Computational Intelligence

AN INTERNATIONAL JOURNAL

Computational Intelligence, Volume 00, Number 0, 2012

A COMPUTATIONAL MODEL FOR PLOT UNITS

AMIT GOYAL,[1] ELLEN RILOFF,[2] AND HAL DAUMÉ III[1]

[1] *Department of Computer Science, University of Maryland, College Park, Maryland, USA*
[2] *School of Computing, University of Utah, Salt Lake City, Utah, USA*

This research revisits *plot units*, which were developed in the 1980s as a conceptual knowledge structure to represent the affect states of and emotional tensions between characters in narrative stories. We present a fully automated system, called AESOP, that generates plot unit representations for narrative texts. AESOP performs four steps: affect state recognition, character identification, affect state projection, and link creation. We also identify a type of knowledge that seems to be missing from existing lexical resources: verbs that impart positive or negative polarity onto their patients (e.g., “eat” imparts negative polarity because being eaten is bad, whereas “fed” imparts positive polarity because being fed is good). We develop two techniques to automatically harvest these “patient polarity verbs” (PPVs) from a Web corpus, and show that the PPVs improve affect state recognition. Finally, we evaluate AESOP’s performance on a set of fables, and present several analyses to shed light on the capabilities and limitations of current natural language processing technology for plot unit generation.

Received 1 August 2010; Revised 17 February 2011; Accepted 17 February 2011

Key words: plot units, affect state analysis, emotion polarity, conceptual knowledge structures.

1. INTRODUCTION

Our research revisits Lehnert’s *plot units* (Lehnert 1981), which were proposed in the 1980s as a conceptual knowledge structure to explicitly represent the affect states of and emotional tensions between characters in narrative stories. Lehnert argued that affect states and the relations between them were central to story understanding and summarization. Plot units were demonstrated to be useful for narrative text summarization in both computer science and psychology studies (e.g., Lehnert 1981; Lehnert, Black, and Reiser 1981; Reiser, Lehnert, and Black 1981; Lehnert, Alker, and Schneider 1983), but the early computational models of plot units relied on large amounts of manual knowledge engineering.

In recent years, the field of natural language processing (NLP) has seen tremendous growth and interest in the computational analysis of emotions, sentiments, and opinions. Much of this work has focused on application areas, such as sentiment analysis of consumer reviews (e.g., Morinaga et al. 2002; Pang, Lee, and Vaithyanathan 2002; Dave, Lawrence, and Pennock 2003; Nasukawa and Yi 2003; Turney and Littman 2003), product reputation analysis (e.g., Morinaga et al. 2002; Nasukawa and Yi 2003; Yi et al. 2003), spam filtering (e.g., Spertus 1997), and tracking sentiments toward events (e.g., Das and Chen 2001; Tong 2001). Our work aims to delve into more complex meaning representations related to emotions and affect for narrative text understanding.

This paper presents a new study and computational model for plot unit analysis that explores whether state-of-the-art NLP tools and resources can be effectively harnessed to produce plot unit representations. Our research includes three contributions: (1) we conduct a manual annotation study to shed light on the types of knowledge that are necessary to identify the affect states in plot units, (2) we introduce a system called AESOP that produces plot unit representations by exploiting existing lexical resources combined with rules that assign affect states to characters and create links between them, and (3) we present two

Address correspondence to Amit Goyal, Department of Computer Science, University of Maryland, College Park, Maryland, USA; e-mail: amit@umiacs.umd.edu

techniques to automatically acquire verbs that impart polarity on their patients, and show that these verbs improve affect state recognition.

We begin by presenting the results of a manual annotation study on a small set of AESOP's fables. We categorize each affect state based on the type of world knowledge that is needed to recognize it, and discover that most affect states arise from emotions related to the successful completion or failure of a plan or goal. We also analyze the distribution of primitive plot unit structures in the test set, and find that successes, failures, and problems are the most common configurations of affect states.

Next, we present AESOP, a system that automatically produces plot unit representations. AESOP decomposes the task into four steps: *affect state recognition*, *character identification*, *affect state projection*, and *link creation*. For affect state recognition, AESOP exploits a variety of sentiment-related and general-purpose language resources to identify positive, negative, and mental affect states. Each affect state is then assigned to a character using *projection rules* that exploit verb argument structure. Additional affect states are also produced as inferences based on syntactic structure. Finally, AESOP connects the affect states with causal and cross-character links to produce full plot unit structures.

During the course of this research, we realized that many affect states arise from an event in which a character is acted upon in a positive or negative way. For example, "*the cat ate the mouse*" produces a positive affect state for the cat and a negative affect state for the mouse because obtaining food is good but being eaten is bad. This type of world knowledge is not readily available in existing lexical resources. To fill this gap, we developed two methods to automatically identify verbs that impart a positive or negative polarity on their patients (*patient polarity verbs*, or PPVs) from a Web corpus. One method uses pattern-based extractions to identify verbs that frequently co-occur with stereotypically evil or kind agents. The second method uses a bootstrapping algorithm to iteratively acquire PPVs from verb phrase (VP) conjunctions.

We evaluate the plot unit structures produced by AESOP on a small set of two-character fables, and present a detailed analysis of the impact of each lexical resource and subcomponent of our model. We also discuss the limitations of current NLP technology with respect to plot unit analysis, and suggest possible directions for future research. This paper expands upon earlier papers on this work (Goyal et al. 2010a,b) with a new manual annotation study of affect state origin classes and primitive plot units in our data set as well as new experimental results and discussion that analyze AESOP's performance on each class of affect states. Data sets, annotations, and PPV lexicons are freely available and can be downloaded from http://www.umiacs.umd.edu/∼amit/AESOP.html.

2. OVERVIEW OF PLOT UNITS

The motivation for plot units was the belief that "emotional reactions and states of affect are central to the notion of a plot or story structure" (Lehnert 1981). In contrast to top−down representations, such as story grammars (e.g., Rumelhart 1975; Thorndyke 1997), plot unit structures are built in a bottom−up fashion. The lowest level components are *affect states* that "emphasize emotional reactions to events and states" (Lehnert 1983). Plot units use three types of affect states: positive (+), negative (−), and mental (M) states, which represent mental events of null or neutral emotionality. Each affect state is attributed to a character in the story. Although affect states are *not* events per se, affect states often arise from events. If an event affects multiple characters, it can produce an affect state for each character.

Different configurations of affect states form *primitive plot unit structures*, which consist of two affect states associated with the same character and a *causal link* between them. Plot

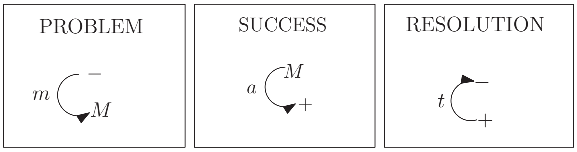
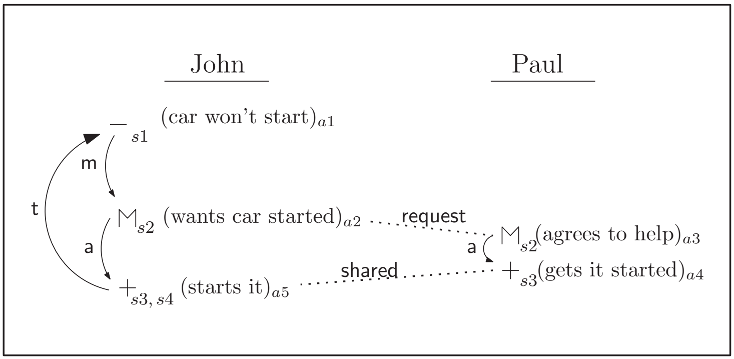


FIGURE 1. Three primitive plot unit structures.

John and Paul
(s1) When John tried to start his car this morning, it wouldn't turn over. (s2) He asked his neighbor Paul for help. (s3) Paul did something to the carburetor and got it going. (s4) John thanked Paul and drove to work.

(a) "John and Paul" Story


John
Paul
$-_{s1}$ (car won't start)$_{a1}$
m
t
M_{s2} (wants car started)$_{a2}$
request
a
a
M_{s2}(agrees to help)$_{a3}$
shared
$+_{s3}$(gets it started)$_{a4}$
$+_{s3,\, s4}$ (starts it)$_{a5}$

(b) Plot Unit Representation for "John and Paul" Story

FIGURE 2. A short story and its complete plot unit representation from Lehnert's paper (Lehnert (1981)).

units can include four types of causal links: motivations (m), actualizations (a), terminations (t), and equivalences (e). Figure 1 shows three examples of primitive plot unit structures. The PROBLEM structure depicts a character who was in a negative state that motivated a mental state (e.g., "*John lost his job so he decided to rob a bank*"). The SUCCESS structure depicts a character who was in a mental state that led to a positive state (e.g., "*Jill proposed to George and he accepted*"). The RESOLUTION structure depicts a character who was in a negative state that eventually resolved to become a positive state (e.g., "*Lee was fired but soon got a new job*"). In total, there are 15 possible primitive plot unit configurations (Lehnert 1981).

