance of our system on detecting social events in *Alice in Wonderland* achieves an F-measure of 61%, the un-weighted network built using these detected social events is not statistically distinguishable from the un-weighted gold network according to popularly used network measures.

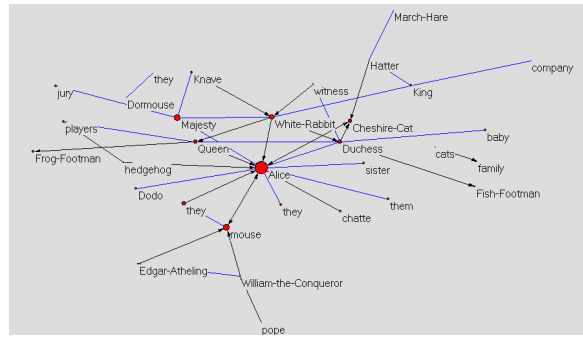
Figure 1 shows two figures exemplifying the meaning of social events and social networks. In the first figure, there are three entity mentions: Alice, Rabbit and her (co-referential with Alice). There is an OBS event between Alice and Rabbit triggered by the word in bold – *saw*. The direction of the event is from the observer to the one

being observed. In the second figure there are two entity mentions: *Alice* and *Mouse*. There is a bi-directional interaction link between the Alice and Mouse triggered by the word *asked*.

Figure 2 shows the network extracted from an abridged version of *Alice in Wonderland* (Agarwal et al., 2012). Figure 3 shows the output of running the Hubs and Authority algorithm (Kleinberg, 1998) on the network. In information retrieval, an *authority* is a webpage that many *hubs* point to and a *hub* is a webpage that points to many *authorities*. In our network, webpages are synonymous to characters. Figure 3a shows the hubs in decreasing order of hub weights. Figure 3b shows the authorities in decreasing order of authority weights. We see that the main character of the story, *Alice*, is the main authority but not the main hub. This network may be used for other

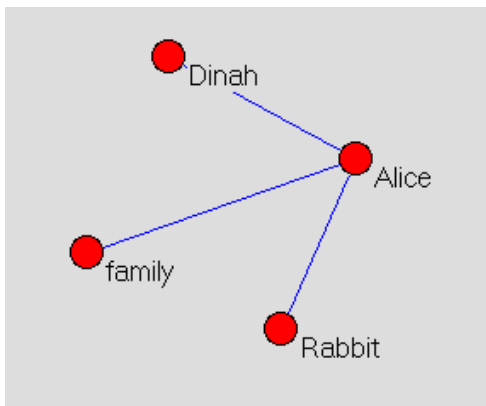


(a) Hubs in order of decreasing hub weight: Mouse, White Rabbit, Alice

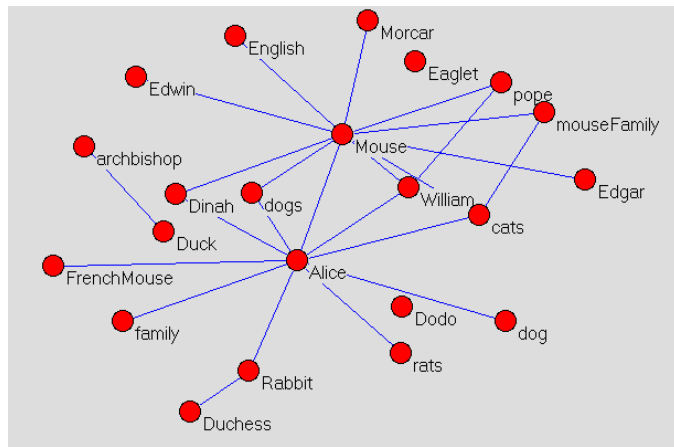


(b) Authorities in order of decreasing authority weights: Alice, Majesty (King), Mouse, White Rabbit

Figure 3: Hub and authority weights of characters. Larger the node, higher its weight.



(a) Network at the end of Chapter 1



(b) Network at the end of Chapter 3

Figure 4: Dynamic network plots of *Alice in Wonderland*

types of social network analyses such as finding communities.

In Agarwal et al. (2012), we argued that a static network does not bring out the true nature of a network. For example, even though the centrality of the *Mouse* in a static network is high, a dynamic network analysis shows that the mouse is central only in one chapter of the novel (Chapter 3 – *The drying ceremony*). Figure 4 shows the the network at the end of chapter 1 and chapter 3.

3 System details and Web demo

SINNET is fully implemented in Java. Following is a list of external off-the-shelf tools used by our current pipeline: Jet sentence splitter, Jet NER (Grishman et al., 2005), Stanford parser (Klein and Manning, 2003), SVM-Light-TK (Moschitti, 2006),

Input to SINNET may be provided in two formats: as raw text or text with entity annotations.

