A Data-Driven Approach to Ontology Discovery

Charlotte Shabarekh, Ian Yohai, Brian Riordan
Aptima, Inc.
Woburn, MA
cshabarekh@aptima.com, iyohai@aptima.com, briordan@aptima.com

ABSTRACT

A central challenge in the intelligence community is managing and effectively integrating large amounts of disparate information sources for concise presentation of knowledge to analysts. Currently, the high volume of incoming intelligence imposes a substantial burden on the analyst to understand the inconsistent, noisy data, potentially leading to missed intelligence about entities and their relationships.

To address this need, we have developed a data-driven approach that unifies disparate mentions to individuals and relationships to provide the analyst with an overview of the social network hidden in large, noisy databases. Our approach automatically discovers systems of related concepts structured in data to learn ontologies that are optimal for representing the knowledge encapsulated in the database. Taking advantage of recent advances in nonparametric Bayesian clustering (Kemp et al., 2006), the system analyzes streams of data to disambiguate references to the same entity and to identify groups of semantically related entities. The tool thus fuses knowledge across the datastore to create concise profiles of entities for use in analysis and an improved ontology for use in semantic search engines.

We evaluated our approach on operational sensor data collected during the JFCOM-sponsored Empire Challenge 2010 military training exercise. The EC10 dataset mirrors operational tactical intelligence datasets, and is characterized by a high level of sparsity and noise (missing and incomplete data, inconsistent manual coding). In preliminary experiments, our system produced high precision semantic clusters of entities by resolving disparate references to entities and uncovering hidden relationships. On the task of resolving entity references, compared with a baseline to *k*-means clustering algorithm, our approach yielded a 38% improvement in purity and a 6 % improvement in F-measure. These results indicate that our approach is better able to "connect the dots" across disparate documents to produce consolidated entity profiles than widely used clustering methods.

ABOUT THE AUTHORS

Ms. Charlotte Shabarekh is a Research Scientist at Aptima, Inc., with expertise in Natural Language Processing, Knowledge Representation and Probabilistic Modeling. She applies Machine Learning algorithms to provide solutions to problems in Semantic Behavioral Analysis, Pattern Recognition and Network Analysis. Ms. Shabarekh holds a M.S. in Computational Linguistics from State University of New York at Buffalo and a B.A. in Linguistics from State University of New York at Albany. She is a member of the Association of Computational Linguistics.

Dr. Ian Yohai is a Quantitative Social Scientist at Aptima, Inc., where he provides expertise in the statistical analysis of social science data. He has extensive experience with methods for missing data, causal inference, multilevel modeling, and the analysis of survey data. Dr. Yohai received his Ph.D. and A.M. in political science from Harvard University, and A.B. in politics with highest honors from Princeton University. He is a member of the American Political Science Association (APSA) and the American Association for Public Opinion Research (AAPOR).

Dr. Brian Riordan is Modeling and Simulation Scientist at Aptima, Inc. with expertise in machine learning, statistical natural language processing, and cognitive modeling. He employs machine learning approaches to human action understanding and imitation learning, modeling the dynamics of topics in text over time, and pattern recognition of human networks in multi-source intelligence data. Dr. Riordan holds a Ph.D. in Linguistics and Cognitive Science from Indiana University, an M.A. in Computational Linguistics from Indiana University, and a B.A. in Linguistic Anthropology and East Asian Studies from New York University.

An Automated Approach to Data-Driven Ontology Discovery

Charlotte Shabarekh, Ian Yohai, Brian Riordan
Aptima, Inc.
Woburn, MA
cshabarekh@aptima.com, iyohai@aptima.com, briordan@aptima.com

INTRODUCTION

Accessing Disparate Data Sources

Managing information retrieved from multiple sources is a critical challenge facing the intelligence community. Currently, the high volume of near-constant incoming intelligence imposes a substantial burden on the analyst to review and understand the raw data, which can lead to analyst overload and fatigue. Building and maintaining a common picture and shared situational awareness is a difficult but necessary step in identification of high value targets and key events appearing across the data sources. Failure to adequately "connect the dots" across multiple datastores can result in enormous information loss, placing the warfighter and intelligence community at a severe disadvantage.

