SCUBA: An Agent-Based Ontology Creation and Alignment Method for Socio-Cultural Modeling

Donald R. Kretz
Raytheon IIS
Garland, TX
donald_r_kretz@raytheon.com

William D. Phillips
Raytheon NCS
Largo, FL
bill.phillips@raytheon.com

Bruce E. Peoples
Raytheon IIS
State College, PA
bruce_e_peoples@raytheon.com

***Abstract* -- An otherwise promising business, political, or military strategy can be crippled by an incomplete understanding of the social-cultural factors that define and influence a region. Such omissions are sometimes due to oversight, but often stem from a fundamental lack of understanding of how to model such difficult and unfamiliar concepts. The information required to generate useful contextual models is typically available but vast, and manual interpretation of detailed text is time-consuming, highly subjective, and requires specialized skills. The SCUBA project achieved a balanced human-computer modeling paradigm to 1) automate the creation of social and cultural ontologies from selected source materials using previously-developed tools, 2) apply a variety of nominal, semantic, structural, and statistical matching techniques to align multiple ontologies using an agent-based multimodel, and 3) evaluate the effectiveness of the generation and alignment processes using precision, recall, and various other measures of effectiveness. Preliminary results of our initial agent-based experiments were promising – by applying ensembles of multiple matching techniques, we achieved significant improvements in alignment F-scores and other measures of performance while dramatically reducing the amount of time required to manually produce coordinated, useful domain models.**

Keywords-ontology; ontology alignment; social ontology; cultural ontology; ensemble alignment; agent-based alignment

I. INTRODUCTION

Having an incomplete understanding of the social-cultural factors that define and influence a region can cripple an otherwise promising business, political, or military strategy. Too often, models that guide strategy development and operational planning do not include critical social and cultural elements. These omissions can be blamed partly on oversight, but often stem from a fundamental lack of understanding of how to model such difficult and unfamiliar concepts. The information required to generate useful contextual models is typically available but is often distributed across vast repositories. Furthermore, the manual interpretation of detailed text is time-consuming, highly subjective, and requires specialized skills. We believe that socio-cultural awareness is best achieved by a system that combines multiple information sources using a variety of automated extraction, mediation, and analysis tools, but guided by a human knowledge engineer in an interactive paradigm called *balanced cooperative modeling* [1].

We apply *ontology* as our modeling method of choice. An ontology can conceptualize a complex domain in a way that both humans and machines can understand, but the use of ontology in this context presents us with two important challenges. First, manual ontology creation is a time-consuming and highly subjective process, particularly when attempting to model abstract social and cultural concepts. While formal models are required to conform to strict rules involving provable logic and model consistency, they will always incorporate some amount of bias. Every human modeler will have a slightly different perspective of the same small part of the world, and will make different value judgments about what parts are important and how those parts interrelate. Striving for added richness by adding more information only complicates this problem and adds to the severity of the "knowledge acquisition bottleneck" [2]. We believe, therefore, that by applying automated *ontology generation* against various corpora of domain-relevant materials, we can generate a useful first approximation of a domain model. An automated generation process will "learn" from the information it can "study". The model it constructs will, therefore, be representative of the "world" described in the input material it receives.

The second challenge involves the alignment of multiple models. Accommodating multiple domain ontologies is usually necessary to capture the complexities of domains having socio-cultural dimensions and to leverage existing models. There has been a loosely-associated body of work in this area that we collected under the general heading of "*ontology alignment theory*". Our interpretation of this theory is essentially built on the principle of approximation – because any ontology is an approximate representation of its real-world domain, generating and aligning multiple ontologies that all represent the same domain yields a richer higher-order approximation of the real world (i.e., removes some of the subjectivity or bias associated with applying a single model).

As described by Euzenat and Shvaiko [3], the *matching operation* accepts ontologies as inputs, and produces an ontology as its output (see Figure 1). The input ontologies (O_1 and O_2) are independent domain ontologies, perhaps derived from different sources of information or developed by different ontology engineers. Optionally, a third ontology (Ω) may be included as input – this may be an upper ontology or may be the composite ontology (or alignment) produced by a

The authors are grateful to have received funding for SCUBA under the 2010 Raytheon Innovation Challenge.

previous matching operation. The latter case suggests that matching operations can be chained for continued refinement by feeding the output from one operation (i.e., an aligned ontology) as input to the next matching process.

$$\Omega' = f(O_1, O_2, \Omega, p, r)$$

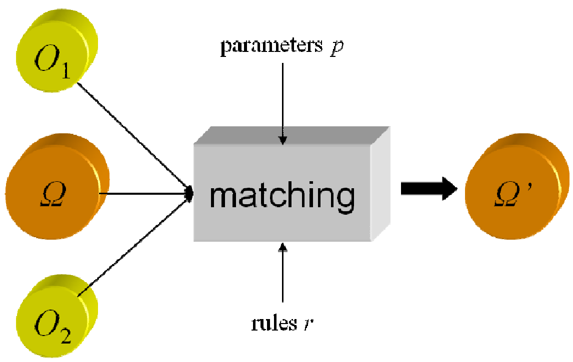

Figure 1. Matching operation, from Euzenat and Shvaiko [3]

In addition to the ontologies, a couple of additional inputs are provided. First, a set of rules directs the matcher to perform certain types of comparisons (i.e., which entities or attributes to compare, what sort of comparison to make, etc.). These rules are derived from a set of basic matching techniques described later in this paper. To accompany the rules, a set of parameters informs the matcher what limits or constraints to impose on the rules. For example, a rule might cause a name similarity technique to be performed using a fuzzy string comparison on a "name" property, but a parameter might indicate that only values having a confidence value higher than 50% are to be considered a match. Finally, the result of this operation is an ontology, referred to as *Ω prime*, that expresses the set of correspondences between the entities in O_1 and O_2.

