Actor-Role Analysis: Ideology, Point of View, and the News*

Warren Sack
MIT Media Laboratory
20 Ames Street, E15-450
Cambridge MA 02139
USA
email: wsack@media.mit.edu
phone: 617/253-0323
fax: 617/258-6264

Abstract A representation of *ideological point of view* is articulated and a method for detecting the point(s) of view expressed in a news story is described. A version of the method, *actor-role analysis*, is encoded in a computer program, *SpinDoctor*, which can automatically detect the point(s) of view represented in some news stories. Results obtained by testing SpinDoctor on a corpus of news stories are reported.

1.0 Introduction

This paper describes a computational technique for text analysis, specifically, for the analysis of news texts. It improves upon existing work in artificial intelligence (AI) natural language processing (NLP) by extending several key concepts from semiotics and narrative theory (e.g., Fredric Jameson's (Jameson, 1987) use of Greimasian semiotics to represent ideological limits). Previous work

* Appears in *New Perspectives on Narrative Perspective*, Will Van Peer and Seymour Chatman (eds.) (Albany, NY: State University of New York Press, 2001): pp., 189-204.

in AI news story understanding has largely been used to build tools which can summarize stories and categorize them according to the events they describe (e.g., the technologies developed for the Message Understanding Conferences[1] (MUC)). These sorts of technologies are built around the assumptions that (1) events reported as facts in news stories should be "understood" as facts; (2) the style of a story, i.e., the way in which the story is told, is not of interest; and, (3) the source of a story should not influence its analysis. These assumptions are obviously unrealistic. By making these simplifying assumptions most existing story understanding systems function as gullible "readers."[2] Everyone knows that one should not believe everything in the news. Reading more selectively requires one to determine not only what events are being reported in a story, but also whose story, whose *point of view*, is expressed in a news article. To build a less gullible story understander it will be necessary to encode in it a means to recognize point of view.

By point of view I mean *ideological* point of view as opposed to, for example, *psychological* point of view. Ideological point of view characterizes the political slant of an entire story while psychological point of view (e.g., as it is used by Wiebe, 1994) characterizes the source of a given sentence or statement contained within a story.

---

[1] The Message Understanding Conferences are sponsored by the Defense Advanced Research Projects Agency (DARPA) to test leading artificial intelligence natural language processing programs against one another. In this paper some of the MUC3 (1991) test corpus will be examined.

[2] There are some early exceptions (e.g., Abelson and Carroll, 1966) and this neglect has not been complete in closely related areas (in dialog and argument understanding, e.g., Allen and Perrault, 1980; Birnbaum, Flowers and McGuire, 1980; Alvarado, 1990; intelligent tutoring systems, e.g., Farrell and Bloch, 1988; and, in the field of language generation, e.g., Hovy, 1988).

This paper describes *SpinDoctor*, a computer program designed to detect ideological point of view in news stories. To detect point of view, SpinDoctor implements a critical reading strategy called *actor-role analysis* (Sack, 1994b). Actor-role analysis was developed around the following observation: one means of detecting point of view is to examine how certain people, who appear again and again in the news (i.e., news *actors*), are described or portrayed (i.e., are assigned *roles*). Thus, for example, if one is given a news story which mentions Oliver North -- (in)famous for his role in the Iran-Contra affair and recent senatorial campaign -- and the story assigns North the role of *patriot* (via the use of certain adjectives and verbs), one can be quite certain that the point of view expressed in the story is significantly to the right (in the spectrum of US politics) than that expressed by another news story which assigns North the role of *villain* or *criminal*.

Two aspects of actor-role analysis, as it is implemented in the SpinDoctor system, might be of especial interest to researchers concerned with textual analysis:

(1) *A representation for ideological point of view*: Although practically all, contemporary, AI systems for NLP are capable of finding actors and roles in texts to fill in scripts, frames, or templates (e.g., Jacobs and Rau, 1993), none of these systems assign any political significance to the pairing of certain actors with certain roles (e.g., North as patriot versus North as criminal). By contrast, we maintain that sets of actor-role pairs are an interesting and implementable representation for differing ideological points of view. The proposed actor-role representation of ideological point of view accords with some recent work by Lakoff (1991) and generalizes and improves upon previous AI work on

representation of ideology (e.g., Abelson and Carroll, 1966; Carbonell, 1978). Actor-role analysis can also be seen as a computational analog of the Greimasian *actor-network* method employed by sociologists to study controversies in the literatures of technoscience (e.g., Latour, 1987).