Compounding the challenge, the intelligence reports populating these datastores are obtained through multiple, asynchronous sensors which are not standardized in terms of reporting processes and terminology. Therefore, the datastores are wrought with *inconsistent*, *incomplete* and sometimes *incorrect* information, which makes both automated and manual extraction of actionable intelligence extraordinarily challenging. For example, entities appearing in reports may be referred to by name, by physical attributes, by the activities they perform or by other observable characteristics. Even in situations where there are *named entities* whose proper name is reported, there is great variation in terms of aliases and spelling. This challenge, termed *entity resolution* (Garcia-Molina, 2005), arises in the absence of Artificial Intelligence's Unique Name Assumption (UNA; Russell & Norvig, 2003), when each name does not have an unambiguous mapping to a unique entity.

One added complication in the intelligence domain is the prevalence of non-English, Arabic script names, which are transcribed into roman letters in a multitude of different, non-standardized ways (i.e *Muammar Gadafi* vs. *Moammar Qadaffi*). Furthermore, Arabic surnames have unique characteristics which are not reliably reported, such as the *nisbah* name which uses the definite article *al* to indicate the family origin (i.e. Mu'ammar *al*-Qadafi) or *patronymic* names based on the name of one's father (i.e. Moammar *Mohammed* Qadafi).

Accessing Inconsistent Data with Search Engines

While automated knowledge management and information retrieval technologies have emerged to assist the analyst in extracting intelligence from large databases, these systems are constrained by the same data issues faced by human analysts. Modern enterprise search engines, which allow users to query a database or set of databases, have in general made accessing data easier, but accessing the *relevant* data remains a challenge. For instance, in traditional keyword search, a query for *Muammar Gadafi* would not return reports containing *Moammar Mohammed Al-Qadafi*. While internet search engines such a *Google* and *Bing* will do query alteration (Manning et al., 2008) behind the scenes to search for known variants of common names, many enterprise search engines do not, thus putting the burden on the analyst to specify comprehensive queries (i.e. *Muammar Gadafi* OR *Moammar Mohammed Al-Qadafi*). In the case of lesser known names, even the search engines that do query alteration will defer to the users' knowledge of a domain to specify the best query. While specifying a good search query seems simple—after all, the vast majority of Google's millions of queries a day are keyword searches—it is actually a challenging feat in the intelligence world, requiring domain knowledge, familiarity with the document store and identification of entities of interest in order to ensure good *recall* of relevant reports and documents.

Ontology-Based Search Engines

Ontology-based search (Petschner et al., 1999) has overcome some obstacles in the intelligence search domain, particularly reducing the burden of keyword query generation, but has also introduced new ones. Extending beyond traditional search, ontology-based search relies not just on the presence or absence of keywords in documents; it actually uses *semantics*, or meaning, to retrieve relevant search results both at the

document and data levels. Therefore, many *semantic search* applications use ontologies to *index* databases on searchable semantic terms. Indexing can be done by associating database elements, such as documents or Resource Description Framework (RDF) *triples*, with ontology terms that appear in them. Searches are performed by executing queries based on the ontology and those items in the index which are tagged with the same ontology terms are returned. Continuing with the *Gadafi* example, all variants of his name would be indexed on the same ontology term or terms – such as *Libyan Leader* – which would also appear on the ontology used for querying. Therefore, a single query on any known variant of Gadafi's name would return all data elements tagged with *Libyan Leader*, producing high recall and eliminating the need to specify all name and spelling variants in the query.

