Good News or Bad News? Conducting Sentiment Analysis on Dutch Text to Distinguish Between Positive and Negative Relations

Wouter van Atteveldt
Jan Kleinnijenhuis
Nel Ruigrok
Stefan Schlobach

ABSTRACT. Many research questions in political communication can be answered by representing text as a network of positive or negative relations between actors and issues such as conducted by semantic network analysis. This article presents a system for automatically determining the polarity (positivity/negativity) of these relations by using techniques from sentiment analysis. We used a machine learning model trained on the manually annotated news coverage of the Dutch 2006 elections, collecting lexical, syntactic, and word-similarity based features, and using the syntactic analysis to focus on the relevant part of the sentence. The performance of the full system is significantly better than the baseline with an F1 score of .63. Additionally, we replicate four studies from an earlier analysis of these elections, attaining correlations of greater than .8 in three out of four cases. This shows that the presented system can be immediately used for a number of analyses.

KEYWORDS. Sentiment analysis, valence, polarity, political communication, automatic content analysis, semantic network analysis

Content analysis, the systematic analysis of textual content such as news media, parliamentary debates, presidential speeches, or blogs, plays a key role in answering a number of research questions in political science. In texts from the political domain, the main objects, or *concepts*, of interest are generally actors (such as states, parties, and politicians) and issues (such as employment, peace, and healthcare). Two aspects of the discourse about actors and

Wouter van Atteveldt is an interdisciplinary Ph.D. student at the Departments of Communication Science and Artificial Intelligence at the Vrije Universiteit Amsterdam (the Netherlands). He hopes to defend his Ph.D. thesis on methods for automatically extracting and representing media data in fall 2008.

Jan Kleinnijenhuis is professor of mass communication. His research deals with news selection and news effects. In collaboration with the University of Amsterdam, Kleinnijenhuis conducts recurrent research of the media and public opinion during the Dutch national election campaigns.

Nel Ruigrok is a researcher at the Netherlands News Monitor, and research fellow in the Amsterdam School of Communications Research at the University of Amsterdam. Her research interests are the role of media in the political arena, especially during times of conflict.

Stefan Schlobach is assistant professor at the Vrije Universiteit. His core research area is knowledge representation and reasoning with a focus on the creation of expressive formal ontologies, and their application to practical problems.

Address correspondence to: Wouter van Atteveldt, De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands (E-mail: wouter@vanatteveldt.com).

Journal of Information Technology & Politics, Vol. 5(1) 2008
Available online at http://jitp.haworthpress.com

doi:10.1080/19331680802154145

issues dominate the literature: attention and polarity (also called valence or direction: the positive/negative aspects of relations or evaluations). Jones & Baumgartner (2005), for example, argue that politics is all about attention. Therefore, they conduct large-scale content analysis projects to study the agenda setting, or prioritization, of specific issues, specific attributes of those issues, specific solutions, and specific alternatives. Content analysis aimed at the measurement of attention is called "thematic" content analysis (Roberts, 1997). In political communication, this is the dominant approach to the study of agenda setting and is also used in the study of framing, which is often operationalized as emphasis on specific aspects of an issue (Entman, 1993; Scheufele, 1999). Thematic content analysis can be automated largely by means of keywords and Boolean and distance-based combinations of keywords, measuring the (co-)occurrence in the texts of the relevant actors and/or issues (Evans & Andersen, 2006; Purpura & Hillard, 2006).

The research question to be answered in this article is whether the polarity of political discourse can be analyzed automatically. There are different ways in which polarity is important in political communication, having to do with the relations between actors and issues and their evaluative and performance descriptions. These relations and descriptions surface in a number of political studies. For example, the polarity of relations between actors, ranging from war to cooperation, is at the heart of international events research, starting from the seminal COPDAB-project (COnflict and Peace DAta Bank) (Azar, 1980; Schrodt & Gerner, 1994). The polarity of relations between issues, ranging from negative to positive causation, or from dissociation to association, is the core of the cognitive map approach (Axelrod, 1976; Dille & Young, 2000). The polarity of relations between actors and issues is used to determine the issue positions of political actors (Kleinnijenhuis & Pennings, 2001; Laver, Benoit, & Garry, 2003; Laver & Garry, 2000). The polarity of evaluative descriptions or (moral) judgments, such as *Republicans are trustworthy* or *Unemployment is awful* is the topic of evaluative text analysis (Hartman, 2000; Janis & Fadner, 1943; Osgood, Saporta, & Nunnally, 1956). Performance descriptions are statements about real-world developments or attributions of success and failure such as *John is gaining in the polls* and *Unemployment is rising*. Although not always distinguished from evaluations, performance descriptions are in fact different from evaluations since the latter can often be seen as indirect expressions of relations between actors or issues: If John thinks Republicans are trustworthy, one can deduce a positive relation between John and Republicans, while John stating that the Republicans are winning in the polls has no implication for his opinion about that party. The polarity of the performance descriptions of actors, the attribution of success and failure, constitutes the core of attribution theory in psychology, and of the bandwagon effect (Lazarsfeld, Berelson, & Gaudet, 1944), political momentum (Bartels, 1988), and horse-race news coverage (Iyengar, Norpoth, & Hahn, 2003) in political science. Performance descriptions of issue developments in the real world have been studied especially with regard to media reports of economic developments, such as reports about an increase or a decrease in employment (Hetherington, 1996; Soroka, 2006).

Most of these studies focus on a specific kind of relation or description. In some cases, multiple aspects of relations between actors and issues need to be considered. Diehl (1992), for example, argues that studying pro–con positions on salient issues can enhance the understanding of conflict and cooperation between states. Monge & Contractor (2003) argue for simultaneously testing theories at different levels—actors, dyadic and triadic patterns, and the whole network—to arrive at a better understanding of how these theories complement each other. Semantic network analysis, a branch of content analysis (Krippendorff, 2004, p. 292), explicitly extracts both the attention for, and the polarity of, relationships between both actors and issues to arrive at a single network (Popping, 2000; Roberts, 1997; Van Cuilenburg, Kleinnijenhuis, & De Ridder, 1986). This yields content analysis data that are useful for studying a large variety of different aspects of the coded texts. For example, Kleinnijenhuis, Van Hoof,

Oegema, & De Ridder (2007) show how news about issue positions, news about relations between political parties, and news about party performance each exert a differential effect on the shift in party preferences during a political campaign.

Extracting the network of positive and negative relationships between actors and issues can be done manually, either with the text of the unit of observation, by asking coders what these relationships are after a careful reading of a text (Azar, 1980), or with the sentence as the unit of observation by dissecting each sentence as one or more positive or negative relations between objects (Osgood et al., 1956; Van Cuilenburg et al., 1986). These processes are time-consuming and expensive, making it difficult to obtain datasets that are sufficiently large, thereby impeding data-intensive research such as internationally comparative and longitudinal research. Typically, concessions are made through analyzing only part of the texts, although such limitations may result in a loss of validity in the case of detailed research questions (Althaus, Edy, & Phalen, 2001).

Automatically extracting positive and negative relations and descriptions is not an easy task. Polarity can be expressed using verbs such as *support*, adjectives such as *good*, or nouns such as *winner*. Often, whether a word has a positive or negative meaning is dependent on context, such as *cool relations* versus *cool plans*; a special case of this is multiword units such as *to push one's buttons* or *to lead up the garden path*, which contain a negative sentiment even though the individual words are generally neutral. To make matters worse, positive and negative expressions contain more infrequently used words than non-polarized text (Wiebe, Wilson, Bruce, Bell, & Martin, 2004), making it very difficult to create word lists for such expressions, either manually, or by automatic extraction or machine learning from manually annotated material. As a consequence, there are currently no automatic semantic network analysis methods that extract a polarized network from text. Approaches to automating the extraction of positive or negative relationships are often based on counting positive words on the one hand, and negative words on the other, such as in extracting issue positions from texts (Laver et al., 2003), in extracting evaluations (Fan, 1996), and in extracting real-world developments, such as attributions of economic success or failure (Shah, Watts, Domke, Fan, & Fibison, 1999). Schrodt & Gerner (1994) use syntactic information for extracting relations, but restrict themselves to conflict and cooperation between actors in sentences with a limited syntactic complexity, such as headlines. Network content analysis methods inspired by social network theory (Wasserman & Faust, 1994) largely focus on the attention for specific relations between actors rather than their polarity (Corman, Kuhn, McPhee, & Dooley, 2002; Diesner & Carley, 2004).

Within computational linguistics, recent years have seen the emergence of sentiment analysis, a field that aims to identify and classify subjective aspects of languages, especially expressions of positive or negative sentiment (Choi, Breck, & Cardie, 2006; Kim & Hovy, 2006; Shanahan, Qu, & Wiebe, 2006; Wiebe et al., 2004). Sentiment analysis uses a variety of linguistic means, such as elaborate thesauri, part-of-speech taggers, lemmatizers, syntactic parsers, and statistical natural language processing methods to assess whether a text contains subjective sentiment and whether that sentiment is positive or negative.

