Content Analysis Through the Machine Learning Mill

Vivi Nastase* Sabine Koeszegi† Stan Szpakowicz*‡

*School of Information Technology and Engineering, University of Ottawa
800 King Edward Avenue, Ottawa, Ontario K1N 6N5, Canada
{vnastase,szpak}@site.uottawa.ca

†Faculty of Business, Economics and Statistics, University of Vienna
Bruenner Strasse 72, A-1210 Vienna, Austria
sabine.koeszegi@univie.ac.at

‡Institute of Computer Science, Polish Academy of Sciences
Ordona 21, 01-237 Warszawa, Poland

Abstract

We present an analysis of partial automation of content analysis using machine learning methods. We use a decision-tree induction system to learn from manually categorized negotiation transcripts of electronic buyer-seller negotiations. The data we use were gathered using the Web-based negotiation support systems Inspire and SimpleNS. We experiment with various ways of representing the data to find the solution that gives the best results. The experiments show that we can identify, in relatively small data sets, linguistic features of interest for the detection of negotiation behaviour and negotiation-specific topics.

1 Introduction

The analysis of textual messages collected in the course of a negotiation promises to reveal useful information. In contrast with the analysis of surveys on negotiation behaviour, the direct examination of behaviour tells us what negotiators *actually do*, rather than what they plan to do or they thought they did. Furthermore, the analysis of actual behaviour not only lets us understand negotiation processes better, but also predict whether there will be an agreement. More knowledge about the effects of behaviour on negotiation processes also helps determine how better to train negotiators to negotiate effectively (Weingart et al., 2004). All in all, language and discourse analyses have become important analytical tools for the study of negotiations (Putnam, 2003).

Recently there appeared several publications on the methodological issues of negotiation research (Druckman, 2005; Putnam, 2003; Weingart et al., 2004). The journal *International Negotiation* has a special issue on this topic (Carnevale and Drel, 2005). In these publications,

content analysis has received much attention as an important research method in negotiation analysis. Content analysis, which draws on the concept of grounded theory (Glaser and Strauss, 1967), was developed specifically for investigating problems where the content of communication serves as the basis of inference (Holsti, 1969). The method, which arose from communication research (Krippendorff, 1980), is applied to systematic analysis of textual material (Mayring, 2002).

Content analysis typically includes the following steps (Srnka and Koeszegi, 2007).

Transcription The qualitative material is collected and transcribed into textual material, usually from audio or visual sources.

Unitization The textual material is divided into units for further analysis. At this stage, researchers decide what types of units (speaking turn, sentence, thought, and so on) to use for coding and analysis.

Categorization Categories relevant to the research questions are developed and revised through an iterative process of analysis.

Coding The unitized data is categorized, that is, each unit is assigned a category.

Ideally, more than one researcher (coder) performs every stage, and the results of each coder are compared at each step to assure inter-coder reliability of the findings (Srnka and Koeszegi, 2007). The whole process of content analysis is often seen as an insurmountable task, since – if performed with scientific rigour – it is extremely labour-intensive (Weingart et al., 2004). That is why qualitative research has been restricted, in most cases, to small samples, as in-depth qualitative analysis proceeds at human pace. We want to see whether automatic content analysis methods are feasible, given small manually annotated training data sets. If automation is possible, we will be able to tag larger data sets automatically.

Content Analysis systems available on the market include QDA Miner[1], N6 and NVivo[2]. Such systems help process the data by converting them into a standard format and performing tasks designed to assist the user in manual annotation: word-frequency and co-occurrence analysis, pattern identification and various statistical tools. Our method differs from typical content analysis tools in that we monitor how well a system can learn from a small amount of manually annotated data, so that it then can run unassisted on larger data sets. We expect that it will be possible to install a learning component in a semi-automatic content analysis system; to this end, we study how machine learning techniques help in the task of content analysis.