Plot units also include *cross-character links* that connect affect states for two characters when a single event affects them both. The complete plot unit representation of a story is a graph structure that typically contains many primitive plot unit structures, often overlapping, and with connections between them. The plot unit graph represents all of the affect states of each character in a story, chronologically, as the plot unfolds.

Figure 2(a) shows a short story from Lehnert's paper (Lehnert 1981), and Figure 2(b) shows its complete plot unit representation. The story has two characters, John and Paul, who experience a series of affect states, depicted chronologically in the two columns. The first affect state ($a1$) is a negative state for John because his car will not start (from sentence $s1$). Although $s1$ does not explicitly say that John is unhappy, anyone reading this story would assume that he is in a negative state based on world knowledge that it is undesirable to have a car that won't start. The second sentence produces one mental state for John ($a2$) and one mental state for Paul ($a3$), representing John's request to Paul for help. The two M states are connected with a cross-character link because the "asked" event produces mental states for both characters (John initiated the request and Paul received it). John's negative state ($a1$) motivated John's mental state ($a2$), thus an m causal link connects these states. Together, states $a1$, $a2$, and the m link form the PROBLEM primitive plot unit structure.

Sentence $s3$ says that Paul fixed John's car, which is represented by two positive affect states, one for Paul ($a4$) and one for John ($a5$), because both characters were presumably happy with the outcome. Because this single event affected both characters, a cross-character link connects the two + states. Each of the M states is also connected to its respective + state with an *actualization* (a) link, indicating that the positive state was a successful realization (completion) of the plan represented by the mental state. This configuration is the SUCCESS primitive plot unit structure. Figure 2(b) includes one SUCCESS structure for John and a corresponding SUCCESS structure for Paul.

Finally, the plot unit graph also has a *termination* (t) link pointing from John's positive state to his negative state, indicating that the negative state has been resolved and he is now in a positive state. This configuration of states is the RESOLUTION primitive plot unit structure.

Sentence $s4$ represents a speech act in which John thanks Paul for his help. However, this sentence does not contribute any additional information to the plot unit representation. The "thank" event provides additional evidence that John is in a positive state, but if sentence $s4$ was omitted from the story the plot unit representation would be the same. This example illustrates that stories often contain redundancy, i.e., multiple clues that can be used to reach the same conclusion.

We present one more example of a plot unit representation using a fable from our development set. The "The Father and Sons" fable appears in Figure 3(a) and our annotation of its plot unit structure is shown in Figure 3(b). This fable focuses on two characters, the "Father" and the "Sons"; the "Sons" are referred to collectively throughout the story so we consider them to be a single (group) entity.

The first affect state ($a1$) is a negative state for the sons because they are quarreling. This state is *shared* by the father (via a cross-character link) who has a negative annoyance state ($a2$). The father decides that he wants to stop the sons from quarreling, which is a mental event ($a3$). The causal link from $a2$ to $a3$ with an m label indicates that his annoyed state "motivated" this decision. His first attempt is by exhortations ($a4$). This produces an M ($a3$) linked to an M ($a4$) with an m (motivation) link, which represents subgoaling. The father's overall goal is to stop the quarreling ($a3$) and to do so, he creates a subgoal of exhorting the sons to stop ($a4$). The exhortations fail, which produces a negative state ($a5$) for the father. The a causal link indicates an "actualization," representing the failure of his plan ($a4$).

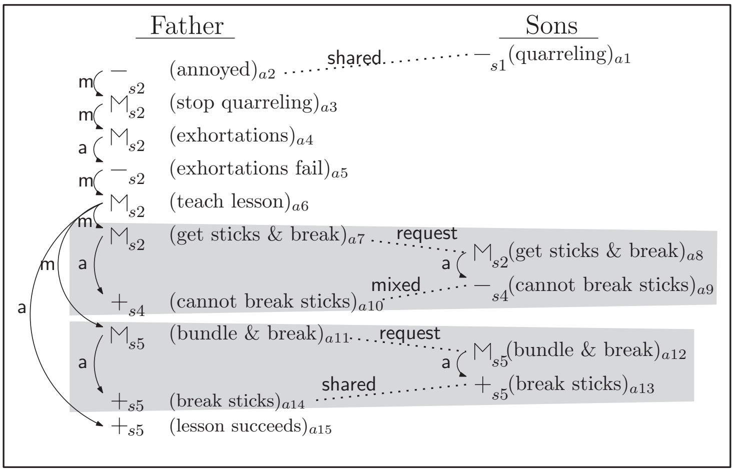
This failure motivates a new subgoal: teach the sons a lesson ($a6$). The m link from $a5$ to $a6$ creates a "problem" primitive plot unit structure. At a high level, this subgoal has two parts, indicated by the two gray regions ($a7$–$a10$ and $a11$–$a14$). The first gray region begins with a cross-character link (M to M), which indicates a request (in this case, to break a bundle of sticks). The sons fail at this, which upsets them ($a9$) but pleases the father ($a10$). The second gray region depicts the second part of the father's subgoal; he makes a second request ($a11$–$a12$) to separate the bundle and break the sticks, which the sons successfully do, making them happy ($a13$) and the father happy ($a14$) as well. This latter structure (the second gray region) is an HONORED REQUEST plot unit structure. At the end, the father's plan succeeds ($a15$) which is an actualization (a link) of his goal to teach the sons a lesson ($a6$).

3. A MANUAL ANALYSIS OF AFFECT STATES AND PLOT UNITS

We began this research with the hope that recent work in emotion and sentiment analysis would supply us with effective tools to recognize the affect states required for plot units.

The Father and His Sons
(s1) A father had a family of sons who were perpetually quarreling among themselves. (s2) When he failed to heal their disputes by his exhortations, he determined to give them a practical illustration of the evils of disunion; and for this purpose he one day told them to bring him a bundle of sticks. (s3) When they had done so, he placed the faggot into the hands of each of them in succession, and ordered them to break it in pieces. (s4) They tried with all their strength, and were not able to do it. (s5) He next opened the faggot, took the sticks separately, one by one, and again put them into his sons' hands, upon which they broke them easily. (s6) He then addressed them in these words: "My sons, if you are of one mind, and unite to assist each other, you will be as this faggot, uninjured by all the attempts of your enemies; but if you are divided among yourselves, you will be broken as easily as these sticks."

(a) "Father and Sons" Fable



(b) Plot Unit Representation for "Father and Sons" Fable

FIGURE 3. A fable and its complete plot unit representation from our development set.

However, we quickly realized that the affect states in plot units can arise from a wide variety of language phenomena and many types of knowledge. Consequently, we embarked on a manual annotation study to catalog the types of affect states and plot unit structures that occur in our data set, and the kinds of knowledge that they originate from.

3.1. Data Set

Plot unit analysis of narrative text is enormously complex, thus we began with relatively short texts that seemed appropriate for this task: fables. Fables have two desirable attributes: (1) they have a small cast of characters, and (2) they typically revolve around a moral, which is exemplified by a concise plot. Even so, fables are challenging for NLP due to anthropomorphic characters, flowery language, and sometimes archaic vocabulary. We collected 34 of AESOP's fables from a Web site,[1] choosing fables that have a true plot (some only contain quotes) and exactly two characters. We divided them into a development set of 11 stories, a tuning set of 8 stories, and a test set of 15 stories. The 15 stories in our test set contain 85 sentences and 242 individual clauses.

[1] www.pacificnet.net/∼ johnr/aesop/

Creating a gold standard is a substantial undertaking, and training nonexperts to produce them did not seem feasible in the near term. Thus, the authors iteratively refined manual annotations on the development and tuning texts until we produced similar results and had a common understanding of the task. Then, two authors independently created annotations for the test set, and a third author adjudicated the differences. The gold standard contains complete plot unit annotations, including affect states, causal links, and cross-character links. Each affect state was also annotated with provenance indicating which clause (or set of clauses) it originated from.

3.2. Identifying the Origin of Affect States

A goal of this work was to gain a better understanding of where affect states come from in terms of language and world knowledge. Thus, we examined the fables in our development and tuning sets and defined six categories that seemed to cover most of the affect states that we found. We will refer to these six categories as *affect origin classes*. Three of these classes (E, S, and PG-C) correspond to positive and negative affect states, and three of them (PG-D, PG-S, and PG-I) correspond to mental affect states. Note that four of the six categories represent affect states that arise from the plans and goals of characters.

(E) *Direct Expressions of Emotion*: this category covers +/− affect states that arise from expressions that explicitly represent an emotional state. For example, "*Max was disappointed*" produces a negative affect state for Max, and "*Max was pleased*" produces a positive affect state for Max.

(S) *Situational Affect States*: this category covers +/− affect states that represent good or bad situations that characters find themselves in. World knowledge suggests that the character is in a positive or negative emotional state because of their situation. For example, "*Wolf, who had a bone stuck in his throat, . . .*" produces a negative affect state for the wolf because the wolf is in an undesirable situation. Similarly, "*The Old Woman recovered her sight . . .*" produces a positive affect state for the old woman because she is presumably happy about this situation.