There are two significant and intrinsically linked challenges associated with semantic search using ontologies. The first is indexing the database elements with the best ontology elements and the second is developing the ontology itself. Ideally, each one of the data elements are straightforwardly mapped to at least one ontology term. However, often the ontologies do not completely represent the information in the datasets, so indexing elements becomes a non-trivial challenge where some elements are not mapped to any ontology terms or semantically similar elements are mapped to different ontology terms because of inconsistent terminology. While the ontologies are often manually built by experts with domain knowledge, they are restrictive and static, and do not always represent the concepts in the index. As new concepts are identified as being critical or new relationships are added to the database, they must be added to the ontology and the original documents must be re-indexed to capture the new additions – tasks which are resource heavy. This lack of flexibility puts a burden on the analyst to identify ways to access the critical data while adhering to the ontological constraints. Therefore, in order for ontology based search engines to provide maximum utility to the user, the ontology must completely contain the concepts of the database being searched.

DATA DRIVEN ONTOLOGY DISCOVERY

Since manual specifications of ontologies can lead to incomplete representations of the associated data, we took a data-driven approach to ontology generation. By using the data to generate the ontology, we not only ensure that the ontology optimally represents the data, but we also implicitly associate each data element with ontology terms to generate an index used for search. Thus, our approach mitigates the traditional challenges of developing an ontology and using it to index a database.

We attempt to uncover the underlying, emergent data structures in the disparate documents in the datastores being searched using the Infinite Relational Model machine learning algorithm (Kemp et al., 2006). These data structures will identify the important entities and relationships that appear in the datastore and organize them into optimal structures (i.e., tree, ring, dominance hierarchy) to encode the relationships between them. The resulting data structures are essentially flexible, data-specific ontologies that implicitly fuse knowledge across disparate documents through dynamically discovered relational links. Leveraging the graph-based network structure of social networks hidden in the datastores, our approach provides a context in which entities exist and interact, in accordance with the notion that each concept draws meaning from its relationships with other concepts (Kemp et al, 2010).

In this paper, we describe an automated approach for fusing inconsistent, incomplete data that appears across searchable repositories. Our approach produces an ontology of entities and semantic relationships that is unified across disparate references and inconsistent terminology. The resulting ontology can be used to provide an overview of the social network hidden in the datastores for analysts and can be used as the backbone of an ontology-based semantic search engine to enable relevant search results of noisy databases.

Our approach consisted of two steps. The first step was data pre-processing where we converted raw structured and unstructured reports from our data set into sets of entity attributes (e.g., biometrics, names) and entity relationships (e.g., work or familial links) for processing by the model. In the second step, we use the combination of entity attributes and relationships to resolve entity references and discover latent relationships between entities. The outputs of the system are: 1) entity clusters, in which entity references who share common attribute data or relationships appear in the same cluster; and 2) inferred higher-order relationships between resolved entities. For instance, if *Azar* and *Hazara* are resolved to the same entity, then all relationships that are associated with each reference are joined. So, if *Azar* had a relationship with *Keyhon*, then the system discovers that *Hazara* also has a relationship with *Keyhon*. We describe the details of the data preprocessing and ontology induction algorithm in the following sections.

DATA

Dataset Characteristics

The dataset that we used for evaluation was the Empire Challenge 2010 (EC10) unclassified dataset. It embodies the inconsistencies of entity reporting described above and also exhibits sparse and contradictory observations. EC10 was a two week live action military exercise hosted by Joint Forces Commanders (JFCOM) at Ft. Huachuca in Sierra Vista, Arizona, in August, 2010 during which multiple audio, imagery and human sensors were deployed to evaluate Intelligence, Surveillance and Reconnaissance (ISR) technologies.

