

A Data Fusion Approach to Indications and Warnings of Terrorist Attacks

David McDaniel^{1a}, Gregory Schaefer^a

^aSilver Bullet Solutions, Inc., 1901 Ft. Myer Dr., Suite 501, Arlington, VA, USA 22209-1620

ABSTRACT

Indications and Warning (I&W) of terrorist attacks, particularly IED attacks, require detection of networks of agents and patterns of behavior. Social Network Analysis tries to detect a network; activity analysis tries to detect anomalous activities. This work builds on both to detect elements of an activity model of terrorist attack activity – the agents, resources, networks, and behaviors. The activity model is expressed as RDF triples statements where the tuple positions are elements or subsets of a formal ontology for activity models. The advantage of a model is that elements are inter-dependent and evidence for or against one will influence others so that there is a multiplier effect. The advantage of the formality is that detection could occur hierarchically, that is, at different levels of abstraction. The model matching is expressed as a likelihood ratio between input text and the model triples. The likelihood ratio is designed to be analogous to track correlation likelihood ratios common in JDL fusion level 1. This required development of a semantic distance metric for positive and null hypotheses as well as for complex objects. The metric uses the Web 1Terabyte database of one to five gram frequencies for priors. This size requires the use of big data technologies so a Hadoop cluster is used in conjunction with OpenNLP natural language and Mahout clustering software. Distributed data fusion Map Reduce jobs distribute parts of the data fusion problem to the Hadoop nodes. For the purposes of this initial testing, open source models and text inputs of similar complexity to terrorist events were used as surrogates for the intended counter-terrorist application.

Keywords: Data Fusion, Hadoop, Mahout, Semantic Distance, Probability Mass, Activity Model Matching

1. INTRODUCTION

In intelligence production and analysis clusters, many text entries (e.g., imagery/FMV annotations, chat, emails, SIGINT) can be generated about real-world activities of interest (AoI). AoI can be simple, complex, and typically overlap. AoI almost always have spatial and temporal parts. They can include categories and/or membership in other categories, e.g., roadside bomb as a type of IED. It is impractical for analysts and operators to find and correlate all text references to AoI for many reasons.

Statistical relational learning¹, probabilistic description logic^{2,3}, semantic distance measures^{4, 5}, ontology and model matching^{6, 7, 8}, distributed data fusion⁹, and big data technologies^{10, 11} offer elements that can be used to link this disparate data. In the work described herein, we leveraged many leading approaches and off-the-shelf tools to research the JDL track fusion approach to compute correlation of input text to activity models using a semantic distance algorithm that is analogous to spatial distance in the track correlation problem. Commodity laptops were loaded and configured with off-the-shelf big data and natural language processing tools: Mahout, OpenNLP, HBase. Openly available knowledge base sources were used: WordNet and the Web1T 5-gram corpus. The Web 1T 5-gram corpus is from the Linguistic Data Consortium at the University of Pennsylvania and consists of 1-5 grams and their observed frequency counts derived from 1 trillion word tokens of text from publicly accessible web pages¹². Mahout's built-in clustering is aided by activity models to which to cluster and the custom semantic distance algorithm. The semantic distance algorithm builds on Google Similarity Distance⁴ and Jaccard distance³, using the grammed corpus to compute text relationship likelihoods in a mathematically principled manner. Even with the big data technology, processing response time was a concern so offline pre-computation, indexing, and other techniques were considered for implementation.

¹ davem@silverbulletinc.com; phone 1 619 253-9040; fax 1 253 663-2023 www.silverbulletinc.com

2. BACKGROUND ON DATA FUSION LEVELS

The Joint Directors of Laboratories (JDL) defined the four levels of data fusion shown in Figure 1 (from¹³) in the early 1990's. They have stood the test of time well with some modifications proposed (e.g.,¹⁴) but mostly what could be considered refinement, e.g., sometimes a Level 0 is included. A summary and generalization derived from^{13, 14, and 15} is shown in Table 1. For this paper, focus will be on Level 1 and the specific algorithm class will be track-to-track correlation.