(PG-D) *Direct Expressions of Plan/Goal*: this category covers M affect states that arise from a plan or goal that is explicitly stated. For example, "*the lion wanted to find food*" would generate a mental affect state.

(PG-S) *Speech Acts*: this category covers M affect states that come from a speech act between characters. For example, "*the wolf asked an eagle to extract the bone*" is a directive speech act that reveals the wolf's plan to resolve its negative state (having a bone stuck in its throat). Commitments and refusals also fall into this category.

(PG-I) *Inferred Plans/Goals*: this category accounts for M affect states that arise from plans or goals that are inferred from an action. For example, when reading "*the lion hunted deer*," most people would infer that lion's goal is to obtain food, which is represented by a mental state. Similarly, most people would assume that "*the serpent spat poison into the man's water*" indicates that the serpent has a plan to kill the man.

(PG-C) *Plan/Goal Completion*: this category includes +/− affect states that represent the completion (successful or failed) of a plan or goal. For example, if an eagle extracts a bone from a wolf's throat, then both the wolf and the eagle will have positive affect states because both were successful in their respective goals. Similarly, if a man tries to kill a mouse but the mouse escapes, then the failure of his plan is represented with a negative affect state.

TABLE 1. Distribution of Affect Origin Categories; Parentheses Show Number of States.

	+/− Affect States (84)			*M Affect States (59)*		
Category	E	S	PG-C	PG-D	PG-S	PG-I
Frequency	3	30	51	9	31	19
Overall %	0.02	0.21	0.36	0.06	0.22	0.13
Normalized %	0.04	0.36	0.61	0.15	0.53	0.32

We manually categorized each affect state in our test set to indicate its origin based on the source text. Table 1 shows the distribution of the affect origin classes in the test set. The Frequency row shows the number of affect states in each category, the Overall % row shows the distribution of each category across all affect states, and the Normalized % row shows the distribution of each category across the affect states of the same type (+/− or M). We can make several observations:

- By far the most common category was PG-C, which comprised 36% of all affect states and 61% of the +/− affect states.
- Seventy-seven percent of the affect states are related to plans and goals, although only 6% were explicitly expressed as such in the text.
- Only 4% of the +/− affect states come from direct expressions of emotion. As we will see in Section 5.1, this analysis probably explains why sentiment analysis resources did not help with affect state recognition as much as we anticipated.

We conclude that recognizing the plans and goals of characters is essential for affect state recognition, and that knowledge about the goodness/badness of events with respect to characters is also an important part of the problem.

3.3. Analysis of Primitive Plot Units

We also used the manual annotations to examine the relative frequency of different primitive plot unit structures in our test set. Figure 4 visually depicts all 15 legal primitive plot unit configurations. The first row shows all legal structures that can have a or m causal links, the second row depicts all possible structures with t causal links, and the third row shows all possible structures with e links. Each cell contains two boxes showing statistics from our test set: the left box shows its absolute frequency and the right box shows the relative frequency of the plot unit across the test set. We can make several observations from Figure 4:

- The SUCCESS plot unit was the most common, occurring 37 times and making up 39% of all instances of primitive plot units.
- Fifty-four percent and 27% of structures are, respectively, generated by forward actualization (a) and motivation (m) links.
- Only 19% of the links are the backward termination (t) and equivalence (e) links.[2]

These results show that just five of the 15 primitive plot unit structures (the first row), using just two types of causal links (a and m), account for 81% of all primitive plot units

[2] The zeros in the figure mean that there were no such structures found in our test set.

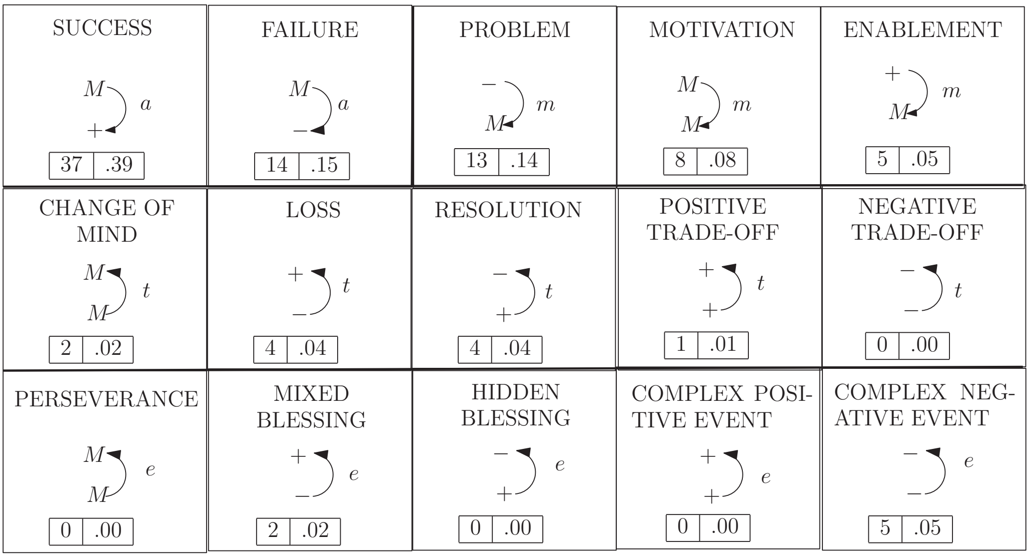


FIGURE 4. Distribution of primitive plot unit structures.

in our test set. Furthermore, a and m links often connect affect states that are close by (chronologically), but the t and e links are more challenging to generate because they tend to connect states that are far apart. Consequently, this study focuses exclusively on generating a and m causal links. We leave the generation of the backward t and e links for future work.

4. AESOP: AUTOMATICALLY GENERATING PLOT UNIT REPRESENTATIONS

Our system, AESOP, automatically creates plot unit representations for narrative text. AESOP decomposes the process into four steps: affect state recognition, character identification, affect state projection, and link creation. During affect state recognition, AESOP identifies words that may be associated with positive, negative, and mental states. AESOP then identifies the main characters in the story and applies *affect projection rules* to map the affect states onto these characters. During this process, some affect states are inferred based on verb argument structure. Finally, AESOP creates cross-character links to connect affect states shared by different characters and causal links to connect affect states associated with a single character. We also present two corpus-based methods to automatically produce a new resource for affect state recognition: a *PPV lexicon*.

4.1. Plot Unit Creation

In this section, we describe the four-step process that AESOP uses to create plot unit representations: affect state recognition, character identification, affect state projection, and link creation.

4.1.1. Affect State Recognition. The basic building blocks of plot units are *affect states*. In recent years, many publicly available resources have been created for sentiment analysis and other types of semantic knowledge. We considered a wide variety of resources and ultimately decided to experiment with five resources that most closely matched our needs:

- FrameNet (Baker, Fillmore, and Lowe 1998): We manually identified 87 frame classes that seem to be associated with affect: 43 mental classes (e.g., COMMUNICATION and NEEDING), 22 positive classes (e.g., ACCOMPLISHMENT and SUPPORTING), and 22 negative classes (e.g., CAUSE HARM and PROHIBITING). We use the verbs listed for these classes to produce M, +, and − affect states. These frame classes yielded 343 M verbs, 117 + verbs, and 287 − verbs.
- MPQA Lexicon[3] (Wilson, Wiebe, and Hoffmann 2005a): We used the words listed as having positive or negative polarity to produce +/− states, when they occur with the designated part-of-speech. This lexicon contains 380 + verbs and 866 − verbs.
- OpinionFinder[4] (Wilson et al. 2005b) (Version 1.4): We used the +/− labels assigned by its contextual polarity classifier (Wilson et al. 2005a) to create +/− states and the MPQASD tags produced by its Direct Subjective and Speech Event Identifier (Choi, Breck, and Cardie 2006) to produce mental (M) states.
- Semantic Orientation Lexicon[5] (Takamura, Inui, and Okumura 2005): We used the words listed as having positive or negative polarity to produce +/− affect states, when they occur with the designated part-of-speech. This lexicon contains 229 + verbs and 286 − verbs.
- Speech Act Verbs: We used a list of 228 speech act verbs from Wierzbicka (1987) to produce M states.

4.1.2. Character Identification. For the purposes of this work, we made two simplifying assumptions:

(1) There are only two characters per fable.[6]
(2) Both characters are mentioned in the fable's title.

The problem of coreference resolution for fables is somewhat different than for other genres, primarily because the characters are often animals (e.g., *he* = *owl*). Thus, we hand-crafted a simple rule-based coreference system. First, we apply heuristics to determine number and gender based on word lists, WordNet (Miller 1990), and part-of-speech tags. If no determination of a character's gender or number can be made, we employ a process of elimination. Given the two character assumption, if one character is known to be male, but there are female pronouns in the fable, then the other character is assumed to be female. The same is done for number agreement.

Finally, if there is only one character between a pronoun and the beginning of a document, then we resolve the pronoun with that character and the character assumes the gender and number of the pronoun. Finally, WordNet is used to obtain a small set of nonpronominal, nonstring-match resolutions by exploiting hypernym relations, for instance, linking *peasant* with *man*.