The dataset includes SALUTE reports (standing for Size, Activity, Location, Unit, Time, Equipment) which consist of unstructured text fields, Entity ID Messages which are automated reports generated by various sensors, and associated imagery. The roughly 650 SALUTE Reports contain short descriptions (phrases or a few sentences) of observations about entities and vehicles which were manually written by humans manning the sensors. Statements about entities contain descriptions of actions, interactions with other entities, and attributes such as clothing or physical characteristics. The amount of information available varies significantly from report to report. In some cases, entities are named, although multiple spellings are common (e.g., *Pacul* vs. *Pakul*). In other cases when the identity of the entities is unknown, the reports contain vague references to an unspecified number of entities performing an action, or there is little or no attribute information. The data with respect to vehicles contain similar variability. Some reports specify the brand and color of a vehicle, while others just reference a "vehicle" performing an action.

The EC10 dataset thus exhibits all of the data noisiness that we are trying to address. Since the SALUTE reports' primary descriptive field contained unstructured text, there was little consistency of vocabulary across reports. Surprisingly, even reports generated by the same sensor used inconsistent terminology, possibly due to different members of a team generating the reports. For instance, four reports, generated within thirty minutes of one another, refer to movement of entities as *approaching*, *coming*, *heading* and *from*. Furthermore, this inconsistent terminology was not limited to predicates, but also affected locations, such as when the *Bazaar* was referred to as the *Market*, *Village A* was also called *Wakil Kalay* and *Village B* was referred to as *Darwishan* and *See-Thru-the-Wall (STTW) Site*. Name variation also occurred with person entities such as *Romeo* vs. *Feda* or *Headdress Guy* vs. *Osmund*. Note that name variation is different from spelling variation, where name variation is when an entity is associated with two or more different names and spelling variation is when an entity is associated with one name that has multiple spellings (e.g., *Osmund* vs. *Osmond*).

The EC10 dataset was not only noisy, but also sparse. Approximately 12% of the SALUTE reports contained references to one or more *named* entities; an additional 32% of the reports contained references to one or more *unnamed* entities (e.g., "individual" or "subject"). Since each report could contain references to more than one entity, we identified 316 separate unnamed entities. Some form of attribute information was available for 33% of the entity references, with clothing, gender, and associated vehicles being the most common attributes. Finally, just over a third of the reports contained some information relating to vehicles or named locations, such as the *Bazaar*.

Data Processing

Individual SALUTE reports were parsed and manually coded to create RDF style statements of entities and events. Two different types of coding were carried out: *attribute* coding and *relationship* coding. Attribute coding was done by extracting all biometric and clothing descriptions of entities appearing in the documents. A vector of these attributes was created including features such as *carriesWeapon*, *wearsEyeglasses*, and *blueShirt.* Relationship coding involved identifying entities who appeared together in reports and constructing a matrix of these relationships (i.e. *Feda associates_with Azar*).

As discussed above, the SALUTE reports were extremely sparse, with 32% containing generic references to individuals. In order to reduce the sparsity of entity descriptions in the dataset, we manually augmented sparse text data with salient information available in associated imagery. Images which were referenced in SALUTE reports were used sparingly to provide additional attribute data when the text reports did not provide enough information.

Ground Truth

From the dataset, we identified five key entities who appear frequently in the resulting dataset. These five key entities – *Azar, Feda, Farshard, Ghetti and Keyhon* – were analyzed and their relationships and attributes were used to create a ground truth for evaluation of the ontology discovery system. (see Results section). Each of the key entities appear in the dataset by name, but also appear as spelling variants (i.e. *Hazar* for *Azar*), name variants (i.e. *Hollywood* for *Ghetti*) and generic

references (i.e. *Asian male* for *Farshard*). Additionally, we identified five non-key entities, which appeared infrequently and with sparse attribute and relationship data – *Osmond, Ahmed, Rehman, Pakol, and Jaladin*. The remaining 200+ entities appearing in the dataset are unresolved generic references which could be references to one of the key entities, but not enough information was available in the reports or associated imagery to make an association.

