Automatic Extraction of Social Networks from Literary Text: A Case Study on *Alice in Wonderland*

Apoorv Agarwal
Dept. of Computer Science
Columbia University
New York, NY, U.S.A.
apoorv@cs.columbia.edu

Anup Kotalwar
Microsoft, Inc.
Redmond, WA, U.S.A.
ankotalw@microsoft.com

Owen Rambow
CCLS
Columbia University
New York, NY, U.S.A.
rambow@ccls.columbia.edu

Abstract

In this paper we present results for two tasks: *social event detection* and *social network extraction* from a literary text, *Alice in Wonderland*. For the first task, our system trained on a news corpus using tree kernels and support vector machines beats the baseline systems by a statistically significant margin. Using this system we extract a social network from *Alice in Wonderland*. We show that while we achieve an F-measure of about 61% on social event detection, our extracted unweighted network is not statistically distinguishable from the un-weighted gold network according to popularly used network measures.

1 Introduction

Social network analysis affects a wide range of academic disciplines and practical applications: psychology (Seidman, 1985; Koehly and Shivy, 1998), anthropology (Sanjek, 1974; Johnson, 1994; Hage and Harary, 1983), political science (Knoke, 1990; Brandes et al., 2001), literary theory (Moretti, 2005), management (Tichy et al., 1979; Cross et al., 2001; Borgatti and Cross, 2003), and crime prevention and intelligence (Sparrow, 1991). In the past, social networks were constructed through interviews, surveys and experiments. With the advent of the internet and online social networks, researchers have started constructing networks using meta-data that reflects interactions, such as self-declared friendship linkages, sender-receiver email linkages, comments on a common blog-post, etc. However, these methodologies of creating social networks ignore a vastly rich network expressed in the unstructured text of such sources. Moreover, many rich sources of social networks remain untouched simply because there is no meta-data associated with them (literary texts, news stories, historical texts). There have been recent efforts to extract social networks from text by mining interactions between people expressed linguistically in unstructured text (Elson et al., 2010; He et al., 2013). However, these approaches are restricted to extracting interactions signaled by quoted speech.

In this paper, we present results for extracting a social network from *Alice in Wonderland* that is not restricted to interactions signaled by quoted speech. We define a social network for a fictional text as follows: nodes are characters and links are *social events*. Two nodes in the network are connected if the characters engage in a social event. We introduced the notion of *social events* in our previous work (Agarwal et al., 2010), in which we presented our annotation scheme for annotating social events on the Automatic Content Extraction (ACE-2005) corpus. We presented a preliminary system for social event detection and classification in Agarwal and Rambow (2010). This system was trained and tested only on the ACE-2005 corpus. A priori, it is unclear if a system trained on a news corpus will be able to extract a high quality social network from a text from a very different genre (literary fiction). There are many syntactic and lexical differences between these genres. For example, news corpora have almost no questions, very little dialog presented as direct speech, and very little use of the first and second person pronouns. The vocabulary in literature can also be very different (relating to, say, whaling, passion, or teenage angst rather than current events). In this paper, we make two novel contributions. First, in an intrinsic evaluation, we show that our system without any domain (or genre) adaptation performs reasonably well on a new genre. Second, in an extrinsic evaluation, we show that the social network that our system extracts is not statistically distinguishable from the underlying gold network

in terms of various standard and popularly used network analysis metrics.

The paper is structured as follows: In section 2 we describe our notion of social events and the annotated data we use for training and testing in our experiments. In section 3, we briefly describe the tree kernel structures used by our system to detect social events in text. Section 4 presents the social network analysis metrics we use to evaluate the quality of the predicted network. In section 5, we present the experiments and results. We discuss some related work in section 6 and conclude in section 7 and mention future directions of research.

2 Social Events and Data

In Agarwal et al. (2010), we defined a **social event** as an event in which two people *interact* such that for at least one person, the interaction is **deliberate** or **conscious**. Put differently, at least one person must be aware of the interaction.