4.1.3. Affect State Projection. Plot unit representations are not just a set of affect states, but they are structures that capture the chronological ordering of states for each character as the narrative progresses. Consequently, every affect state needs to be attributed to a character, or discarded. Because most plots revolve around events, we use verb argument structure as the primary means for projecting affect states onto characters.

[3] www.cs.pitt.edu/mpqa/lexiconrelease/collectinfo1.html

[4] www.cs.pitt.edu/mpqa/opinionfinderrelease/

[5] www.lr.pi.titech.ac.jp/∼takamura/pndic_en.html

[6] We only selected fables that had two main characters.

We developed four *affect projection rules* that orchestrate how affect states are assigned to the characters based on verb argument structure. We used the Sundance parser (Riloff and Phillips 2004) to produce a shallow parse of each sentence, which includes syntactic chunking, clause segmentation, and active/passive voice recognition. We normalized the VPs with respect to active/passive voice (i.e., we transform the passive voice constructions into an active voice equivalent) to simplify the rules. We made the assumption that the Subject of the VP is its AGENT and the Direct Object of the VP is its PATIENT.[7] The rules only project affect states onto AGENTS and PATIENTS that refer to a character in the story. The four projection rules are presented below:

Rule 1: AGENT **VP**: This rule applies when the VP has no PATIENT or the PATIENT corefers with the AGENT. All affect tags assigned to the VP are projected onto the AGENT. Example: "*Mary **laughed***(+)," projects a + affect state onto Mary.

Rule 2: **VP** PATIENT: This rule applies when the VP has no agent, which is common in passive voice constructions. All affect tags assigned to the VP are projected onto the PATIENT. Example: "*John was **rewarded***(+)," projects a + affect state onto John.

Rule 3: AGENT **VP** PATIENT: This rules applies when both an AGENT and PATIENT are present, do not corefer, and at least one of them is a character. If the PATIENT is a character, then all affect tags associated with the VP are projected onto the PATIENT. If the AGENT is a character and the VP has an M tag, then we also project an M tag onto the AGENT (representing a shared, cross-character mental state). Example: "*John **asked***(*M*) *Paul for help*," projects an M affect state onto both John and Paul.

Rule 4: AGENT **VERB1** to **VERB2** PATIENT: This rule has two cases: (a) If the AGENT and PATIENT refer to the same character, then we apply Rule #1. Example: "*Bo decided to teach himself...*" (b) If the AGENT and PATIENT are different, then we apply Rule #1 to **VERB1** and Rule #2 to **VERB2**.

Finally, if an adverb or adjectival phrase has affect, then that affect is mapped onto the preceding VP and the rules above are applied. For all of the rules, if a clause contains a negation word, then we flip the polarity of all words in that clause.

4.1.4. Inferring Affect States during Projection. Recognizing plans and goals depends on world knowledge and inference, and is beyond the scope of this paper. However, we identified two cases where affect states often can be inferred based on syntactic properties.

- The first case involves VPs that have both an AGENT and PATIENT, which corresponds to Projection Rule #3. If the VP has polarity, then Rule #3 assigns that polarity to the PATIENT, not the AGENT. For example, "*John killed Paul*" imparts negative polarity on Paul, but not necessarily on John. Unless we are told otherwise, one assumes that John *intentionally* killed Paul, and so in a sense, John accomplished his goal. Consequently, this action should produce a positive affect state for John. We capture this notion of accomplishment as a side effect of projection rule #3: if the VP has +/− polarity, then we produce an *inferred positive affect state* for the AGENT.
- The second case involves infinitive VPs of the form: "AGENT VERB1 TO VERB2 PATIENT" (e.g., "*Susan tried to warn Mary*"). The infinitive VP construction suggests that the AGENT has a goal or plan that is being put into motion (e.g., *tried to, wanted*

[7] Of course this is not always correct, but it works reasonably well, especially in our fables.

to, attempted to, hoped to, etc.). To capture this intuition, in rule #4 if VERB1 does not already have an affect state assigned to it then we produce an *inferred mental affect state* for the AGENT.

4.1.5. Creating Causal and Cross-Character Links. Plot unit structures include *cross-character links* to connect states that are shared across characters, and *causal links* to connect states for a single character. As an initial attempt to create complete plot units, AESOP produces links using a few simple heuristics. AESOP adds a cross-character link when two characters in a clause have affect states that originated from the same word. AESOP adds a causal link between each pair of chronologically consecutive affect states for the same character. Currently, AESOP only produces forward causal links (m and a) and does not produce backward causal links (e and t). For forward links, only five configurations produce legal plot unit structures: $M \xrightarrow{m} M$, $+ \xrightarrow{m} M$, $- \xrightarrow{m} M$, $M \xrightarrow{a} +$, $M \xrightarrow{a} -$. Thus, when AESOP adds a causal link, the types and ordering of the two affect states uniquely determine which label it should get (m or a).

4.2. Generating PPV Lexicons

Affect states often arise from actions that are good or bad for the character that is acted upon. For example, "*the cat ate the mouse*" produces a negative state for the mouse because being eaten is undesirable. Similarly, "*the man fed the dog*" produces a positive state for the dog because being fed is desirable. Many of these verbs are considered to have neutral emotional polarity, however, because they represent basic activities (e.g., eat, feed). Nonetheless, these verbs carry important world knowledge about how the activities affect their participants. Being *fed*, *paid*, or *adopted* is generally a desirable positive event for the entity that is acted upon, while being *eaten*, *chased*, or *hospitalized* are generally undesirable negative events for the entity that is acted upon.

We are not aware of any existing resources that identify verbs which produce a desirable/undesirable state for their patients even though the verb itself has neutral emotionality. We will refer to verbs that impart positive or negative polarity on their patients as *PPVs*. We try to fill this gap by using two corpus-based techniques to automatically harvest PPVs from a Web corpus.

4.2.1. PPV Harvesting with Evil/Kind Agents. The key idea behind our first approach is to identify verbs that frequently occur with evil or kind agents. Our intuition was that an "evil" agent will typically perform actions that are bad for the patient, whereas a "kind" agent will typically perform actions that are good for the patient.

We manually identified 40 stereotypically evil agent words, such as *monster*, *villain*, *terrorist*, and *murderer*, and 40 stereotypically kind agent words, such as *hero*, *angel*, *benefactor*, and *rescuer*. We searched the Google Web 1T N-gram corpus[8] to identify verbs that cooccur with these words as probable agents. For each agent term, we applied the pattern "* by [a,an,the] AGENT" and extracted the matching N-grams. Then we applied a part-of-speech tagger to each N-gram and saved the words that were tagged as verbs (i.e., the words in the * position).[9] This process produced 811 negative (evil agent) PPVs and 1,362 positive (kind agent) PPVs.

The first and third columns of Table 2 show examples of top 20 **Evil Agent** PPVs and the top 20 **Kind Agent** PPVs. Most of the evil agent PPVs are quite negative, from the patient's

[8] www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13.

[9] The POS tagging quality is undoubtedly lower than if tagging complete sentences but it seemed reasonable.

TABLE 2. Top 20 PPVs in Each Lexicon.

Evil Agent	Neg Basilisk	Kind Agent	Pos Basilisk
scam	sodomize	approve	harbor
injure	molest	chair	lampoon
hijack	punk	ordain	romance
raid	badger	assign	fete
chase	belittle	reproduce	laud
slam	persecute	prepare	reverence
overrun	ridicule	appoint	reward
drill	mock	pimp	commemorate
stalk	terrorize	lecture	glorify
dog	pillage	accept	adore
assassinate	imprison	swear	aspire
devour	rack	supervise	idolize
damage	brutalise	complete	commend
beset	intimdate	grant	venerate
ambush	ignominiously	judge	congratulate
bomb	humiliate	authorize	admire
victimize	harass	sort	praise
ravage	vilify	institute	honour
brightside	murder	forgiven	honor
ipod	brutalize	author	understood

perspective. However, the kind agent PPVs are a mixed bag: some of the words are positive for the patient (e.g., approve, accept) but many neutral words were generated as well (e.g., assign, prepare, and supervise). As we will discuss in Section 5.2, only the evil agent PPVs were ultimately used in AESOP.

4.2.2. PPV Bootstrapping over Conjunctions. Our second approach for acquiring PPVs is based on an observation from sentiment analysis research that conjoined adjectives typically have the same polarity (e.g., Hatzivassiloglou and McKeown 1997). Our hypothesis was that conjoined verbs often share the same polarity as well (e.g., "*abducted and killed*" or "*rescued and rehabilitated*"). We exploit this idea inside a bootstrapping algorithm to iteratively learn verbs that cooccur in conjunctions.

Bootstrapping begins with 10 negative and 10 positive PPV seeds. The seed words that we used are shown below:

−: abused beat caught criticized demoted killed mocked penalized punished robbed
+: adopted complimented empowered honored kissed loved promoted rewarded thanked trusted

First, we extracted triples of the form "$w1$ *and* $w2$" from the Google Web $1T$ N-gram corpus that had frequency ≥100 and were lower case. Because we wanted only verbs, we discarded conjunctions that contained a stopword, a word with a nonverbal suffix,[10] numbers, or nonwords (e.g., YYYZ). We separated each conjunction into two parts: a primary VERB ("$w1$") and a CONTEXT ("*and* $w2$"), and created a copy of the conjunction with the roles of

[10] We checked for eight such suffixes, such as "−tion" and "−ology."