This article uses sentiment analysis techniques to automatically determine the polarity of relations between actors and issues in Dutch political newspaper articles. Specifically, we use a machine learning strategy with a number of lexical features based on an existing Dutch thesaurus and extracted automatically from a large unannotated corpus, using syntactic analysis to focus the machine learning on the relation rather than the whole sentence. We use an earlier, manual semantic network analysis of the Dutch 2006 parliamentary elections (Kleinnijenhuis, Scholten, et al., 2007) to train the machine learning model and test the model at the sentence level. Finally, we validate the usefulness of the method for answering political research questions by replicating a number of analyses from the original study, and comparing the results derived from the automatically extracted network with those derived from the manual annotation on the

level of analysis (e.g., a week of news about an actor) rather than the level of sentences.

The contribution of this article is threefold: First, we present a system for automatically determining the polarity of relations between, and descriptions of, actors and issues in text. This is an important step in automating semantic network analysis, and allows the automatic extraction of the data needed for many interesting political research questions. Second, we show that existing sentiment analysis methods can be used for extracting data that is relevant for answering political science questions. The existing sentiment analysis literature focuses on the ability to extract a linguistic phenomenon at the level of sentences, and we show that this can be used for analyzing political phenomena at the level of political analysis. This serves as a use case and external validation for sentiment analysis techniques, and gives the political analysts an indication of the utility of these techniques for their research. Finally, sentiment analysis is generally focused on the English language, and although a number of articles apply these methods to other languages, this is the first explicit sentiment analysis study conducted on Dutch, showing that the methods developed for English can be translated to that language.

In the next section, we describe the relational content analysis method that we want to automate and formulate the tasks that the system needs to perform. This is followed by a brief summary of the relevant techniques in sentiment analysis and an explanation of how we used these techniques to build our system. Subsequently, we present the performance at the sentence level and give a short analysis of which techniques performed well. Finally, we conduct the four case studies mentioned above, analyzing the performance of the system at the level of analysis and showing its usefulness for political research.

CLASSIFYING NET RELATIONS

The NET Method

The data on which this article is based is the manual analysis of the 2006 parliamentary election campaign in the Netherlands described in Kleinnijenhuis, Scholten, et al. (2007). This analysis uses a semantic network analysis method called the NET method (Network analysis of Evaluative Texts) (De Ridder & Kleinnijenhuis, 2001; Van Cuilenburg et al., 1986). The NET method creates a network of positive and negative relations between actors and issues. This results in the sentence types listed in Table 1, categorized into relations, evaluative descriptions, and performance descriptions. Relations are links between actors and issues, for example affinity or causality. Performance descriptions are represented as links between the abstract value Reality and an actor or issue: *Balkenende leads in the polls* is represented as $Reality \pm Balkenende$. Similarly, evaluations are represented as links from an actor or issue to the abstract value Ideal: *Inflation should be avoided* is represented as $Inflation \sqsubset Ideal$. Representing these descriptions as relations with abstract variables means that all relevant information is encoded in a single network, making it easier to analyze the resulting network, for example using graph theory to make inferences about the latent content (Van Cuilenburg et al., 1986) or representing

TABLE 1. NET Relation Types with Examples

Task	Sentence type	Subject	Object	Example
Relation	Support / criticism	Actor	Actor	Balkenende criticizes Bos.
	Issue position	Actor	Issue	Bos proposes fiscalizing pensions.
	Causation	Issue	Issue	Inflation leads to increased unemployment.
	Consequences	Issue	Actor	Unemployment torments goverment.
Performance	Success/failure	Reality	Actor	Balkenende leads in today's polls.
	Developments	Reality	Issue	The oil price has risen to a record high.
Evaluation	Actor eval.	Actor	Ideal	Bos is untrustworthy.
	Issue eval.	Issue	Ideal	High inflation should always be avoided.

and querying the network using semantic Web techniques (Van Atteveldt, Schlobach, & Van Harmelen, 2007).

For a concrete example, consider the (translated) article in Figure 1. The headline of this article is a negative evaluation of the PvdA and VVD, which is coded as two negative relations from those concepts to the Ideal. The next three sentences are coded as a negative issue position of Bos (the PvdA leader) on *SmallerGovernment* and a positive position on *ReformingDismissal* (making it easier for companies to fire employees). The next two sentences are coded as the opposite relations from the PvdA: in favor of *SmallerGovernment* and against *ReformingDismissal*. The first sentence of the last paragraph is coded as a positive relation from Bos to the VVD, while the last sentence is coded as a positive relation from PvdA to CDA. The resulting network is visualized next to the article in Figure 1, with the black arrows representing negative relations and the wider gray arrows positive relations. This picture shows the confusion within the PvdA expressed in the article: The party and its leader disagree on two issues and with whom to cooperate, leading to an unclear profile in the political arena.

Task Definition

Given a relation or description expressed in a text, the task of the system is to determine whether the relation is positive, negative, or neutral. This is part of an ongoing effort to fully automate semantic network analysis. Identifying and extracting the relations is performed by another module that is being developed, which currently attains an acceptable F1 score of .65 and correlations between manual and automatic analysis of up to .83 for a number of use cases (Van Atteveldt, Kleinnijenhuis, & Ruigrok, 2008). Since we have a corpus manually analyzed with semantic network analysis, we can study the polarity separately from the identification of the relations by using the manually identified relations as input for the system described in this paper.

We define three different tasks on the basis of the sentence types described above. The *Relation* task is based on the first four sentence types, which all represent a relation between two concepts occurring in the text. The *Performance* task uses the Success/Failure and (real-world) Developments sentences, which are both indicators of how well a person or issue is doing. The last two sentence types are used in the *Evaluation* task, in which a (moral) judgment is passed on an actor or issue. More formally, the three tasks are as follows:

- *Relation*: Given two concepts located in a sentence, is their relation positive (cooperative, supportive), negative (conflictive, critical), or neutral?
- *Performance*: Given a concept located in a sentence, is it described as successful

FIGURE 1. Example (translated) newspaper article with visualized network.

PvdA and VVD keep on blundering

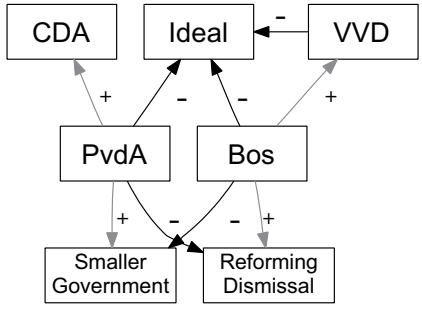
De Telegraaf, 14 November 2006

...

Bos also caused confusion about what he wants with the government. It had to be bigger, he said some months ago. Dismissal rights also had to be reformed. In its manifesto, the PvdA suddenly wants a small government. And dismissal rights will not be changed.

...

For months he has been informally probing the VVD for cooperation! But in the meantime there is a PvdA manifesto that is aimed at governing with the CDA.



(increasing, winning), failing (decreasing, losing), or neither?
- *Evaluation*: Given a concept located in a sentence, is it evaluated positively (as good, sincere, or beautiful), negatively (as evil, wicked, or ugly), or neutrally?

SENTIMENT ANALYSIS

The work conducted in sentiment analysis (Shanahan et al., 2006; Wiebe et al., 2004) is highly relevant to our task. This field of computational linguistics aims at identifying and classifying subjective language, defined as the "language used to express *private states* in the context of a text or conversation" (Wiebe et al., 2004, p. 5). This section will survey some of the techniques used in this field, on which we base the system described in the next section.

A number of sentiment analysis papers try to create lists of subjective words by starting with a *seed set* of words with a known evaluative value, and then by expanding this set. For example, Hatzivassiloglou & McKeown (1997) use "Adj_1 and/but Adj_2" patterns on a large corpus to cluster adjectives, assuming that *and* connects similar adjectives while *but* conjoins adjectives with opposite polarity. Hatzivassiloglou & Wiebe (2000) expand this system by adding gradable adjectives (adjectives that can be modified with a grading adverb such as *very*) as an indicator of subjectivity and test whether these adjectives help in identifying subjectivity at the sentence level. Wiebe (2000) uses distributional similarity of syntactic relations to further expand this set. Two adjectives are distributionally similar if they appear in the same contexts, which in this case means having the same syntactic relations with other words (Lin, 1998). Wiebe et al. (2004) test various subjectivity clues, including unique words, N-grams, and distributional similarity, on a number of data sets. Baroni & Vegnaduzzo (2004) use "A near B" patterns using an Internet search engine to expand a seed set based on co-occurrence. Finally, Riloff & Wiebe (2003) learn "extraction patterns" from sentences containing known subjective words, creating lists of syntactic patterns such as a specific *verb–infinitive* or *active verb–preposition* combinations.

Word lists suffer from the inability to consider the specific context in which words are used. An alternative approach is to use a machine learning algorithm to discover patterns in large sets of training sentences whose polarity has been manually annotated. In machine learning, two important choices are the learning algorithm and the characteristics of the text to use as features (or independent variables). For example, Wilson, Wiebe, & Hoffmann (2005) use a program called BoosTexter, which uses decision rules to determine the polarity of words in context. As features, they use a thesaurus, words from the general inquirer (Stone, Bayles, Namerwirth, & Ogilvie, 1962), the patterns from Riloff & Wiebe (2003), and a number of syntactic features such as whether a word is in the subject or object clause. Breck, Choi, & Cardie (2007) identify and classify subjective statements using a support vector machine using words, the verb categories defined by Levin (1993), and the word lists derived by Wilson et al. (2005) as input features. Choi et al. (2006) train two different conditional random field (CRF) models, one for extracting opinions and opinion sources, and one for determining the relation between the two. This second model is a zero order CRF (which is equivalent to the maximum entropy model used in this article) trained on a number of lexical and syntactic features, such as whether the sentence is active or passive, the syntactic path between the opinion and its possible source, and a number of specific patterns called "syntactic frames" that can match the grammatical structure.