Automated analysis of negotiation transcripts has already proved promising. Sokolova et al. (2004) and Sokolova and Szpakowicz (2005) analyze textual data of electronic negotiations. The purpose is to explore how language is used differently in different situations. The authors seek to discover patterns that distinguish negotiations with different outcomes (successful or unsuccessful) or negotiators playing different roles (buyers or sellers), and to associate behaviour with such linguistic patterns. For example, they observed that sellers use more frequently certain linguistic patterns which express persuasion, while for buyers substantive

[1] http://www.provalisresearch.com
[2] http://www.qsr.com.au

expressions predominate (Sokolova et al., 2004). Successful negotiations also have their own linguistic "fingerprint" (Sokolova and Szpakowicz, 2005).

Automated text analysis methods face two major problems. The data collected in computer mediated communication are unedited, so there is much noise, both lexical and grammatical, introduced by careless users and non-native speakers of the language of interaction. Only part of this noise can be eliminated automatically. The other problem is how to assign meaning to units. It is a challenge to analyse the meaning of a unit by itself, but the problem is further complicated by the fact that communication is context-dependent. On the one hand, the actual sequence of interaction is relevant – the previous and subsequent utterances (Brett, 1998; Mayring, 2003). On the other hand, the social context of the interaction itself determines how communication is meant *and* interpreted: the role of the communicator (Friedman and Gal, 1991; Sokolova et al., 2004); their social status (Dubrovsky et al., 1991; Weisband et al., 1995); the gender of the communicator and the addressee of the message (Adrianson, 2001; Eagly, 1987; Koeszegi et al., 2006; Lee, 2003); the cultural background of the communicators (Adler, 1993; Brett, 1998; Graham et al., 1994). Finally, non-verbal cues – gestures, tone of voice, intonation and so on – influence the way in which utterances are or should be interpreted.

In the case of computer-mediated communication, many of these inter-personal and social context cues are reduced to some extent, so it is necessary to rely on the cues available in the actual message (Kiesler, 1986; Sproull and Kiesler, 1986). Identifying communication units and their meaning in the context in which they appear is one of the tasks that are relatively easy for humans, but very hard to automate. We contend, however, that a system can learn from manually categorized data to recognize linguistic indicators on which people may focus to recognize and classify text units, such as clauses or phrases, and their meanings. We believe that manual and automatic content analyses are complementary, and that we can build on the manual content analysis method and annotated data to achieve automation. This will allow us to test research hypotheses on larger data sets, thus reducing the effect of biases and regularities that may exist in smaller data sets.

In the present study we work with a sample of 98 participants in 49 electronic negotiation simulations. The negotiators are Austrian, Canadian and Taiwanese students enrolled in negotiation courses or Master of Information Science degrees. The sample is small enough to be handled with high accuracy and to require only a reasonable effort in terms of time and personnel. In the first step, two well-trained, independent coders manually analysed the negotiation transcripts. Next, we used Machine Learning (ML) systems to find out from this data what represents a specific type of communication or behaviour, and to investigate the possibility of partially automating the task of content analysis.

2 Data

The focus of the work that we present here is the analysis of strategic behaviour during electronic negotiations. We analyze messages exchanged in negotiations conducted via the electronic negotiation support systems Inspire (Kersten and Noronha, 1999) and SimpleNS (Kersten, 2004). All interaction between negotiation partners is stored in a transcript. We collect and study the textual messages exchanged. While other information is available as well, we focus on the linguistic aspect of the interaction.

Two independent coders unitized the process transcripts; every message was divided into smaller fragments. Breaking the messages into thought units – phrases, sentences or clauses that convey only one idea or thought – allows us to investigate negotiation strategies and tactics. They reflect the way in which negotiators use language to establish relationships and try to reach an agreement. Next, the coders categorized these thought units according to the following adapted Bargaining Process Analysis framework (BPA III) for e-negotiations (Srnka and Koeszegi, 2007). We show several examples of each of nine categories.

1. **Substantive negotiation behaviour** – make an offer or concession, reject an offer;
2. **Task-oriented behaviour** – request or provide information;
3. **Persuasive argumentation** – explain one's own position or requests;
4. **Tactical behaviour** – communicate with the intention to influence the negotiation partner, like exert pressure or make promises;
5. **Affective behaviour** – express positive or negative emotions;
6. **Private communication** – release identity information, communicate about private topics;
7. **Procedural communication** – communicate about the negotiation system or about time issues;
8. **Text-specific communication units** – 'e.g.' or 'p.s.' or 'This is my offer:';
9. **Communication protocol** – address, closing and signature.