MODEL

The ontology was discovered using the Infinite Relational Model (IRM; Kemp et al., 2006). The IRM infers a *theory* to explain the data – a system of structured categories and relationships, neither of which is directly observable in the data. The inferred theory functions as an ontology for a domain consisting of interrelated concepts. Using nonparametric Bayesian inference, the IRM algorithm discovers novel categories for groups of entities, and predicts the relationships between categories that are most probable. As an example, consider the situation where an analyst is presented with information derived from SALUTE or other HUMINT (human intelligence) reports about a number of individuals. The information indicates that certain individuals interact with others, but there is little background information about many of the individuals. Working largely from this data, the analyst may want to come up with a theory that explains the pattern of interaction among the individuals. Constructing such a theory is challenging—especially when there are many individuals and many interactions between them—for at least two reasons. First, with few distinguishing attributes for the individuals, there is little basis to categorize them into groups or roles. Second, with little knowledge other than the interactions, it is difficult to discern types of interaction. The IRM seeks to learn the categories in the domain, the numbers of categories, and the relationships among the categories. Rather than tackling each of these problems one at a time, the IRM attempts to capitalize on the fact that in many cases entities may be categorized largely by their relationships to other entities, and so aims to discover the categories and relationships for the data simultaneously.

The IRM takes as input relationships between entities (e.g., *subject-predicate-object* triples in RDF statements) or entities and their attributes (e.g.,biometrics, clothing articles). These input requirements map intuitively to RDF statements derived from text documents. The output of the IRM algorithm is an ontology or relational system that specifies sets of discovered categories and the relationships between them, along with the probabilities of observing these relationships.

In the simplest case, the input data may contain a set of entities—e.g., individual people—and a single binary relationship, *interacts*(e_i, e_j) – for example, *associates(Azar, Feda).* Representing this data as a matrix, each cell encodes whether entity e_i has been observed to interact with entity e_j.

The IRM framework has several important properties for ontology discovery:

- The IRM discovers categories not only for entities in the data, but also the predicates that describe and relate them. Most clustering algorithms can only cluster entities into categories.
- The number of categories is not fixed. The IRM framework allows a potentially infinite number of categories to be discovered in the data. The IRM incorporates a prior probability that discovers the optimal number of semantic clusters from the data, favoring smaller numbers of clusters. Because the number of categories is allowed to grow with the size of the data, IRM can create more complex representations as needed.

EXPERIMENTS AND RESULTS

To evaluate the approach, we performed three experiments focused on the entity resolution task. In the first experiment, we evaluated the algorithm's ability to cluster entity mentions using only entity attributes (not relations). The attributes that we extracted from the SALUTE reports consisted mainly of physical descriptors such as race, gender, and clothing. The purpose of the second experiment was to evaluate the effect of sparsity in entity attributes on the algorithm. In this second experiment we manually augmented the attribute data for the five key entities to enforce consistency across reports, but continued to use the raw attributes for the other entities. The results of this experiment were compared with the results of Experiment 1 to gauge trends in entity resolution performance with decreasing sparsity. In the third experiment, we evaluated the algorithm's performance under a compound scenario where we 1) included entity relationships in addition to entity attributes and 2) employed a semi-supervised learning procedure, in which the outputs of one clustering run were used to label one key entity's mentions, and this augmented data was input to a second clustering run for entity resolution. Each experiment is described below.

Experiment 1: Entity Attributes

Experiment 1, which used just raw attribute observations from the data as input to the IRM,

produced two main findings. First, we observed a homogenous semantic cluster containing all known references to *Azar*. This implies that *Azar* had one or more unique attributes that distinguished her from the other entities. In the other entity clusters, we observed heterogeneity. For example, one cluster contained references to multiple entities, including *Feda*, *Farshard*, *Ghetti*, and *Keyhon*, among others. These results are not surprising given the sparsity of the feature data, and the fact that only entity attributes (and not relations) were included. Still, the ability to pull out one key entity from these sparse data is an important finding, especially in light of Experiment 3, described below.