Considering the points made above, we set two primary goals for our project: *Eliminate the knowledge acquisition bottleneck through semantic parsing and extraction of domain concepts from data sources into multiple ontologies and contexts,* and *bridge the gap between multiple, heterogeneous generated ontologies and a single domain ontology.* In our initial phase, we chose to apply the previous work of others in generating ontologies from text using readily available tools (citations to follow). Our investigation, instead, focused on the effective alignment of ontologies through various techniques of mapping. Here, too, we borrowed from the work of others for specific techniques and algorithms (citations to follow). However, we began with the premise that ontologies have characteristics that make them more or less suited for effective alignment with certain other ontologies. Rather than approach the problem using a single technique or by applying complex n-way comparisons, we formulated three key observations that guided our efforts:

Observation #1: Certain pairs of ontologies are more effectively aligned with one another than with other ontologies

Observation #2: Certain matching techniques produce more useful alignments for certain ontology pairs than others

Observation #3: The selection of candidate ontology pairings and matching techniques can be guided by heuristics and aided by the inspection of model metacharacteristics

This paper presents SCUBA, an agent-based framework for ontology alignment based on the observations stated above. We will describe the methodology we applied, as well as provide some initial results.

II. Methodology

The objective of SCUBA was to develop a balanced human-computer modeling paradigm to 1) automate the creation of social and cultural ontologies from selected source materials, 2) apply a variety of nominal, semantic, structural, and statistical matching techniques to align multiple ontologies, and 3) evaluate the effectiveness of the generation and alignment processes. Since our work was mainly focused on the alignment framework, we will concentrate most of this section on that effort.

A. Ontology Generation

In answering the challenge of ontology generation, we relied on the groundbreaking work of a number of others, particularly Maedche and Volz[4] and Cimiano and Völker [5]. We used the common academic ontology generator Text2Onto [6] to generate ontologies from bodies of text we obtained from various sources, including the Yale University Human Relations Area Files (HRAF) [7], Yale University's Outline of Cultural Materials (OCM) [8], the United Nations Development Programme – Human Development Reports [9] and others. Documents were clustered by geographic area, and a separate ontology was generated for each area. The team also generated 95 separate ontologies utilizing the 54 Cultural and 41 Social text files obtained from open source materials. All ontologies were created in the Web Ontology Language (OWL) format.

Additionally, the team manually generated a set of "gold standard" ontologies to compare with the automatically generated models. Seven ontologies were created based on the Department of Defense (DoD) PMESII-PT paradigm (Political, Military, Economic, Social, Information, Infrastructure, Physical Environment, Time) [10] using Protégé [11]. An ontology for Time was not created. To provide instance data for the PMESII-P ontologies, the team developed a method to automate the merging of Yale HRAF instance data with Yale OCM codes in the developed ontologies, saving weeks of manual labor.

B. Ontology Alignment

1) General Approach

As stated, we focused most of our work on ontology alignment. We again borrowed heavily from the body of prior research in specific ontology matching techniques, most of which were collected and documented by Euzenat and Shvaiko [3]. In order to investigate our own hypotheses, however, we constructed a customized agent-based framework using the Java Agent DEvelopment Framework (JADE) [12]. We used agents to develop automated workflows for the two main component processes: selecting the optimal set of alignment candidates and most promising match techniques, and performing the matching operation by applying the rules to the alignment candidates (Figure 2). Specific match techniques were encoded as composable sets of agent behaviors. An agent-oriented design allowed us to apply a technique known as "ensemble forecasting", which is common in highly specialized domains such as weather prediction. Yilmaz [14] refers to this idea as a *multimodel*, or a set of component models that, together, define the behavior of a more complex process. Using ensemble forecasting or multimodeling, various combinations of matching algorithms ("behaviors") were applied against concept pairs, then evaluated in order to determine the strength of the match. An average, or *ensemble mean*, of the different behaviors inspired greater confidence because it essentially smoothed the performance peaks and troughs introduced by model imperfections or context sensitivities. For example, the concepts "car" and "automobile" produce very low results for all name-based match behaviors, but semantic match behaviors rate them as nearly identical. Hence, while any one technique for matching two concepts is inherently unreliable, an ensemble mean that accounts for the strengths and weaknesses of all match techniques yielded a higher-confidence correlation. The matches can be used to produce a merged ontology in any format desired; e.g., a set of OWL assertions (i.e., "sameAs" or equivalentClass") between matched concepts, or Semantic Web Rule Language (SWRL) rules to bridge the aligned models.

