18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Use of Background Knowledge in Natural Language Understanding for Information Fusion

Stuart C. Shapiro
Department of Computer Science and Engineering
Center for Multisource Information Fusion
and Center for Cognitive Science
University at Buffalo, Buffalo, New York
shapiro@buffalo.edu

Daniel R. Schlegel
Department of Biomedical Informatics
Center for Multisource Information Fusion
and Center for Cognitive Science
University at Buffalo, Buffalo, New York
drschleg@buffalo.edu

***Abstract*—Tractor is a system for understanding English messages within the context of hard and soft information fusion for situation assessment. Tractor processes a message through text processors, and stores the result, expressed in a formal knowledge representation language, in a syntactic knowledge base. This knowledge base is enhanced with ontological and geographic information. Finally, Tractor applies hand-crafted syntax-semantics mapping rules to convert the enhanced syntactic knowledge base into a semantic knowledge base containing the information from the message enhanced with relevant background information. Throughout its processing, Tractor makes use of various kinds of background knowledge: knowledge of English usage; world knowledge; domain knowledge; and axiomatic knowledge. In this paper, we discuss the various kinds of background knowledge Tractor uses, and the roles they play in Tractor's understanding of the messages.**

Keywords*—*background knowledge, natural language understanding, soft information fusion, message understanding, information extraction, hard+soft information fusion.

I. An Overview of Tractor

Tractor is a system for message understanding within the context of a multi-investigator, multi-disciplinary, multi-university effort on "Hard and Soft Information Fusion" [1]. Information obtained from physical sensors such as pan tilt zoom (PTZ) cameras, light detection and ranging (LIDAR) sensors and acoustic sensors are considered hard information. Information from humans expressed in natural language is considered soft information. Tractor [2], [3] is a computational system that understands isolated English intelligence messages in the counter-insurgency domain for later fusion with each other and with hard information, all to aid intelligence analysts in performing situation assessment. In this context, "understanding" means creating a knowledge base (KB), expressed in a formal knowledge representation (KR) language, that captures the information in an English message.

Tractor has been introduced and discussed in a previous set of papers [1], [2], [3], [4], [5], [6], [7]. An overview of the entire Hard and Soft Information Fusion project, and the architecture of the process is given in [1]. An introduction to Tractor and its initial architecture is given in [2]. An introduction to the Context-Based Information Retrieval (CBIR) subprocess of Tractor, its proposed use of spreading activation, and how spreading activation algorithms might be evaluated is given in [5]. A general overview of the role of contextual information in information fusion architectures is given in [4]. Tractor's use of propositional graphs for representing syntactic and semantic information is introduced in [6]. The rules that map syntactic to semantic information are discussed in [3]. The design of a test and evaluation framework for Tractor and the larger Hard and Soft Information Fusion system is given in [7], along with some preliminary results.

The proposed use of spreading activation for CBIR [5] was not, in fact, pursued. Nevertheless, background knowledge is used by Tractor, and Tractor adds relevant background knowledge to the information it extracts from each message. In this paper, we discuss what background knowledge is used by Tractor, and where in Tractor's processing background knowledge is used. This discussion both completes the elucidation of how Tractor understands natural language messages for information fusion, and indicates where changes would need to be made for domains other than the one for which Tractor was initially developed.

Tractor takes as input a single English message. The ultimate goal is for Tractor to output a KB representing the semantic information in that message. Later systems of the larger project combine these KBs with each other and with hard information. Combining KBs from different messages and different hard sources is done via a process of data association [1], [8] that operates by comparing the attributes of and relations among the entities and events described in each KB. It is therefore important for Tractor to express these attributes and relations as completely and accurately as possible. Doing this requires the use of background knowledge—knowledge that is not explicitly included in the text. Background knowledge includes: knowledge of how the natural language is used; knowledge of the world; knowledge of the domain being discussed in the text; and knowledge of the axioms of the relations that are used (explicitly or implicitly) in the text.