These papers all focus on the English langauge. Mihalcea, Banea, & Wiebe (2007) try to directly translate subjectivity clues from English to Romanian using an online dictionary, but this has limited success. Mathieu (2006) presents and evaluates a computational semantic lexicon of French emotive verbs. The NTCIR Information Retrieval workshop in 2006 had an opinion extraction task in Chinese and Japanese as well as English (Seki et al., 2007), leading to a number of papers focusing on these languages such as Kanamaru, Murata, and Isahara (2007) and Xu, Wong, and Xia (2007), who use machine learning methods for subjectivity in Japanese and Chinese texts, respectively. To the knowledge of the authors, no explicit sentiment

analysis work has been performed on Dutch, although there is related work such as an investigation of subjective verbs (Pit, 2003) and work on automatically expanding lexical resources (Tjong Kim Sang & Hofmann, 2007).

METHOD

The system described in this paper will use a machine learning approach similar to the work discussed above (e.g., Breck et al., 2007; Choi et al., 2006; Wilson et al., 2005). In the Task Definition section above, we defined three tasks: classifying relations, classifying performance descriptions, and classifying evaluative descriptions. For each of these tasks we train and test a maximum entropy model (Berger, Della Pietra, & Della Pietra, 1996) based on lexical and syntactic features. Maximum entropy models are log-linear models built to maximize the entropy in the model within the constraints set by the training data that have been used successfully for a number of natural language processing tasks (e.g., Abney, 1997; Ratnaparkhi, 1998), including the work by Choi et al. (2006) described above. Nonetheless, other machine learning methods, such as support vector machines or higher order conditional random fields, could have been used as well, and it would be interesting to test whether higher performance can be attained with other methods.

In machine learning, the learning algorithm is presented with a set of cases together with their actual class (the polarity according to the manual analysis). From these cases, called the training data, the learning algorithm creates a model of the relation between characteristics of the input data and the class. This model is subsequently tested on the test data: a set of cases not used in training with the actual class hidden from the model. Comparing the class assigned by the model with the actual class gives an indication of the performance of the model. The remainder of this section describes which features (characteristics of the input data) are considered, the strategy for collecting these features from the text, the procedure and measures used to test performance, and the corpus that is used for training and testing.

Features

An important choice in using machine learning models such as maximum entropy is which characteristics of the text, called *features*, are given as input to the model. Since the model can only use information contained in these features, the choice of features strongly influences the performance of the model. Model features are similar to the independent variables in the statistical modeling such as regression analysis, but the focus of machine learning is on finding the best model, not on understanding the underlying phenomenon. Consequently, the value of the parameters attached to the features is generally not of interest, and machine learning models can have very large numbers of features.

Below, we list the features that are used in our model. The first two feature groups are based on the output of linguistic preprocessing of the text, such as lemmatizing and parsing. A problem with such features is the *data scarcity problem*: It is quite likely that a word used for expressing polarity has not been encountered in the training data. To overcome this problem, we included lexical information to group words with similar meaning together: The third feature group is based on existing lexical information in the form of a thesaurus, while the last three feature groups are based on finding clusters of similar words in a reference corpus that has not been manually analyzed. In each of the descriptions below, the description *the young senator* will be used as an example.

Lexical and POS Features

Similar to Choi et al. (2006), we use the frequencies of lemmata and part-of-speech (POS) tags as reported by the Alpino parser as features (Van Noord, 2006). For the example description, this would yield {lemma:the, lemma:young, lemma:senator, pos:Determiner, pos:Adjective, pos:Noun}.

Syntactic and Surface Bigrams

We use all adjacent lemma pairs in the selected part of the sentence as features. Moreover, we include the syntactic dependency relations between words reported by Alpino (Van

Noord, 2006) as features: In a sentence such as *John trusts Republicans*, John is the subject of trusts and Republicans are the object of trusts, yielding the dependency relations *John-subject-trust* and *Republicans-object-trust*. In the example description *The young senator*, the surface and syntactic bigrams are {bigram:the_young, bigram:young_senator, dependency:the-determiner-senator, dependency:young-modifier-senator}.

Brouwers Thesaurus

Brouwers (1989) thesaurus is a manually created general-purpose Dutch thesaurus, comparable with *Roget's Thesaurus for English* (Kirkpatrick, 1998). Brouwers lists around 123,000 single-word entries, including around 20 thousand verbs and 16 thousand adjectives and adverbs. These words are categorized into 997 categories, where a single lemma can be a member of multiple classes. We look up all lemmata in the thesaurus, and use the frequency count of each found category as a feature. For example, the word *young* falls into Brouwers' categories *age* and *incompetent*, and *senator* is categorized as *authority* yielding the features set {brouwers:age, brouwers:incompetent, brouwers:authority} for the example description.

Mutual Information on an Unannotated Corpus

The intuition behind co-occurrence–based methods such as mutual information is that words that frequently occur together probably have similar meanings. For this feature, we create clusters of words based on pointwise mutual information on an unannotated corpus similar to the work by Grefenstette, Qu, Evans, & Shanahan (2006) and Baroni & Vegnaduzzo (2004). Specifically, for each pair of words belonging to the same category (noun, verb, adjective/adverb), we determined the number of documents containing either or both terms, and calculate the mutual information as the log of the intersection divided by the product of the individual document counts. We then transform this to a distance metric by subtracting from the theoretical maximum log(|*D*|), yielding:

$$dist_{MI}(w_1, w_2) = log(|D|) - log\left(\frac{|D| \cdot |D_{w1} \cap D_{w2}|}{|D_{w1}| \cdot |D_{w2}|}\right) = log\left(\frac{|D_{w1}| \cdot |D_{w2}|}{|D_{w1} \cap D_{w2}|}\right)$$

Using this distance metric, we created 500 word clusters using a K-means clustering algorithm, and used each of these clusters as a feature. In the example description, suppose *young* is contained in cluster 131 and *senator* is in cluster 265, we would get the feature set {mutual:131, mutual:265}.

Distributional Similarity Based on Syntax Trees

Whereas co-occurrence is based on two words appearing in the same document, distributional methods are based on two words appearing in similar contexts. Following Lin (1998) and Wiebe (2000), we constructed a classification using the distributional similarity of the syntactic relations entered into by adjectives. In particular, we computed the distance between pairs of adjectives based on the cosine of the relationship frequency vectors for each adjective. Similar to the mutual information feature, we used this distance to create 500 clusters that are used as features:

$$dist_{DS}(w_1, w_2) = \frac{\sum_{r \in relations} fr(w_1, r) \cdot fr(w_2, r)}{\sqrt{\sum_r fr(w_1, r)^2 \cdot \sum_r fr(w_2, r)^2}}$$

where *relations* is the set of all (syntactic relation, object) pairs, and *fr(w,r)* is the frequency with which *w* is the subject of the relation *r*.

Conjunction Patterns on an Unannotated Corpus

A problem with using distributional methods for determining polarity is that antonyms often occur in similar contexts. Similarly, subjective

texts often contain a large number of both positive and negative words, making co-occurrence–based methods difficult. Hatzivassiloglou & McKeown (1997) explicitly look for words with the same polarity in an unannotated corpus by looking for conjunctions of adjectives using *and* or *but*, relying on the fact that words of different polarity cannot be conjoined by *and* (**a corrupt and legitimate regime*) and vice versa for *but*. We applied this to a corpus of unannotated text, looking for ".. en .." (*and*) and ".. maar .." (*but*) for all pairs of adjectives and verbs. From this we compute a distance metric as follows:

$$dist_{CP}(w_1, w_2) = \frac{1}{1 + e^{\frac{1}{10} \cdot (|w_1 en w_2| - |w_1 maar w_2|)}}$$

Strategies for Feature Collection

The features described in the previous section are all collected from words and word pairs in the text containing the relation or description to be classified. Since a sentence can contain multiple relations or descriptions, it might be better to collect features only from the part of the sentence containing the relation or description rather than the whole sentence. This section describes three strategies to focus the feature collection on the relevant part of the sentence.

Strategy 1: Sentence

The first strategy simply collects features from the whole sentence, functioning as a baseline.