Figure 2. Primary SCUBA workflows

2) *Agents, Behaviors, and Ensembles*

In the SCUBA framework, a community of agents interacts to perform the high-level operations of candidate selection and ontology matching. Each agent determines the types of behaviors it needs in order to perform its current task, and loads them dynamically. Agents serve in a variety of roles:

- OA - *Ontology Agent*: perform as a proxy for an ontology by mediating access to its concepts as well as responding to inquiries about its metacharacteristics (e.g., depth, breadth, number of concepts, etc.).
- EA - *Evaluation Agent*: make a judgment as to the relatedness of available ontologies along some relevant dimension (e.g., domain relevance, semantic similarity, etc.).
- HA - *Heuristic Agent*: determine which ontology pairs make good candidates for matching, which matching behaviors should be applied, and manage the execution of selection and matching workflows.
- MA - *Matching Agent*: creates mappings of the concepts and relationship types between two ontologies.
- SA - *Similarity Agent*: calculates the similarity between concepts.
- UA - *Utility Agent*: performs supporting tasks such as data and ontology storage/retrieval, job ID management, etc.

Each matching algorithm or technique was implemented as a behavior. In JADE parlance, a behavior is a set of actions to be performed. Coding each set of actions in a separate component, rather than in the agent itself, allowed each agent to select and compose the behaviors it wished to use to complete an assigned task. There are many techniques for performing the matching operation, and some are better at matching certain types of entities and properties than others. Furthermore, a better match might result in some cases if more than one technique is applied at the same time ("matcher composition").

- Name-based ("terminological") techniques compute some measure of similarity based on strings containing names, descriptions, comments, etc. Comparisons based on simple or fuzzy string comparisons would match "George Bush" with "George Bush", "George W. Bush", or "G. W. Bush". Matching can also be performed using synonyms ("newspaper" matches "periodical") or other language-based methods like lemmatization, which would match houses to house, mice to mouse, etc.
- Semantic techniques rely on deductive methods to justify their matching results. A semantic model could contain a very rich set of relations, with inferred associations between ontologies. For example, "brain injury" and "head injury" might be inferred to be synonymous based on the fact that a "brain" is "part-of" a "head".
- Structural techniques take into account an entity's attributes or properties, as well as other related entities, when performing a match. For example, a constraint-based rule would match "Book" and "Volume" if each contained the key properties of author, year, publisher, and title. Similarly, a graph-based rule would match "Book" and "Volume" if the two concepts had the same (or similar) subclasses, like "Novel", "Textbook", and "Children's".
- Extensional techniques are applied not to concepts, but to instances. Typically applied when other techniques contain little name or structure overlap, these techniques entail matching two concepts based on their membership; i.e., the objects that belong to

each particular class. For example, book titles are unique enough that, with some estimable probability, two instances having same title or label are likely to be the same object. If the object is classified differently in two separate ontologies, a match between concepts then becomes possible.

3) Workflow Heuristics

Heuristics are encoded inside an Heuristic Agent and govern the selection and matching workflows. Many such heuristics can be encoded simultaneously in one or many agents, and a single HA can construct complex heuristic workflows from multiple matching behaviors chosen from different categories. For instructional purposes, the following example is used throughout the rest of this section to describe what happens in each step of the process:

Agent: HA01
Behavior: MinDepth-MaxDepth
Other agents: EA01, OA1-OA$_n$, MA01, SA01-SA03
Behavior Description: Inspect each ontology for its depth. As candidates, choose the ontology with the minimum depth to be matched with the ontology having the maximum depth. Perform an alignment of the two candidates using an average of all available name-based and semantic matching techniques. Evaluate the results using an F-score statistic.

In the example, HA01 performs candidate selection by directing EA01 to evaluate ontologies according to their depth. EA01 requests a depth statistic from each of the OAs, and reports the results – the ontologies having the least and greatest depths – back to HA01. The candidate ontologies have now been identified, and the first phase is complete. HA01 then moves into the ontology matching phase. The agent directs MA01 to match the selected ontologies using all of the name-based and semantic behaviors. MA01 manages the next level of orchestration, directing a set of SAs to perform an alignment, assigning each to use one of the specific matching behaviors. For example, if there are defined behaviors for Levenschtein distance (name-based), Jaro-Winkler (name-based), and WordNet similarity (semantic), the MA tasks three SAs – one per behavior - to align the concepts in the candidate ontologies and record the results of their work. MA01 then computes the ensemble mean and reports its result back to HA01. Once the match process is complete, other components can refer to the scores in order to produce a number of possible outcomes: a merged ontology, a set of rules mapping pairs of similar concepts, ontology entries reflecting class equivalencies, etc.

4) Workflow 1: Candidate Selection

All available ontologies are evaluated and compared according to a subset of predetermined set of criteria (e.g., depth, breadth, domain relevance, number of concepts, etc.). From this observation, the most suitable pairs are selected for alignment. Additionally, matching techniques are chosen to maximize the effectiveness of the alignment process for the types of ontologies chosen as candidates. Figure 3 describes the roles of agents and behaviors in the candidate selection process.