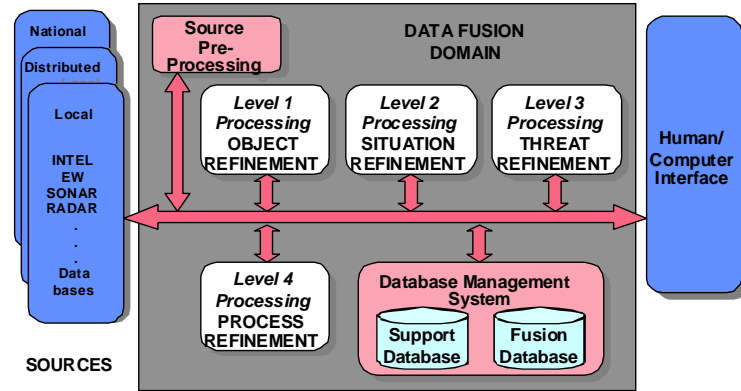


Figure 1. JDL Fusion Levels

Table 1. Fusion Levels

JDL 1991	Steinberg, Bowman, and White	Generalized
	Level 0 Sub-Object Data Assessment: estimation and prediction of signal/object observable states on the basis of pixel/signal level data association and characterization;	<i>Signal processing and feature extraction</i>
Level One Fusion Processing - Object Refinement. Level one processing combines parametric data from multiple sensors sources to determine the position, kinematics, state and other attributes or identity of low level entities.	Level 1 Object Assessment: estimation and prediction of entity states on the basis of observation-to-track association, continuous state estimation (e.g. kinematics) and discrete state estimation (e.g. target type and ID);	<i>Single object estimation</i>
Level Two Fusion Processing - Situation Refinement. Level two processing develops a description or interpretation of the current relationships among objects and events in the context of the operational environment. The results of this processing is a determination or refinement of the battle/operational situations.	Level 2 Situation Assessment: estimation and prediction of relations among entities, to include force structure and cross force relations, communications and perceptual influences, physical context, etc.;	<i>Object and event association</i>
Level Three Fusion Processing - Threat Strategic Refinement. Level three processing develops a threat-an extra-organizational oriented perspective of the data to estimate enemy extra-organizational capabilities, identify threat opportunities, estimate enemy extra-organizational intent, and determine levels of danger .risk.	Level 3 Impact Assessment: estimation and prediction of effects on situations of planned or estimated/predicted actions by the participants; to include interactions between action plans of multiple players (e.g. assessing susceptibilities and vulnerabilities to estimated/predicted threat actions given one's own planned actions);	<i>Prediction</i>
Level Four Fusion Processing - Process Refinement. Level four processing monitors and evaluates the ongoing fusion process to refine the process itself, and guides the acquisition of data to achieve optimal results. These interactions among the data function levels and with external systems or the operator to accomplish their purpose.	Level 4 Process Refinement (an element of Resource Management): adaptive data acquisition and processing to support mission objectives.	<i>Feedback control</i>

3. TECHNICAL APPROACH

The architecture for the technical approach is shown in Figure 2. Clustering is a powerful way of discovering new relationships. For the best results, this system would first process the text using natural language techniques to provide the algorithm with as much information from the documents as possible. Documents containing linguistic variations of operator and pre-developed activity models containing EEOI are input to the clustering algorithm, in effect creating classification clusters. Then, a sophisticated distance function is used to accurately measure similarity. The upper left of the diagram shows documents being acquired, e.g., FMV and imagery annotations, intelligence analyst chats, and database records. Natural Language Processing (NLP) is performed to extract entities, tokenize, and lemmatize and, optionally, co-reference and synonymize. Clusters are generated by Mahout using a custom semantic distance algorithm described later in this paper. High-likelihood relationships between the acquired data sources are maintained in the data store along with their cluster scores.

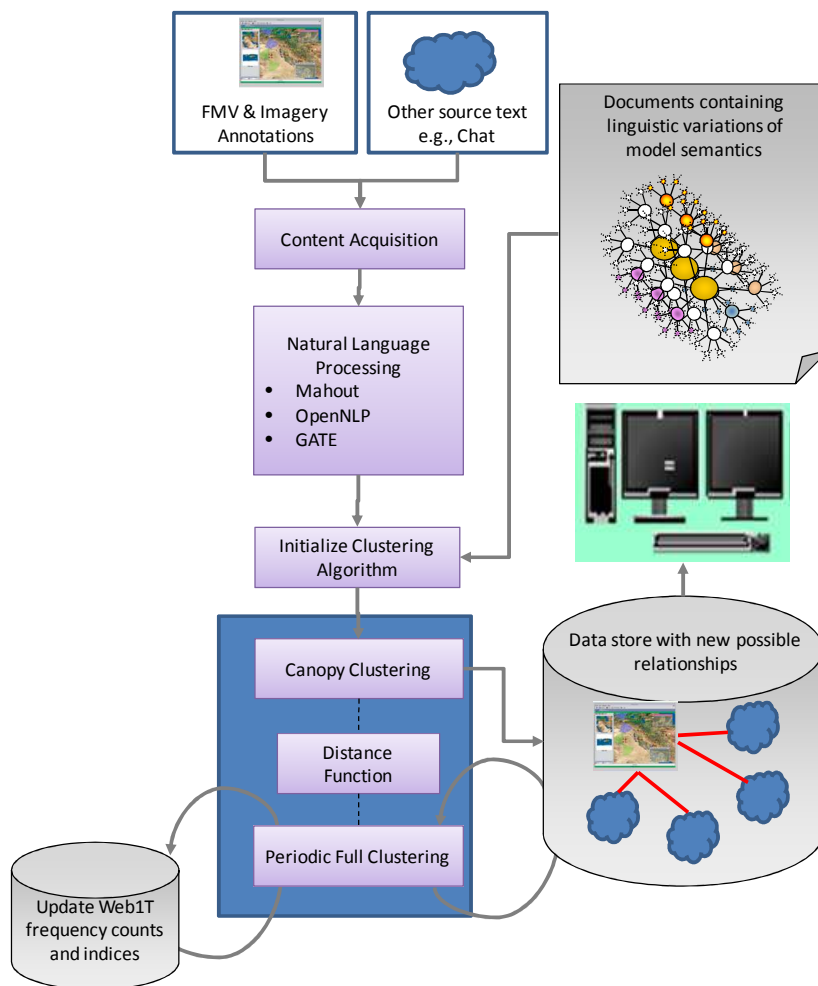


Figure 2. Model-Guided Clustering

Key design elements are described in the following subparagraphs.