The architecture of Tractor is shown in Fig. 1. Each English message is input to a set of "processing resources" (PRs) operating within GATE, the General Architecture for Text Engineering [9]. Most of these PRs are from the ANNIE (a

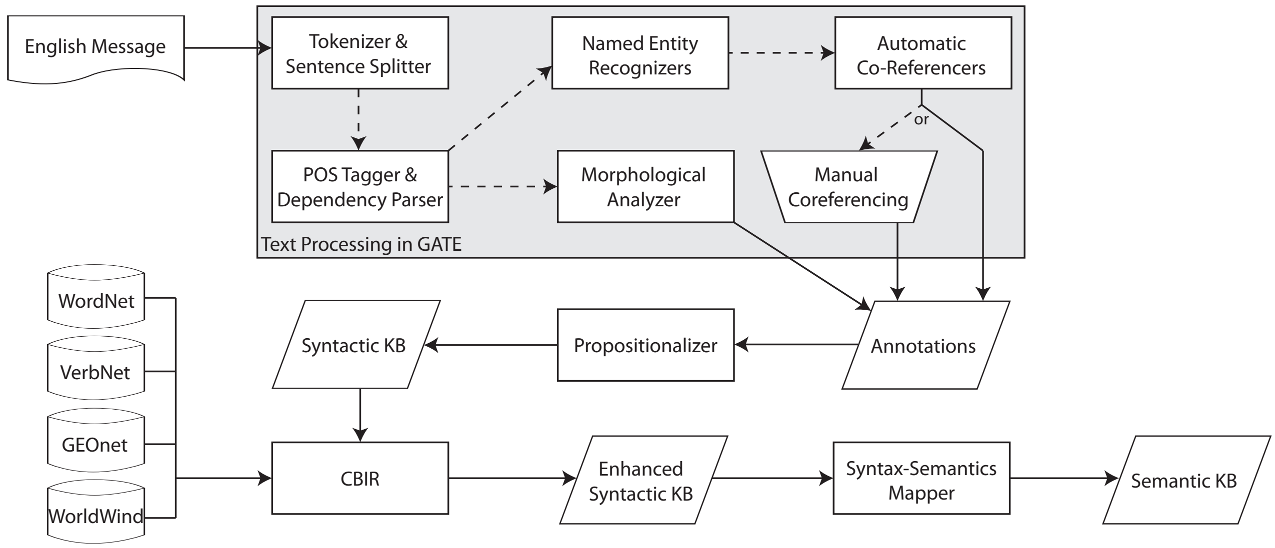

Fig. 1. Tractor Architecture

Nearly-New Information Extraction System) suite [10]. Shown in Fig. 1 are: the ANNIE English Tokenizer and Sentence Splitter that divide the input into linguistic units; the Stanford Dependency Parser, for part-of-speech tagging and parsing; the GATE Morphological Analyser for finding the root forms of inflected nouns and verbs; a group of named-entity recognizers, to be discussed in Sect. II; a group of PRs that perform co-reference resolution; and the optional GATE Co-reference Editor for manual corrections of and additions to the results of the automatic coreference resolution PRs.

The results of GATE processing, with or without the Co-reference Editor, is a set of "annotations", each consisting of an ID, a start and end position within the message's text string, a Type, and a set of attribute-value pairs. The Propositionalizer examines the annotations, and produces a "syntactic KB" consisting of a set of assertions in the SNePS 3 knowledge representation language [11], [12], [13].

The syntactic KB is enhanced by CBIR (Context-Based Information Retrieval) [14], [4], [5] with additional ontological information, to be discussed in Sect. IV, and geographic information, to be discussed in Sect. V. The enhanced syntactic KB is operated on by the syntax-semantics mapping rules using the background knowledge discussed in Sect. VI.

Tractor and the larger information fusion system of which it is a part have been developed by experimenting with several datasets, particularly the Synthetic Counterinsurgency (SYNCOIN) [15] dataset. All examples in this paper have been drawn from these datasets.

II. Background Knowledge in Text Processing

During text processing in GATE (*see* Fig. 1), background knowledge is used for named-entity recognition by two PRs: the ANNIE Gazetteer, for lexicon-based named-entity recognition; and the ANNIE NE Transducer, for rule-based named-entity recognition. "This process of named entity recognition refers to the combined task of finding spans of text that constitute proper names and then classifying the entities being referred to according to their type" [16, p. 727].