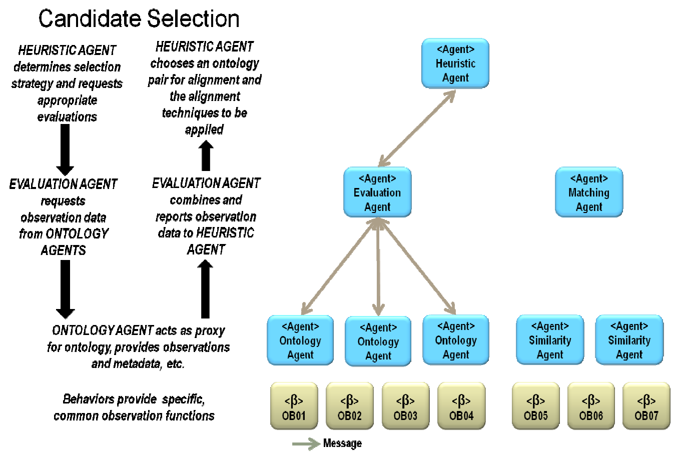

Figure 3. Candidate selection workflow

5) Workflow 2: Ontology Matching

Once the candidate ontologies and techniques are chosen for alignment, the matching process is carried out using the agents and behaviors described above. Figure 4 illustrates the ontology matching process.

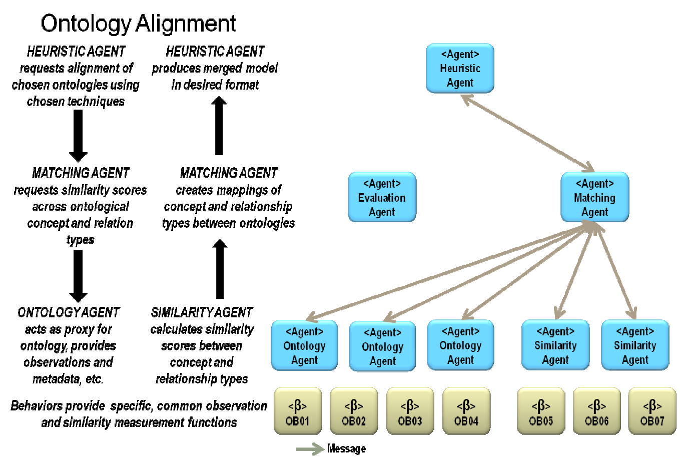

Figure 4. Ontology alignment workflow

6) Scoring and Evaluating

All of the concept matching results were recorded in a database for later use. The entries included the two concepts being matched and their ontologies of origin, the behavior used to perform the match, and the similarity score that was normalized to range between -1.0 (known to be different) and 1.0 (known to be the same). A score of 0.0 indicated complete uncertainty. The scoring results were used to compute the ensemble mean over each discrete concept match (i.e., by averaging the scores of all behaviors that were applied to each of the match pairs).

C. Demonstration

A military planning exercise was chosen as a scenario to demonstrate SCUBA, since this type of event is typically a time consuming, manual, and ad hoc process that can take hours to days depending on size of the mission and echelon of command. War planners skim through available classified sources of information such as Signal Intelligence (SIGINT), Communications Intelligence (COMINT), and Human Intelligence (HUMINT), but typically spend the majority of their effort analyzing Open Source (OSINT) or simply searching the Internet[1]. As a result, critical information and cross relationships between pieces of information are commonly missed due to time constraints and the limits of human processing ability. Compounding the difficulty of the research effort, the number of data sources is necessarily limited by time and staff and not all information may be up-to-date.

1) *Military Planning Scenario & Decision Making Model*

The Military Decision Making Process (MDMP) Model [13] is a standardized mission planning and decision making model used by the US Army and combatant commands (COCOMs) to support counterinsurgency operations (COIN). The formal tactical planning process of counterinsurgency operations is performed by the commander's staff utilizing the MDMP model. In plain language, MDMP identifies the problem, develops solutions, compares alternatives, and recommends a best decision to the commander.

a) *Mission Analysis*

Mission analysis is crucial to the MDMP. It allows the commander to begin the battlefield visualization. The outcome of mission analysis is a tactical problem definition that feeds the process of determining feasible solutions. Mission Analysis consists of 17 steps, not necessarily sequential, and results in a formal staff briefing to the commander. Figure 5 depicts the breakdown of the MDMP model and green shading is used to highlight the relevant steps for the SCUBA demonstration.

b) *Initial Intelligence Preparation of the Battlefield (IPB)*

IPB is a systematic, continuous process of analyzing the threat and the effects of the environment on the unit. It identifies facts and assumptions that determine likely threat COAs. The IPB supports the commander and staff and is essential to estimates and decision making. It provides the basis for intelligence collection and synchronization to support COA development and analysis. Furthermore, it is a dynamic process that continually integrates new intelligence information.