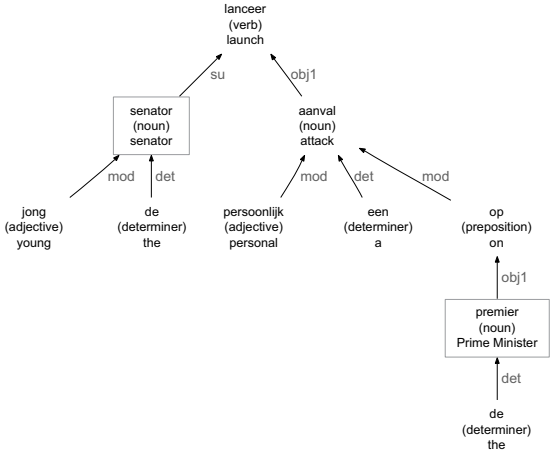
Strategy 2: Predicate

In the second strategy, feature collection is restricted to the predicate expressing the relation or description. For the *relation* task, we define the predicate as being all nodes on the direct path between the subject and object in the dependency tree, and all modifiers and related verbs of these nodes. For the *performance* and *evaluation* tasks, the predicate comprises all nodes directly connected to the target node and all modifiers of these nodes. As an example, consider the fictive sentence *De jonge senator lanceerde een persoonlijke aanval op de premier* (*The young senator launched a personal attack on the Prime Minister*), of which the dependency graph produced by Alpino (Van Noord, 2006) is given in Figure 2. *Senator* and *premier* (*Prime Minister*) are identified as actors by the preprocessing, in this case the manual annotation. To determine the predicate expressing the relation between these actors, we take the shortest path between them through the dependency graph: *Lanceer aanval op* (*launch attack on*). Subsequently, this set of words is expanded by adding all their modifiers and auxiliary verbs, yielding *lanceer een persoonlijke aanval op* (*launch a personal attack on*). For the evaluation or performance description of the first concept, senator, we first select all direct parents and children: *Lanceer jong de* (*launch young the*). This set is then expanded with all modifiers of these words, in this case none. In the predicate strategy, features are only collected from the words in the predicate.

Strategy 3: Combination

The third strategy is a combination of the other two strategies. It creates two distinct sets of features: one for the predicate and one for the remainder of the sentence. For example, the combination strategy applied to the performance description of senator in Figure 2 would have

FIGURE 2. Grammatical analysis of example sentence.



De jonge senator lanceerde een persoonlijke aanval op de premier
The young senator launched a personal attack on the Prime Minister

separate features for *lemma young inside predicate*, which would have value one, and *lemma young outside predicate*, which would be zero. The lemma *personal*, which is excluded in the predicate strategy, is included in the *out of predicate* set in this strategy. This gives the machine learner access to the part of the sentence outside the predicate while still allowing the model to focus on the features in the predicate, for example, by giving a higher weight to specific words in the grammatical context of the evaluation, performance description, or relation.

Procedure and Measure for Testing Performance

We trained a maximum entropy model for each of the tasks described above. In order to find out which features and feature collection strategies worked best for each task, we trained and tested models with different configurations. To obtain an unbiased estimation of the performance of each configuration, we used separate data for training and testing in a procedure called 20-fold cross-validation. This means that we split the data into 20 sets, used 19 for training, and the remaining set for testing. This was then repeated 20 times with a different set as testing set each time, and the results are averaged over these sets.

As performance metric, we used the average F1 score, which is the harmonic average of *precision* and *recall* (Manning & Schütze, 1999, pp. 267–271). The F1 score is calculated per target class (positive, negative, neutral) by counting how many cases in the test set belonged to that class according to the trained model and the manual gold standard. *True positives* belong to the target class according to both model and gold standard; *false positives* are misclassified by the model as belonging to the class (errors of the first kind), and *false negatives* are miscassified by the model as not belonging to the target class (errors of the second kind). *Precision* is then defined as how often the model was correct when it classified a case as beloning to the target class, and *recall* is the percentage of cases actually belonging to the target class (according to the gold standard) that was found by the model. The F1 score is defined as the harmonic average of those two measures, and can be reported either per target class or as an average over all classes:

$$\textbf{Precision} \quad pr = \frac{True\,Positive}{True\,Positive + False\,Positive}$$

$$\textbf{Recall} \quad re = \frac{True\,Positive}{True\,Positive + False\,Positive}$$

$$\textbf{F1 Score} \quad \mathrm{F1} = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The F1 score is an estimation of the performance on unseen data, based on 20 random samples of data that are each unseen to the model trained on the remaining 19 sets. In order to test whether the performance difference between two configurations can be due to chance, we perform a t-test of difference of means between the sets of 20 scores from each category.

The Corpus

The main corpus used in the article consists of the coverage of the Dutch parliamentary election campaign of 2006. This corpus has been annotated manually using the NET method as described above. It includes all articles mentioning a political actor or one of the main issue keywords in the five largest daily Dutch national newspapers from September 1st until November 22nd (election day). In total, the corpus contains 13,954 articles. Of these articles, the headlines and lead paragraphs were manually annotated using the NET method. This resulted in a total of 16,455 relations, 5,816 performance descriptions, and 4,722 direct evaluations of actors and issues.

Unfortunately, these annotations did not specify which words in a sentence represent the concepts used in the annotations. In order to extract features from a relation or description, we need to know which words represent the used concept(s). Therefore, we used only the concepts where we could find a word in the sentence that matched the label of the concept. The precision of this matching is high: a manual evaluation of a small sample ($N = 61$) indicates

a precision of 97%. Unfortunately, the recall is low (59% on the same sample), especially since many different words can be used to refer to the same issue (recall on political actors was 77%, while on issues and other actors it was 51%). As a result of this limitation, our actual corpus consists of 5,348 relations where both subject and object could be matched, and 3,025 and 2,316 performance and evaluation statements where the described concept could be matched. Although this is a small sample, we have no reason to assume that it is biased except for the fact that actors will be over-represented. Hence, we expect the results on this sample to generalize fairly well.

In the coding instructions for the manual analysis, coders were told to omit neutral relations, so the corpus does not contain explicit neutral cases. Since it is important that the system can distinguish between neutral and polarized statements, we created neutral statements from the polarized statements as follows: For the relation task, we added a neutral relation between every pair of identified concepts between which no relationship was annotated. For the performance and evaluation tasks, we added a neutral statement for a random subset of concepts that was not annotated with a performance or evaluation. This resulted in 3,359, 3,292, and 3,466 neutral statements for the relations, performance descriptions, and evaluative descriptions, respectively.

Reference Corpora

In addition to this annotated corpus, we use two reference corpora as sources of unannotated material for the last three features described above. The first reference corpus consists of the fully parsed sentences of the non lead paragraphs of the 2006 election corpus, which were not manually analyzed. This corpus is used for the *distributional similarity* feature.

In order to use the *mutual information* and *conjunction patterns* features described above, we also used an unparsed, unannotated reference corpus. This corpus consisted of articles from the Dutch elections in 1994, 1998, 2002, and 2003 (600,000 articles); a broad range of articles on government policy and media exposure in 2003 and 2004 (1.4 million articles); a stratified sample of all news in 2006 and 2007 (60,000 articles); and the news surrounding two recent events (the referendum campaign for the European Constitution and the news on two anti-immigration populists, Geert Wilders and Rita Verdonk; 63,000 articles). In total, this corpus consists of around 750 million words in 2 million articles.

RESULTS

Table 2 lists the performance of the system on each of the three tasks defined in the Task Definition section. The *full model* row gives the F1 score of the model using all features described above and using the best strategy for that task. As will be shown below, for the relation task this was the *combination* strategy, while the *sentence* strategy was best for the evaluation and performance tasks. For comparison, the performance of two baseline models is also given. The *guess baseline* is the performance attained by blindly guessing each category with a chance proportional to its frequency in the whole corpus. The *lemma baseline* is a maximum entropy model trained using only the lemma frequencies in the whole sentence as features. Precision and recall were very close to the F1 score and to each other for each of the results presented in this section, so to avoid redundancy they are not reported.

The full model improves on both baseline models for each task, performing between .09 and .025 better than the Lemma baseline; this improvement is highly significant according to

TABLE 2. Performance of the Full Model and Baselines (F1 Scores)

Model	Relation	Performance	Evaluation
Guess baseline	.114	.134	.154
Lemma baseline	.541	.534	.534
Full model	.631***	.559***	.580***
N	8,681	5,988	5,773

***Significantly better than Lemma baseline at $p < .001$ level based on t-test on 20 folds.

a t-test on difference of means. Total performance on the *relation* task is .631, which is comparable to the results reported by Wilson et al. (2005): They report an average F1 Score of .728 on identifying whether a relation is polarized, and .628 on classifying the resulting relation (ignoring their extra category of sentence that contains both positive and negative polarity).

Feature Contributions

As stated in the Method section, we use three different strategies for collecting features from the sentences: the *sentence, predicate*, and *combination* strategies. In order to test which of these strategies performed best for each task, we calculated the performance of the model using all features from the three different strategies. The top three rows in Table 3 present the results of this comparison. For each task, the F1 score of the best performing strategy is set in boldface, and for the other strategies the performance is listed along with the significance of the difference from the best model. For the relation task, the model using separate features for the predicate and the remainder of the sentence is significantly and substantially better than using the whole sentence. This indicates that the syntactic analysis is useful to determine the predicate that expresses the relationship using this information. Looking at the performance and evaluation statements, the model using the whole sentence is the best model. For performance statements, this difference is not significant with respect to the combined model, while for the evaluation statements both differences are significant. From this, we conclude that our definition of the predicate for these descriptive statements is probably suboptimal, and we hope that this can be improved by looking at these statements more closely.

The lower part of Table 3 lists the contribution of the different feature sets to the full classifier. For each feature, we have calculated the performance decrease if we leave that feature out, and list this difference and the significance of this difference. This is a rather strict test because of the overlap between the different features, so the individual scores are quite low.