3.1 Ontology activity model containing Events and Entities of Interest

In the model-guided clustering mode, incoming data is injected with specialized documents that give hints to the clustering system about what is being looked for. A model of AoI could define relationships between activities, entities, events, agents, resources, etc. that could aid clustering. These documents are designed to contain heavily weighted information regarding an Activity model, or a model about how something is done. The document representing this concept is inserted into the cluster to potentially become a centroid of other documents related to the concept that would gather around. The types of relationships defined could be sequential / temporal, input-output, whole-part, and so on. Examples of activity models are shown in Figure 3. Although these examples are about Blue force operations, models of Red force operations can be made similarly. The model is serialized and formatted. Our in-house tools we have developed over the years can transform these models to XML, RDF, or OWL. Then it is subjected to the NLP, e.g., to lemmatize and tokenize the model terms.

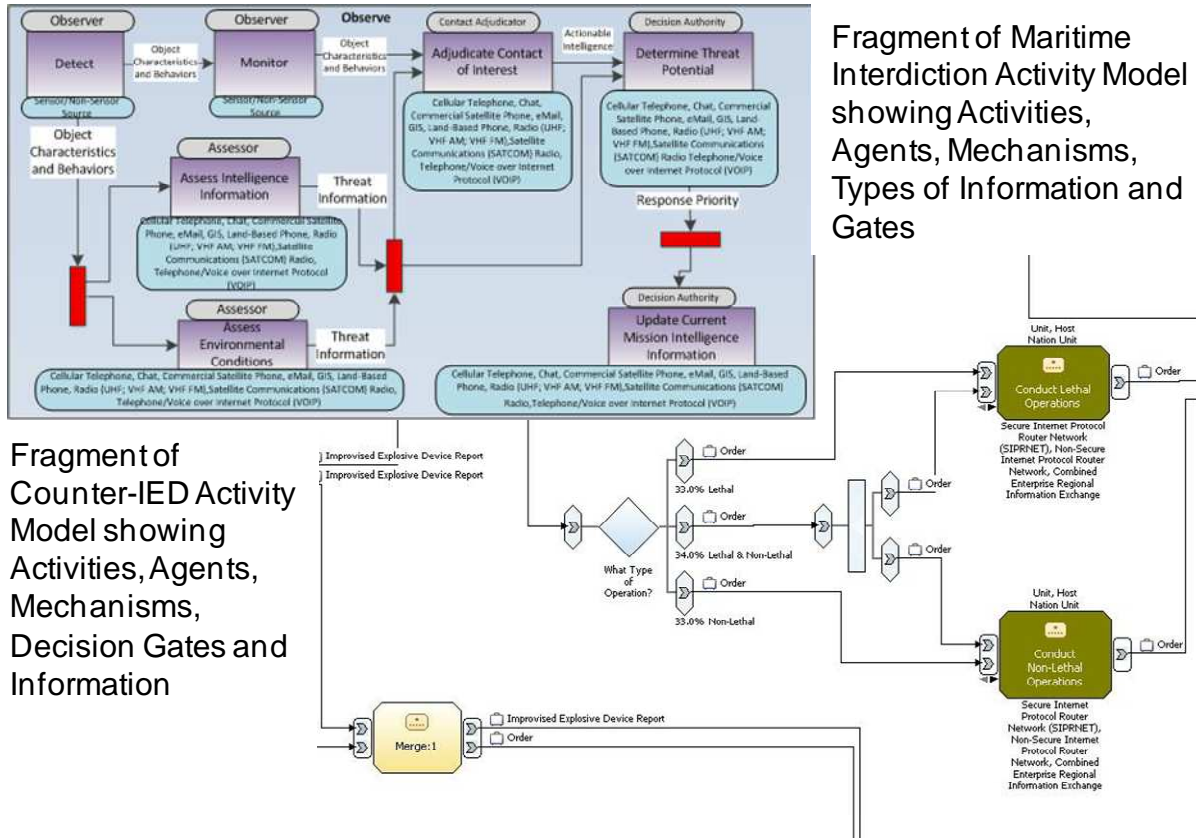


Figure 3. Fragment of Maritime Interdiction and Counter-IED Activity Models

3.2 Natural Language Processing

After gathering the data, one must be able to make sense of it to have the best chance of finding new relationships in that set. Also, it is important to best prepare the data for the eventual processing by the clustering algorithm. At the simplest level, the text or database record or query result must be serialized and formatted to be ingested by Apache Mahout, the machine learning library our system will be using for clustering. This library itself contains simple functions that can analyze text. More parsing and processing may be needed in the form of semantic, syntactic and pragmatic enhancements to the text. These advanced language techniques may require other natural language processing (NLP) components such as UIMA, the Unstructured Information Management Architecture, and supervised machine learning (SML) and gazetteer/lexicon-based libraries like OpenNLP and GATE. UIMA provides the ability to manage a document pipeline. Text would come into the pipeline and be efficiently process by any number of pluggable NLP engines. This is important because the gazetteer and SML approaches have different strengths and weaknesses and can complement each other nicely. The GATE gazetteer has an easy way for categorizing text using user-defined dictionaries. This is important for targeting a particular problem domain. OpenNLP's supervised learning model is based on probabilistic theory. The system is trained with contextual examples and can discover new concepts. Both allow for a controlled feedback loop. These components are able to more precisely split sentences and tokenize the documents. Parts of speech tagging of the text could provide semantic clues and ease entity recognition. The stemming/lemmatizing of the entities will help yield better results since the algorithm will be provided with the word's core. Wordnet can take the document pieces and increase the likelihood of matches with synonyms and hyperhymes, customizable per domain. Even temporal analysis and coreferencing can be tackled by this system and Hadoop can optimize the load by chunking the data over many nodes, leveraging the industry standard parallel processing framework. The last step is placing the text into a normalized vector space model. With Mahout, normalization uses what is known in statistics as a p-norm. In the end, a complicated document of unstructured data with potentially hidden intelligence will have been seeded and converted into a vector of meaningful parts or features, giving the clustering algorithm the best possible chance of finding new relationships and other information previously unknown.