> [Toujan Faisal], 54, {said} [she] was {informed} of the refusal by an [Interior Ministry committee] overseeing election preparations.

In the above example, the people (or groups of people) involved in social events are *Toujan Faisal* and the *Interior Ministry committee*. There are two social events in this example: one described by the word *said*, in which *Toujan Faisal* is *talking about* the committee, and the other described by the word *informed*, in which *Toujan Faisal* presumably has a mutual interaction with the committee.

We annotated two corpora for social events: 1) The Automatic Content Extraction (ACE) data-set[1] (Agarwal et al., 2010) and 2) the *Alice in Wonderland* data-set[2] (Agarwal et al., 2012).

For each pair of entity mentions in a sentence, if the annotators annotate a social event, we count this as a positive example for the task of social event detection. If no social event is annotated between a pair of entity mentions, we count this as a negative example. Note that we only consider pairs of entity mentions that correspond to different entities; our annotation scheme disregards self-interactions (talking to oneself).

[1]Version: 6.0, Catalog number: LDC2005E18

[2]http://www.gutenberg.org/ebooks/19551

We use all of ACE data for training and refer to this data-set as **ACE-train**. We use all of *Alice in Wonderland* data for testing and refer to this data-set as **Alice-test**. The distribution of these data-sets is presented in Table 1.

Data-set	# of social events	# of No-event
ACE-train	396	1,101
Alice-test	81	128

Table 1: The distribution of social events in the training and test sets used for experiments

3 SINNET: Social Interaction Network Extractor from Text

In Agarwal and Rambow (2010), we presented a preliminary system that extracts social events from news articles. We used Support Vector Machines (SVM) in conjunction with tree kernels for detecting social events between pairs of entities, called target entities, that co-occur in a sentence. Following is a brief description of the tree structures that we used for building our models. We used the Stanford parser's (Klein and Manning, 2003) phrase structure and dependency tree representations. Of the following tree structures, 1-3 have previously been proposed by Nguyen et al. (2009) for the relation extraction task, while we introduced the fourth structure in Agarwal and Rambow (2010) for social event detection task.

1. PET: This refers to the smallest phrase structure tree that contains the two target entities.

2. Grammatical Relation (GR) tree: This refers to the smallest dependency tree that contains the two target entities. We replace the words (in the nodes of the tree) with their corresponding grammatical roles. For example, in Figure 1, if we replace *Toujan_Faisal* by *nsubj*, *54* by *appos*, *she* by *nsubjpass* and so on, we will get a GR tree.

3. Grammatical Relation Word (GRW) tree: We get this tree by adding the grammatical relations as separate nodes between a node and its parent. For example, in Figure 1, if we add *nsubj* as a node between *T1-Individual* and *Toujan_Faisal*, add *appos* as a node between *54* and *Toujan_Faisal*, and so on, we will get a GRW tree.

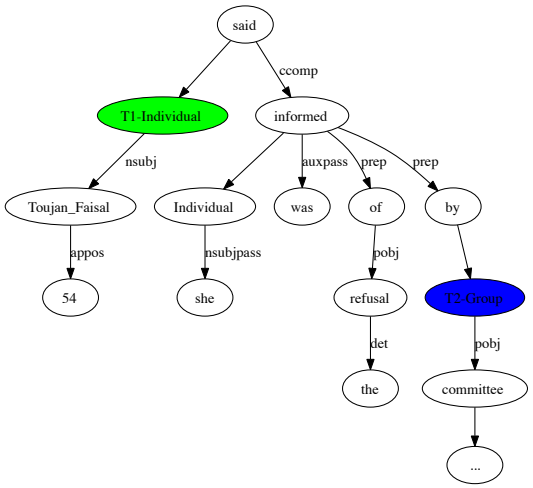

Figure 1: Dependency parse tree for the sentence (in the ACE corpus): *[Toujan Faisal], 54, said [she] was informed of the refusal by an [Interior Ministry committee] overseeing election preparations.*

4. Sequence in GRW tree (SqGRW): This is the sequence of nodes from one target to the other in the GRW tree. For example, in Figure 1, this would be *Toujan_Faisal nsubj T1-Individual said ccomp informed prep by T2-Group pobj committee*.