The ANNIE Gazetteer uses lists of names of people, neighborhoods, cities, countries, car models, car companies, newspapers, other companies, religious groups, ministries, etc. There are currently 138 lists, each of which contains a list of names each of which is the name of an entity of some given type, and possibly a subtype. The lists are compiled into a finite-state machine used to recognize occurrences of these names in the messages. We have extended the Gazetteer lists that were supplied with GATE to recognize names of entities specific to our domain, such as the person Dhanun Ahmad Mahmud Ahmad, the neighborhood Abu T'Shir, and the organizations BCT and ISG.

The ANNIE Gazetteer is supplemented by the ANNIE NE Transducer, which uses sets of rules written in the Java Annotation Patterns Engine (JAPE) language. For example, one set of rules recognizes dates and times written in various formats, another recognizes email addresses, and another recognizes world-wide web addresses. There are also JAPE rules that recognize named entities due to their context. For example, the rule named "PersonContext1" looks for a capitalized word not already known to be a person, followed by the word "from", followed by an Organization, and declares the capitalized word to be a Person. So, if the text had "Mohammad from ING", and Mohammad were not already in the name lexicon, it would be identified as a Person anyway. The rule named "PersonTitle1" looks for a title followed by an unknown named entity, usually consisting of capitalized words, and declares that to name a person. For example, "Admiral Morrison" would be declared to be a person even if "Morrison" were not in the gazetteer.

We have extended the JAPE rules that were supplied with GATE to recognize: weights and heights in common formats; groups of persons identified by a listing in the message, such as "Dhanun Ahmad Mahmud, Mu'adh Nuri Khalid Jihad, Sattar

'Ayyash Majid, Abd al-Karim, and Ghazi Husayn" ; names of persons, locations, and organizations with common Arabic prefixes such as "al-"; decade time periods such as "20s"; and dates in context such as "morning of 01/23".

The named-entity recognizers accomplish two tasks. One is to specify the categories of various entities mentioned in the messages for use by data association. The other is to provide single tokens for entities that have multiple-word names. For example, even though the Tokenizer splits "Iraqi National Police" into three tokens, the Gazetteer recognizes that this three-word phrase is the single name of an organization, for which it creates a single token which could be referred to as "they" later in some message.

The named-entity recognizers add semantic information to the set of annotations which become the syntactic KB. Recent investigations have shown that about 32% of the assertions in the syntactic KBs are such semantic information [3].

III. Background Knowledge in the Propositionalizer

The Propositionalizer examines the annotations produced by the GATE PRs, and produces a set of SNePS 3 assertions. The stages of the Propositionalizer are: annotation merging; correction of minor errors in syntactic categories; canonicalization of dates and times; and processing the structured portion of semi-structured messages. For this last stage, the Propositionalizer uses our knowledge of the format of our corpus of messages. Most of them have structured headers, generally consisting of a message number and date, and sometimes a time and either "ET" for "event time", or "RT" for "report time". A message reporting a call intercept generally lists a description or name of the caller and of the recipient, duration, medium (e.g., cell phone or text message), and intercepting analyst. An example message that is not a call intercept is

> 127. 02/10/10 - ET: 0545hrs – The safe-house on Horajeb Road, //MGRSCOORD: 38S MB 44709 79949//, was raided suddenly and unexpectedly by INP forces.

An example message reporting a call intercept is

> 248. 3/16/10 - RT: 0700hrs - |C:| || satellite phone| |P1:| caller| Dhanun Ahmad|||| |P2:| receiver| Dhanun Ahmad's handler's voice drop-box|||| |A:| "Ahmad said he is in al-Kut after driving through the night. He could not stay on the phone long, as the man traveling with him was watching closely."

Semi-structured message matchers are created using a rule language designed specifically for this task. Rule files are made up of a sequence of named rules, and code for mapping the named segments of a matched message to SNePS 3 assertions. The rule files are built, much like in a software compiler, into a tree which is matched against each message. When a message matches, the code associated with the matcher executes, producing SNePS 3 assertions. Matchers are given priorities to define the order in which they are checked against a message, allowing for the creation of progressively more simple (or broad) matchers.

IV. Enhancing with Ontological Information

CBIR (Context-Based Information Retrieval) [14], [4], [5] adds relevant ontological and geographic information to the syntactic KB. The information is "relevant" in the sense that, although CBIR has access to large databases of ontological and geographical information, it adds to the syntactic KB only those data that are connected to the terms already in the syntactic KB. For example, it would add ontological information above the term "truck" only to the KB of a message that mentions a truck, and geographic information about Baghdad only to the KB of a message that mentions Baghdad.