3.3 Initialize Clustering Algorithm

Before execution of a set of documents by the clustering algorithm, a minimum set must be gathered, queued up and undergo NLP and vectorization. This initial minimum number of documents required is customizable for performance since the first full run needs to setup the baseline and will be slower than subsequent runs that could potentially be realtime. From these documents, initial clusters are created, either randomly or selected purposefully. The Mahout machine learning libraries provide code to more easily run this analysis, including Canopy Clustering described in the next section. Natively, it is able to run 5 different clustering algorithms: K-Means, Fuzzy K-Means, K-Means Spectral, Canopy and Latent Dirichlet Allocation (LDA) clustering. Other than using NLP to optimize feature selection, customizing distance measurement is an important step to improving clustering. Our proprietary distance function is described in paragraph 3.1.5.

3.4 Canopy Clustering

In order to provide an analyst with a tool to present the most reliable information in a timely manner, the clustering system may need to handle rapid updates. Content could be acquired using a real-time architecture like Storm, processing elements of a data feed immediately through NLP and then sending it to a specialized algorithm called canopy clustering. Canopy clustering, one of a group of approximate clustering algorithms can estimate the number of clusters in a data set as well as the approximate location of the cluster centroids, the geometric centers of these two-dimensional regions. Its name is derived from the overlapping clusters known as canopies that the input set of points are divided into using two distance thresholds and a fast distance measure. Canopy clustering's strength lies in its ability to create clusters extremely quickly, even with a single pass over the data, although this may not give accurate and precise clusters. The algorithm begins with a set of points and an empty list of canopies, iterating over the data set to create new canopies in the process. During each of the iterations, it removes a point from the data set and adds a canopy to the list with that point as its center. Next, it loops through the rest of the points, one at a time, calculating the distances to all the canopy centers in the list. If the distance between the point and any canopy center is within the first distance threshold, it is added into that canopy. If the distance is within the second distance threshold, it is removed from the list, preventing it from forming a new canopy in subsequent loops. The process is repeated until the list of points is empty. This approach prevents all points close to an already existing canopy from becoming the center of a new canopy, which would be redundant.

3.5 Distance Function

The distance function employs the following concepts and notation:

g^s	gram of size s
G^s	Web1T gram of size s
$freq$	frequency from Web1T
P	probability $f(freq)$
pmf	probability mass function $\Sigma(P)$
smf	semantic mass function $\Sigma(P)^{-1}$
pmr	probability mass ratio pmf_{\cap}/pmf_{\cup}
smr	semantic mass ratio smf_{\cap}/smf_{\cup}
Λ	likelihood ratio smr_{H_1}/smr_{H_0}
$dist$	distance $1/(\Lambda)$

3.6 Utterances, Grams, and Subgrams

A gram is an ordered tuple of tokenized terms. For Phase I, gram likelihoods come from the University of Pennsylvania Linguistic Data Consortium's Web 1T 5-gram corpus which Silver Bullet has purchased and installed on a Hadoop cluster. This data set, contributed by Google Inc., contains English word n-grams and their observed frequency counts. The length of the n-grams ranges from unigrams (single words) to five-grams. The n-gram counts were generated from approximately 1 trillion word tokens of text from publicly accessible Web pages. The frequency counts tend to be inversely proportional to gram size meaning, for example, that there are more tokens of a given unigram than combinations with additional grams. This simply says the AND'd counts are less than the singletons.

The frequency counts provide a source for a-priori values:

$$P(G^s) = \frac{freq(G^s)}{\alpha^s}$$

where:

G^s is a Web1T gram of size s (i.e., with s tokens) (1)

(Note: G will always refer to a Web1T gram while g refers to any gram, Web1T or in an input document.)

$freq$ is the Web1T frequency count

α^s is a normalizing constant over the Web1T frequencies for s -grams

$$= \sum_{G_i^s \in \text{Web1T}, s} freq(G_i^s)$$

The reason the normalization factor is specific to gram size is that the frequency counts are specific to the gram size. Although the utterances that are processed will be more than 5 grams, we have come up with a way to leverage the 5-gram a-priori's using subgram processing. The limitations due to the use of the 5-gram for Phase I is also mitigated by the fact that grams decrease rapidly in frequency as n grows.