We also use combinations of the aforementioned structures. For example, PET_GR_SqGRW refers to a kernel that considers a linear combination of three structures (PET, GR and SqGRW) for calculating similarities between examples. We use the Partial Tree kernel, first proposed by Moschitti (2006a), to calculate similarities between these tree structures.

In this paper, we use a Bag of Words Model (BOW) as a baseline. In the BOW model, each sentence is represented as a vector of three feature spaces. The first feature space encodes the presence and absence of words between the start of sentence and the start of the first target entity mention. The second feature space encodes the presence and absence of words between the end of the first target entity mention and the start of the second target entity mention. The third feature space encodes the presence and absence of words between the end of the second target entity mention and the end of the sentence. This feature space has previously been used by GuoDong et al. (2005) for the relation extraction task on ACE. We use stemming and remove stop words from our feature space.

4 Social Network Analysis Metrics

In this section we briefly discuss some of the most popular social network analysis (SNA) metrics used by researchers to mine information from networks. We evaluate the social network extracted by our system with the gold network using these metrics. At a broad level, SNA researchers are interested in measuring importance of nodes in the network and in finding community structures in the network. To measure the importance of nodes, they use the notion of centrality. Following are the centrality measures that are often used in the literature (Freeman, 1979):

1. **Degree centrality** of a node in the network measures the number of incoming and outgoing links from the node. Degree centrality is viewed as an index of a node's *communication activity*.

2. **Betweenness centrality** of a node in the network measures the frequency with which a point falls between pairs of other nodes on the shortest paths connecting them. Nodes with high betweenness centrality are strategically located on the communication paths linking pairs of others, thus having the potential of influencing the group by withholding or distorting information (Bavelas, 1948; Shaw, 1954; Shimbel, 1953).

Another aspect of social networks that SNA researchers are interested in has to do with finding communities in the network and structural properties of networks. Following are some basic metrics used for this task:

1. **Graph density**: The density of a graph is the ratio of the number of edges to the number of possible edges. This measures how close the network is to being complete.

2. **Connected components**: a connected component of an undirected graph is a subgraph in which any two vertices are connected to each other by some path. The number of connected components is an indication of the overall connectivity of the network.

3. **Triads**: A triad is a set of three parties connected pair-wise to each other. In his seminal work, Simmel (1950) argued that triads are a fundamental unit of sociological analysis. He argued that three actors in a triad may allow qualitatively different social dynamics that cannot be reduced to individuals or dyads.

5 Experiments and Results

We present experiments and results for two tasks: social event detection and social network extraction. We use the same system for both tasks; the first task is an intrinsic evaluation of our system, while the second task presents an extrinsic evaluation of our system. In the following subsections, we describe the individual tasks, their experimental set-up followed by a discussion of results.

5.1 Social Event Detection

Task description: Given a pair of entity mentions in a sentence, we evaluate how well we identify the occurrence of a social event between these two entities. This is a binary task with two classes: presence/absence of a social event. We evaluate using F-measure on the presence of social events.

Tree structure	P	R	F
BOW	34.62	77.78	47.91
PET	58.54	59.26	58.90
GRW	49.14	70.37	57.87
SqGRW	49.59	74.07	59.41
PET_GR	56.32	60.49	58.33
PET_GR_SqGRW	56.82	61.73	59.17
GR_SqGRW	54.37	69.14	60.87
GRW_SqGRW	51.30	72.84	60.20
GR_GRW_SqGRW	50.47	66.67	57.45

Table 2: Results for training on ACE-train and testing on Alice-test. P refers to Precision, R refers to Recall, F refers to F1-measure. Terminology used for the tree structures is explained in detail in Section 3.