CBIR first looks up in WordNet [17] all the common nouns that are in a syntactic KB, and adds to the KB the synsets of the nouns, their hypernyms, the hypernyms of their hypernyms, *etc.,* all the way to the top of the ontology. Then it looks up in VerbNet [18] all the verbs in the KB, and adds all their classes, parent classes, *etc.* At the top of the VerbNet hierarchy, CBIR looks up all the member verbs of the highest level classes in WordNet, and adds the connected WordNet hierarchy to the VerbNet hierarchy.

Although VerbNet and WordNet are often viewed as hierarchies of words, and thus in the syntactic realm, WordNet synsets are groups of synonymous words "expressing a distinct concept" [17] and the hypernym relation is a semantic relation between concepts. VerbNet classes are an extension of Levin classes [19], which add subclasses to "achieve syntactic and semantic coherence among members of a class" [17]. Thus, the VerbNet and WordNet hierarchies added by CBIR constitute an ontology in the semantic realm. The addition of this ontology adds to the categorization of entities and events begun by the named-entity recognizers. These categories are used by the syntax-semantics mapping rules so that they apply to classes of entities and events, not just to specific ones. This will be discussed further in Sect. VI. In addition, the ontology is used by the scoring algorithms of the data association routine to assess the semantic distance between entities and events mentioned in different messages.

V. Enhancing with Geographic Information

CBIR looks up every proper noun that is in the message in the NGA GEOnet Names Server database [20]. To reduce the confusion caused when one name is the name of multiple places, we use our knowledge of our domain to restrict the database to places in Iraq. The information found is added to the KB for the message. For example, looking up Badrah, CBIR finds that it is a second order administrative division, its MGRS (Military Grid Reference System) coordinates are 38SNB8399760885, its latitude is 33.08333, and it longitude is 45.90000.

If CBIR finds MGRS coordinates, but no latitude and longitude (This particularly happens when MGRS coordinates are explicitly included in a message.), it converts the MGRS coordinates to latitude and longitude using NASA's World Wind software [21].

The geographic information added by CBIR is used by the data association system.

VI. Background Knowledge in Syntax-Semantics Mapping

The main job of the syntax-semantics mapping rules is to convert syntactic information, created largely by the parser, into semantic information. For example, in the sentence, "Coalition forces in the Shi'a neighborhood of Abu T'Shir arrested a man after he was observed directing the offload of heavy weapons," the phrase "a man" is parsed as the direct object of the verb "arrested," whereas in the sentence (from a later message), "A man arrested in the Shi'a neighborhood of Abu T'Shir has been identified as Abdul Jabar," the phrase "arrested in the Shi'a neighboorhood of Abu T'Shir" is parsed as a participial modifier of "a man." In both cases, however, the syntax-semantics mapping rules recognize that the man is the theme of the arrest event, helping data association to recognize that the men mentioned in the two sentences are the same.

Additionally, the mapping rules: supplement VerbNet and WordNet by adding additional ontological assertions; convert some idioms to standard usages; make purely syntactic changes, such as changing a passive construction to an active one; and draw inferences, making the conclusions explicit in the KB. In the rest of this section, we will discuss several examples of the use of background knowledge by the mapping rules.

Graded, descriptive adjectives provide linguistic values for attributes of instances of categories such that the adjective and the category imply the attribute [22, pp. 48ff]. The mapping rules use a database of $adjective \times category \rightarrow attribute$ mappings to find the correct attribute. For example, "a young man" is interpreted as a man whose age is young, and "a large gathering" is interpreted as a group whose cardinality is large.

Sometimes a possessive construction indicates ownership, sometimes the part-of relation, and sometimes a weaker association. The mapping rules use a mereological database of parts and wholes to interpret, for example, the phrase "the man's arm" as an arm that is part of the man, rather than an arm that is owned by the man.

Count nouns (like "car") denote categories whose instances occur in discrete units that can be counted. Mass nouns denote substances (like "wood) that objects may be made of. Some nouns can be used both ways ("a piece of a cake" *vs.* "a piece of cake"). The mapping rules use a list of mass nouns, so that, for example, "a man with dark hair" is correctly interpreted as a man who has as a part something which is made of hair whose color is dark. (Notice that this interpretation also makes use of the mereology and the database of graded, descriptive adjectives.)