Experiment 2: Manually Augmented Attributes

In this experiment we manually augmented the attribute matrix for the five key entities, *Azar, Feda, Farshard, Ghetti* and *Keyhon*, while leaving the raw attributes in place for the other entities. The rationale for this experiment was to evaluate the performance of the IRM on "cleaner" data. In order to perform the augmentation, the features were analyzed and a consistent set of observations was created that uniquely captured the features for each of the five key entities.

In sharp contrast to the Experiment 1 results, we obtained five homogenous clusters, one for each key entity. All known references to the key entities throughout the dataset appear in the appropriate cluster. For example, all known references to *Feda* appear in one cluster; all known references to *Farshard* appear in another cluster, and so on. In addition to the five key entity clusters, we also obtained five other clusters that contained references to non-key entities or entities that could not be identified from the data.

Experiment 3: Stacked IRM Approach

In the third computational experiment, we apply the lessons learned from the first two experiments, while also incorporating the relation data. In this "stacked" approach, the outputs of one clustering run are used to bootstrap the clustering on a second run. As noted above, the first step of the stacked IRM approach involved using the raw attribute data as in Experiment 1, then using the results to improve the clustering when relationships were included as well.. Relationships took the form of binary indicators denoting association between two entities. There were a large number of such associations in the data where several unidentified individuals were linked only to each other (and not to identifiable entities). For example, a typical SALUTE report stated: "an Asian male was seen with a female in a blue shirt." In this case, we assigned unique identifiers to each entity (e.g., Unknown27 and Unknown28). For this reason, these two entities were not explicitly connected to any other (identifiable) entities in the dataset.

Recall from the Experiment 1 results that we were able to pull out a homogenous *Azar* cluster. On the second step of Experiment 3, we used the results of Experiment 1 experiment to label references to *Azar* in the relationship data. Since *Azar* was associated quite frequently with *Feda* and *Farshard*. unknown references to *Feda* and *Farshard* could thus be tied together through their relationships with *Azar*.

Experiment 3 produced two relatively homogenous clusters containing *Feda* and *Farshard*. All known references to these entities appear in the two clusters, and only four entities were misclassified. Similarly, we obtained a *Ghetti-Keyhon* cluster containing most, but not all, references to those two individuals. Like *Feda* and *Farshard*, these two entities often frequently appear together in the data. Finally, we also obtain a homogenous *Azar* cluster, as in the first experiment.

These results illustrate the importance of taking advantage of both entity attribute and relationship data. For example, in one cluster we grouped together generic entities *Unknown29* and *Unknown30*. These entities were only linked to each other, and not to any other entities, in the raw relationship matrix. The analyst would have no way of knowing that these entities were in fact *Feda* and *Farshard* (which we knew from our construction of ground truth). Nevertheless, the IRM was able to properly group these two entities in the *Feda* and *Farshard* cluster.

Metrics and Comparison to *k*-means

In order to provide a point of comparison for our system's results, we compared the results on the entity resolution task with a *k*-means clustering algorithm. *K*-means partitions *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest mean, or centroid. For EC10, the observations were attributes of entities (i.e. biometrics, clothing) and relations between entities. The means are initially sampled at random; the algorithm re-estimates the mean at each iteration. Formally, given a set of observations $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, where each observation is a *d*-dimensional real vector, *k*-means clustering aims to partition the *n* observations into *k* sets ($k \leq n$) $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ where $\boldsymbol{\mu}_i$ is the mean of points in S_i.

Using the ground truth constructed for the five key entities referenced above, we scored the results of the *K*-means and IRM-based approaches using standard clustering metrics– *Purity*, *Inverse Purity* and *F*-