Experimental set-up: For all our experiments, we use the SVM-Light-TK package (Moschitti, 2006b) for training models. We use the default parameters of the package to avoid over-fitting. Since we are interested in knowing how well we do at finding the social events, we report Precision, Recall and F-measure of the class of interest (the minority class) instead of % accuracy. For training, we set the $-j$ parameter of the package to the ratio of the number of negative examples to the number of positive examples in the training data-set. The $-j$ parameter assigns a weight to the minority class. Since the SVM optimizes for accuracy, if we do not set this parameter at the time of training, the learner may learn a trivial hyperplane classifying all the examples as negative (the majority class), thus achieving a high accuracy. By assigning a weight to the examples in the minority class, we increase the cost of mis-classifying these examples, thus forcing the learner to find a non-trivial hyperplane. We use the model trained just on bag-of-words (BOW) as a baseline.

Discussion of results: Table 2 presents the results for models trained on ACE-train and tested on Alice-test. We use the tree kernel structure combinations described in section 3. The results show that building a model using tree kernels outperforms the bag-of-words baseline model by an absolute 10% F1-measure. This difference is statistically significant with $p < 0.05$.

5.2 Social Network Extraction

In this section, we provide results to establish that the un-weighted network extracted using social event detection models is *close* to the true underlying network.

Task description and experimental set-up: Using our social event detection models, we build an unweighted network of entities in *Alice in Wonderland*. For this task, we report the *distance* between the SNA metrics in the predicted and gold networks. Table 3 summarizes the metrics we use for our evaluation and elaborates on the meaning of *distance*. We use two baseline systems for our evaluation:

1. **B-Simple**: For this baseline, we create a network by linking all pairs of entity mentions (of different entities) that appear in the same sentence.

2. **B-BOW**: This is the network extracted by building a model that uses bag-of-words features for training.

Discussion of results: Table 4 shows results for the SNA metrics (Section 4) for the kernel combinations used to extract a network from *Alice in Wonderland*. The terminology used in Table 4 is explained in Table 3.

Symbol	Name and explanation
P, R, F, %A	Precision, Recall, F1-measure and Accuracy
p	McNemar's two-sided p-value significance test. We linearize the adjacency matrix of the predicted and gold network and test if these two vectors are significantly different.
D, B	Degree and Betweenness centrality. For each of these centrality measures, we find the centrality of nodes in the network, represented by a vector $\vec{v}$, where the i^{th} component of the vector is the centrality of the i^{th} node. We then calculate the Euclidean distance between the predicted and gold vectors, which measures the difference in degree centralities of the nodes in the two networks.
S, CC, T	Network density, number of connected components and number of triads respectively. For these measures, we calculate the difference between the gold and the predicted network. For example, a value of 6 in the column labeled CC and the row labeled Alice in Table 2 is the difference in number of connected components found in the predicted network and the gold network.

Table 3: Terminology used to present results in Table 4

	Stats					Centrality		Community		
System	P ↑	R ↑	F ↑	%A ↑	p ↑	D ↓	B ↓	S ↓	CC ↓	T ↓
B-Simple	0.40	1.00	0.57	97.58	0.0000	19.67	0.57	0.0245	23	439
B-BOW	0.47	0.87	0.61	98.15	0.0000	13.00	0.34	0.0145	14	165
PET	**0.77**	0.65	**0.71**	**99.12**	0.12	**6.93**	0.15	0.0026	2	**0**
GRW	0.61	0.68	0.65	98.78	0.38	8.31	0.10	0.0019	1	49
SqGRW	0.64	**0.81**	**0.71**	98.93	0.01	8.77	0.11	0.0045	6	34
PET_GR	0.72	0.60	0.66	98.96	0.15	7.75	0.12	0.0026	2	28
PET_GR _SqGRW	0.72	0.65	0.68	99.01	0.41	7.87	0.10	0.0016	3	18
GR_SqGRW	0.67	0.70	0.68	98.93	**0.75**	8.77	**0.06**	**0.0008**	1	23
GR_GRW _SqGRW	0.63	0.68	0.66	98.83	0.55	8.31	0.10	0.0013	**0**	6

Table 4: Results comparing the two baseline systems (B-Simple and B-BOW) with the models trained on the tree kernel structures discussed in Section 3. ↑ means greater value is better. ↓ means lesser value is better. Network density of the gold network is 0.0166. The number of connected components in the gold network are 34. The number of triads in the gold network are 103.