A common noun, especially one that is the head of a noun phrase, usually denotes an entity that is an instance of the category expressed by the noun. However, the named-entity recognizers recognize certain nouns as job titles, and in that case, the mapping rules identify the noun as denoting an entity that fills the role. For example, in the sentence, "The assistant said the man she treated was covered in dust," "the man" is understood to denote an instance of the category man, but "the assistant" is understood to denote a person who fills the role of assistant. Similarly, a plural common noun, such as "heavy weapons" is understood to denote a group whose members are instances of the category expressed by the noun ("weapon"), but a plural job title, such as "BCT analysts" is understood to denote a group whose members fill the role expressed by the job title ("analyst").

Noun phrases that name vehicles have their own peculiar structure that can include color, model year, make, model, and body style. For example, all are included in the phrase "his black 2010 Ford Escape SUV." The named-entity recognizers within GATE recognize colors, years, car companies, car models, and car body styles, and a special mapping rule relates each appropriately to the named entity.

If a movement event is modified by a movement preposition whose object is a location, then the location is understood to form the path of the movement. For example, in the sentence, "Dillinger was last seen *driving* his black 2010 Ford Escape SUV westward *down Indianapolis Road* at 1:20pm on 3/17/2013," Tractor understands that Indianapolis Road forms the path of the driving event. Moreover, because "westward" is a direction and an adverbial modifier of "driving," the direction along the path is understood to be westward.

If a search of some place uncovers some object, then the object was located in the searched place. For example, Tractor infers from "a search of his car netted IED devices" that the IED devices were located in the car.

A. *Understanding Noun-Noun Modification*

The modification of a noun by another noun is used to express a wide variety of semantic relations. However, certain cases are recognized by the mapping rules from the categories of the nouns. (Though exceptions might still occur.) If both nouns denote locations, then the location of the modifying noun is located within the location of the head noun. For example in "Rashid, Baghdad," Rashid is understood as a neighborhood within Baghdad.

However, buildings and other facilities are also locations. (One can be in or next to a building.) So if the head noun denotes a facility, then the facility is understood as being in the location of the modifying noun. For example, "Second District Courthouse" is interpreted as a courthouse located in the Second District.

If the modifying noun is a location, but the head noun is not, then the entity denoted by the head noun is understood as headquartered in the location expressed by the modifying noun. For example "A Baghdad company" is interpreted as a company headquartered in Baghdad.

If neither noun is a location, but both are proper nouns, then they are both assumed to be names of the denoted entity. For example, "Ahmad Mahmud" is interpreted as a person who has both "Ahmad" and "Mahmud" as names, as well as having the full name "Ahmad Mahmud."

If the head noun denotes a person and the modifying noun denotes the name of a religious group (recognized by the named-entity recognizers), then a mapping rule asserts that the person is a member of the religious group and has that religion. For example, "a Sunni munitions trafficker" is understood to be a munitions trafficker whose religion is Sunni and who is a member of the religious group whose name is "Sunni."

If both nouns denote groups, then the head noun is understood to denote a group that is a subgroup of the group denoted by the modifying noun. For example, “BCT analysts” is interpreted to denote a group of analysts all of whom are members of the organization named “BCT”.

If the modifying noun denotes an organization, but the head noun does not, then the entity denoted by the head noun is understood to be a member of the organization. For example “the ISG affiliate” is interpreted to be someone filling the role of affiliate within the organization named “ISG.”

B. Understanding Copulas

If there is a copula between a subject and a noun, then the subject is understood to be co-referential with an entity that is an instance of the category that the predicate noun denotes. For example, “the rented vehicle is a white van” is interpreted to mean that one entity is both a rented vehicle and a white van.

Typically, one end of a dimension has a linguistic value that can be used in neutral questions to ask what value some entity has on that dimension. For example, “How old is he?” is a neutral question about the person’s value on the age dimension without implying that the person is old, but “How young is he?” also suggests that the person is young. Similarly, “How tall is she?” is a neutral question, whereas “How short is she?” suggests that she is short. The neutral value can be used in a copula to say that the subject entity has that value on the implied scale, for example, “Dillinger is old,” is interpreted to mean that Dillinger’s age has the linguistic value “old,” but can also be modified by a specific value to indicate the value on the implied scale. For example, “Dillinger is 30 years old” is interpreted to mean that Dillinger’s value on the age attribute is 30 years. One wouldn’t normally say something like “Dillinger is 20 years young,” or “Betty is 5 feet short.”