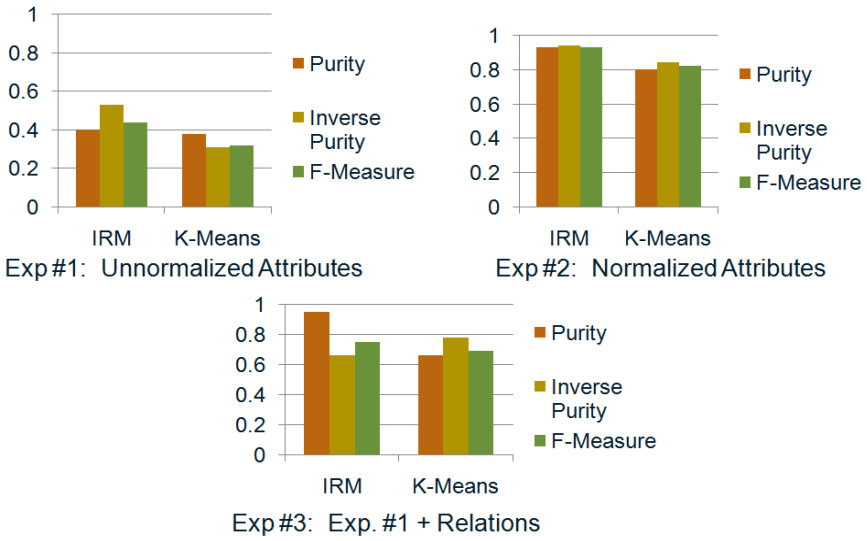

Figure 1. Results from Experiments 1- 3.

Measure. Purity and inverse purity measured the homogeneity of the clustering to determine how often items were grouped with similar items As evaluation metrics, purity and inverse purity are limited in that they do not penalize for many small, homogenous clusters. For instance, purity is 1 (perfect score) if each entity gets its own cluster. Thus, we cannot use purity to trade off the quality of the clustering against the number of clusters (Manning et al., 2008). Therefore, we also use the F-measure to balance purity and inverse purity.

Figure 1 plots the results for each of the three experiments, on each of the three metrics, for both IRM and *k*-means. For Experiment 1, using the entity attribute data, the IRM outscored *K*-means in Purity, Inverse Purity and F-Measure While there was only a 2% improvement in Purity using the IRM Inverse Purity improved over 20%, yielding an overall improvement of 12% for the F-Measure.

For Experiment 2, using the manually augmented attribute data for the five key entities, the IRM also out-performed the *K*-means algorithm with at least a 10% improvement for Purity, Inverse Purity and F-Measure. Both the IRM and *K*-means algorithms performed well on the fully normalized data, which was expected given the consistent, complete attributes used as input to the IRM. However, given the nature of *K*-means, we were forced to specify the number of clusters a priori that it needed to produce. In this case, we set the number of clusters to six, which we knew from our construction of ground truth was the "correct" answer – there were five key entities plus one cluster for the non-key entities or entities that could not be identified. Therefore, as in the other experiments, *K*-means had an advantage over the IRM, which needed to determine the optimal number of clusters on its own.

Finally, in Experiment 3, where we employed the two-step "stacked" IRM approach, the IRM far outscored *K*-means in purity. The inverse purity score of *K*-means was much higher, but that is due to the fact that it produced far fewer clusters, and many entities sharing attributes or relations were clustered together by chance. The high inverse purity score brings the *K*-means F-measure score to within range of the IRM models, but with lower scores overall. The tradeoff with high inverse purity is low purity which is clear from the K-Means results.

CONCLUSION

The experiments presented in the previous section show that our approach – notably the IRM – can produce precise results, with clusters of entities that are closely related semantically. The high level of purity of the resulting clusters – even in the presence of much noise – show promise that this solution can be adapted to present analysts with a semantic network of related entities. The IRM outperformed *k*-means when in precision, achieving on average 38% total gain across the three purity metrics. While the IRM's recall is low relative to *k*-means (6% total loss), the combined F-measure balancing precision and recall showed a 6% total gain, suggesting that the IRM is more effective overall than traditional clustering algorithms such as *k*-means. The F-measure is the metric that we believe best measures the value to the analyst. Currently, analysts must deal with low recall for search queries, with only

exact matches being returned. While both IRM and *k*-means can boost recall, only the IRM can moderate the increased recall with a balanced precision score, ensuring that not only will analysts receive more results, but also that those extra results are *relevant* and meet their information requirements. We are encouraged by these results and believe that they demonstrate the potential of the approach on operational data.