IPB defines the battlefield or operational environment in order to identify the characteristics of the environment that influence friendly and threat operations, help determine the area of interest, and identify gaps in current intelligence. IPB describes the battlefield's effects, including the evaluation of all aspects of the environment with which both sides must contend, to include terrain and weather and any infrastructure and demographics in the area of operations (AO). IPB evaluates the threat by analyzing current intelligence to determine how the threat normally organizes for combat and conducts operations under similar circumstances.

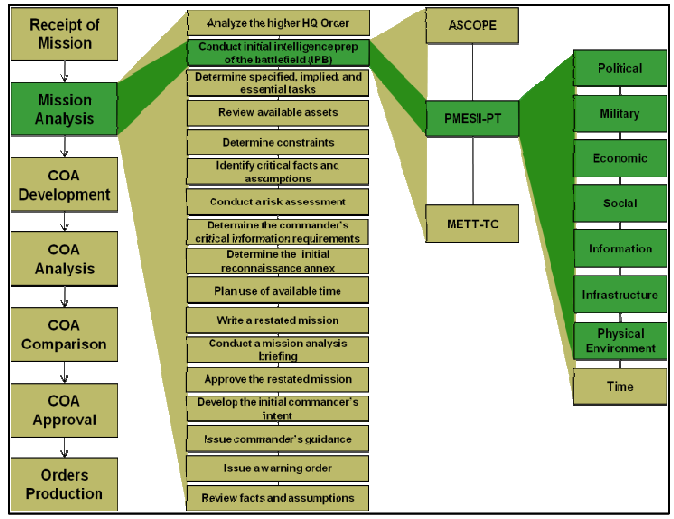

Figure 5. Military Decision Making Process Model

c) *PMESII Ontology Structure*

PMESII-PT, or for brevity PMESII, is a framework used to describe and understand the operating environment [10]. PMESII provides structure to the IPB process, and facilitates the organization of facts and assumptions about actors operating in an AO. Each letter in the PMESII acronym corresponds to a specific variable of interest to the war planner: P – Political, M – Military, E – Economic, S – Social, I – Information, I – Infrastructure, P – Physical Environment, and T – Time.

2) *Military Planning Using SCUBA*

As illustrated in Figure 5, the eighth variable, Time, was not modeled in this scenario since the time element was already embedded in the instance data populating the other variables. The DoD currently uses an expansion of the PMESII model that includes about 60 sub-categories. By merging PMESII with the Yale OCM model the SCUBA team extended the level of fidelity to approximately 900 super class and class concepts providing much greater model fidelity. An example of this expansion for the Social PMESII variable can be seen in blue in Figure 6.

When SCUBA executes, it ingests data from text documents, extracts domain relevant concepts, and links those concepts both vertically within individual PMESII variables and horizontally across the PMESII model. Instance (source) data is connected to each concept, which allows later review by the war planner or intelligence analyst. The main advantage of this paradigm is that instead of a planning staff performing manual keyword search queries across a variety of databases, a single lookup within SCUBA will provide the analyst or operational planner with all relevant information on a historic, social, or cultural topic of interest.

---

[1] This process was described to the SCUBA team during a December 2010 visit to the Joint Operations Center at US Central Command Headquarters, MacDill AFB, FL.