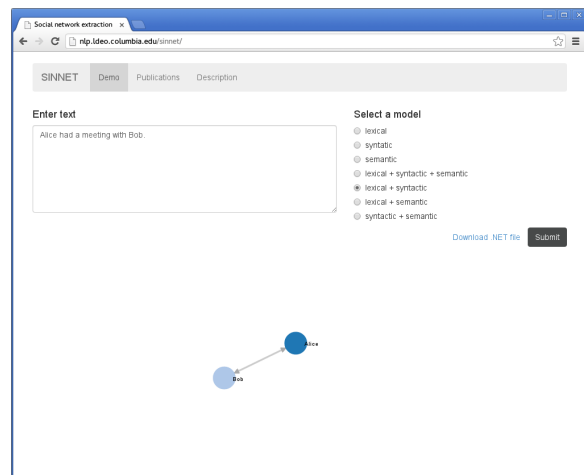


Figure 5: Image of our web demo

If the text is input as raw text without any entity annotations, SINNET first runs an off-the-shelf named entity recognizer and co-reference resolution (NER) tool. Currently, we run Jet (Grish-

man et al., 2005), but an interface makes it easy to plug-in any other NER tool. Once the text is annotated with entity mentions, for each sentence, for each entity mention pair per sentence, we create *test* examples in the format that our models accept. We use tree kernels with Support Vector Machines (SVM) for our models. Details of our system may be found in Agarwal and Rambow (2010). Any sentence splitter may be plugged in. Currently, we are using Jet’s sentence splitter. Finally, the examples are fed to the models for prediction. The output is stored as a list of entities and their relations in a standard graph format. Currently, the output formats include graph modeling language (gml) and Pajek’s .net format (Batagelj and Mrvar, 1998).

In many situations, the input text may already have entity mentions annotated and co-referenced. In these situations, SINNET will accept these gold entity mention annotations instead of running the NER tool. The rest of the processing remains the same as above.

Figure 5 shows an image of our web demo.² The demo has a text box for entering text. We have various models that use features from three levels of natural language abstractions: lexical, syntactic and semantic. Users of the web demo are given the option of selecting the type of model used for making predictions. We have seven models in place: lexical, syntactic, semantic and all combinations of these three types. Once the user inputs a text and selects the type of model, we display the extracted network and make the file with the extracted network (which is in a standard graph format such as .gml/.net) available for download. Our web demo has two other tabs: one listing the publications relevant to SINNET and the other mentioning technical details and capabilities of our web demo.

Acknowledgments

This paper is based upon work supported in part by the DARPA DEFT Program. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government. Kotalwar participated in the work described in this paper while at Columbia University.

References

- Apoorv Agarwal and Owen Rambow. 2010. Automatic detection and classification of social events. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1024–1034, Cambridge, MA, October. Association for Computational Linguistics.
- Apoorv Agarwal, Owen C. Rambow, and Rebecca J. Passonneau. 2010. Annotation scheme for social network extraction from text. In *Proceedings of the Fourth Linguistic Annotation Workshop*.
- Apoorv Agarwal, Augusto Corvalan, Jacob Jensen, and Owen Rambow. 2012. Social network analysis of alice in wonderland. In *Proceedings of the NAACL-HLT 2012 Workshop on Computational Linguistics for Literature*, pages 88–96, Montréal, Canada, June. Association for Computational Linguistics.
- Apoorv Agarwal, Anup Kotalwar, and Owen Rambow. 2013. Automatic extraction of social networks from literary text: A case study on alice in wonderland. In *The Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*.
- Vladimir Batagelj and Andrej Mrvar. 1998. Pajek-program for large network analysis. *Connections*, 21(2):47–57.
- David K. Elson, Nicholas Dames, and Kathleen R. McKeown. 2010. Extracting social networks from literary fiction. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 138–147.
- Ralph Grishman, David Westbrook, and Adam Meyers Proc. 2005. NYU’s english ace 2005 system description. In *ACE Evaluation Workshop*.
- Hua He, Denilson Barbosa, and Grzegorz Kondrak. 2013. Identification of speakers in novels. *The 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*.
- Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pages 423–430.
- Jon Kleinberg. 1998. Authoritative sources in a hyperlinked environment. In *Proc 9th ACM SIAM Symposium on Discrete Algorithms*, pages 668–677.
- Alessandro Moschitti. 2006. Making tree kernels practical for natural language learning. In *Proceedings of European chapter of Association for Computational Linguistics*.

²Available at <http://nlp.ldeo.columbia.edu/sinnet/>