Next Steps

Our next steps include incremental ontology learning, improved data preprocessing, and further experimentation. First, we will extend the IRM algorithm to support continuously growing datastores by implementing an incremental version. Rather than doing a complete retraining of the models on every element in a datastore every time new data is added, incremental learning allows the addition of new information to the already learned models, eliminating the potentially time-consuming process of retraining using all of the data. In addition to saving time, this also produces more consistent results, combining what was learned in previous runs with newly available data. Unlike manually specified, static ontologies, our approach is designed to generate flexible ontologies that dynamically change to reflect changes in the data from which they are derived.

To better deal with the sparsity inherent to operational datasets, more sophisticated data preprocessing capabilities are needed. We plan to explore statistical methods for data imputation. Finally, we intend to employ this approach on additional operational datasets for continued experimentation and development.

This completed system will reduce the burden on analysts to process large amounts of inconsistent information and provide them access to disambiguated information about entities and their social networks. When complete, we imagine a real-time system that consists of four components that cascade into each other in an iterative loop, simulating the information dissemination workflow from raw data to actionable intelligence. The envisioned system will:

1) monitor a database of human intelligence reports from multiple sensor sources

2) process the reports into a searchable index

3) induce an ontology for the index, and

4) present the discovered ontology to the user for analysis and searching of the database contents.

The work we have done so far is only a small piece of this envisioned system, but we believe that it lays the groundwork for an ontology-based semantic search engine. The "next steps" outlined above, combined with the promising results from early experimentation will help us get one step closer to retrieving actionable intelligence from noisy datastores.

ACKNOWLEDGEMENTS

The work presented in this paper was funded by the Office of Naval Research, Code 30, under a Small Business Innovative Research (SBIR) Phase I Contract (Contract # N00014-10-M-0453). We would like to thank Mr. Chris Kirkos and Mr. Martin Kruger for their support.

REFERENCES

Amigo, E., Gonzalo, J., Artiles, J., and Verdejo, F. (2008). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, *Vol. 12*, 461-486.

Garcia-Molina, H. (2005). Handling data quality in entity resolution. *Proceedings of the 2nd international workshop on Information quality in information systems.*

Guha, R., R. McCool and E. Miller (2003). Semantic Search. *Proceedings of the 12th international conference on World Wide Web*, 700-709.

Kemp, C., Tenenbaum, J. B., Griffiths, T. L., Yamada, T. & Ueda, N. (2006). Learning systems of concepts with an infinite relational model. *Proceedings of the 21st National Conference on Artificial Intelligence.*

Kemp, C., Tenenbaum, J. B., Niyogi, S. & Griffiths, T. L. (2010). A probabilistic model of theory formation. *Cognition.* 114(2), 165-196.

Kok, S., & Domingos, P. (2008). Extracting semantic networks from text via relational clustering. *Proceedings of the Nineteenth European Conference on Machine Learning*, 624-639.

Manning, Christopher D., Prabhakar Raghavan and Hinrich Schütze (2008). *Introduction to Information Retrieval*, Cambridge: Cambridge University Press.

Powers, S. (2003). *Practical RDF*, Sebastopol, CA: O'Reilly Media.

Pretschner, A. and Gauch, S. (1999). Ontology Based Personalized Search. *Proceedings of the 11th IEEE Intl Conf on Tools with Artificial Intelligence*, 391-398.

Stuart J. Russell and Peter Norvig (2003). *Artificial Intelligence: A Modern Approach, Second Edition*, Upper Saddle River, NJ: Prentice Hall.

Zhao, Ying and George Karypis (2005). Hierarchical Clustering Algorithms for Document Datasets. *Data Mining and Knowledge Discovery, Vol. 10, No. 2*, 141 - 168.