The first five features do not paint a very clear picture. For the relation task, the individual lemmata can be left out without decreasing performance, indicating that the information in the lemmata is captured in the other features. For the other tasks, however, the lemmata are the largest contributors. The reverse holds for

TABLE 3. Performance of Strategies and Features (F1 Scores)

	Relation	Performance	Evaluation
Strategy			
Sentence	.562***	**.559**	**.580**
Relevant	.603***	**.521*** **	**.479*** **
Combined	**.631**	(.556)	.546***
Features			
Lemmata	(.003)	.016***	.016***
POS	.022***	(–.004)	.010**
Bigram	.015***	.009*	(–.003)
Dependency	.012***	(.007)	.006*
Thesaurus	(–.003)	.016***	.011***
Distr. Sim. (Adjectives)	(.004)	(–.007)	(.000)
Distr. Sim. (Nominals)	.007**	(–.004)	(–.006)
Distr. Sim. (Verbs)	.007*	(.006)	(.001)
Mutual Information	.014***	–.010*	(–.002)
Conjunction Patterns	.012***	(–.007)	.006*
N	8,681	5,988	5,773

The best model is set in boldface in the strategy rows. Cells in the features rows indicate performance degradation when leaving that feature out. Significance of differences based on t-test with 20 folds. Significant at $p < .001$ level; **Significant at $p < .01$ level; *Significant at $p < .05$ level; (..) not significant

the part-of-speech and surface bigram features: For the relation task they are the strongest contributors, while for the other tasks they contribute less or not at all. Interestingly, Brouwer's thesaurus works well for the performance and evaluation tasks, but does not contribute significantly to the relation task. Looking at the last five features, which are all based on word similarities using the unannotated corpus, we see that for the relation task all features contributed significantly. The largest gain came from the simplest method, the mutual information based on NEAR queries, and the conjunction patterns also scored well. For the distributional similarity features, which are specified per part-of-speech, we can see that the contribution of the adjectives is barely significant, while the contributions of the verbs and nouns are highly significant. This is not surprising, since relationships between concepts will often be expressed using verb phrases and noun–verb combinations such as *give support* or *pick a fight*. It does underscore that the traditional focus on adjectives in sentiment analysis might not be suited for determining the polarity of relations. For the descriptive tasks, the clustering methods perform worse: For evaluations only, the conjunction patterns improve significantly, while for the performance task the mutual information even decreases performance significantly.

TABLE 4. Performance of the Model on the Different Classes (F1 Scores)

Class	Relation	Performance	Evaluation
Negative	.818	.466	.591
Neutral	.697	.759	.776
Positive	.576	.473	.362
N	8,681	5,988	5,773

Error Analysis

In order to improve the performance of the system, it is interesting to see whether we can detect patterns in the errors made by the system. Table 4 lists the performance of the model on each target class. This paints an interesting picture: For the relation task, the conflicts or negative issue positions are easier to detect than the neutral or positive ones. Possibly, the language in which criticism is expressed is less ambiguous than that in which positive sentiments are expressed. For the performance and evaluation tasks, the picture is different: The performance on the neutral category is much higher than that on the other categories. Apparently, detecting sentiment in these descriptions is easier than classifying the sentiment. Worst performance was attained on positive evaluations, which is probably due to the low frequency of these statements (11%; see Table 5).

Table 5 lists the confusion matrix per task in table percentages. Each cell contains the percentage of cases that belonged to a certain class according to the manual annotation and were assigned a certain class by the system. The bottom row and last column for each task show the total percentage of cases assigned to a class according to the system and manual annotation, respectively. For example, the first data column shows that 29% of all cases in the relation task were assigned the negative class by the system, out of which 18% were also assigned that class by the manual annotation. The remaining 11% were divided over classes assigned the neutral (6%) and positive (5%) class by the manual annotation. This table shows that there is no systematic bias in the errors made by the system: the marginal distributions of the predicted classes are very similar to the marginal

TABLE 5. Confusion Matrix per Task (Percentages)

Manual	Relation (*N* = 8,681)				System Performance (*N* = 5,988)				Evaluation (*N* = 5,773)			
	Neg.	Neut.	Pos.	Tot.	Neg.	Neut.	Pos.	Tot.	Neg.	Neut.	Pos.	Tot.
Negative	18	6	6	30	10	8	5	23	15	12	2	29
Neutral	6	29	7	42	6	43	5	55	9	49	2	61
Positive	5	7	17	29	5	8	10	22	3	5	3	11
Total	29	42	29	100	21	59	20	100	27	66	7	100

distributions of the actual classes. Moreover, for every task and target class, the mistakes seem divided over the other classes according to the marginal distribution.

Finally, Table 6 lists the performance of the model for each statement type, following the statement types described in section two with an additional distinction between political actors (ministers, parliamentarians) and other actors (such as citizens and pressure groups). In each task, the performance on statements involving political actors is highest followed by the performance on issues; performance on other actors is worst. Possibly the language used in these cases is simply less explicit or more diverse, but it is also possible that it is an effect of the higher frequency of political actors, caused in part by the overrepresentation of actors due to the label matching problem described in the Corpus section. This suggests that it could be interesting to either train separate models for the different actor types or to use features to allow the model to distinguish between them.

VALIDATION

In the previous section, we calculated the performance of our system by comparing the outcome with the manual annotations at the level of measurement. For political analysis, a much more important question than how well it performs on individual sentences, however, is how well it answers the questions it was designed for (cf. Krippendorff, 2004, p. 243): measuring how actors and issues are framed and portrayed. Since the precise answer depends on the (political) research question, we will take a number of analyses performed previously on the Dutch 2006 campaign data (Kleinnijenhuis, Scholten, et al., 2007) and test how well the results of these analyses based on the outcome of the system match the results based on manual analysis. Specifically, we will look at the overall tone of the news during the campaign, the issue positions taken by political parties, the patterns of conflict and support between parties in different periods, and whether newspapers differ in their attribution of success to the different parties.

TABLE 6. Performance on Different Statement Types

Task	Statement type	*N*	F1 score
Relations			
	Conflict (politicians)	3,420	.680
	Conflict (other actors)	2,258	.592
	Issue positions (polticians)	1,276	.552
	Issue positions (other actors)	836	.477
	Issue causality	627	.544
	Other	264	.526
Perform			
	Success / Failure (politicians)	2,962	.581
	Success / Failure (other actors)	608	.467
	Real world developments (issues)	2,416	.554
Evaluate			
	Evaluation of politicians	3,179	.577
	Evaluation of other actors	1,227	.523
	Evaluation of issues	1,366	.524

Overall Tone of the News

It is often claimed that news is becoming more negative, especially during campaigns (Patterson, 1993). In that light, it is interesting to look at the tone of the news operationalized as the average polarity of all statements. Figure 3 shows the graphs of the tone of the news for three news types: evaluations, issue positions, and conflict. For each graph, the two lines represent the results computed on the basis of the manual annotations and on the system output.

In the topmost graph, we can see that direct evaluations become more negative very slowly after the second week, going from −.11 to −.16. The issue positions also become less positive over time, meandering from around .3 in the first weeks to .1 in the last. Interestingly, the conflict news, defined as all relations between actors, seems to become more positive, going from −.25 to around neutral. This is probably because the beginning of the campaign was characterized by a strong clash between the leaders of the PvdA (Social Democrats) and CDA (Christian Democrats), while in the last weeks there was a détente between the left-wing parties. The lines of the manual annotation and system output follow each other very closely. In fact, the two are correlated with a