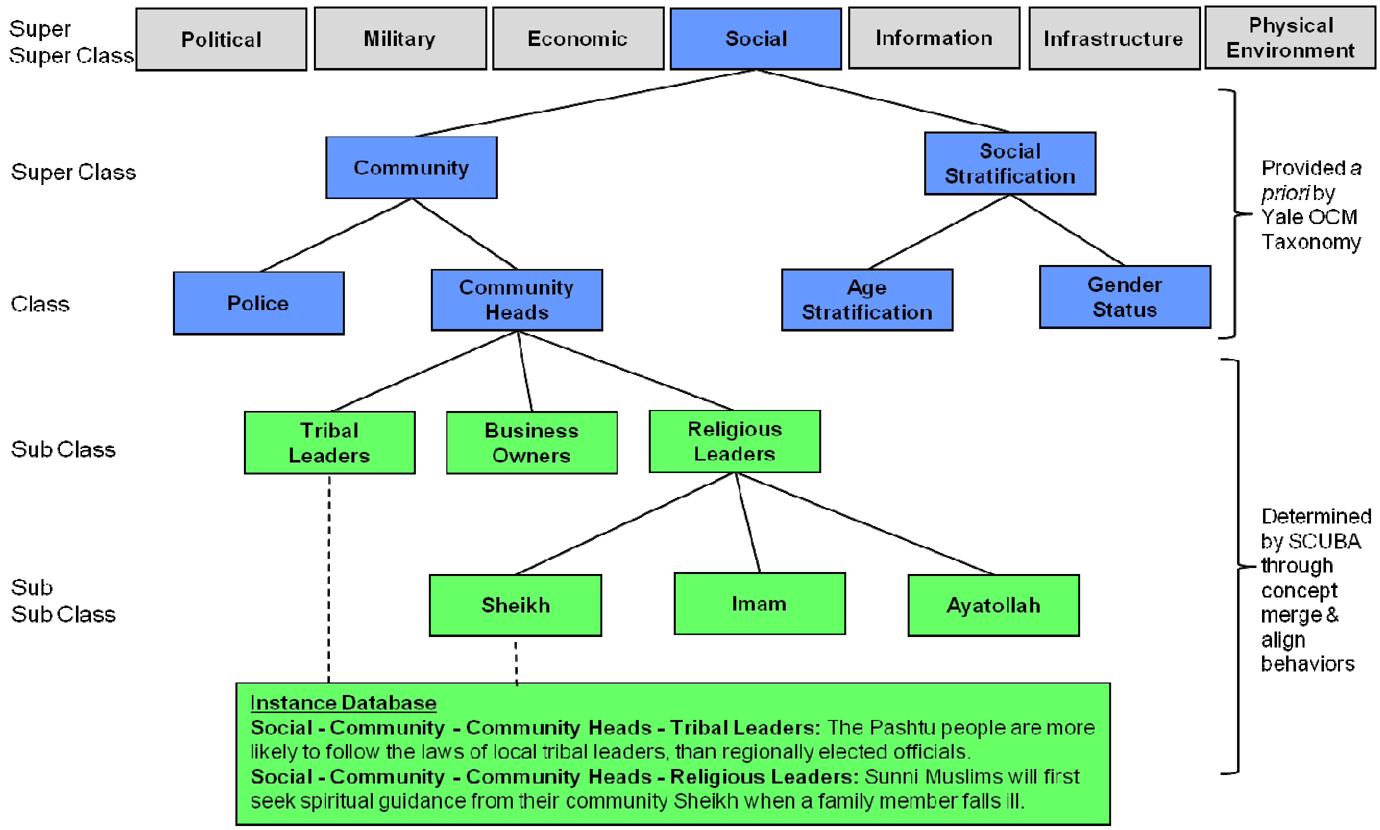

Figure 6. PMESII ontology structure

While the focus of SCUBA is in the socio-cultural domain, an expanded PMESII model was created in order to demonstrate horizontal relationships between variables. The result is a major improvement over existing systems that are highly specialized and restricted in scope. Additionally, when this same effort is performed manually, the PMESII variables are commonly divided between staff officers. This practice produces information stovepipes and complicates the task of identifying cross-variable effects. In contrast, SCUBA facilitates such understanding.

Continuing with the example in Figure 6, the concept class hierarchy in green are those identified and created by SCUBA. Notice that in addition to aligning similar concepts, SCUBA can create class hierarchies, merge similar concepts into a single class, and link original instance data to each relevant concept.

A small portion of the merged PMESII Ontology generated by SCUBA using open source socio-cultural information of Afghanistan was displayed in Raytheon's hyperbolic semantic graph tool and is shown in Figure 7. Notice the equivalence relationship identified between "Military Organization" and "Militia". Also, "Districts" in one ontology was aligned with "District" (no 's') in another ontology. This was all performed automatically by the SCUBA agents and behaviors with no human in-the-loop. In the case of Districts/District, the SCUBA heuristic relied on structural matching techniques. The match occurring between Military Organization and Militia was a combination of structural and semantic matching. The remainder of the figure illustrates multiple concepts arising from a single paragraph: Military Organization, Districts, and Police, as well as, additional instance data on each of those concepts arising from other source material.

III. RESULTS

The team identified dozens of possible evaluation metrics, many of which were used in the candidate selection process. As an overall measure of effectiveness, however, we report our results in terms of F-scores using the formulas below.

$$F = 2 * \frac{precision * recall}{precision + recall} \quad (1)$$

$$precision = \frac{tp}{tp + fp} \quad (2)$$

$$recall = \frac{tp}{tp + fn} \quad (3)$$

The F-score is a measure of a test's accuracy which considers both the precision ("exactness") and recall ("completeness") of the test. In the models we chose for testing, over 60,000 comparisons were made between