Each main category summarizes up to 7 sub-categories. In total, there are 42 sub-categories. The first four main categories constitute core negotiation behaviour. *Affective behaviour* and *communication about private topics* are relationship categories. *Procedural, text-specific, communication units* and *communication protocol* are categories intended to coordinate or structure the electronic negotiation process.

Coders received training and instructions before the analysis process started. Each coder ran the unitization and coding process individually. In a first run, the coders unitized the data and compared results. The 49 negotiation transcripts were divided into 5,246 communication units. Guetzkow's U measures the reliability of the *number of units* that two independent coders identify. It is calculated as follows (Weingart et al., 2004):

$$U = \frac{O_1 - O_2}{O_1 + O_2}$$

where O_i is the number of units identified by coder $i, i = 1, 2$. In our study U = 0.016. Because U accounts only for the number of units, but not the units as such, we also measured the inter-coder unitizing reliability indicating *textual consistency of the identified units* (Weingart et al., 1990); it reached 93%.

Dimension	Main category	Number of examples
Content	Substantive negotiation behaviour	909
	Task-oriented behaviour	904
	Persuasive argumentation	366
	Tactical behaviour	357
Relationship	Affective behaviour	541
	Private communication	76
Process	Communication protocol	1,417
	Procedural communication	269
	Text-specific communication units	407
	Total	5,246

Table 1: Distribution of communication units in nine categories

Next the identified units were assigned a category. The distribution of the categories appears in Table 1. After a first round of coding, the coders compared categorization units and discussed the differences. Then they went through another round of coding and again compared the results. Finally, after a discussion, they assigned a category to each communication unit. After the final coding round, intercoder agreement was calculated using Cohen's κ:

$$\kappa = \frac{P_{ij} - P_i \times P_j}{1 - P_i \times P_j}$$

where P_{ij} is the observed proportion of inter-coder agreement, and $P_i \times P_j$ reflects the chance proportion of inter-coder agreement (Brennan and Prediger, 1981; Cohen, 1960). In this experiment we had κ=0.91. A score above 0.7 is considered to show good agreement. All reliability values for the manual coding are extremely satisfactory (Brett, 1998; Weingart et al., 1990).

Finally, the remaining differences between coders were resolved through discussions, so that each unit was categorized unequivocally into one main category and one sub-category. Table 1 shows the frequency of the communication units in each main category after the final coding round. On this categorized data, we trained an automatic system to detect language patterns that indicate the classes of interest.

3 Experiments

The experiments we present here were all performed on the data set described in the preceding section. There were 5246 thought units extracted from 49 negotiations.

In (partial) content analysis, an automatic system faced three tasks:

- identification of fragments of text that represent communication units;
- recognition of the topic of each unit;

- extraction of patterns of communication.

In this paper we present experiments related to the second task. In order to learn to recognize the topics of textual units extracted from negotiation messages, we seek, using a ML system, words or expressions that unambiguously identify specific topics. To this end, we represent each text/thought unit by the words it contains. Two facts about our data complicate the learning process: (i) there is much lexical noise – the texts are unedited and written by second- language users of English, (ii) we consider all words in the text, so the number of attributes that describe our data is quite large – 2724 unique words from 49 negotiations.

We have experimented with several methods of reducing the dimensionality of the data:

- spell-checking,
- spell-checking and lemmatising,
- spell-checking, lemmatising and stemming,
- using a manually built ontology and experimenting with various levels of granularity,
- using language patterns discovered in other analyses of the same data.

The third method worked best, reducing the dimensionality of the data to 1693.