(2) *An algorithm for anaphoric resolution*: Actor-role analysis incorporates an anaphoric resolution algorithm. It is shown how careful attention to actor-role pairings assists in the resolution of anaphoric reference. By noting, for example, that an instance of the pronoun "he" is cast in the role of *victim* and that, earlier in the same story, "Lieut. Rodriguez" is also cast in the role of *victim*, SpinDoctor postulates the resolution of the instance of the pronoun "he" to the proper name "Lieut. Rodriguez."

This paper contains three main sections:

(i) *Data structures*: First, the actor, role and point of view data structures employed in SpinDoctor are described. An example of the use of sets of actor-role pairs to represent ideological point of view is given and then actor-role pairs are briefly compared to other AI work on the representation of ideology.

(ii) *Analysis process*: SpinDoctor's analysis process is presented and several interesting subroutines isolated for discussion. In particular, the subroutine *coreference-actors* is described in detail to show how anaphoric resolution is handled by SpinDoctor.

(iii) *Performance test*: Finally, results obtained by testing SpinDoctor on one hundred news stories from the MUC3 corpus are reported.

2.0 Data Structures

The abstract data structures used by the SpinDoctor system are described in this section. Particular examples of the data structures are taken from a database created to test SpinDoctor's performance on news stories from El Salvador (from the MUC3 corpus). Results of that test are given in a later section.

2.1 Actors

The actor data structures have three properties associated with them:

- *Names*: Each actor can be recognized according to one or more names. Usually the names listed are proper names or titles.
- *Pronouns*: Declared pronouns assist SpinDoctor in its search for anaphoric referents. Certain actors might refer to groups of people and so each such actor could be referred to as "it." Alternatively, an actor (e.g., the actor *army-officer*, which is meant to refer to anyone with a military title (e.g., Gen., Col., Lieut., etc.)) could be referred to as "he" or "she"; while an actor defined to represent a specific individual (e.g., *bishop-romero*) would have pronouns of only one gender associated with it (e.g., he, him, etc.).
- *Member-of*: Almost all actors are declared to be a *member-of* a larger set of actors. Thus, for example, *bishop-romero* is declared to be a *member-of* the actor called *clergymen* which, in turn, is declared to be a *member-of* the *catholic-church*. Those actors which are not a *member-of* any other actor (e.g., the *catholic-church*.) are called *top-level actors*.

Here is an example definition for the actor Bishop Romero:
(defactor *bishop-romero*

:member-of clergyman

:names ((monsignor romero) (msgr romero) (bishop romero)
(oscar arnulfo romero))

:pronouns (*he*))