Table 4 shows that all the systems trained using tree kernels are better than the two baselines across all SNA metrics. In terms of F-measure, both the baselines perform significantly worse than the tree kernel based systems. In terms of p-value, both the baselines are significantly different from the gold network, whereas none of the tree kernel based systems are significantly different from the gold network. In terms of the distance between the vectors of centrality measures (D, B) for the predicted and gold network – the distance is larger for the baselines, which means that the difference in centrality measures of nodes in the baseline system and the gold network is larger than the difference in centrality measures of nodes in the other systems and the gold network. The difference in network densities of the baseline and gold network is also larger than the difference in network densities (S) of the other systems and the gold network. The same is the case with the number of connected components (CC) and the number of triads (T). Using these results, we conclude that the network predicted by our system that uses tree kernels performs well in terms of extracting an unweighted, undirected network from *Alice in Wonderland*. In particular, the tree structure derived from the phrase structure tree (PET) performs the best on most of the SNA metrics.

6 Literature Survey

With the advent of the internet and social media, researchers have got access to different forms of communication such as Email (Klimt and Yang, 2004; Rowe et al., 2007), online discussion threads (Hassan et al., 2012), Slashdot, Epinions, and Wikipedia (Jure Leskovec and Kleinberg, 2010). There have also been approaches of extracting networks based on Information Retrieval techniques – Jing et al. (2007) extract a network from conversational speech data. The events they are interested in are custody, death, hiding, liberation, marriage, migration, survival and violence. Tang et al. (2008) aim at extracting and mining academic social networks. Aron Culotta and McCallum (2004) extract social networks and contact information from email and the Web. Mori et al. (2006) mine networks based on the collective context in which entities appear.

Our notion of social network is different from the aforementioned work. We are interested in extracting *interaction* networks from unstructured text. In terms of our goals, our work is closest to the work by Elson et al. (2010) and He et al. (2013). Elson et al. (2010) and He et al. (2013) are also interested in extracting a social network from literary texts. However, they restrict their definition of *interaction* to interactions that are signaled by quotation marks. Our system does not have this limitation and is therefore able to extract interaction links appearing even in reported speech (non-dialogue text).

7 Conclusion and Future Work

In this paper, we have addressed the problem of extracting a social network from literary narrative text. We have used our previous system that detects social events to extract a network from *Alice in Wonderland*. This system was trained on news articles and has never been tested out of domain. Our evaluation on *Alice in Wonderland* has two components: a standard intrinsic evaluation in terms of the detection of social events, and an extrinsic evaluation which measures how well the un-weighted network formed by the extracted social events mirrors the gold social network. For the extrinsic evaluation, we use various network measures such as centrality or density. We show that while we achieve an F-measure of about 61% on the intrinsic evaluation, our extracted network is not statistically distinguishable from the gold network according to the various network measures.

In the future, we will apply our system to more literary texts. We are currently acquiring annotations on 19th century novels such as *Emma* by Jane Austen. We will also apply our system to other genres such as historical documents.

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.

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.

Ron Bekkerman Aron Culotta and Andrew McCallum. 2004. Extracting social networks and contact information from email and the web. *In First Conference on Email and Anti-Spam (CEAS)*.

A. Bavelas. 1948. A mathematical model for group structures. *Human Organization*, 7:16–30.

Stephen P. Borgatti and Rob Cross. 2003. A relational view of information seeking and learning in social networks. *Management science*.