We define an m -gram to be a subgram of an n -gram, where $m < n$, to be any m -gram formed by eliminating zero or more tuple elements from the n -gram. For example, (A, B, C) is a subgram of (A, B, C, D) . Note that it is not a subgram of (D, A, C, B) because grams are tuples, not sets. Note also that an m -gram can be a subgram of an n -gram multiple times, e.g., (A) is a subgram of (A, B, C, A) twice. Annotated set notation will be used to denote a subgram as $g^m \subseteq^{\text{gram}} g^n$.

A proper subgram of an utterance will be defined to be a Web1T subgram of the utterance that is not a subgram of any other Web1T subgram of the utterance. It will be denoted $g \subset^{s\text{-gram}} U$ where U is the utterance and s is the maximum Web1T gram size being employed. Thus,

$$G^m \subset^{s\text{-gram}} U \text{ iff } \neg \exists G^n \subseteq^{\text{gram}} U \wedge G^m \subset^{s\text{-gram}} G^n \quad (2)$$

where $n, m \leq s$

For example, in the utterance (A, B, C, D, E) , if (A, B) were a Web1T 2-gram but so was (A, B, C) , then (A, B) would not be a proper subgram of the utterance. Proper subgrams are important in the likelihood calculations later on.

For $m < n$, an n -gram has $\binom{n}{m}$ m -gram subgrams. For any n -gram utterance, it cannot have a frequency greater than any of its 5-gram subgrams, that is, $freq(g^n) \leq \min_{\text{all } (G^5) \text{ subgrams of } g^n} (freq(G^5))$. Or equivalently, it cannot have a probability less than any of its 5 gram subgrams. Some additional notation that we have found useful:

$|U|$ is the norm, i.e., number of subgrams in the utterance

$|U|_s$ is the norm, i.e., number of proper subgrams in the utterance
with respect to Web1T gram size s -gram

$$U_i \cap^{gram} U_j = \left\{ \text{all subgrams } g_x \text{ of both } U_i \text{ and } U_j \mid \left| U_i \cap^{gram} U_j \right| \leq \min \left(\left| g_x \subseteq U_i \right|, \left| g_x \subseteq U_j \right| \right) \right\} \quad (3)$$

$U_i \cap^{s\text{-gram}} U_j = \left\{ \text{all proper subgrams of both } U_i \text{ and } U_j \right\}$ where proper is
with respect to Web1T gram size s -gram

$\left| U_i \cap^{s\text{-gram}} U_j \right|$ is the norm, i.e., number of proper subgrams in the intersection

with respect to Web1T gram size s -gram

Note: union is defined similarly.

3.7 Distance Functions

Mahout can be given a distance algorithm as the basis for the clustering. We propose an innovative algorithm with which we have been experimenting that uses Google Similarity Distance (GSD) [4] as the foundational measure to leverage towards the set-theoretic Jaccard similarity distance [3]. The essential idea of GSD is using the Google distribution, meaning the number of web pages returned for a search term. The similarity of terms is simply.

$$\frac{|u_i \cap u_j|}{|\Omega|}$$

where:

u are the sets of web pages that contain an occurrence each of the search terms (utterances) u

Ω is the set of all Google web pages

$|x|$ is the cardinality of a set

(4)

Of course it is not quite that simple because of normalization and independence problems. In a related vein, the Jaccard similarity coefficient [3] can be expressed as:

$$\text{sim}(o_k, o_l) = \frac{P(o_k \cap o_l)}{P(o_k) + P(o_l) - P(o_k \cap o_l)}$$

where:

o are objects (concepts)

P are probability masses of objects (concepts)

the intersection represents the common elements of the concepts

(5)

The theoretical requirement for distance may be expressed as a likelihood ratio as is done in track association:

$$\Lambda = \frac{P(\text{same target} | \text{track}_u, \text{track}_v)}{P(\text{different targets} | \text{track}_u, \text{track}_v)} = \frac{P(\text{track}_u, \text{track}_v | \text{one target}) P(\text{one target})}{P(\text{track}_u, \text{track}_v | \text{two targets}) P(\text{two targets})}$$

(6)

Often this becomes:

$$\Lambda = \frac{P(\chi^2(\hat{x}_1, P_1, \hat{x}_2, P_2))}{P(\chi^2((\text{expected track density})^{-1}, P_1, P_2) | v)}$$

(7)

where:

\hat{x} is the estimated track state

P is the estimated track covariance

A way to think of the numerator is as the proportion of probability mass in the intersection of the joint probability density functions of the uncertainty as represented by the merged covariance matrices (see, e.g., ^{16,17}). The denominator is similarly a proportion of the probability mass in an intersection but it is between the expected (mean) minimum distance between two targets. That is, pick any target, find its closest target and then do this repeatedly to get a mean. This is inversely proportional to the track density. The likelihood ratio then can be said to be how much the tracks overlap compared to how much you would expect targets to overlap. Figure 4 illustrates this for 1-dimensional tracks.