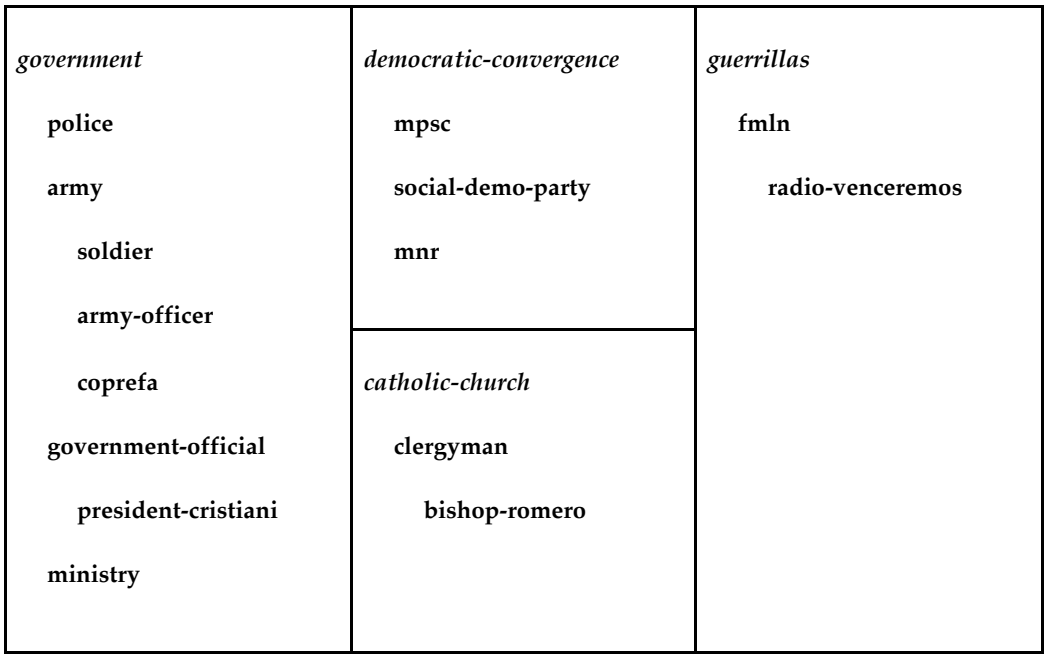

Figure 1: Actors and member-of relations between actors in a database for the analysis of Salvadoran news stories.

Figure 1 lists the actors defined for a database created for the analysis of Salvadoran news stories. The *member-of* relations between actors are denoted by indentations. The four top-level actors (*government*, *democratic-convergence*, *catholic-church*, and *guerrillas*) are shown in italics.

2.2 Roles

In the SpinDoctor system, roles are defined as sets of simple patterns that are lists of words and variables. Variables are role names and are prefixed with a question mark. For example, here is part of (about one-fourth of) the definition of the *criminal* role used in SpinDoctor:

(defrole *criminal* ?criminal murdered ?victim)
(defrole *criminal* ?criminal commit genocide)
(defrole *criminal* ?criminal participated in the crime)
(defrole *criminal* ?criminal is responsible for this crime)
(defrole *criminal* ?criminal were responsible for the massacre)
(defrole *criminal* ?criminal staged this assassination)
(defrole *criminal* the arrest of ?criminal)
(defrole *criminal* ?victim was abducted by ?criminal)
(defrole *criminal* ?judge decides to punish ?criminal)

Note that one pattern can help define two roles; e.g., the first pattern shown above mentions both the *?criminal* and *?victim* roles. SpinDoctor's analysis process attempts to match each of the variables to a noun phrase. The following section on SpinDoctor's analysis process will explain how these patterns are matched. However, no claim is made that either this representation of roles or the means by which they are matched is especially interesting since most AI-NLP systems do some matching of this sort, in a more or less sophisticated manner, to fill in the roles of templates. SpinDoctor's roles work simply because news

stories contain a large number of idiomatic phrases which are frequently repeated word-for-word in many stories.[3]

2.3 Points of View

Figure 2 shows the definitions for the four points of view used to analyze Salvadoran news. A point of view corresponding to each of the top-level actors was defined. However, nothing about the architecture of the system requires the names of the points of view to be related to the names of the top-level actors. Points of view are represented as pairings of top-level actors and roles.

```
(defpov
 government
 (government ?source)
 (government ?military)
 (guerrillas ?terrorist)
 (guerrillas ?criminal))
```

```
(defpov
 democratic-convergence
 (democratic-convergence
  ?source)
 (democratic-convergence
  ?victim))
```

```
(defpov
 guerrillas
 (guerrillas ?source)
 (guerrillas ?military)
 (government ?criminal))
```

[3] An earlier version of SpinDoctor (Sack, 1994a) used a more sophisticated version of role matching which included a syntactic parse and morphological analysis of the text of the stories and which, therefore, allowed the word order and, for example, verb tenses in the role patterns to be automatically varied. This extra level of sophistication was found to be unnecessary to demonstrate the central features of actor-role analysis: the determination of whose point of view is represented by a news story and the representation of ideological points of view.

```
(defpov
 catholic-church
 (catholic-church
  ?source)
 (catholic-church
  ?victim))
```

Figure 2: Definitions of four points of view useful for the analysis of Salvadoran news stories.