Predicate adjectives that do not imply a specific attribute dimension are interpreted as simple properties of the subject. For example “he is secretive” is interpreted to mean that he has the property “secretive”, and “he is apolitical” is interpreted to mean that he has the property “apolitical.”

C. Making Inferences

The ontology includes a category of symmetric relations so that a particular representational scheme can be used for them [23]. For example, “match” is symmetric, so the relation expressed in the sentence “The trigger devices netted in the arrest of Dhanun Ahmad Mahmud Ahmad on 01/27/10 match materials found in the truck of arrested ISG affiliate Abdul Wahied.” is represented in such a way that both “the devices match the materials” and “the materials match the devices” are represented. (That is, they match each other.)

Concrete participants in an act performed at some location were at that location at the time of the act. For example, “Ahmad Mahmud was arrested at Expressway on 20100127” is understood to imply that Ahmad was located at the Expressway on 20100127.

If someone drives a vehicle at some time, then the vehicle is not only the object of the driving, it is also the location of the driver at that time. For example, in the sentence, “Dillinger was last seen driving his black 2010 Ford Escape SUV westward down Indianapolis Road at 1:20pm on 3/17/2013,” Dillinger is understood both to be the driver of the SUV and to be located in the SUV at 1320 on 20130317.

The location relation is transitive. So, when interpreting the above sentence, Tractor understands that Dillinger was on the Indianapolis Road at 1320 on 20130317. The subgroup relation is also transitive. So if Ahmad is a member of one group that is a subgroup of another group, then Ahmad is a member of both groups.

VII. Evaluation

A recent study [3] found that about 62% of the syntax-semantics mapping rules fired on a test corpus of messages that were part of the SYNCOIN dataset, but were not used in developing the rules. The conclusion was that the rules then in use were “reasonably general.” This generality is largely due to the rules firing based on the categories, VerbNet classes, and WordNet synsets in the messages, rather than on specific words.

The same study found that the syntactic KBs for the test corpus were almost 70% syntactic, whereas the final semantic KBs were nearly 99% semantic (about 91% not counting the semantic assertions added by CBIR). The conclusion was that the rules then in place were “converting a large part of the syntactic information into semantic information, and doing so in a way that generalizes from the training sets to test sets.”

We have developed a “grading rubric” to measure the correctness and completeness of the semantic KBs produced by Tractor against manually produced “gold standard” semantic KBs.

For semantic analysis of natural language messages, the notion of “ground truth” does not apply, because regardless of the actual situation being described in the message, if the writer of the message described the situation poorly, no one would be able to reconstruct the situation from the poor description. Instead, the correctness of the system should be judged by comparing its performance to a human’s performance on the same task. First, a human, or group of humans, produces an *answer key* to serve as a “gold standard.” The system, or, in fact, any “performer” to be graded, must produce an answer submission that can be compared to the answer key. Finally, a “grader” compares the performers submission to the answer key.

Understanding a message is demonstrated by producing the following entries in the answer key or answer submission:

1) A list of the entities and events mentioned in the message, and the categories they are instances of. If several mentions co-refer to a single entity or event, that entity or event is listed only once.
2) A list of attributes of the entities and events. Entity attributes include the name, sex, and height of a person. Event attributes include the date, time, and location of the event.
3) A list of relations among the entities and events. Examples of relations are: one entity is located in some other entity (which must be some location); the agent of some action-event is some person.

Grading involves comparing the entries in the answer key to the submitted answers and judging when they agree. We call the entries in the answer key expected entries, and the entries in the submission found entries. An expected entry might or might not be found. A found entry might or might not be expected. However, a found entry might still be correct even if it wasnt expected. For example, some messages in our corpus explicitly give the MGRS coordinates of some event or location, and MGRS coordinates are also found in the NGA GeoNet Names Server database and added to the KB. If MGRS coordinates were not in the message, but were added, they would not have been expected, but may still have been correct. The grade depends on the following counts: $a =$ the number of expected entries; $b =$ the number of expected entries that were found; $c =$ the number of found entries; $d =$ the number of found entries that were expected or otherwise correct. These counts are combined into evaluation measures adapted from the field of Information Retrieval: $R = b/a$, Recall, the fraction of expected entries that were found; $P = d/c$, Precision, the fraction of found entries that were expected or otherwise correct; and $F = 2RP/(R+P)$, the harmonic mean of R and P. R, P, and F are all interesting, but F can be used as a summary grade.