Applying this to utterances and concepts can be stated as:

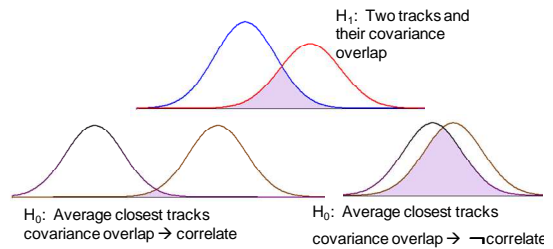


Figure 4. Probability Mass Overlaps in the Track Correlation Likelihood Ratio

$$\Lambda = \frac{P(\text{same concept } C_u | U_i, U_j)}{P(\text{different concepts } C_u, C_v, u \neq v | U_i, U_j)} = \frac{P(U_i, U_j | C_u) p(C_u)}{P(U_i, U_j | C_u, C_v, u \neq v) p(C_u, C_v)} \quad (8)$$

where:

C_u, C_v are concepts to which utterances refer

U_i, U_j are utterances

The numerator is the chance that two similar but different utterances could refer to the same concept while the denominator indicates the chance that the two similar but different utterances could arise from two distinct objects.

3.8 Likelihoods and Similarity Measures for Unigrams

This section introduces the semantic distance formula used in initial experiments using only the Web 1T unigrams. This section also provides basic concepts that are extended to the 5-grams in the next section. Starting with the temporary simplification of the probabilities of the unigrams being equal, then the question analogous to the track correlation problem is, what is the probability mass overlap of the utterances and what is the expected utterance overlap?

In the track correlation likelihood ratio, the numerator (positive hypothesis) can be thought of as measuring the probability mass of the overlapping covariances. The analog for utterances in our uniform probability example would be a *probability mass ratio (pmr)* version of the Jaccard similarity measure:

$$pmr(U_i, U_j) = \frac{\sum_m p}{\sum_{n_i} p + \sum_{n_j} p - \sum_m p} \quad (9)$$

In other words, *pmr* is the probability mass in the intersection divided by the total. Figure 5 illustrates this concept. Analogous to the track correlation example above, the uncertainty pertaining to the two utterances is how many different utterances there are about the same concept. At the unigram level, it is a function of the synonyms. For example, if there are ten synonyms for a concept, then the probability that the same one would show up in two 1-gram utterances about the concept is $1/10^{\text{th}}$. WordNet could be used to lookup the synonyms but it would not have which ones are more likely than others and, in the case of utterances greater than 1 gram, nonsense, and even syntactically wrong, utterances might result. Fortunately, the Web1T database contains the synonyms, i.e., it has not been lemmatized, much less de-synonymized. In the simple $1/10^{\text{th}}$ example, what this means is that we can set any match to the probability of 1 knowing that the Web1T database is roughly 10 times bigger than it would be if were lemmatized and de-synonymized, thereby implicitly resulting in $1/10^{\text{th}}$ because the Web1T probabilities are a tenth smaller than they would be in a lemmatized and de-synonymized database. The likelihood ratio for this unigram constant probability example is:

$$\Lambda = \frac{\frac{\sum_m p}{\sum_{n_i} p + \sum_{n_j} p - \sum_m p}}{E \# \text{matches}(p, n_i, n_j)} \quad (10)$$

$$\frac{(n_i + n_j) - E \# \text{matches}(p, n_i, n_j)}{E \# \text{matches}(p, n_i, n_j)}$$

The likelihood ratio is how much the utterances overlap in mass divided by the chance that the matching would happen just by chance. That is, given two random utterances, the probability that they would have m matching unigrams appears to be binomial. But, 1) matching is trials without replacement, and 2) the sample pairs from the two utterances are not independent. Once a match is declared, the matching grams cannot be matched again with other grams. For example, (the, the, the, it, it) and (the, it) result in match set {the, it}, not {the, the, the, it, it}. This is because matching is a set derived from tuples. Tuple

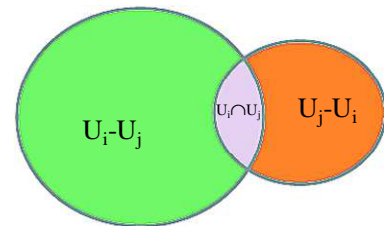


Figure 5. Probability Mass Ratio Venn Diagram

elements are distinguished by their tuple positions. There is no concept of position in a set. The dependence between the trials is shown in the example in Figure 6. Once a match is declared, not only is that trial removed from the sample space, so also are all the joint variables for the match declaration rendered known, i.e., impossible. This results in the following expectation:

$$E \# \text{matches}(p, n_i, n_j) = \frac{0 \cdot (1-p)^{n_i \cdot n_j} + \sum_{m=1}^{\min(n_i, n_j)} m \cdot P(m)}{\min(n_i, n_j) + 1}$$

where:

m is the number of matches

p is the uniform probability for this example

$$P(m) = \left(\prod_{t=0}^{m-1} (n_i - t)(n_j - t) \right) \cdot p^m (1-p)^{(n_i-m)(n_j-m)} \quad (11)$$