The news corpus used to test the system (the MUC3 corpus) was gathered in the late-eighties/early-nineties when El Salvador was in the midst of a civil war between the *government* and the *guerrillas* (the FMLN-FDR). Caught in the middle of the fighting were both the *Catholic Church* and the *Democratic Convergence* (a coalition of center-left political groups).

The government's point of view can be detected by looking for news articles in which the government was the *source* (i.e., played the role of *source*) of the story, the guerrillas were characterized as *criminals* and *terrorists*, and, in which, the violent actions of the government and military were characterized as legitimate *military* actions (rather than *criminal* actions).

The stories which represented the guerrilla's point of view often had the guerrillas as the *source* of the story and characterized the guerrilla's violent actions as legitimate *military* actions while assigning the role of *criminal* to the Salvadoran government and armed forces. In other words, the government, in these news stories, called the guerrillas *terrorists* and *criminals* while the guerrillas reciprocated by calling the government *criminals*. Displays of

complementary epideictic rhetoric of this sort were the means by which the points of view could be readily discerned.

Priests of the Catholic Church and leading politicians in the Democratic Convergence (a center-left organization) were murdered. Neither the Catholic Church nor the Democratic convergence had any military presence. It is not surprising then that neither of these parties engaged in the sort of name calling that characterizes the stories representing the government and the guerrillas. Stories representing the Catholic Church tended to avoid blaming anyone for the deaths of its members. The Democratic Convergence also tended to not to cast anyone in the role of criminal when discussing the death of one of its members. Instead, these stories can be recognized by, for example, the use of the Catholic Church as the *source* of a story and the assignment of one or more members of the Catholic Church as *victim*(s).

One might think that all points of view are simply determined by the source of the story. Although the source of a story is important it is not definitive of the point of view because, for example, more than one source can be represented in a story and the "top-level source", i.e., the broadcaster or publisher of the story, is not necessarily in agreement with the point of view represented in the story (e.g., when the Catholic Church broadcasts its point of view on the Salvadoran government radio station). Also, although the source is a powerful way of categorizing radio, TV, or newspaper stories, it is less effective in the electronic environments for which SpinDoctor is being developed. For example, in looking through news stories posted to electronic bulletin boards or sent as email, potentially millions of individuals could be a source and so the recipient

of a story may be familiar with the point of view expressed, but not necessarily with the source who sent the story.

2.3.1 Related Work in Artificial Intelligence and Cognitive Science

During the Gulf War, the linguist George Lakoff wrote an essay on the metaphors used by journalists to justify the war (Lakoff, 1991). In the essay he points out how the war has been cast by the American media in a fairy tale-like format where the US plays the role of hero, Iraq the role of villain, and Kuwait is the maiden in distress. Actor-role analysis can be seen as an attempt to create a computational model of and elaborate the insights sketch out by Lakoff in his short essay.

Actor-role analysis can also be seen as a generalization of some early AI work on the representation of ideology. For example, Carbonell's work (Carbonell, 1978) attempted to represent ideologies as pairings of actors with goal trees. Thus, a "conservative" American ideology was represented as a description of a set of plans and goals that the actor Russia could take to achieve domination of the world, and a set of plans and goals that the United States could take to prevent Russia from doing so. Contrasting ideologies assigned different actors different goal trees. Goals and plans (i.e., goal trees) can be seen as a limited form of the roles used in SpinDoctor. However, to equate roles with goal trees would be tantamount to making Carbonell's mistake of equating ideology with teleology when, in fact, ideology can encompass more than just plans and goals. Although one might want to argue that goal trees are a good representation for everything, when the issues at stake are more ones of, for

example, description, rather than actions (e.g., when the Salvadoran government radio broadcasts call the FMLN terrorists), then something like SpinDoctor's role data structures are certainly more flexible and, thus, appropriate than Carbonell's goal trees. Carbonell's goal trees were incapable of representing subtle differences in style and description because his parser was built to coerce natural language text into Schankian conceptual dependency primitives which, by design, wiped out any interesting differences in wording in the original texts. To avoid this problem SpinDoctor has been designed to use multiple idiomatic phrases in its role definitions rather than some set of language "primitives."