We learn using decision trees and Naïve Bayes from the Weka package (Witten and Frank, 2005). The decision tree induction tool gives results easy to understand and analyse. The probabilistic learner picks out interesting combinations of features. The decision tree learner, J48, uses information in the data to build decision trees. At each step, it chooses a feature – among those that represent the data – which produces the most ordered (pure) split of the data set in that node. For a data set S, and a feature F with the set of possible values VF, the *information gain* from splitting set S by feature F is:

$$Gain(S, F) = Entropy(Set) - \sum_{v \in VF} \frac{|S_v|}{|S|} Entropy(S_v)$$

$|S_v|$ is the cardinality of the subset of instances where feature F takes value v, $|S|$ if the total number of instances in set S, and the entropy measures how disordered a set is:

$$Entropy(S) = \sum_{i=1}^{c} p_i log_2 p_i$$

p_i is the proportion of instances in dataset S that take the i-th value of the target attribute. High entropy values mean a disordered dataset, that is, there is an approximately equal mixture of classes. Low entropy values mean a relatively pure dataset, with one predominant class.

The machine learning tool builds a classifier based on the training data. It is then run on the test data, and the performance is measured using precision, recall and accuracy.

For a class C, **precision** shows how many examples, out of all those that the classifier assigns to class C, are classified correctly. If *TP(C)* is the number of examples that belong to class C and which the classifier handles correctly (true positives), and *FP(C)* is the number of examples that the classifiers assigns, incorrectly, to class C (false positives), precision $P(C)$ of class C is defined as follows:

$$P(C) = \frac{TP(C)}{TP(C) + FP(C)}$$

$TP(C) + FP(C)$ is the total number of examples that the classifier assigns to class C.

For a class C, **recall** shows how many examples, out of all those that belong to class C, are classified correctly. If *TP(C)* is the number of true positives, as defined above, and *FN(C)* is the number of examples that the classifier assigns incorrectly to other classes than class C (false negatives), the recall $R(C)$ of a class C is defined as follows:

$$R(C) = \frac{TP(C)}{TP(C) + FN(C)}$$

$TP(C) + FN(C)$ is the total number of examples in class C ($TP(C) + FN(C) = |C|$).

The **accuracy** is the number of examples classified correctly (for all classes represented in the dataset), out of the total number of examples in the dataset.

$$Acc = \frac{\sum_{i=1}^{n} TP(C_i)}{\sum_{i=1}^{n} |C_i|}$$

We perform tenfold cross-validation experiments. The data is randomly split into 10 equal parts. There are 10 rounds of experiments in which 9 parts of the data are used for training and one part for testing. The same splits are used for training and testing with both the decision tree and the Naïve Bayes learners. We then report the cumulative results of the 10 experiments. Cross-validation helps evaluate the performance of the classifier by giving a more balanced, more accurate image of its learning capabilities. Using one training and one test set may give (accidentally) very good results due to some specific split of the data into training and test sets. By performing experiments with different partitions of the data, we avoid such biases.

Table 2 shows the cumulative results of the 10 testing evaluations on our data set. We use decision trees and Naïve Bayes from Weka (Witten and Frank, 2005). P_S, R_S show the precision and recall for the representation based on the stemmed and lemmatized words. P_{OS}, R_{OS} show the precision and recall for the representation that uses the manually built ontology to group together stemmed and lemmatized words.

The results show that the ML systems can learn, and the approach is very promising, especially considering that we have learned all 9 classes together. The precision baseline was computed as the probability of assigning a class C to an example, which is equal to the proportion of the number of examples in class C. Also, the data set is quite imbalanced – the ratio of examples from one class to the rest of the examples in the data set is much greater than 1, which would signal perfect balance. The extreme ratio values are 1:2.7 for the most populated class (*communication protocol* – 1417 examples) and 1:68.02 for the least populated

Class	Baseline	Decision tree				Naïve Bayes				Inter-coder
		P_S	R_S	P_{OS}	R_{OS}	P_S	R_S	P_{OS}	R_{OS}	agreement
Substantive negotiation behaviour	17.32	68.4	81.5	68.6	84.4	76.5	76	79.9	72.2	94.43
Task-oriented behaviour	17.23	48.3	50.7	45.1	43.5	42.8	66.2	45.3	50.4	87.42
Persuasive argumentation	6.97	45.3	41	48.5	45.6	65.4	37.2	59.7	45.4	83.90
Tactical behaviour	6.80	27.4	16	31.7	16.5	36.3	16.2	39.6	21.3	60.08
Affective behaviour	10.31	70.3	55.6	40.9	45.8	62.6	51.4	43.7	45.1	91.37
Private communication	1.44	56.3	35.5	62.2	30.3	0	0	100	3.9	82.89
Communication protocol	27.01	74.8	92.9	67.5	89.1	66.8	92.7	61.3	91.6	97.54
Procedural communication	5.12	37.6	28.6	27	14.1	63.5	12.3	46.7	18.2	80
Text-specific communication units	7.75	72.5	41.4	37.5	6.6	64	17.9	33.3	13.5	98.21