For a 5X3:

$$P(1) = 5 * 3 * p^1 (1-p)^{(4)(2)} = 15p (1-p)^8$$

$$P(2) = 4 * 2 * p^2 (1-p)^{(3)(1)} = 8p^2 (1-p)^3$$

$$P(3) = 3 * 1 * p^3 (1-p)^{(2)(0)} = 3p^3$$

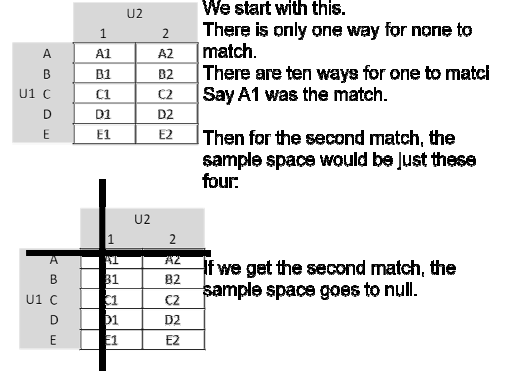


Figure 6. Dependent Trials

Since the Web 1T unigram probabilities are not equal, instead of probability mass, it is more meaningful to switch to unusualness, what might be called semantic mass, the idea being that infrequent grams have more semantic significance than frequent ones. The similarity (H_1) has to be modified as:

$$smr(H_1) = \frac{\sum_{\eta} (P(G_{\eta}))^{-1}}{\sum_t (P(G_t))^{-1}} \quad (12)$$

where:

G_{η} is a unigram $\in U_i \cap U_j$

G_t is a unigram $\in \left(U_i \cup U_j \right) = \left(U_i - U_j \right) + \left(U_j - U_i \right) + \left(U_i \cap U_j \right)$

And, the random chance (H_0) has to be modified as:

$$smr(H_0) = \frac{E \# \text{matches}(\bar{p}, n_i, n_j) \cdot \bar{s}}{(n_i + n_j) \cdot \bar{s} - E \# \text{matches}(\bar{p}, n_i, n_j) \cdot \bar{s}}$$

where:

\bar{p} is the average frequency for unigrams

\bar{s} is the average semantic mass for unigrams ($=1/\bar{p}$)

$E \# \text{matches}(\bar{p}, n_i, n_j)$ is as defined above

(13)

The likelihood ratio is then $\Lambda = \frac{smr(H_1)}{smr(H_0)}$. The inverse of this appears to behave like a distance so that

$dist(U_i, U_j) = (\Lambda)^{-1}$. That is, high likelihoods indicate semantic closeness because, 1) they are similar (approaching 1) and the degree of match by chance is unlikely (approaching 0).

3.9 Extension to 5-grams

The extension to 5-grams requires consideration of proper subgrams, e.g., in the constant probability example, the H_0 becomes:

$$smr(H_0) = \sum_{s_g=1}^5 \frac{E \# matches(n_i^{s_g} n_j^{s_g} \bar{p}^{s_g})}{(n_i^{s_g} + n_j^{s_g}) - E \# matches(n_i^{s_g} n_j^{s_g} \bar{p}^{s_g})} \quad (14)$$

where:

\bar{p}_{s_g} is the average frequency for s_g -grams, i.e., $1/\#s_g$ -grams

$n_i^{s_g}$, the number of s_g proper subgrams in $|U_i|_{s_g}$

The H_1 with all the Web1T gram frequencies:

$$smr(H_1) = \frac{\sum_{s_g=1}^5 \sum_{\eta} \left(P(G_{\eta}^{s_g}) \right)^{-1}}{\sum_{s_g=1}^5 \sum_{t} \left(P(G_t^{s_g}) \right)^{-1}} \quad (15)$$

where:

$G_{\eta}^{s_g}$ is a s_g -gram $\in U_i \cap U_j$

$G_t^{s_g}$ is a s_g -gram $\in \left(U_i \cup U_j \right) - \left(U_i \cap U_j \right)$

The likelihood ratio remains $smr(H_1)(smr(H_0))^{-1}$ and $dist(U_i, U_j) = (\Lambda)^{-1}$.

4. PRELIMINARY EXPERIMENT DESIGN

In order to test and present unrestricted results, the test case we used was an open domain activity model for Search and Rescue (SAR) initially developed by the Object Management Group (OMG). It is probably less complicated than an actual terrorist activity model but our results should be extensible. We encoded the activity model into RDF triples statements. After tokenization and lemmatization, it had 272 unigrams. Then we picked some news stories were for testing, listed in Table 2.

Table 2. News Stories Picked for Initial Testing

Name	Topic	Unigrams
Crimea	Change course in Crimea or face costs, West warns Russia, Simferopol, Ukraine (CNN)	612
Malaysia	Malaysian plane may have flown long after last contact (CNN)	478
Pistorius	Gruesome shooting scene photos sicken Oscar Pistorius at murder trial, Pretoria, South Africa (CNN)	420
NYC	Amid search for missing in NYC blast, loved ones, friends reflect on those killed, New York (CNN)	490
SXSW	2 killed when intoxicated driver plows car into crowd at SXSW, police say, Austin, Texas (CNN)	334
Student	New Jersey student who sued parents returns home (CNN)	268
Diplomat	Charges against Indian diplomat who was strip searched dismissed, New York (CNN)	185
Overtime	Obama says it's time to change overtime rules, Washington (CNN)	183
Winter	Winter's last hurrah? Storm spreads snowy misery from Illinois to Maine (CNN)	182
Holder	Holder: I back a plan to reduce some drug-related sentences, Washington (CNN)	125

Since one of the stories was about the Malaysian aircraft loss, we were expecting high scoring to the SAR activity model.