Interestingly enough earlier work by one of Carbonell's teachers, Robert Abelson, did not confuse ideology and teleology and did not crucially depend up language primitives (e.g., Abelson and Carroll, 1965; i.e., before the beginning of Abelson's collaborations with Schank). However, while Carbonell's system, POLITICS, was capable of handling natural language texts, Abelson's earlier systems accepted only logical propositions. Nevertheless, Carbonell's system only handled a few small example texts while SpinDoctor has been applied to hundreds of stories that are each hundreds of words long.

3.0 Analysis Process

3.1 Analyze-story

Analyze-story is the top-level routine of the SpinDoctor system. Given a news story analyze-story processes it and returns the name of the point of view which is represented by the story.

The analysis process followed by analyze-story is as follows: (1) Given a news story analyze-story calls the subroutine *find-actor-role-bindings* to find which noun phrases (i.e., actors) play which roles in the story. (2) Then, it determines which actors are of a similar group or are identical to other actors. The subroutine which performs this task is called *coreference-actors*. Coreference-actors does both (a) anaphoric resolution (e.g., determining who or what is being referenced when pronouns like "he," "she," and "it" are used); and, (b) actor grouping (e.g., inferring that "an army spokesman," "Maj. Rodriguez," and "Gen. Bustillo" are all a part of the same group (e.g., the armed forces). (3) After coreferencing the actors into groups, analyze-story uses the subroutine *construct-high-level-weighted-actor-role-bindings* to construct a profile of the analyzed story which describes how often (within the given story) top-level actors were assigned different roles. (4) Using weighted actor-role bindings allows the final subroutine, *identify-pov*, to distinguish, for example, stories in which the government is cited as a source once and the guerrillas are cited ten times from stories in which the guerrillas are cited as a source less often than the government. Identify-pov matches the weighted actor-role bindings against the point of view definitions, determines which point(s) of view match(es) best and outputs the name of the one (or more) best-matching point(s) of view.

3.2 Find-actor-role-bindings

The purpose of the find-actor-role-bindings function is to determine which noun phrases (i.e., actors) that occur in a story fill variables in the role patterns matched to the same story. Thus, the input to find-actor-role-bindings is a news

story and the output is a set of actor-role bindings. Find-actor-role-bindings first calls the subroutine *find-noun-phrases* (to identify the noun phrases in the story) and then calls the subroutine *find-roles* (to match all of the defined role patterns against the text of the story).

3.2.1 Find-noun-phrases

Given the text of a story find-noun-phrases first employs the Xerox part-of-speech tagger (Cutting et al., 1993) to tag the words of the story. After tagging the text of the story, find-noun-phrases attempts to group together words into noun phrases. Thus, after processing, a sentence like "Bill went to the movies" would become (noun-phrase "Bill") "went to" (noun-phrase "the movies").

3.2.2 Find-roles

For each role pattern, find-roles outputs a list of bindings between the pattern and possible positions in the text of the story. The role variables, like ?victim and ?criminal, are assumed to be filled by noun phrases. Find-roles returns a range of positions within the text in which the actor that fills a role might occur.

Find-actor-role-bindings combines the output of find-roles with the output of find-noun-phrases to create extended actor-role bindings. Extended-bindings are a data structure with four slots:

- a *noun-phrase* slot which is filled by a list of words and their start and end positions in the text; this slot represents the actor;

- a *role-variable* slot which names the role against which the actor was matched by find-actor-role-bindings;
- a *coreferents* slot which, at this point in the analysis of the story is empty;
- an *actor-groups* slot which is also empty but which will later be filled by a list of groups (social, political, economic) that the actor might belong to; thus, for example, President Cristiani might be grouped together with other government officials.