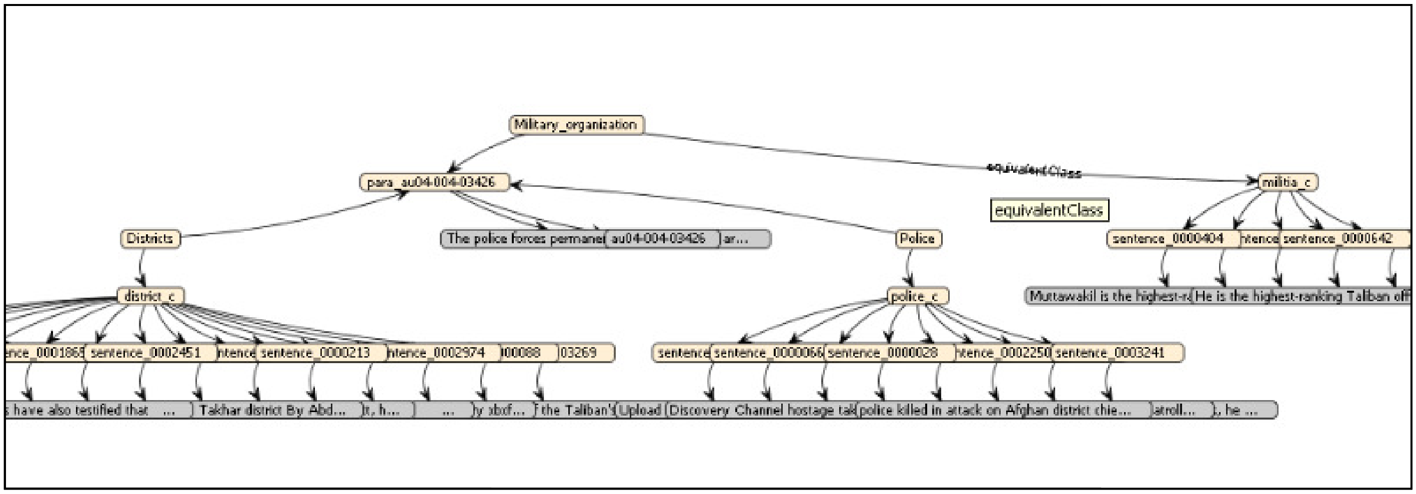

Figure 7. PMESII ontology structure in hyperbolic browser

concepts. Using name-based comparison alone, precision was typically high (~90%), while recall was much lower (~20 - 30%). Because the name-based approach suffered from a high number of false negatives, the F-scores averaged only ~40% (see Figure 8). However, when semantic matching was combined with name-based matching, there was a dramatic reduction of false negatives - this resulted in significant increase in recall (~80%) and brought the average F-score to above 80% (all differences were significant) – see Figure 9. Even greater improvement is expected when additional behaviors are added. Based on these results, we are encouraged by the prospect of evolving information alignment and interoperability from a manual, costly chore to an effective semi-automated process.

A. *Measures of Performance*

In order to determine whether the automated align-and-merge methodology defined by SCUBA demonstrated any improvement over existing ontology generation tools alone, the SCUBA merged ontology was compared against an ontology generated using Text2Onto [5][6]. In both cases, the same data set and initial taxonomy were used. The comparison was made across 12 measures of performance (MOPs) that fall within three general measurement dimensions: Structural, Usability, and Timeliness.

Structural MOPs

- Measure of Concept Count – Total number of concepts in the ontology.
- Measure of Concept Instance Count – Number of linked paragraph instances over all concepts.
- Measure of Relationship Type Count – Total number of unique relationships in ontology, i.e. 'is a part of', 'is equivalent to', etc.
- Measure of Relation Instance Count – Number of relationship links between concepts.
- Measure of Maximum Depth – Levels of concept hierarchy within the ontology.
- Measure of Degree Centrality - Measure used often in social network theory - average number of relationships linked to each concept.

Usability MOPs

- User Recognition – Survey score indicating how similar ontology structure is with current models.
- Fitness for User – Survey score indicating how easy it is for the user to load and navigate among the concepts in the ontology

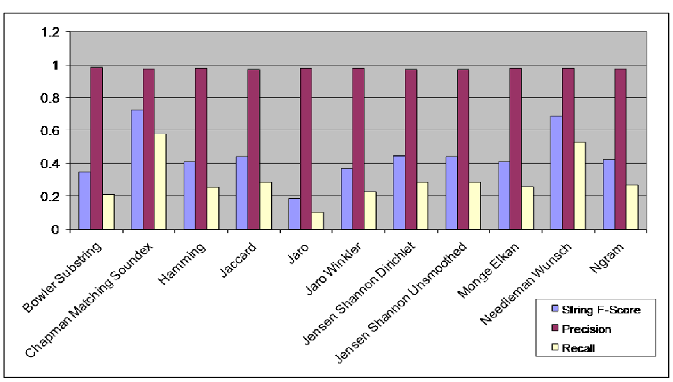

Figure 8. Results for string-only alignment

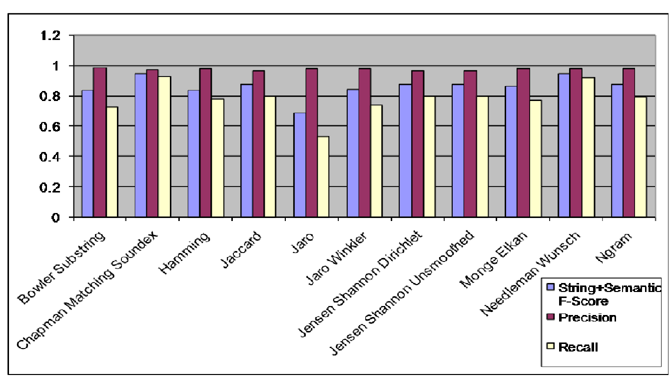

Figure 9. Results for String + Semantic Alignment

Timeliness MOPs

- Speed to Build Ontology – Time to create the ontologies.
- Time to Perform Alignment – Time to perform alignment between 2 or more ontologies.

Figure 10 depicts these MOPs for each of the ontologies. It is immediately clear that the Human Generated ontology was most recognizable with information in a format most easily used while the purely machine generated, Text2Onto ontology scored lowest in this area. SCUBA scored well in this area because it was based on the same PMESII model used by military planners. Conversely, the Human Generated ontology took longest to build and was much smaller than the faster and larger generated Text2Onto ontology. These results were in line with our expectations. What we intended to see was whether SCUBA could create ontologies that were at least as large/deep as those created by software algorithms or humans, but were richer and more usable similar to those generated by humans.