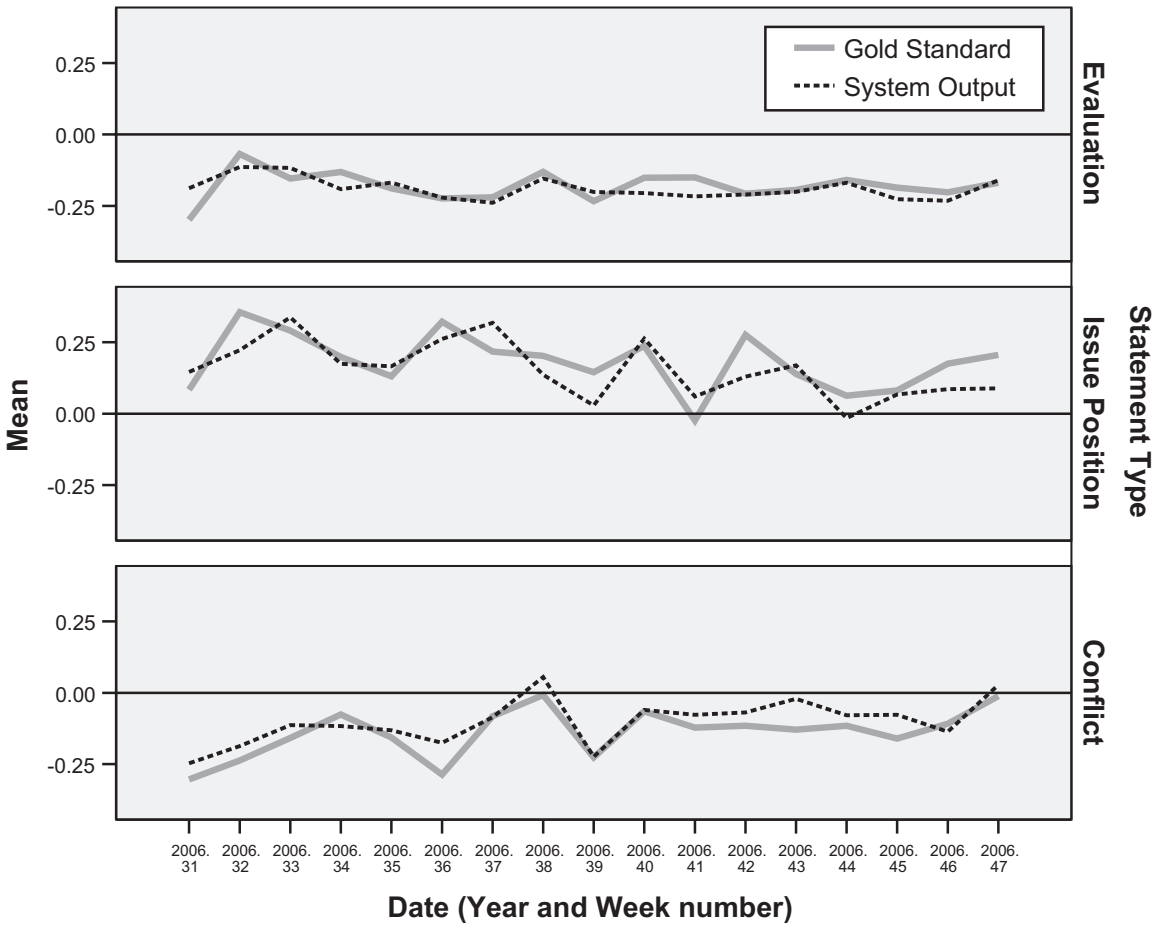


FIGURE 3. The overall tone of the news according to the manual annotation and system output.

coefficient of .9, albeit based on only three scores for 17 weeks ($N = 51$). From this we can conclude that the current system is certainly capable of finding patterns in the overall tone of the news.

Issue Positions

Another aspect of political campaigns is the issue positions taken by different actors. There are a number of theories that predict a relation between (perceived) issue positions and voting behavior, such as directional voting (Rabinowitz & MacDonald, 1989) and spatial/proximity voting (Westholm, 1997). This section analyzes party issue positions, replicating Table 4.2 in Kleinnijenhuis, Scholten, et al. (2007, p. 75), although the actual results may differ since this is based on the smaller selection of matched sentences as described in the Corpus section.

Table 7 shows the average issue position of three parties on three issue categories: leftist issues (such as job security, welfare), rightist issues (such as taxation, defense), and administrative reforms (such as the referendum and elected mayor). For each issue category, three columns are given: the number of issue statements from the party on that issue, and the average polarity according to the manual analysis and according to the system. On the left side of the table, we see that the PvdA (Social Democrats) are in favor of leftist issues while the conservative VVD are against, with the CDA (Christian Democrats) taking the middle ground. Looking at rightist issues, the reverse happens: The VVD is in favor and the PvdA

TABLE 7. Issue Positions on a Number of Issues (–1 ..1)

Party	Leftist			Rightist			Reforms		
	N	Manual	System	N	Manual	System	N	Manual	System
PvdA	71	.3	.3	37	–.1	.0	4	.8	.5
CDA	72	.2	.1	111	.3	.2	23	.3	.5
VVD	56	–.1	.1	101	.1	.3	34	–.1	.1

Example: There were 72 statements in which the CDA took a position on leftist issues; according to the manual analysis, the CDA was somewhat positive (+.2), while according to the model output this is only +.1.

against. There is a discrepancy here between the manual annotation and the system: According to the manual annotation, the CDA is slightly more rightist than the VVD, while the system places the VVD to the right of the CDA.

On the last issue presented here, administrative reforms, the PvdA is strongly in favor according to the manual annotation, while the more conservative CDA and VVD are slightly above and below zero. According to the system, however, the CDA is also quite positive, placing them side by side with the PvdA. An important example of a reform issue during the election was whether to hold a referendum on the new EU constitutional treaty, of which the PvdA was in favor. The greater divergence on this issue between manual annotation and system output is probably due to the lower number of statements on which this is based, allowing for less room for individual errors.

Comparing the manual annotation and our system, we see that they diverge, although they are generally in the same range. This is confirmed by a correlation analysis on the full selection of seven parties and 14 issues, which shows a weighted correlation coefficient of .72 (N = 98). Generally speaking, the result of the model will be similar to the results based on manual annotation, but there will be small errors if one looks at the details, especially for the parties and issues receiving little attention; this latter point is reflected in the fact that the correlation coefficient that is not weighted for frequency is only .58. If we were to trace the development of issue positions over time, for example per week, the weighted and unweighted correlations also drop, to .58 and .54, respectively. Thus, in general the system is good enough to answer questions on party issue positions, but it performs less well on smaller parties and issues or for smaller time periods.

Political Conflict

An interesting part of campaigns in multiparty systems is the pattern of support and criticism between the parties, as the various parties balance discrediting and ignoring the other parties while also keeping possible future coalitions in mind. Here, we shall replicate the analyses presented graphically in Figures 5.1–5.3 of Kleinnijenhuis, Scholten, et al. (2007, pp 84–89), which show the network of party relations for three periods.

Table 8 shows a sample of this network based on the manual annotations and system output. Each row represents the relation between two specific parties. For each relation, the number of statements (N) and average polarity according to the manual annotation and according to the system is given for three time periods. The top two rows show the mutual relations between the CDA and PvdA, which were seen as the main contestants for becoming the largest party and had strong negative relations during the whole period. The third row shows the internal support and criticism within the PvdA. In the first period, the PvdA has internal problems after forcing two Turkish candidates to withdraw because they refuse to acknowledge the Armenian genocide. According to the manual annotations, they continue to have some internal problems, although the system actually measures a moderate positive internal relation in the second period. The bottom rows show the relation between the PvdA and more extreme Socialist Party. The PvdA

TABLE 8. Support and Criticism in Three Periods (from –1 +1)

Subject	Object	1 Sept–15 Oct			16 Oct–13 Nov			14 Nov–22 Nov		
		N	Manual	System	N	Manual	System	N	Manual	System
CDA	PvdA	65	–.6	–.6	79	–.8	–.5	23	–.8	–.7
PvdA	CDA	78	–.8	–.6	70	–.6	–.5	34	–.7	–.4
PvdA	PvdA	47	–.2	–.3	25	–.1	.6	7	–.1	.2
PvdA	SP	0	–	–	10	–.7	.3	17	.6	.4
SP	PvdA	3	–.3	–.3	3	–1.0	–1.0	7	.7	.4

Example: In the second period there are 79 statements in which the CDA expresses an opinion about the PvdA. According to the manual analysis, this opinion was strongly negative (–.8), while according to the system output this was slightly less so (–.5).

completely ignores the SP in the beginning while there is some sharp but low-frequency criticism during the second period. In the last period, relations turn positive, although the system measures it as being slightly less positive than the manual annotation.

If we compare the full network based on the manual annotation and system output for these three periods, we find a weighted correlation of .77 ($N = 21$), which is certainly acceptable. If we do this comparison per week rather than in three periods, this drops to .66 ($N = 119$). From this we conclude that the performance of the system is certainly good enough to analyze party relations over fairly large time spans, and probably good enough for more detailed analysis.

Party Performance and Newspaper Preferences

During campaigns, a significant portion of the news is always devoted to the horse race: Who is winning in the polls; who won the last debate; who has the best chances of becoming Prime Minister? Because of the bandwagon effect (Lazarsfeld et al., 1944), for a candidate to be portrayed as successful is often a self-fulfilling prophecy (Bartels, 1988). Hence, it is interesting to investigate which newspapers portray which parties as being successful or failing. This replicates the analysis presented in Kleinnijenhuis, Scholten, et al. (2007, table 6.4, p 104).

Table 9 shows the performance of the main parties according to three newspapers. According to the manual annotations, *de Volkskrant*, a left-wing quality newspaper, portrays the CDA and PvdA as somewhat successful and neutral, respectively, while the system classifies in the reverse order. According to manual annotation and the system, the VVD is depicted as failing. The popular conservative newspaper *De Telegraaf* portrays the CDA as being fairly successful and the PvdA as failing. The system completely misses this, portraying the CDA as

TABLE 9. Performance of the Main Parties According to Three Newspapers

Party	de Volkskrant			De Telegraaf			Trouw		
	N	Manual	System	N	Manual	System	N	Manual	System
1 CDA	91	.2	.0	82	.3	.0	81	–.2	–.2
5 PvdA	82	.0	.2	40	–.5	.2	34	–.6	–.2
7 VVD	79	–.2	–.3	67	–.1	.0	50	–.3	.1

Example: *De Volkskrant* contained 91 evaluative statements about the CDA. According to the manual analysis, these statements were on average slightly positive (+.2), while according to the system output they were neutral (.0).

neutral and the PvdA as successful. The left-wing confessional newspaper *Trouw* also portrays the PvdA as failing, but is also negative about the performance of the other parties. The system measures a less extreme failing for PvdA, and a slight success for the VVD.

It seems from Table 9 that the performance of the system on this case is worse than for the other use cases, which is confirmed by a correlation coefficient of only .46. The most likely explanation for the difficulty the system has with this task is that the classifier for performance statements was the weakest. This underscores the importance of optimizing the used classifier, even if performance on higher levels of aggregation can be higher than on the sentence level.