Table 2: Learning results

	Stemmed and lemmatized words (S&L)		Ontology and words (S&L&Ont)	
	J48	NB	J48	NB
Misclassified examples (%)	37.15	39.36	43.11	42.75

Table 3: Misclassified examples, a summary

class (*private communication* – 76 examples). Imbalance is a hard problem in ML; most tools, including decision tree and Naïve Bayes learners, are sensitive to it. Despite this, we had good results.

We consider the percentage of misclassified examples in each of the four experiments, presented in Table 3. We observe that we get the best results with decision trees, on the representation based on lemmatized and stemmed words. Reducing the dimensionality of the data even more using a manually built ontology had a negative effect on the results. A finer level of granularity improved them, but overall they were weaker than the results with spell-checked, lemmatised and stemmed unigrams (43.11% versus 37.15% average overall error for the decision tree learner J48 in Weka, and 42.75% versus 39.36% with Naïve Bayes). Only three classes (*substantive negotiation behaviour, persuasive argumentation, tactical behaviour*) showed an improvement when we used generalized concepts from our ontology. The cumulative results of the 10 test runs are presented in columns labelled P_{OS}, R_{OS} for the two ML tools. Additionally, in the last column we show the percentages of inter-coder agreement in each category of the manual content analysis procedure. This allows us compare human and machine performance.

4 Discussion and future work

In all experiments, we had the highest precision and recall for the classes *communication protocol* and *substantive negotiation behaviour*. There may be several reasons for these results.

- These are two of the most populated classes. The system learns better when more

examples are available.

- There are clear patterns in language that convey behaviour and attitudes within these two topics. In order to test whether this is the case, we will split our original problem in which we tried to learn all 9 classes together into 9 binary classification problems. We will analyze the classifiers built by the decision tree learners. An ML system builds small trees if it can abstract well from examples and identify general phenomena. Large trees typically signal overfitting, that is, new examples tend not to be covered by the learned pattern that highly depends on training data.