The Sunni Criminal Thread (part of SYNCOIN) was used for development and evaluation of Tractor. The message set contains 114 messages, of which 37 were used for development, and the remaining 77 for evaluation. The only deviation from a standard development/evaluation set methodology is that we used the evaluation set in a cursory manner to ensure coverage of the additional ontological classifications in the syntax-semantics mapper. Where the development set was examined in close detail in the development of mapping rules, the evaluation set was not. Manual co-referencing was used for both datasets. Grading was done by a student not involved with the development of Tractor.

Table I presents the results of running Tractor on the Sunni Criminal Thread development and test sets. We found that precision, recall, and f–measure were very similar for the two datasets. This reinforces earlier results which indicate that Tractor generalizes well across datasets within the domain. Given the relatively high precision and recall scores, we can now also conclude that the semantic understanding by Tractor is quite good. Most of the semantic content of the messages is understood, and few mistakes are made in the understanding.

TABLE I. RESULTS OF GRADING TRACTOR ON BOTH DEVELOPMENT AND TEST SETS.

Set	Count	R		P		F	
		Avg	StDev	Avg	StDev	Avg	StDev
Development	37	0.83	0.09	0.82	0.10	0.82	0.08
Test	77	0.83	0.09	0.83	0.09	0.83	0.08

VIII. CONCLUSIONS

Understanding a natural language text involves identifying the entities and events mentioned in the text, and also making explicit the attributes of those entities and events, and the relations among them. Doing this requires the use of background knowledge—knowledge that is not explicitly included in the text. Background knowledge includes: knowledge of how the natural language is used; knowledge of the world; knowledge of the domain being discussed in the text; and knowledge of the axioms of the relations that are used (explicitly or implicitly) in the text. We have discussed specific examples of how background knowledge is used by Tractor, a system for understanding English messages within the context of hard and soft information fusion for situation assessment.

Examples of the use of knowledge of how English is used include the analysis of various noun-noun modification, and inferring the correct attribute when a noun is modfied by a graded descriptive adjective.

Examples of the use of knowledge of the world include the addition of geographic information, and the fact that the driver of a vehicle is located in the vehicle.

Examples of the use of knowledge of the domain being discussed are the inclusion of named entities mentioned in the datasets in the lists used by the named-entity recognizers, and the focus of the NGA GEOnet Names Server database on names of places in Iraq.

Examples of the use of knowledge of the axioms of the relations used in the text are the use of the transitivity of the is-located-at relation and the subgroup relation.

Use of this background knowledge allows Tractor to explicate the attributes of entities and events it identifies in the texts, and the relations among them. Evaluation has shown that the vast majority of these are identified correctly, with few mistakes.

ACKNOWLEDGMENTS

This work has been supported by a Multidisciplinary University Research Initiative (MURI) grant (Number W911NF-09-1-0392) for "Unified Research on Network-based Hard/Soft Information Fusion", issued by the US Army Research Office (ARO) under the program management of Dr. John Lavery.

REFERENCES

[1] G. A. Gross, R. Nagi, K. Sambhoos, D. R. Schlegel, S. C. Shapiro, and G. Tauer, "Towards hard+soft data fusion: Processing architecture and implementation for the joint fusion and analysis of hard and soft intelligence data," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*. ISIF, 2012, pp. 955–962.

[2] M. Prentice, M. Kandefer, and S. C. Shapiro, "Tractor: A framework for soft information fusion," in *Proceedings of the 13th International Conference on Information Fusion (Fusion2010)*, 2010, p. Th3.2.2.

[3] S. C. Shapiro and D. R. Schlegel, "Natural language understanding for soft information fusion," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*. ISIF, 2013, 9 pages, unpaginated.

[4] J. Gómez-Romero, J. Garcia, M. Kandefer, J. Llinas, J. M. Molina, M. A. Patricio, M. Prentice, and S. C. Shapiro, "Strategies and techniques for use and exploitation of contextual information in high-level fusion architectures," in *Proceedings of the 13th International Conference on Information Fusion (Fusion 2010)*. ISIF, 2010.