When breaking down the Speed to Build by individual algorithms, the SCUBA string-based matching agents and ontology behaviors were executed on par with Text2Onto, while the semantic matching agents took considerably more time to execute. This is understandable because the semantic algorithms are more complex with the purpose of determining additional positive matches through synonym, lemmatization, and morphological comparisons. This significantly improved the accuracy of the results, as boldly illustrated in Figure 9, however there is a corresponding increase in ontology generation run time. We believe this is reasonable (it is still significantly lower than the Human Generated ontology) and can be further reduced by adding computing resources.

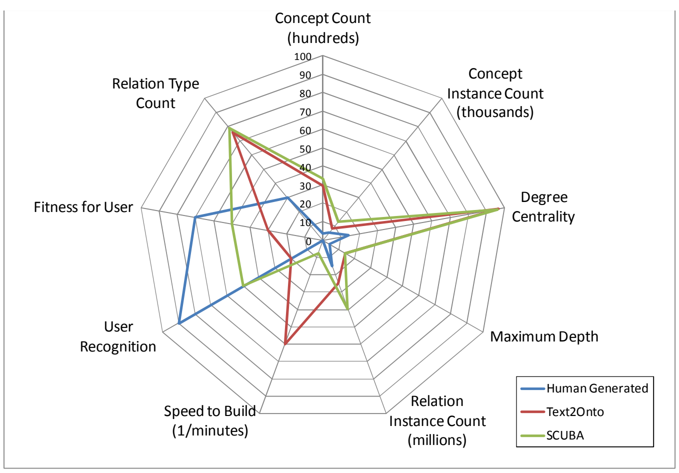

Figure 10. MOP comparison between Text2Onto and SCUBA

Regarding the other MOPs, SCUBA either met or exceeded the performance of Text2Onto. For example, for Concept Instance Count and Relation Instance Count, SCUBA identified close to 50% more concept and relationship instances than Text2Onto. This is an indication that the strategy of generating multiple smaller ontologies, and then aligning and merging the results into a larger composite ontology can improve information quality. Thus, SCUBA seems ideally suited for cases where information is spread across numerous and small data sources, or in cases where narrowly specific ontologies are merged with broader, more general ones.

IV. Summary

In this paper, we presented SCUBA, an agent-based ontology creation and alignment framework developed to address the shortcomings of current socio-cultural modeling efforts. SCUBA achieves a balanced human-computer modeling paradigm to 1) automate the creation of social and cultural ontologies from selected source materials, 2) apply a variety of nominal, semantic, structural, and statistical matching techniques to align multiple ontologies in a multimodeling environment, and 3) evaluate the effectiveness of the generation and alignment processes. Preliminary results of our initial agent-based experiments were promising – by applying ensembles of multiple matching techniques, we achieved significant improvements in alignment F-scores and other evaluation measures while dramatically reducing the amount of time required to manually produce coordinated, useful domain models.

Acknowledgment

The authors acknowledge the valuable contributions of the SCUBA development team at Raytheon IIS (State College, PA and Garland, TX) and Raytheon NCS (Largo, FL).

References

[1] K. Morik. Balanced cooperative modeling. *Machine Learning*, 11:217–235, 1993.

[2] B. G. Buchanan and D. C. Wilkins (editors). *Readings in Knowledge Acquisition and Learning: Automating the Construction and Improvement of Expert Systems.* Morgan Kaufmann, San Mateo, CA., 1993.

[3] J. Euzenat and P. Shvaiko, *Ontology Matching*. Springer-Verlag, Berlin Heidelberg (DE), 2007.

[4] A. Maedche and R. Volz, "The ontology extraction maintenance framework Text-To-Onto," in Proc. ICDM'01 Workshop on Integrating Data Mining and Knowledge Management, 2001.

[5] P. Cimiano and J. Völker. Text2Onto - A Framework for Ontology Learning and Data-driven Change Discovery. In Andres Montoyo, Rafael Munoz, Elisabeth Metais, Proc. 10th Int. Conf. on Applicat. of Natural Language to Inform. Syst. (NLDB), volume 3513 of Lecture Notes in Computer Science, pp. 227-238. Springer, Alicante, Spain, June 2005.

[6] http://code.google.com/p/text2onto/

[7] http://www.yale.edu/hraf/

[8] http://www.yale.edu/hraf/Short_OCM_List_121503.pdf

[9] http://hdr.undp.org/en/

[10] Field Manual 2-0, Intelligence. 2010. HQ, Dept. of the Army

[11] http://protege.stanford.edu/

[12] http://jade.tilab.com/

[13] Field Manual 101-5: Staff Organization and Operations. Dept. of the Army.

[14] L. Yilmaz, A. Lim, S. Bowen, and T. "Requirements and Design Principles for Multisimulation with Multiresolution, Multistage Multimodels" in *Proceedings of the 2007 Winter Simulation Conference,* Washington, DC, 2007.