$w1$ and $w2$ reversed. For example, “*rescued and adopted*” produces:

$$\text{VERB} = \text{“}\textit{rescued}\text{”}\ \text{CONTEXT} = \text{“}\textit{and adopted}\text{”}$$
$$\text{VERB} = \text{“}\textit{adopted}\text{”}\ \text{CONTEXT} = \text{“}\textit{and rescued}\text{”}$$

Next, we used the Basilisk bootstrapping algorithm (Thelen and Riloff 2002) to learn PPVs from these conjunction contexts. Basilisk identifies semantically similar words based on their cooccurrence with seeds in contextual patterns. Basilisk was originally designed for semantic class induction using extraction patterns, and has also been used to learn subjective and objective nouns (Riloff, Wiebe, and Wilson 2003).

Basilisk first identifies the pattern contexts that are most strongly associated with the seed words. Words that occur in those contexts are labeled as *candidates* and scored based on the strength of their contexts. The top five candidates are selected and the bootstrapping process repeats, using the newly learned words as additional seeds during the next iteration.

Basilisk produces a *lexicon* of learned words as well as a ranked list of the best pattern contexts. Because we bootstrapped over verb conjunctions, we also extracted new PPVs from the conjunction contexts. We ran the bootstrapping process to create a lexicon of 500 words, and we collected verbs from the top 500 pattern contexts as well.

The second and fourth columns of Table 2 show the top 20 positive and top 20 negative verbs harvested by Basilisk. The negative PPVs are of high quality (i.e., consistently negative from the patient’s perspective). However, in Section 5.2 we will see that the Negative Basilisk PPVs were essentially subsumed by the Evil Agent PPVs because the Evil Agent PPV lexicon was much larger. Thus, in the end AESOP only uses the Evil Agent PPVs to recognize negative affect states. However, the Positive Basilisk PPVs were of higher quality than the Kind Agent PPVs, and we did use them to produce positive affect states in AESOP. Table 2 shows several examples of verbs that are positive for their patients, such as “*harbor*,” “*romance,*” “*laud*,” and “*reward*.”

5. EVALUATION

We evaluated AESOP’s performance on our test set of 15 fables with gold standard annotations. We measured the accuracy of the affect states and links separately in terms of recall (R), precision (P), and F-measure (F). In our gold standard, each affect state is annotated with the set of clauses that could legitimately produce it. In most cases (75%), we were able to ascribe the existence of a state to precisely one clause. During evaluation, a system-produced affect state is counted as correct only if it was generated from the correct clause. For affect states that could be ascribed to multiple clauses in a sentence, the evaluation was done at the sentence level.

Coreference resolution is a difficult problem that is far from solved, thus we created gold standard coreference annotations for our fables and used them in most of our experiments. This allowed us to evaluate our approach to plot unit generation without coreference mistakes factoring in. In Section 5.5, we show comparative results using automatic coreference resolution.

5.1. Affect State Evaluation Using Only External Resources

Our first set of experiments evaluates the quality of the affect states produced by AESOP using only external lexical resources. The top half of Table 3 shows the results for each resource independently. FrameNet produced the best results, yielding the highest recall. The

TABLE 3. Evaluation Results for AESOP Using External Resources; The Number in Parentheses Is the Number of Gold Affect States.

Affect state	M (59)			+ (47)			− (37)			All (143)		
Resource(s)	*R*	*P*	*F*	*R*	*P*	*F*	*R*	*P*	*F*	*R*	*P*	*F*
FrameNet	0.49	0.51	0.50	0.17	0.57	0.26	0.14	0.42	0.21	0.29	0.51	0.37
MPQA Lexicon	0.07	0.50	0.12	0.21	0.24	0.22	0.22	0.38	0.28	0.15	0.31	0.20
OpinionFinder	0.42	0.40	0.41	0.00	0.00	0.00	0.03	0.17	0.05	0.18	0.35	0.24
SOLex	0.07	0.44	0.12	0.17	0.40	0.24	0.08	0.38	0.13	0.10	0.41	0.16
SpeechAct	0.36	0.53	0.43	0.00	0.00	0.00	0.00	0.00	0.00	0.15	**0.53**	0.23
Frame + MPQA	0.44	0.52	0.48	0.30	0.28	0.29	0.27	0.38	0.32	**0.35**	0.40	0.37
Frame + OpnFinder	0.53	0.39	0.45	0.17	0.38	0.23	0.16	0.33	0.22	0.31	0.38	0.34
Frame + SOLex	0.49	0.51	0.50	0.26	0.36	0.30	0.22	0.42	0.29	0.34	0.45	**0.39**
Frame + SpeechAct	0.51	0.48	0.49	0.17	0.57	0.26	0.14	0.42	0.21	0.30	0.49	0.37

Note: Boldface values denote best precision (P), recall (R) and f-measure (f) using different resources.

TABLE 4. Evaluation Results for AESOP with PPVs; The Number in Parentheses Is the Number of Gold Affect States.

Affect state	M (59)			+ (47)			− (37)			All (143)		
Resource(s)	*R*	*P*	*F*	*R*	*P*	*F*	*R*	*P*	*F*	*R*	*P*	*F*
− Evil PPVs	0.07	0.50	0.12	0.21	0.40	0.28	0.46	0.46	0.46	0.22	0.44	0.29
− Neg Basilisk PPVs	0.07	0.44	0.12	0.11	0.45	0.18	0.24	0.45	0.31	0.13	0.45	0.20
− Evil & NegBas PPVs	0.05	0.43	0.09	0.21	0.38	0.27	0.46	0.40	0.43	0.21	0.39	0.27
+ Kind PPVs ($\theta > 1$)	0.03	0.33	0.06	0.28	0.17	0.21	0.00	0.00	0.00	0.10	0.19	0.13
+ Pos Basilisk PPVs	0.08	0.56	0.14	0.02	0.12	0.03	0.03	1.00	0.06	0.05	0.39	0.09
FrameNet	0.49	0.51	0.50	0.17	0.57	0.26	0.14	0.42	0.21	0.29	**0.51**	0.37
Frame + SOLex	0.49	0.51	0.50	0.26	0.36	0.30	0.22	0.42	0.29	0.34	0.45	0.39
Frame + SOLex + Evil	0.49	0.54	0.51	0.30	0.38	0.34	0.46	0.42	0.44	0.42	0.46	0.44
Frame + Evil	0.49	0.54	0.51	0.28	0.45	0.35	0.46	0.46	0.46	0.41	0.49	**0.45**
Frame + Evil + PosBas	0.49	0.53	0.51	0.30	0.41	0.35	0.49	0.49	0.49	**0.43**	0.48	**0.45**

Note: Boldface values denote best precision (P), recall (R) and f-measure (f) using different resources.

bottom half of Table 3 shows AESOP's results when combining FrameNet with each of the other resources. In terms of F score, the only additive benefit came from the Semantic Orientation Lexicon (SOLex), which produced a better balance of recall and precision and an F score gain of +2.

The analysis of affect origin in Section 3.2 probably explains why FrameNet produced better results than the sentiment lexicons. Most affect states are derived from situational states and plan/goal states, which are often expressed using verbs. Relatively few affect states originate from explicit expressions of emotion.

5.2. Affect State Evaluation with PPVs

Our second set of experiments evaluates the quality of the new PPV lexicons that we harvested from a Web corpus. The top portion of Table 4 shows the results for the negative

PPVs. The PPVs harvested by the Evil Agent patterns (Evil PPVs) produced the best results, yielding recall and precision of 0.46 for $-$ states. Note that M and $+$ states are also generated from the negative PPVs because they can be inferred during affect projection (Section 4.1.4). A $+$ affect state can also be produced by a $-$ PPV when it occurs in a negated context and its polarity is flipped.

The negative PPVs harvested by Basilisk achieved similar precision but lower recall. We see no additional recall and some precision loss when both negative PPV lists are combined. The precision drop is likely due to redundancy, which can create spurious affect states. If two different words refer to the same event, then only one affect state should be generated. However, since AESOP does not attempt event coreference, if both words produce an affect state, then one of them will be scored as spurious.

The middle section of Table 4 shows the results for the positive PPVs. Both positive PPV lexicons were of dubious quality, thus we tried to extract a high-quality subset of each list. For the Kind Agent PPVs (Kind PPVs), we computed the ratio of the frequency of the verb with evil agents versus kind agents and only saved verbs with an evil:kind ratio (θ) > 1, which yielded $1,203$ PPVs. For the positive Basilisk PPVs (PosBas), we used only the 100 top-ranked lexicon and pattern verbs, which yielded 164 unique verbs. Table 4 shows that the positive PPVs did generate several correct affect states (including a $-$ state when a $+$ PPV was negated), but also many spurious states.