[5] M. Kandefer and S. C. Shapiro, "Evaluating spreading activation for soft information fusion," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*. ISIF, 2011, pp. 498–505.

[6] M. Prentice and S. C. Shapiro, "Using propositional graphs for soft information fusion," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*. ISIF, 2011, pp. 522–528.

[7] G. A. Gross, K. Date, D. R. Schlegel, J. J. Corso, J. Llinas, R. Nagi, and S. C. Shapiro, "Systemic test and evaluation of a hard+soft information fusion framework: Challenges and current approaches," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*. ISIF, 2014, 8 pages, unpaginated.

[8] A. B. Poore, S. Lu, and B. J. Suchomel, "Data association using multiple frame assignments," in *Handbook of Multisensor Data Fusion*, 2nd ed., M. Liggins, D. Hall, and J. Llinas, Eds. CRC Press, 2009, ch. 13, pp. 299–318.

[9] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, N. Aswani, I. Roberts, G. Gorrell, A. Funk, A. Roberts, D. Damljanovic, T. Heitz, M. A. Greenwood, H. Saggion, J. Petrak, Y. Li, and W. Peters, *Text Processing with GATE (Version 6)*. The University of Sheffield, Department of Computer Science, 2011. [Online]. Available: http://tinyurl.com/gatebook

[10] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications," in *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.

[11] S. C. Shapiro and W. J. Rapaport, "The SNePS family," *Computers & Mathematics with Applications*, vol. 23, no. 2–5, pp. 243–275, January–March 1992, reprinted in [24, pp. 243–275].

[12] S. C. Shapiro, "An introduction to SNePS 3," in *Conceptual Structures: Logical, Linguistic, and Computational Issues*, ser. Lecture Notes in Artificial Intelligence, B. Ganter and G. W. Mineau, Eds. Berlin: Springer-Verlag, 2000, vol. 1867, pp. 510–524.

[13] D. R. Schlegel and S. C. Shapiro, "Visually interacting with a knowledge base using frames, logic, and propositional graphs," in *Graph Structures for Knowledge Representation and Reasoning*, ser. Lecture Notes in Artificial Intelligence, M. Croitoru, S. Rudolph, N. Wilson, J. Howse, and O. Corby, Eds. Berlin: Springer-Verlag, 2012, vol. 7205, pp. 188–207.

[14] M. Kandefer and S. C. Shapiro, "An F-measure for context-based information retrieval," in *Commonsense 2009: Proceedings of the Ninth International Symposium on Logical Formalizations of Commonsense Reasoning*, G. Lakemeyer, L. Morgenstern, and M.-A. Williams, Eds. Toronto, CA: The Fields Institute, 2009, pp. 79–84.

[15] J. L. Graham, "A new synthetic dataset for evaluating soft and hard fusion algorithms," in *Proceedings of the SPIE Defense, Security, and Sensing Symposium: Defense Transformation and Net-Centric Systems 2011*, 2011, pp. 25–29.

[16] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2nd ed. Upper Saddle River, NJ: Pearson Education, Inc., 2009.

[17] Princeton University, "About WordNet," 2010, http://wordnet.princeton.edu.

[18] M. Palmer, "VerbNet: a class-based verb lexicon," http://verbs.colorado.edu/~mpalmer/projects/verbnet.html.

[19] B. Levin, *English verb classes and alternations: A preliminary investigation*. University of Chicago press, 1993.

[20] National Geospatial-Intelligence Agency, "NGA GEOnet Names Server," 2014, http://earth-info.nga.mil/gns/html/.

[21] National Aeronautics and Space Administration, "NASA World Wind," 2011, http://worldwind.arc.nasa.gov/java/.

[22] K. J. Miller, "Modifiers in WordNet," in *WordNet: An Electronic Lexical Database*, C. Fellbaum, Ed. Cambridge, MA: MIT Press, 1998, pp. 47–67.

[23] S. C. Shapiro, "Symmetric relations, intensional individuals, and variable binding," *Proceedings of the IEEE*, vol. 74, no. 10, pp. 1354–1363, October 1986.

[24] F. Lehmann, Ed., *Semantic Networks in Artificial Intelligence*. Oxford: Pergamon Press, 1992.