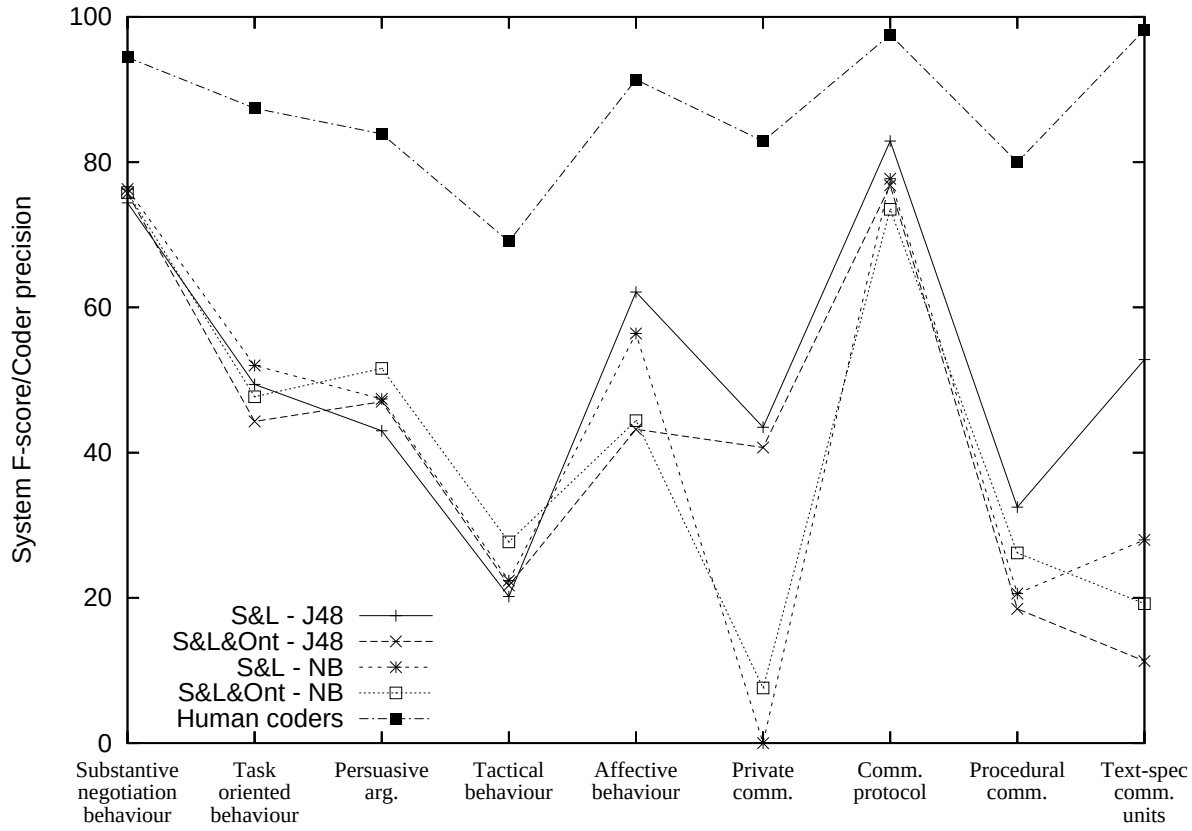

Figure 1: System and human coder performance

We had the lowest precision in the *tactical behaviour* category. Tactical behaviour is a relatively well populated class (357 examples, comparable with *persuasive argumentation*). The fact that despite having relatively many examples the system was not able to find patterns means that this category is harder to identify based only on the words in the text units. As argued before, context plays an important role in interpreting social communication. This is even more true for communication that is intended to influence another person. Since this manipulation should not be obvious, tactics are often very subtle. To identify a tactic is difficult for human coders, too. This is evidenced by the low inter-coder agreement score, compared to the other categories. We should find a way to incorporate context in our data representation.

The manually built ontology slightly increased the precision and recall for the *private communication* and *tactical behaviour* categories. This suggests that there are a variety of ways to express tactical behaviour through language.

We observe an interesting correlation between the inter-coder agreement and the performance of the system, as shown in Figure 1. The coders have the highest agreement for the same classes for which the ML system performed best, and also have the lowest agreement for the classes that the ML system also found harder to learn.

Based on the experiments presented in this paper, we conclude that one can train a system to detect subtle communication using manually annotated data. Automation of content analysis using ML methods is a promising method, especially for large data sets (precision increases with the number of text units). As more manually categorized data become available, we plan to extend the experiments presented here, and to refine our search for linguistic patterns. Once a satisfactory level of performance has been reached, we can deploy a machine-based system on the full collection of messages extracted from Inspire, and test hypotheses that arise from manual analysis of small sets of data.

The interesting correlation between the inter-coder agreement and the system performance suggests another possible continuation of this work. It may be worthwhile to compute agreement of the system with each individual coder. The coder that has the highest agreement with the system will be asked to rate the patterns that the system found (similarly to what she focused on during the coding process, on a predetermined scale). This can give us insights into what the human coders and the system focus on in this content analysis task. It can also help reveal more easily "thought patterns" of coders and where the discrepancies between them lie.

References

N. J. Adler. 1993. Do cultures vary? In T.D. Weinshall, editor, *Societal Culture and Management*, pages 23–46. Walter de Gruyter & Co., Berlin.

L. Adrianson. 2001. Gender and computer-mediated communication: Group processes in problem solving. *Computers in Human Behaviour*, 17:71–94.

R. L. Brennan and D. J. Prediger. 1981. Coeficient kappa: Some uses, misuses, and alternatives. *Educational and Psychological Measurement*, 41:687–699.

J. M. Brett. 1998. Inter- and intra-cultural negotiation: U.s. and japanese negotiators. *Academy of Management Journal*, 5(41):495–510.