3.3 Coreference-actors

The function coreference-actors starts with the output of find-actor-role-bindings and begins by matching a set of actor "name" patterns against the noun phrases identified as actors by find-actor-role-bindings. For example, Maj. Rodriguez can be categorized as member of the Army by noting that his title "Maj." is a rank that Army officers have. The lists in the "names" slot of the actor definitions are the patterns that are used to group actors.

Coreference-actors then applies a set of constraints in order to try to resolve pronoun references and in order to partition all of the actors into mutually exclusive sets. It thereby identifies that, for example, a reference to "Maj. Rodriguez", a reference to "Rodriguez", and an instance of "he" might all represent the same actor and, furthermore, that references to "Rodriguez" and "General Bustillo" are both instances which represent the more inclusive actor named the "Army."

The following constraints are used by coreference-actors:

(1) the "member-of" relations which detail which groups of actors include other groups; e.g., "army-officers" is contained in the more inclusive actor group "army;"

(2) the positions of the actors in the text of a story; the control structure of coreference-actors has been designed to apply this constraint by first searching through all of the other actors which are positioned above the current actor (i.e., towards the beginning of the story) in its search for a coreferent for the current actor before it searches through all of the actors positioned below the current actor;

(3) the role(s) assigned to an actor; e.g., if "Maj. Rodriguez" is determined to play the role of ?source in an article, then it is expected that other noun phrases representing unknown actors (e.g., "the spokesman" or the "interviewee") which are also identified as ?sources would be more likely to be matched to "Maj. Rodriguez" than unknown actors which play other roles (e.g., the role of ?victim).

(4) the pronouns associated with certain known actors; e.g., president-cristiani is defined in the database of SpinDoctor to have a list of appropriate pronouns which include "he" and "him" but not "she" or "it;"

(5) the words in the noun phrases which constitute the actors; e.g., "Maj. Rene Rodriguez Hurtado" matches the actor "Maj. Rene Rodriguez" better than it matches "Gen. Bustillo."

Coreference-actors has been written to use all of the above constraints to coreferences actors. However, if a coreferent for an actor cannot be found, then the constraints are dropped one-by-one until only a subset of the constraints

remain and either (a) a coreferent is found, or (b) no coreferent is found even with the smallest allowable set of constraints.

Coreference-actors applies the constraints in a slightly different manner depending upon the form of the current actor. If the current actor is a pronoun, constraint (5) is not applied, e.g., an earlier occurrence of "he" is considered a poor coreferent for the pronoun "he" even though the words of the noun phrases match exactly. For both pronouns and non-pronouns, non-pronoun coreferents are sought. A further specialization exists for the pronouns "I" and "we." "I" and "we" are assumed to refer to actors which fill the role of ?source. Consequently, in the case of "I" and "we" pronouns constraint (3) is dropped in favor of constraint (3a) which favors coreferents which fill the role of ?source. Constraint (3a) is the role of the actor is assumed to be ?source.

The program uses the following "derefinement operators" to slowly relax the constraints if no coreferent can be found for an actor. The derefinement operators are lists of conjuncts of constraints. If the first conjunct of constraints in a list cannot be fulfilled, then the coreference-actors picks the next conjunct in the list and tries again, until the end of the list. Here, the conjuncts are represented by lists of numbers which refer to the numbered constraints given above:

- derefinement operator for non-pronouns:
  ((3 1 5) (1 5) (3 5) (3 1) (5) (1) (3))
- derefinement operator for "I" and "we" pronouns:
  ((3a 1) (3a) (1))
- derefinement operator for other pronouns:
  ((3 1) (3) (1))

Constraints 2 and 4 are not listed as they are members of all of the conjuncts of constraints shown. The procedure coreference-actors propagates the relationships that it finds between actors and so, in the end, actors may be found to have several groups and several coreferents. The effect of coreference-actors is to fill in the "coreferents" and "actor-groups" slots of the extended-binding structures output by the find-actor-role-bindings routine.

3.4 Construct-high-level-weighted-actor-role-pairs

Given the extended-binding structures output by coreference-actors, the routine construct-high-level-weighted-actor-role-pairs further abstracts the actors into their most inclusive actor groups and then counts the number of times actors from each of the top-level actor-groups play each role matched in the text of the story under analysis.