The bottom section of Table 4 shows the impact of the PPVs when combined with FrameNet (Frame) and the Semantic Orientation Lexicon (SOLex). Adding the Evil Agent PPVs (Evil) improves F score by $+5$ with an $+8$ overall recall gain because the recall of the $-$ states increased from 22% to 46% (with no loss of precision). Interestingly, if we remove SOLex and use only FrameNet and our Evil PPVs, precision increases from 46% to 49% and recall only drops by -1. Finally, the last row of Table 4 shows that adding Basilisk's positive PPVs (PosBas) produces an additional 2% recall gain.

Throughout the rest of this paper, unless otherwise noted, the version of AESOP that we use includes only FrameNet, the Evil Agent PPVs, and the Positive Basilisk PPVs.

5.3. Manual Evaluation of PPV Lexicons

We also conducted a manual evaluation to directly measure the quality of our PPV lexicons. We recruited three annotators and developed annotation guidelines that instructed each annotator to judge whether a verb is generally *good* or *bad* for its patient, assuming that the patient is animate. The annotators assigned each verb to one of six categories: $\times$ (not a verb), 2 (always good), 1 (usually good), 0 (neutral, mixed, or requires inanimate patient), -1 (usually bad), -2 (always bad). Each annotator labeled 250 words: 50 words randomly sampled from each of our four PPV lexicons[11] (Evil Agent PPVs, Kind Agent PPVs, Positive Basilisk PPVs, and Negative Basilisk PPVs) plus 50 verbs labeled as neutral in the MPQA lexicon.

First, we measured agreement based on three groupings: *positive* (1 and 2), *neutral* (0), or *negative* (-2 and -1). We computed κ scores to measure interannotator agreement for each pair of annotators,[12] but the κ scores were relatively low because the annotators had trouble distinguishing the positive and neutral cases. Thus, we recomputed agreement using two groupings, *not-negative* (0 through 2) and *negative* (-2 and -1), and obtained κ scores

[11] We sampled from the top-ranked ($\theta > 1$) $1,203$ kind Agent PPVs, the top-ranked ($\theta > 1$) 477 Evil Agent PPVs, the top 164 (unique) positive Basilisk verbs, and the 678 (unique) negative Basilisk verbs.

[12] We discarded words labeled as not a verb.

TABLE 5. Overlap between PPVs and External Resources.

	Total	New	% New
− PPVs	811	560	69%
+ PPVs	164	106	65%

TABLE 6. Link Evaluation Results; Parentheses Show Number of Gold Links.

	Gold Affect States			*Sys Affect States*		
Links	*R*	*P*	*F*	*R*	*P*	*F*
xchar (56)	0.79	0.85	0.82	0.18	0.43	0.25
a (51)	0.90	0.94	0.92	0.04	0.07	0.05
m (26)	1.00	0.57	0.72	0.15	0.10	0.12

of 0.69, 0.71, and 0.74. We concluded that people largely agree on which verbs are bad for their patients, but they do not necessarily agree on which verbs are good for their patients. One reason may be that "bad" verbs often refer to physical harm and these are easy to recognize. In contrast, "good" verbs are often more abstract and open to interpretation (e.g., is being "envied" or "feared" a good thing?).

We used the labels produced by the two annotators with the highest agreement to measure the accuracy of our PPVs lists. Both the Evil Agent and Negative Basilisk PPVs were judged to be 72.5% accurate, averaged over the judges. The Kind Agent PPVs were only about 39% accurate, whereas the Positive Basilisk PPVs were nearly 50% accurate.[13] These results are consistent with our impressions that the negative PPVs are of relatively high quality, whereas the positive PPVs are mixed.

Finally, we confirmed that many of the harvested PPVs are not present in existing lexical resources. Table 5 shows the amount of overlap between the words in our PPV lists and the external resources used in our experiments. Of the 811 negative PPVs harvested with Evil Agent patterns, 560 (69%) are not present in any of the external resources. Of the 164 positive PPVs harvested with Basilisk, 106 (65%) are not present in any of the external resources. Some examples of harvested PPVs that were not present in any other resources are:

−: censor, chase, fire, orphan, paralyze, scare, sue
+: accommodate, harbor, nurse, obey, respect, value

5.4. Link Evaluation

We represent each link as a five-tuple $\langle$*src-clause*, *src-state*, *tgt-clause*, *tgt-state*, *link-type*$\rangle$, where source/target denotes the direction of the link, the source/target-states are the affect type $\{+, -, \text{M}\}$ and *link-type* is one of {a,m,xchar}. A system-produced link is scored as correct if *all* five elements of the tuple match the gold standard annotation.

The *Gold Aff States* column of Table 6 shows AESOP's performance using gold standard affect states. Our simple heuristics for creating links work surprisingly well for xchar and a links when perfect affect states are available. However, the heuristics yield mediocre

[13] However, the human judgments for the + PPVs had low agreement, thus the results for the + PPVs are only suggestive.

TABLE 7. Baselines and Subcomponent Results.

	R	*P*	*F*
Baseline #1 (FrameNet)	0.32	0.32	0.32
Baseline #2 (FrameNet + PPVs)	0.44	0.24	0.31
AESOP w/gold coref	0.43	0.48	0.45
AESOP w/auto coref	0.26	0.56	0.36

precision for m links, albeit with 100% recall. The *Sys Aff States* column of Table 6 shows AESOP's performance using system-generated affect states. Performance is much lower, which is not surprising because AESOP only has 43% recall and 48% precision for affect state recognition, so many states are missing or spurious. For a link to be correct, both the source and target affect states must be present and no spurious states can occur between them.

The cross-character links (xchar) fared better, with 18% recall and 43% precision. These links are generated when the projection rules map the affect state associated with a single verb onto multiple characters. Because precision is comparable to that of the affect states, these results suggest that future work should focus on improving the recall of shared events.

5.5. Baselines and Subcomponent Analysis

We performed additional experiments to evaluate a simple baseline system and to assess the impact of each of AESOP's components individually. Baseline #1 simply uses FrameNet to generate affect states in an exhaustive fashion. The baseline system naively assigns every affect state in a clause to every character in that clause. The first row of Table 7 shows that this baseline achieves 32% recall and precision. Baseline #2 uses both FrameNet and our harvested PPV lists (Evil Agent and + Basilisk PPVs), but also naively assigns each affect state to every character in the same clause. Comparing these baselines, we see that the PPVs increased recall to 44%, but dropped precision to 24%. The third row of Table 7 shows the results of AESOP, which uses the same resources as Baseline #2 but applies its affect projection rules to assign affect states to characters and also infer affect states. AESOP achieves nearly the same recall as Baseline #2, but with double the precision. These results demonstrate that the projection rules substantially improve the quality of the plot units.

The last row of the Table 7 shows AESOP's performance when using automated coreference resolution (Section 4.1.2) instead of gold standard coreference annotations. We see a −17 recall drop coupled with a +8 precision gain. We were initially puzzled by the precision gain but believe that it is primarily due to the handling of quotations. Most fables end with a moral, which is often a quote. Our gold standard includes annotations for characters mentioned in quotations, but our automated coreference resolver ignores quotations. Thus, with automated coreference, we almost never produce affect states from quotations. This is a double-edged sword: some quotes contain important affect states, but many do not. For example, from the Father and Sons fable, "if you are **divided** among yourselves, you will be **broken** as easily as these sticks." Automated coreference does not produce any character resolutions and therefore AESOP produces no affect states. In this case, this is the right thing to do. However, in another well-known fable, a tortoise says to a hare: "although you be as **swift** as the wind, I have **beaten** you in the race." Here, perfect coreference produces multiple affect states, which *are* related to the plot: the hare receives a negative affect state for having been beaten in the race. Handling quotations in a more intelligent way is an important issue for future work.

TABLE 8. Recall with Respect to Affect State Origin Categories; Parentheses Show Number of Gold States.

	+ /− Affect States (84)			*M Affect States (59)*		
Category	E (3)	S (30)	PG-C (51)	PG-D (9)	PG-S (31)	PG-I (19)
Recall	0.00	0.53	0.31	0.33	0.68	0.32

TABLE 9. Analysis of Affect State Origin Performance.

	+/− Affect States (84)			*M Affect States (59)*		
Category	E (3)	S (30)	PG-C (51)	PG-D (9)	PG-S (31)	PG-I (19)
FrameNet only	0.00	0.13	0.10	0.22	0.65	0.26
PPVs only	0.00	0.43	0.12	0.00	0.00	0.00
FN + PPVs	0.00	0.43	0.18	0.22	0.61	0.26
FN + PPVs w/inferred states	0.00	0.53	0.31	0.33	0.68	0.32

Finally, in our gold standard we annotated *pure inference states* that are critical to the plot unit structure but come from world knowledge that is not mentioned in the story. For example, a fable may begin with a wolf luring a horse into a field, and a reader would infer that wolf's motivation was to eat the horse, but this is not explicitly stated in the story. Nonetheless, a mental affect state representing the wolf's intentions is a critical part of the plot unit representation. Of the 157 affect states in our test set, 14 were pure inference states. We ignored these states in our experiments because AESOP has no way to generate them (i.e., they have no provenance in the source text). For the sake of completeness, we added those pure inference states to the evaluation and found that AESOP's recall dropped from 43% to 39%. We conclude that affect states originating purely from inference are a relatively small part of the overall problem, but this is an issue that will need to be addressed in the future.