Peter J. Carnevale and Carsten K.W. De Drel, editors. 2005. *Journal of International Negotiation, Special issue on Methods of Negotiation Research II*, volume 10(1). Brill Academic Publishers.

J. Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

D. Druckman. 2005. *Doing research: methods of inquiry for conflict analysis*. Sage Publications, Inc., London.

V. J. Dubrovsky, S. Kiesler, and B. N. Sethna. 1991. The equalization phenomenon: status effect in computer-mediated and face-to-face decision-making groups. *Human-Computer Interaction*, 6:119–146.

A. H. Eagly. 1987. *Sex differences in social behaviour: a social role interpretation*. Erlbaum, Hillsdale, NJ.

R. A. Friedman and S. Gal. 1991. Managing around roles: building groups in labor negotiations. *Journal of Applied Behavioural science*, 3(27):356–387.

B. G. Glaser and A. L. Strauss. 1967. *The discovery of grounded theory: strategies for qualitative research*. Aldine, Chicago.

J. L. Graham, A. T. Mintu, and W. Rodgers. 1994. Explorations of negotiation behaviors in ten foreign cultures using a model developed in the united states. *Management Science*, 1(40):72–95.

O. R. Holsti. 1969. *Content analysis for the social sciences and the humanities*. Addison-Wesley, Reading, MA.

G. E. Kersten and S. J. Noronha. 1999. Www-based negotiation support: Design, implementation and use. *Decision Support Systems*, 25:135–154.

G. E. Kersten. 2004. E-negotiation systems: Interaction of people and technologies to resolve conflicts. Technical Report INR 08/04, InterNeg Research. http://interneg.org/.

S. Kiesler. 1986. Thinking ahead, the hidden messages in computer networks. *Harvard Business Review*, 1.

S. T. Koeszegi, E.-M. Pesendorfer, and S. W. Stolz. 2006. Gender salience in electronic negotiations. *Electronic Markets*, 16(3). Forthcoming.

K. Krippendorff. 1980. *Content analysis. An introduction to its methodology*. Sage, Beverly Hills, CA.

E.-J. Lee. 2003. Effects of "gender" of the computer on informational social influence: the moderating role of task type. *International Journal of Human-Computer Studies*, 58:347–362.

P. Mayring. 2002. Qualitative content analysis - research instrument or mode of interpretation? In M. Kriegelmann, editor, *The role of the researcher in qualitative psychology*, pages 139–148. Huber, Tübingen.

P. Mayring. 2003. Phases, transitions and interruptions: modelling processes in multi-party negotiations. *International journal of conflict management*, 3/4(14):191–210.

L. L. Putnam. 2003. Dialectical tensions and rhetorical tropes in negotiations. *Organization studies*, 1(25):35–53.

M. Sokolova and S. Szpakowicz. 2005. Analysis and classification of strategies in electronic negotiations. In *Proceedings of Canadian AI 2005*, pages 145–157, Victoria, BC, Canada.

M. Sokolova, V. Nastase, and S. Szpakowicz. 2004. Language in electcronic negotiations: patterns in completed and uncompleted negotiations. In *Proceedings of ICON 2004*, pages 142–151, Hyderabad, India.

L. Sproull and S. Kiesler. 1986. Reducing social context cues: electronic mail in organizational communication. *Management Science*, 11(32):1492–1512.

K. J. Srnka and S. T. Koeszegi. 2007. From words to numbers - how to transform rich qualitative data into meaningful quantitative results: Guidelines and exemplary study. In *Schmalenbach's Business Review*. Forthcoming.

L. R. Weingart, L. L. Thompson, M. H. Bazerman, and J. S. Carroll. 1990. Tactical behaviour and negotiation outcomes. *The international journal of conflict management*, 1(1):7–31.

L. R. Weingart, M. Olekalns, and P. L. Smith. 2004. Quantitative coding of negotiation behaviour. *International Negotiation*, 9:441–455.

S. P. Weisband, S. K. Schenider, and T. Connolly. 1995. Computer mediated communication and social information: Status salience and status differences. *Academy of management journal*, 4(38):1124–1151.

Ian H. Witten and Eibe Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco. 2nd edition.