Each list of groups in an extended-binding structure is a list of actors. So, for example, a binding which contains the noun phrase "President Cristiani" in its actor slot might also contain the list of actors (president-cristiani government-official government) in its actor-groups slot. Construct-high-level-weighted-actor-role-pairs finds the most inclusive group in each of the actor-groups slots; it uses the "member-of" declarations in the actor definitions to determine the most inclusive group. In this example, president-cristiani is a member-of government-officials which is a member-of the government and so government is the most inclusive group in this case. Right now the system does not support multiple member-of declarations for an actor. Obviously this would be a worthwhile and interesting extension.

After construct-high-level-weighted-actor-role-pairs has determined the most inclusive group for each binding it then goes through and counts the number of times each most inclusive actor group is bound to a particular role. For instance, if "President Cristiani" appeared in the role of ?source once, then construct-high-level-weighted-actor-role-pairs would create a weighted actor-role pair like this (assuming that no other government-official appeared as a ?source in the story): (government ?source 1).

The output of construct-high-level-weighted-actor-role-pairs is a list of weighted actor-role pairs.

3.5 Identify-point-of-view

The routine identify-point-of-view performs the final stage of analysis. Given a list of weighted actor-role pairs, identify-point-of-view compares them with each of the points of view defined in the database to determine which point of view matches best to the actor-role pairs found in the story. If one of the actor-role pairs matches an actor-role pair in the point of view (see Figure 2 for a list of point of view definitions), then the weight associated with the pair is added to the score for the point of view. The point of view which has the largest score -- after the weighted actor-role pairs have been matched against all of the points of view -- is output as the point of view which matches the story. If two or more points of view have the same score then they are all output. Thus, at its most discriminating identify-point-of-view returns one point of view, while at its least discriminating, i.e., when it cannot determine the point of view for the story,

identify-point-of-view outputs the list of all of the points of view defined in the database.

4.0 Performance Test

4.1 Method

To build up the actor, role, and point of view representations the first twenty-five stories about El Salvador in the MUC3 corpus were scrutinized. Characteristic textual patterns were noted and then coded into the actor and role definitions. The point of view definitions were then defined by trying to articulate the strongest ideological differences that seemed to be represented in the stories (e.g., that the guerrillas call the government criminals and vice versa).

4.2 Results

After coding the databases using the first twenty-five stories about El Salvador, SpinDoctor was then run on the first one hundred stories in the MUC3 corpus. Tables 1, 2, 3, and 4 summarize the results of that test. The columns labeled *right* contain the number of times in which SpinDoctor was able to pick which one of the four points of view was represented by a given story. The columns labeled *wrong* contain the number of times in which SpinDoctor picked a subset of the points of view (often a subset of one point of view) for the story and none of the elements of the subset were actually represented in the story. This was the case, for example, when the story was written from the point of view of the El Salvadoran government and SpinDoctor incorrectly deduced that the story either represented the Catholic Church or the Democratic Convergence. The columns labeled *vague* count the number of times in which SpinDoctor selected a superset of the points of view actually represented in the story. In such a case, SpinDoctor was not actually wrong, the stories did indeed represent a subset of the points of view selected by SpinDoctor. However, this was the case when, for example, SpinDoctor reported that a story represented both the Catholic Church and the government when, indeed, perhaps it only represented the government's points of view.

Table 1: The Training Set: Twenty-five stories about El Salvador

	right	wrong	vague	totals
number of stories	19	2	4	25
percentage	76	8	16	100

Table 2: Test Set #1: Seventeen unseen stories about El Salvador

	right	wrong	vague	totals
number of stories	6	1	10	17
percentage	35	6	59	100

Table 3: Test Set #2: Fifty-eight unseen stories about other countries

	right	wrong	vague	totals
number of stories	17	0	41	58
percentage	29	0	71	100

Table 4: Totals: One hundred stories seen and unseen

	right	wrong	vague	totals
number of stories	42	3	55	100
percentage	42	3	55	100

As one can see in the Table 1, unsurprisingly, SpinDoctor did best with the subcorpus of stories that were used to create the databases (i.e., the first twenty-five stories about El Salvador). What is more interesting to see is Table 2 which shows how well SpinDoctor did on stories about El Salvador which were not a part of the twenty-five used to create the databases of actors, roles, and points of view. This test result indicates that SpinDoctor could provide a user with a, at least partially effective, news filter to detect point of view. It is important to note that SpinDoctor was often overly vague, but rarely completely wrong in its assessment of point of view.