5.6. Analysis Based on Affect State Origin

In Section 3.2, we presented a study that assigned each affect state to one of six *affect origin categories* based on how it was derived (conceptually) from the source text. Using these annotations, we can now identify which types of affect states AESOP can successfully recognize and which it has the most trouble with. Table 8 shows AESOP's recall for each category of affect states.[14] AESOP recognizes 68% of the PG-S (speech act) mental affect states, and 53% of the S (situational) +/− affect states. The recall for the other affect states is generally around 30%, except for E (direct expressions of emotion) +/− affect states, which are extremely rare (only three such states in our test set) and AESOP does not recognize any of them.

Table 9 shows a detailed breakdown of how each category of affect states is recognized by AESOP. Affect states can be created from three components: FrameNet, PPV lexicons, or inferred by the affect projection rules (Section 4.1.4). The first row of Table 9 shows AESOP's recall when only FrameNet is used. We see that FrameNet is responsible for recognizing nearly all of the speech act affect states (PG-S), and also many of the PG-D,

[14] We cannot measure precision because we do not know which origin category the *incorrect* system-generated affect states belong to.

and PG-I affect states. The second row of Table 9 shows AESOP's recall when only the PPV lists are used. As we would expect, the PPVs identify many situational (S) affect states, contributing knowledge that is not available in FrameNet. The third row shows the results when using both FrameNet and PPVs. The last row in Table 9 shows the results when we also allow the projection rules to infer affect states based on verb argument structure. The inferred affect states produced by the projection rules improve recall in nearly all categories, but show the biggest increase on PG-C states. This makes sense because the heuristic for inferring positive affect states was specifically designed to create positive affect states for agents under the assumption that an activity was successfully accomplished.

6. RELATED WORK

Our work is related to research in narrative story understanding (e.g., Elson and McKeown 2009), goal-based story understanding (e.g., Wilensky 1978), and conceptual knowledge structures (e.g., Schank and Abelson 1977), and contributes to the body of work in creating computational models for conceptual language understanding (e.g., Mooney and DeJong 1985; Fujiki, Nanba, and Okumura 2003; Chambers and Jurafsky 2008, 2009; Kasch and Oates 2010). We focused our work specifically on plot units because they revolve around affect and the emotional and situational tensions between characters. With the growth in freely available languages resources, especially for verb semantics and sentiment analysis, we felt that the time was right to make progress on the automatic generation of plot unit representations. Some preliminary ideas had been proposed for plot unit modeling of single-character stories (Appling and Riedl 2009), but our research is the first effort to fully automate the creation of plot unit structures.

For affect state recognition, our work has benefited from prior research in creating semantic resources such as FrameNet (Baker et al. 1998), sentiment and emotion lexicons (Takamura et al. 2005; Wilson et al. 2005a; Mohammad, Dunne, and Dorr 2009; Velikovich et al. 2010), speech act verbs (Wierzbicka 1987), and opinion classifiers (e.g., Wilson et al. 2005a; Choi et al. 2006). There are also emotion lexicons available that identify emotion categories such as joy, sadness, anger, fear, and surprise (e.g., Mohammad and Turney (2010), General Inquirer (Stone et al. 1966), WordNet Affect Lexicon (Strapparava and Valitutti 2004)). These resources could potentially be useful for plot unit analysis as well. Recent work (Goldberg et al. 2009) on creating "wish detectors" tries to obtain insights into the world's wants and desires. The ideas from this work could help to identify affect states related to plans and goals.

For affect state projection, plot units need to assign affect states to individual characters. Assigning affect states to characters is somewhat analogous to, but not the same as, associating opinion words with their targets or topics (Kim and Hovy 2006; Stoyanov and Cardie 2008). There has been previous work on affect state analysis, such as Alm (2010), which produced a data set of 15,000 sentences with affect labels. Alm (2009) developed a supervised classifier to identify affect states automatically. However, they do not produce affect states with respect to a character. Some aspects of affect state identification are closely related to Hopper and Thompson (1980) theory of transitivity. In particular, their notions of *aspect* (has an action completed?), benefit and harm (how much does an object gain/lose from an action?), and volition (did the subject make a conscious choice to act?).

The PPV harvesting techniques that we created draw heavily from research on pattern-based semantic knowledge extraction (e.g., Hearst 1992; Paşca 2004; Kozareva, Riloff, and Hovy 2008), and bootstrapping techniques that learn from contextual patterns (e.g., Collins and Singer 1999; Riloff and Jones 1999). The Basilisk bootstrapping algorithm (Thelen and

Riloff 2002) that we used has also previously been applied to the problem of identifying subjective nouns (Riloff et al. 2003). Our conjunction-based bootstrapping approach was inspired by Hatzivassiloglou and McKeown (1997)'s observation that conjoined adjectives typically have the same polarity.

7. CONCLUSIONS

This research is the first computational model to fully automate the process of generating plot unit representations. This work represents a step forward toward narrative text understanding and the generation of more complex meaning representations related to emotions and affect. Overall, AESOP achieved 43% recall with 48% precision for affect state generation, which is encouraging given the complexity of the plot unit task, but still leaves ample room for improvement. The quality of the affect states will need to be improved before we can expect to produce complete plot unit structures with high accuracy.

During the course of this work, we identified a type of world knowledge that is largely missing from existing lexical resources: knowing whether an action is good or bad for the entity that is acted upon. To fill this gap, we created two methods to automatically acquire verbs that impart positive or negative polarity on their patients (*PPVs*), and we showed that they improve affect state recognition on the fables. Of the 975 PPVs that we acquired, 666 (68%) of them were not present in any of the preexisting lexical resources that we used in this research. PPVs represent just one kind of world knowledge that is important for affect state recognition; we expect that there are many others as well.

We made several simplifying assumptions in this work, and the problem of generating high-quality plot unit representations remains far from solved. First, AESOP was evaluated only on two-character fables. Most of the language analysis was performed with general-purpose tools (e.g., parsing and affect state recognition), but AESOP relied on perfect coreference resolution and assumed that the two main characters could be identified from the title of the fable. To create plot units for other text genres, identifying the central characters of the story will be a new challenge that must be tackled. Our projection rules also made some simplifying assumptions to identify agents and patients. Semantic role labeling may be needed to identify these roles accurately in other forms of text. Second, most fables have a linear temporal ordering, but that is not necessarily true in other text genres. In particular, the heuristic that we used to produce causal links is based on the assumption that the affect states are ordered chronologically. In future work, more sophisticated recognition of plans and intentions will likely be required to produce causal links more accurately.

Finally, our analysis revealed that many affect states arise from the plans and goals of the characters in a story, and we found that our system has an especially difficult time recognizing these affect states. Plan/goal recognition seems to be a central issue for plot unit generation and a particularly important research direction for future work in this area.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of Department of Homeland Security Grant N0014-07-1-0152, NSF grant IIS-0712764, and the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0172. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA, AFRL, or the U.S. government. Thanks to Peter Jensen, Emily Schlichter, and Clay Templeton for PPV annotations, Nathan Gilbert for help with the coreference resolver, and the anonymous reviewers for many helpful comments.

REFERENCES

ALM, C. O. 2009. Affect in Text and Speech. VDM Verlag Dr. Müller.

ALM, C. O. 2010. Characteristics of high agreement affect annotation in text. *In* Proceedings of the Fourth Linguistic Annotation Workshop, Association for Computational Linguistics, pp. 118–122.

APPLING, D. S., and M. O. RIEDL. 2009. Representations for learning to summarize plots. *In* Proceedings of the Association for the Advancement of Artificial Intelligence Spring Symposium on Intelligent Narrative Technologies II.

BAKER, C. F., C. J. FILLMORE, and J. B. LOWE. 1998. The Berkeley FrameNet Project. *In* Proceedings of Conference on Computational Linguistics and Association for Computational Linguistics (COLING/ACL).

CHAMBERS, N., and D. JURAFSKY. 2008. Unsupervised learning of narrative event chains. *In* Proceedings of the Association for Computational Linguistics, Association for Computational Linguistics, Columbus, OH, pp. 789–797.

CHAMBERS, N., and D. JURAFSKY. 2009. Unsupervised learning of narrative schemas and their participants. *In* Proceedings of the Association for Computational Linguistics, Association for Computational Linguistics, Suntec, Singapore, pp. 602–610.

CHOI, Y., E. BRECK, and C. CARDIE. 2006. Joint extraction of entities and relations for opinion recognition. *In* EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Morristown, NJ, pp. 431–439.

COLLINS, M., and Y. SINGER. 1999. Unsupervised models for named entity classification. *In* Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99), pp. 100–110.

DAS, S. R., and M. Y. CHEN. 2001. Yahoo! for Amazon: Opinion extraction from small talk on the Web. *In* Proceedings of the 8th Asia Pacific Finance Association Annual Conference (APFA), Bangkok, Thailand.