F1 Score and Usefulness

The main conclusion of the results section is that the full model improved significantly over the simpler *lemma* baseline model, but left room for improvement. An interesting question is whether the .03 to .07 increase in F1 score gained by adding all the features of the full model translates into a better answer to the political research questions. To answer this question, we have calculated the same correlations as presented above using the output of the *lemma* baseline (see the Results section above). The results of this comparison are presented in Table 10. For each of the subsections in this section, we list the correlations as reported above and the analogous correlation achieved using the baseline model.

TABLE 10. Comparison of Correlations With Gold Standard of Full Model and Baseline

Section	Analysis	Full model	*Lemma* baseline
6.1	Tone of the news	.933	.907
6.2	Issue positions (whole period)	.718	.516
	Issue positions (per week)	.575	.423
6.3	Political conflict (per period)	.774	.591
	Political conflict (per week)	.656	.527
6.4	Party performance per newspaper	.460	.603

For the tone of the news, the full model outperforms the *lemma* baseline but both score greater than .9. For the analyses of issue positions and political conflict, the performance of the baseline is substantially lower, and for the more coarse-grained analyses the correlation drops from a very acceptable .72 and .77 to a meagre .52 and .59. Interestingly, the baseline model actually outperforms the full model on classifying the success of parties per newspaper, although the performance of the full model on that use case was already fairly poor. These results show two things: First, that the full model presented here is substantially better than the baseline model even if compared on a higher level of analysis. Second, and more important, it shows that a relatively modest increase in F1 score of .07 can lead to increases in correlation of 20% points on a higher level of aggregation and can make the difference between being good and being not quite good enough.

CONCLUSION

This article presents a system that automates an important step in semantic network analysis: the determination of the polarity of relations and descriptions—that is, whether those relations and descriptions are positive or negative. We analyze three types of statements: relations between actors and issues; evaluations of actors and issues; and performance descriptions of actors and issues. Using techniques from sentiment analysis, we train a machine learning model using a number of lexical features, and use syntactic analysis to focus the model on the specific relation or description rather than the whole sentence. The model is trained and tested on a manual semantic network analysis of the Dutch 2006 parliamentary elections (Kleinnijenhuis, Scholten, et al., 2007).

The direct comparison of the automatic analysis with the manual annotations on the level of sentences shows that the system can reproduce these annotations reasonably well and significantly better than a baseline model based on only the lemma frequencies. The syntactic

analysis improves the classification of relations but does not improve the classification of evaluations and performance descriptions.

In order to show that the system is useful for political research, we compare the automatic analysis with the manual semantic network analysis at the level of analysis of politically interesting phenomena. This is performed for four different use cases taken directly from the original election study. We find that the system can immediately be used for determining the overall tone of the news, overall issue positions, and party conflict patterns in the periods used in the original analysis. For determining party performance according to the different newspapers, and for analyzing issue positions and conflict patterns in smaller periods, the system performance is lower due to the underlying model performance and the high granularity (the low number of statements within one unit of analysis). These tests show that it is important to validate automatic content analysis methods on the actual task to be performed rather than rely on sentence-level performance. Moreover, we show that a modest increase in F1 score at the level of measurement can lead to substantially better performance at a higher level of analysis.

To further improve the system, we analyze the errors made by the system and suggest a number of ways to alleviate these. First, we can increase the size of our training corpus by resolving the matching problems described above and by combining the corpus with other existing manually analyzed corpora, such as the other Dutch election campaigns, which have been analyzed using the same method since 1994. Second, the methods used to create clusters of similar words based on the reference corpus did not perform as well as we had hoped. Using a seed set of words that are good indicators of polarity, we can create these clusters in a more focused manner, hopefully leading to better performance. Finally, since using the syntactic analysis to determine the predicate improved performance for the relations but not for the descriptions, we can probably improve performance by modifying which part of the sentence is contained in the predicate for the descriptions.

Although the performance of the system presented here leaves room for improvement, this article presents two meaningful contributions. First, this is the first article that applies sentiment analysis techniques to the Dutch language, showing that the techniques for English also work for Dutch and providing a baseline and infrastructure for future Dutch sentiment analysis research. Second, by duplicating a number of analyses from an existing election study, we provide external validation of the sentiment analysis techniques used and give the political scientist insight into how well these techniques really perform at answering his or her research question.

We think there are three requirements for these techniques to be really useful in political science: (a) Performance must be good enough, (b) the system must be usable for nonexpert users, and (c) it must be applicable or portable to different contexts and languages. This article shows that the performance is already good enough for certain tasks, especially if there is enough textual source material and the required granularity is not too fine, and that there is a lot of potential for performance increase in future work. Moreover, political analysis rarely requires perfect performance, and human coding is deemed acceptable if the accuracy corrected for chance agreement is higher than 67% (Krippendorff, 2004, p. 241). As for the second requirement, the system is fully automated, so it would be easy to create a program or Web service to run it on any text without any technical expertise. Although the processing is fairly time-consuming because of the requirement to fully parse the text, this is measured in seconds per sentence rather than minutes, and with computing power being cheap and likely to get cheaper in the future, this allows for very large corpora to be analyzed at relatively small expense.

The third requirement is more problematic. Whether the system can be immediately applied to other contexts and genres depends on the similarity in language use, and must be a subject of future work. Presumably, the current system will perform better on related domains, such as earlier election studies or other political news rather than forum postings. If we gather a

larger and more diverse corpus, we can use that corpus to determine how domain-specific the techniques are and whether we need to create separate models for different domains.

Applying the techniques discussed here to different languages will certainly require retraining the models. This is only possible if the linguistic tools, such as syntactic parsers, exist for that language. Moreover, it requires sufficient manually coded training data, making it an expensive undertaking. Fortunately, such training data is a natural by product of a semantic network analysis, as long as we make sure that the data from a normal analysis are suitable by keeping the original text and linking the coded relations to the words in the text on which they are based. Additionally, there is a large amount of existing data that can be used as training material by matching it to the original text. In this way, we can create the corpus needed for training new models or improving existing models without incurring additional expenses.

If these three requirements can be met, gathering content analysis data will become much faster and less expensive. This means that larger data sets can be produced, allowing for more complex modeling of interesting interactions between politics, media, and the public. Since content analysis currently is a significant bottleneck for investigations into media functioning and effects, this will greatly decrease the costs and increase the scope of such studies. Moreover, the faster coding means that the analysis can be performed quickly after an event has taken place, making it easier to bring scientific analysis into the public debate on politics and media functioning, potentially contributing to the quality of that debate.

REFERENCES

Abney, S. (1997). Stochastic attribute-value grammars. *Computational Linguistics, 23*, 597–618.

Althaus, S., Edy, J., & Phalen, P.(2001). Using substitutes for full-text news stories in content analysis. *American Journal of Political Science, 45*(3), 707–724.

Axelrod, R. (Ed.). (1976). *Structure of decision: The cognitive maps of political elites*. Princeton, NJ: Princeton University Press.

Azar, E. (1980). The conflict and peace data bank (COPDAB) project. *The Journal of Conflict Resolution, 24*(1), 143–152.

Baroni, M., & Vegnaduzzo, S. (2004). Identifying subjective adjectives through Web-based mutual information. In I. E. Buchberger (Ed.), *Proceedings of KONVENS 2004* (pp. 17–24). Vienna: ÖGAI.

Bartels, L. (1988). *Presidential primaries and the dynamics of public choice*. Princeton, NJ: Princeton University Press.

Berger, A. L., Della Pietra, S. A., & Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics, 22*(1), 39–71.

Breck, E., Choi, Y., & Cardie, C. (2007). Identifying expressions of opinion in context. In M. M. Veloso (Ed.), *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*. Hyderabad, India.

Brouwers, L. (1989). *Het juiste woord: Standaard betekeniswoordenboek der nederlandse taal (7th edition; ed. f. claes)*. Antwerpen, Belgium: Standaard Uitgeverij.

Choi, Y., Breck, E., & Cardie, C. (2006). Joint extraction of entities and relations for opinion recognition. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*. Sydney, Australia.

Corman, S., Kuhn, T., McPhee, R., & Dooley, K. (2002). Studying complex discursive systems: Centering resonance analysis of communication. *Human Communication Research, 28*(2), 157–206.

De Ridder, J., & Kleinnijenhuis, J. (2001). Media monitoring using ceta: The stock-exchange launches of kpn and wol. In M. D. West (Ed.), *Applications of computer content analysis* (Vol. 17, pp. 165–84). New York: Ablex Publishing.

Diehl, P. (1992). What are they fighting for? The importance of issues in international conflict research. *Journal of Peace Research, 29*(3), 333–344.

Diesner, J., & Carley, K. (2004). *Automap1.2 – extract, analyze, represent, and compare mental models from texts* [Technical Report No. CMU-ISRI-04-100]. Pittsburgh, PA: Carnegie Mellon University.

Dille, B., & Young, M. (2000). The conceptual complexity of presidents Carter and Clinton: An automated content analysis of temporal stability and source bias. *Political Psychology, 21*(3), 587–596.

Entman, R. M. (1993). Framing: Toward clarification of a fractured paradigm. *Journal of Communication, 43(4)*, 51–58.