Perhaps what is the most surprising of the test results is that SpinDoctor behaved reasonable well with news stories which were neither from nor about El Salvador, but were rather descriptive of events in other countries in Central and South America. To a certain degree many of the other countries represented in the MUC3 corpus had, like El Salvador, serious internal, civil disputes. Consequently, some of the parties in these disputes are called "the government" and other parties are called "the guerrillas." These structural similarities in the news discourses about various civil disputes in different countries allowed SpinDoctor to sometimes correctly detect the point of view represented by a news story about something that had no direct relation to the politics of El Salvador.

Table 4 shows the totals for all 100 hundred stories analyzed: 42 right, 3 wrong, and 55 overly vague.

5.0 Conclusions and Future Work

In this paper we have shown how ideological point of view can be represented by sets of actor-role pairs. Differing ideological points of view are represented by different actor-role pairs (e.g., Oliver North as patriot versus North as criminal). The computer program, *SpinDoctor*, can recognize the point of view represented by a news story by matching actor and role patterns to the text of the story and then by comparing the actor-role pairings matched in the story with a set of predefined ideological points of view. The *actor-role analysis* method employed by SpinDoctor was compared to earlier AI work on the representation of ideology. SpinDoctor was tested on a portion of the MUC3

corpus and the results of that test showed that SpinDoctor could be used as a news filter to detect point of view.

In future work we will expand SpinDoctor's actor, role, and point of view databases and combine its abilities to recognize point of view with conventional techniques from information retrieval and NLP (e.g., those techniques employed in systems built for the MUC competitions) to create a news filter capable of examining stories according to both events described and the point of view from which the events are described.

References

- Abelson, R.P. and Carroll, J.D. (1965) "Computer Simulation of Individual Belief Systems", *American Behavior Scientist*, 8, 24-30.
- Carbonell, J. (1979) *Subjective Understanding: Computer Models of Belief Systems*. Ph.D. Thesis. New Haven, CT: Yale University Computer Science Department, Technical Report 150.
- Cutting, Kupiec, Pedersen, and Sibun (1993) A Practical Part-of-Speech Tagger,*Proceedings of the Annual Conference of the Association for Computational Linguistics*,
- Jacobs, P. and Rau, L. (1993) Innovations in Text Interpretation, *Artificial Intelligence* , 63(1-2), 143-192.
- Jameson, F. (1987) Foreword, in Greimas, A.J., *On Meaning: Selected Writings in Semiotic Theory* (trans.P.J. Perron & F.H. Collins) (Minneapolis: University of Minnesota Press).

- Lakoff, G. (1991) "Metaphor and War: The Metaphor System Used to Justify the War in the Gulf", *Journal of Urban and Cultural Studies*, Volume 2, Number 1, pp. 59-72.
- Latour, B. (1987) *Science in Action: How to follow scientists and engineers through society* Cambridge, MA: Harvard University Press.
- MUC-3 (1991) *Proceedings of the Third Message Understanding Conference*, San Mateo, CA: Morgan Kaufmann Publishers, Inc.
- Sack, W. (1994b) Indexing Multimedia by Ideology, in *Proceedings of the AAAI Workshop on Indexing and Reuse of Multimedia*, Seattle, WA.
- Sack, W. (1994b) *Actor-Role Analysis: Ideology, Point of View and the News* (Technical Report 94-005) Cambridge, MA: MIT Media Laboratory, Learning and Common Sense Section.
- Wiebe, J. (1994) Tracking Point of View in Narrative, *Journal of Computational Linguistics*, June 1994