DAVE, K., S. LAWRENCE, and D. M. PENNOCK. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. *In* Proceedings of the 12th International World Wide Web Conference (WWW2003), Budapest, Hungary, pp. 519–526.

ELSON, D., and K. MCKEOWN. 2009. Extending and evaluating a platform for story understanding. *In* Proceedings of the Association for the Advancement of Artificial Intelligence 2009 Spring Symposium on Intelligent Narrative Technologies II.

FUJIKI, T., H. NANBA, and M. OKUMURA. 2003. Automatic acquisition of script knowledge from a text collection. *In* Proceedings of the European Association for Computational Linguistics, Association for Computational Linguistics, Morristown, NJ, pp. 91–94.

GOLDBERG, A. B., N. FILLMORE, D. ANDRZEJEWSKI, Z. XU, B. GIBSON, and X. ZHU. 2009. May all your wishes come true: A study of wishes and how to recognize them. *In* NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics. pp. 263–271.

GOYAL, A., E. RILOFF, and H. Daumé III. 2010a. Automatically producing plot unit representations for narrative text. *In* Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 77–86.

GOYAL, A., E. RILOFF, H. Daumé III, and N. GILBERT. 2010b. Toward plot units: Automatic affect state analysis. *In* Proceedings of HLT/NAACL Workshop on Computational Approaches to Analysis and Generation of Emotion in Text (CAET), Los Angeles, CA, pp. 17–25.

HATZIVASSILOGLOU, V., and K. MCKEOWN. 1997. Predicting the semantic orientation of adjectives. *In* Association for Computational Linguistics, Madrid, Spain, pp. 174–181.

HEARST, M. 1992. Automatic acquisition of hyponyms from large text corpora. *In* Proceedings of the 14th International Conference on Computational Linguistics (COLING-92), pp. 539–545.

HOPPER, P. J., and S. A. THOMPSON. 1980. Transitivity in grammar and discourse. Language, **56**:251–299.

KASCH, N., and T. OATES. 2010. Mining script-like structures from the web. *In* Proceedings of North American Chapter of the Association for Computational Linguistics (NAACL) Workshop on Formalisms and Methodology for Learning by Reading (FAM-LbR), Association for Computational Linguistics, Los Angeles, CA, pp. 34–42.

KIM, S., and E. HOVY. 2006. Extracting opinions, opinion holders, and topics expressed in online news media Text. *In* Proceedings of Association for Computational Linguistics and Conference on Computational Linguistics (ACL/COLING) Workshop on Sentiment and Subjectivity in Text, Association for Computational Linguistics, Morristown, NJ, pp. 1–8.

KOZAREVA, Z., E. RILOFF, and E. HOVY. 2008. Semantic class learning from the Web with hyponym pattern linkage graphs. *In* Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08), Association for Computational Linguistics, Columbus, OH, pp. 1048–1056.

LEHNERT, W., H. ALKER, and D. SCHNEIDER. 1983. Affective plot structure of Toynbee's Christus patiens. *In* Proceedings of the Sixth International Conference on Computers and the Humanities.

LEHNERT, W., J. BLACK, and B. REISER. 1981. Summarizing narratives. *In* Proceedings of the Seventh International Joint Conference on Artificial Intelligence. Morgan Kaufmann: San Francisco, CA, pp. 184–189.

LEHNERT, W. G. 1981. Plot Units and Narrative Summarization. Cognitive Science, **5**(4):293–331.

LEHNERT, W. G. 1983. Narrative complexity based on summarization algorithms. *In* IJCAI'83: Proceedings of the Eighth International Joint Conference on Artificial Intelligence. Morgan Kaufmann: San Francisco, CA, pp. 713–716.

MILLER, GEORGE A., RICHARD BECKWITH, CHAISTIANE FELLBAUM, DEREK GROSS and KATHERINE J. MILLER. 1990. "Introduction to WordNet: an on-line lexical database." *In* International Journal of Lexicography **3**(4) pp. 235–244.

MOHAMMAD, S., C. DUNNE, and B. DORR. 2009. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. *In* Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 599–608.

MOHAMMAD, S., and P. TURNEY. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. *In* Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL) Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Association for Computational Linguistics, pp. 26–34.

MOONEY, R., and G. DEJONG. 1985. Learning schemata for natural language processing. *In* Proceedings of the Ninth International Joint Conference on Artificial Intelligence, pp. 681–687.

MORINAGA, S., K. YAMANISHI, K. TATEISHI, and T. FUKUSHIMA. 2002. Mining product reputations on the Web. *In* Proceedings of the 8th Association for Computing Machinery SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2002), Edmonton, Canada, pp. 341–349.

NASUKAWA, T., and J. YI. 2003. Sentiment analysis: Capturing favorability using natural language processing. *In* Proceedings of the 2nd International Conference on Knowledge Capture (K-CAP 2003), Sanibel Island, FL, pp. 70–77.

PAŞCA, M. 2004. Acquisition of categorized named entities for web search. *In* Proceedings of the Thirteenth Association for Computing Machinery International Conference on Information and Knowledge Management, pp. 137–145.

PANG, B., L. LEE, and S. VAITHYANATHAN. 2002. Thumbs up? Sentiment classification using machine learning techniques. *In* Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, Philadelphia, PA, pp. 79–86.

REISER, B., W. LEHNERT, and W. BLACK. 1981. Recognizing thematic units in narratives. *In* Proceedings of the Third Annual Cognitive Science Conference, Berkeley, CA.

RILOFF, E., and R. JONES. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. *In* Proceedings of the Sixteenth National Conference on Artificial Intelligence, pp. 474–479.

RILOFF, E., and W. PHILLIPS. 2004. An introduction to the Sundance and AutoSlog systems. Technical Report UUCS-04-015, School of Computing, University of Utah.

RILOFF, E., J. WIEBE, and T. WILSON. 2003. Learning subjective nouns using extraction pattern bootstrapping. *In* Conference on Computational Natural Language Learning (CONLL), pp. 25–32.

RUMELHART, D. 1975. Notes on a scheme for stories. *In* Representation and Understanding. *Edited by* D. G. Bobrow and A. M. Collins. Academic Press: New York, pp. 211–236.

SCHANK, R. C., and R. P. ABELSON. 1977. Scripts, Plans, Goals and Understanding. Lawrence Erlbaum: Hillsdale, NJ.

SPERTUS, E. 1997. Smokey: Automatic recognition of hostile messages. *In* Proceedings of the Eighth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-97), Providence, RI, pp. 1058–1065.

STONE, P. J., Dexter C. DUNPHY, M. S. SMITH, and D. M. OGILVIE. 1966. The General Inquirer: A Computer Approach to Content. MIT Press: Cambridge, MA.

STOYANOV, V., and C. CARDIE. 2008. Topic identification for fine-grained opinion analysis. *In* Conference on Computational Linguistics (COLING 2008), Coling 2008 Organizing Committee, Manchester, UK, pp. 817–824.

STRAPPARAVA, C., and A. VALITUTTI. 2004. WordNet-Affect: An affective extension of WordNet. *In* Proceedings of International Conference on Language Resources and Evaluation (LREC), Volume **4**, pp. 1083–1086.

TAKAMURA, H., T. INUI, and M. OKUMURA. 2005. Extracting semantic orientations of words using spin model. *In* ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 133–140.

THELEN, M., and E. RILOFF. 2002. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. *In* Proceedings of the Empirical Methods in Natural Language Processing, pp. 214–221.

THORNDYKE, P. 1997. Cognitive structures in comprehension and memory of narrative discourse. Cognitive Psychology, **9**:77–110.

TONG, R. 2001. An operational system for detecting and tracking opinions in on-line discussions. *In* Working Notes of the Special Interest Group on Information Retrieval (SIGIR) Workshop on Operational Text Classification, New Orleans, LA, pp. 1–6.

TURNEY, P., and M. L. LITTMAN. 2003. Measuring praise and criticism: Inference of semantic orientation from association. Association for Computing Machinery (ACM) Transactions on Information Systems (TOIS), **21**(4):315–346.

VELIKOVICH, L., S. Blair-Goldensohn, K. HANNAN, and R. MCDONALD. 2010. The viability of Web-derived polarity lexicons. *In* Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 777–785.

WIERZBICKA, A. 1987. English Speech Act Verbs: A Semantic Dictionary. Academic Press: Sydney, Australia.

WILENSKY, R. 1978. Understanding goal-based stories, Ph. D. Thesis, Yale University, New Haven, CT,.

WILSON, T., J. WIEBE, and P. HOFFMANN. 2005a. Recognizing contextual polarity in phrase-level sentiment analysis. *In* Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 347–354.

WILSON, T., P. HOFFMANN, S. SOMASUNDARAN, J. KESSLER, J. WIEBE, Y. CHOI, C. CARDIE, E. RILOFF, and S. PATWARDHAN. 2005b. OpinionFinder: A system for subjectivity analysis. *In* Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing Interactive Demonstrations, pp. 34–35.

YI, J., T. NASUKAWA, R. BUNESCU, and W. NIBLACK. 2003. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. *In* Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM-2003), Melbourne, FL, pp. 427–434.