Evans, G., & Andersen, R. (2006). The political conditioning of economic perceptions. *The Journal of Politics, 68*(1), 194–207.

Fan, D. (1996). Predictions of the Bush-Clinton-Perot presidential race from the press. *Political Analysis, 6*, 67–105.

Grefenstette, G., Qu, Y., Evans, D., & Shanahan, J. (2006). Validating the coverage of lexical resources for

affect analysis and automatically classifying new words along semantic axes. In J. Shanahan, Y. Qu, & J. Wiebe (Eds.), *Computing attitude and affect in text: Theory and applications* (pp. 93–108). Dordrecht, the Netherlands: Springer.

Hartman, K. (2000). Studies of negative political advertising: An annotated bibliography. *Reference Services Review, 28*(3), 248–261.

Hatzivassiloglou, V., & McKeown, K. (1997). Predicting the semantic orientation of adjectives. In P. R. Cohen & W. Wahlster (Eds.), *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics (ACL-97)* (pp. 174–181). Somerset, NJ: Association for Computational Linguistics.

Hatzivassiloglou, V., & Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING-2000)* (pp. 299–305). San Francisco, CA: Morgan Kaufmann.

Hetherington, M. (1996). The media's role in forming voters' national economic evaluations in 1992. *American Journal of Political Science, 40*(2), 372–395.

Iyengar, S., Norpoth, H., & Hahn, K. (2003). Consumer demand for election news: The horse race sells. *Journal of Politics, 66*, 157–175.

Janis, I., & Fadner, R. (1943). The coefficient of imbalance. *Psychometrika, 8*(2), 105–119.

Jones, B., & Baumgartner, F. (2005). *The politics of attention: How government prioritizes problems*. Chicago: University of Chicago Press.

Kanamaru, T., Murata, M., & Isahara, H. (2007). Japanese opinion extraction system for Japanese newspapers using machine-learning method. In *Proceedings of NTCIR-6 Workshop Meeting*. Tokyo, Japan.

Kim, S.-M., & Hovy, E. (2006). Extracting opinions expressed in online news media text with opinion holders and topics. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text at the Joint COLING-ACL 2006 Conference* (pp. 1–8). Sydney, Australia.

Kirkpatrick, B. (1998). *Roget's thesaurus of English words and phrases*. Harmondsworth, England: Penguin.

Kleinnijenhuis, J., & Pennings, P. (2001). Measurement of party positions on the basis of party programmes, media coverage and voter perceptions. In M. Laver (Ed.), *Estimating the policy positions of political actors* (pp. 162–182). London and New York: Routledge

Kleinnijenhuis, J., Scholten, O., Van Atteveldt, W., Van Hoof, A., Krouwel, A., Oegema, D., et al. (2007). *Nederland vijfstromenland: De rol van media en stemwijzers bij de verkiezingen van 2006*. Amsterdam: Bert Bakker.

Kleinnijenhuis, J., Van Hoof, A., Oegema, D., & De Ridder, J. (2007). A test of rivaling approaches to explain news effects: News on issue positions of parties, real-world developments, support and criticism and success and failure. *Journal of Communication, 57*(2), 366–384.

Krippendorff, K. (2004). *Content analysis: An introduction to its methodology (2nd ed.)*. Thousand Oaks, CA: Sage Publications.

Laver, M., Benoit, K., & Garry, J. (2003). Extracting policy positions from political texts using words as data. *American Political Science Review, 97*(2), 311–331.

Laver, M., & Garry, J. (2000). Estimating policy positions from political texts. *American Journal of Political Science, 44*(3), 619–634.

Lazarsfeld, P., Berelson, B., & Gaudet, H. (1944). *The people's choice: How the voter makes up his mind in a presidential campaign (3rd ed.)*. New York: Duell, Sloan, and Pearce.

Levin, B. (1993). *English verb classes and alternations*. Chicago: University of Chicago Press.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *In Proceedings of COLING-ACL* (pp. 768–774). Montreal, Canada.

Manning, C., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge, MA: MIT Press.

Mathieu, Y. (2006). A computational semantic lexicon of French verbs of emotion. In J. Shanahan, Y. Qu, & J. Wiebe (Eds.), *Computing attitude and affect in text: Theory and applications* (pp. 109–123). Dordrecht, the Netherlands: Springer.

Mihalcea, R., Banea, C., & Wiebe, J. (2007). Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07)* (pp. 968–975). Prague, Czech Republic.

Monge, P. R., & Contractor, N. S. (2003). *Theories of communication networks*. Oxford, England: Oxford University Press.

Osgood, C., Saporta, S., & Nunnally, J. (1956). Evaluative assertion analysis. *Litera, 3*, 47–102.

Patterson, T. (1993). *Out of order*. New York: Knopf.

Pit, M. (2003). *How to express yourself with a causal connective: Subjectivity and causal connectives in Dutch, Qerman and French*. Amsterdam: Rodopi.

Popping, R. (2000). *Computer-assisted text analysis*. Newbury Park/London: Sage.

Purpura, S., & Hillard, D. (2006). Automated classification of congressional legislation. In *Proceedings of the 2006 International Conference on Digital Government Research, ACM International Conference Proceeding Series; vol. 151* (pp. 219–225). New York: ACM.

Rabinowitz, G., & MacDonald, S. (1989). A directional theory of issue voting. *American Political Science Review, 83*, 93–122.

Ratnaparkhi, A. (1998). *Maximum entropy models for natural language ambiguity resolution*. Unpublished doctoral dissertation, University of Pennsylvania, Philadelphia.

Riloff, E., & Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *Conference on empirical methods in natural language processing (EMNLP-03), ACL SIGDAT* (pp. 105–112). Sapporo, Japan.

Roberts, C. W. (Ed.). (1997). *Text analysis for the social sciences: Methods for drawing statistical inferences from texts and transcripts*. Mahwah, NJ: Lawrence Erlbaum.

Scheufele, D. (1999). Framing as a theory of media effects. *Journal of Communication, 29*, 103–123.

Schrodt, P., & Gerner, D. (1994). Validity assessment of a machine-coded event data set for the middle east, 1982–1992. *American Journal of Political Science, 38(3)*, 825–854.

Seki, Y., Evans, D. K., Ku, L.-W., Chen, H.-H., Kando, N., & Li, C.-Y. (2007). Overview of opinion analysis pilot task at NTCIR-6. In *Proceedings of NTCIR-6 Workshop Meeting*. Tokyo, Japan.

Shah, D., Watts, M., Domke, K., Fan, D., & Fibison, M. (1999). Television news, real world cues, and changes in the public agenda 1984–1996. *Journal of Politics, 61*, 914–943.

Shanahan, J., Qu, Y., & Wiebe, J. (Eds.). (2006). *Computing attitude and affect in text: Theory and applications*. Dordrecht, the Netherlands: Springer.

Soroka, S. (2006). Good news and bad news: Asymmetric responses to economic information. *Journal of Politics, 68*(2), 372–385.

Stone, P., Bayles, R., Namenwirth, J., & Ogilvie, D. (1962). The general inquirer: A computer system for content analysis and retrieval based on the sentence as a unit of information. *Behavioral Science, 7*, 484–498.

Tjong Kim Sang, E., & Hofmann, K. (2007). Automatic extraction of Dutch hypernym-hyponym pairs. In *Proceedings of CLIN-06*. Leuven, Belgium.

Van Atteveldt, W., Kleinnijenhuis, J., & Ruigrok, N. (2008). Parsing, semantic networks, and political authority: Using syntactic analysis to extract semantic relations from Dutch newspaper articles. In *Proceedings of the Etmaal van de Communicatiewetenschap*. Amsterdam, the Netherlands.

Van Atteveldt, W., Schlobach, S., & Van Harmelen, F. (2007). Media, politics and the semantic web: An experience report in advanced rdf usage. In E. Franconi, M. Kifer, & W. May (Eds.), *ESWC 2007* (pp. 205–219). Berlin, Germany: Springer.

Van Cuilenburg, J. J., Kleinnijenhuis, J., & De Ridder, J. A. (1986). Towards a graph theory of journalistic texts. *European Journal of Communication, 1*, 65–96.

Van Noord, G. (2006). At last parsing is now operational. In P. Mertens, C. Fairon, A. Dister, & P. Watrin (Eds.), *Verbum ex machina, Actes de la 13e Conference sur le Traitement Automatique des Langues Naturelles* (pp. 20–42). Louvain-la-Neuve, Belgium: Presses Universitatires de Louvain.

Wasserman, S., & Faust, K. (1994). *Social network analysis*. Cambridge, England: Cambridge University Press.

Westholm, A. (1997). Distance versus direction: The illusory defeat of the proximity theory of electoral choice. *American Political Science Review, 91*, 865–883.

Wiebe, J. (2000). Learning subjective adjectives from corpora. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00)*. Austin, TX.

Wiebe, J., Wilson, T., Bruce, R. F., Bell, M., & Martin, M. (2004). Learning subjective language. *Computational Linguistics, 30(3)*, 277–308.

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing HLT-EMNLP-2005*. Vancouver, Canada.

Xu, R., Wong, K.-F., & Xia, Y. (2007). Opinmine—opinion analysis system by for NTCIR-6 pilot task. In *Proceedings of NTCIR-6 Workshop Meeting*. Tokyo, Japan.