U. Brandes, J. Raab, and D. Wagner. 2001. Exploratory network visualization: Simultaneous display of actor status and connections. *Journal of Social Structure*.

Rob Cross, Andrew Parker, and Laurence Prusak. 2001. Knowing what we know:-supporting knowledge creation and sharing in social networks. *Organizational Dynamics*.

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.

Linton C. Freeman. 1979. Centrality in social networks conceptual clarification. *Social Networks*, 1 (3):215–239.

Zhou GuoDong, Su Jian, Zhang Jie, and Zhang Min. 2005. Exploring various knowledge in relation extraction. In *Proceedings of 43th Annual Meeting of the Association for Computational Linguistics*.

P. Hage and F. Harary. 1983. *Structural models in anthropology*. Cambridge University Press.

Ahmed Hassan, Amjad Abu-Jbara, and Dragomir Radev. 2012. Extracting signed social networks from text. In *Workshop Proceedings of TextGraphs-7 on Graph-based Methods for Natural Language Processing*, TextGraphs-7 '12, pages 6–14, Stroudsburg, PA, USA. Association for Computational Linguistics.

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)*.

Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2007. Extracting social networks and biographical facts from conversational speech transcripts. *45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic.*

J. C. Johnson. 1994. Anthropological contributions to the study of social networks: A review. *Advances in social network analysis: Research in the social and behavioral sciences*.

Daniel Huttenlocher Jure Leskovec and Jon Kleinberg. 2010. Signed networks in social media. *In Proceedings of the 28th international conference on Human factors in computing systems*.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pages 423–430.

Bryan Klimt and Yiming Yang. 2004. Introducing the enron corpus. *In proceedings of the First Conference on Email and Anti-Spam (CEAS)*.

D. Knoke. 1990. *Political Networks: The structural perspective*. Cambridge University Press.

Laura M. Koehly and Victoria A. Shivy. 1998. Social network analysis: A new methodology for counseling research. *Journal of Counseling Psychology*.

Franco Moretti. 2005. *Graphs, Maps, Trees: Abstract Models for a Literary History*. Verso.

Junichiro Mori, Takumi Tsujishita, Yutaka Matsuo, and Mitsuru Ishizuka. 2006. Extracting relations in social networks from the web using similarity between collective contexts. In *The Semantic Web-ISWC 2006*, pages 487–500. Springer.

Alessandro Moschitti. 2006a. Efficient convolution kernels for dependency and constituent syntactic trees. In *Proceedings of the 17th European Conference on Machine Learning*.

Alessandro Moschitti. 2006b. Making tree kernels practical for natural language learning. In *Proceedings of European chapter of Association for Computational Linguistics*.

Truc-Vien T. Nguyen, Alessandro Moschitti, and Giuseppe Riccardi. 2009. Convolution kernels on constituent, dependency and sequential structures for relation extraction. *Conference on Empirical Methods in Natural Language Processing*.

Ryan Rowe, German Creamer, Shlomo Hershkop, and Salvatore J Stolfo. 2007. Automated social hierarchy detection through email network analysis. *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 109–117.

R. Sanjek. 1974. What is social network analysis, and what it is good for? *Reviews in Anthropology*.

S. B. Seidman. 1985. Structural consequences of individual position in nondyadic social networks. *Journal of Mathematical Psychology*.

Marvin E. Shaw. 1954. Group structure and the behavior of individuals in small groups. *The Journal of Psychology*, 38(1):139–149.

Alfonso Shimbel. 1953. Structural parameters of communication networks. *The bulletin of mathematical biophysics*, 15(4):501–507.

Georg Simmel. 1950. *The Sociology of Georg Simmel*. The Free Press, New York.

Malcolm K. Sparrow. 1991. The application of network analysis to criminal intelligence: An assessment of the prospects. *Social networks*.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998. ACM.

Noel M. Tichy, Michael L. Tushman, and Charles Fombrun. 1979. Social network analysis for organizations. *Academy of Management Review*, pages 507–519.