5. PRELIMINARY EXPERIMENT RESULTS

For these early test results, several approximations were taken:

- a. Only one model was developed and setup.
- b. Only unigrams were processed.
- c. A “with replacement” formula was used for $\text{smr}(H_0)$ instead of the “without replacement” described herein just to limit formula complexity in this early testing.
- d. The H_0 formula was for the expected overlap between random utterances. As described in the next paragraph, there could be ways to use the database to compute an expected minimum distance, as is done in the track correlation analog.
- e. Because database maintenance was not setup, a match between grams not in the database was considered to imply a frequency of two and a non-match a frequency of one. This, obviously, underscored intersections and underlaps for grams not in the database that were repeated in the input documents.
- f. Only tested with K-Means clustering.
- g. In addition to the SAR activity model, the Mahout clustering algorithm required an additional initial cluster to show the differences between the data. The Other cluster was initialized with all the input news stories. There are probably better choices that would give the inputs something to cluster around when they do not correlate to the models. Because it was large and contained the input texts, it tended to over attract input text so we diluted the mass for all its grams. Further work on the Other cluster is needed as described in the next paragraph.
- h. The Web1T database frequencies were for linguistic variants not lemmatized forms.

Early testing showed that new grams (ones that were not in the Web1T database) had masses that caused very high match or underlap scores. Proper names, dates, and clock times caused strong scoring. Some of this was excessive because we had not yet implemented database maintenance (described below). But some was also caused by the Other cluster having too many matches with the news stories (all). Since a match implies a frequency of at least two, we used two for the frequency for grams not in the database. This reduced the excess match scoring some. For Other’s over-attraction, we diluted all its masses by a constant divisor. In future work, we would like to create Other in a more random but still slightly attracting manner. With these fixes, the news stories clustered to SAR and Other as shown in Table 3.

Table 3. Clustering Choices and Semantic Distances to Clusters

Clustered-to	News Story	Distance to SAR	Distance to Other
Other	Crimea	14.60401132250545	3.0267857835737243E-4
Other	Diplomat	12.418283382110316	1.5572923015108885E-4
Other	Holder	2.783971518832796	0.2737367533233926
Other	Overtime	3.1201080786575544	0.12214595018128663
Other	Pistorius	9.601955716263772	3.607117339160669E-4
Other	Student	5.41586487268464	3.1167233489480214E-4
Other	Winter	2.7848342073216794	3.562664084992547E-4
SAR	Malaysia	1.4240469496447567E-4	11.915843362875526
SAR	NYC	2.7653785658511863E-4	9.425796796175135
SAR	SXSW	1.0871790108830683E-4	11.520216814093697

This seemed reasonable. Some of the score drivers for SAR were:

- a. NYC contained victim, danger and fire

- b. SXSX contained driver, cpr and victim
- c. Malaysia contained aircraft, transmitter, searcher

6. FUTURE WORK

There are several areas where additional work needs to be done, described in the following subparagraphs.

6.1 Database Maintenance

The database can be maintained to accumulate and adapt to new terms, adjust for changing probabilities, and deprecate obsolete terms. When inputs arrive, they should be probably processed to update the frequencies in the database, e.g., increment the gram frequency every time it is matched. In the event that a 1 to 5 gram input occurs that is not in the WebIT database, it could be added on the fly and given a frequency of 1, i.e., a probability of $1/\alpha^5$. Obsolescence, retirement, or archiving should be researched. A way might be to periodically reduce every frequency by trimming percentage. For example, the trigger could be that when the database is $x\%$ over the original size. Then every frequency would be reduced by $x\%$. Over time, the frequency for grams that were not getting matches would tend to zero and could be pruned out. Grams that were being matched often would get frequency increments faster than the trimming.

Multiple corpii could be employed and, if not already grammed and counted, could be tokenized and processed for frequency counts. However, as noted in [4], once large Google samples (e.g., 1 trillion) are taken, they do not vary much across other samples or by the addition of more samples.

6.2 Statistical Testing

Additional activity models and news stories or other text inputs should be tested. Many activity models are available in the Department of Defense Warfare Mission Area (WMA) repository¹⁸ and the Intelligence Community Enterprise Repository and Registry (ER2) on the Joint Worldwide Intelligence Communications System (JWICS, high-side classified). Although the models are usually not in an RDF triples format, there are procedures for converting them.

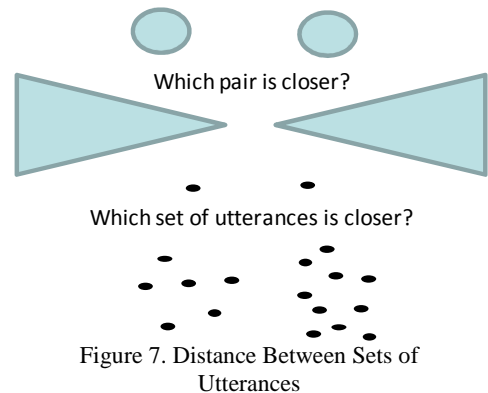
Further testing should be designed to measure correctness of the semantic distance. The analog to the track correlation problem may provide a guide. Early testing is typically simulation based, where “ground truth” targets are created for expected operating environments. These are processed through multi-sensor and multi-platform simulators of varying fidelity, e.g., for detection range and measurement noise. By comparing the correlation decisions to the ground truth targets, various metrics for goodness of the correlator can be computed. The next phase of testing is typically at a land-based test site such as the AEGIS test site at Wallops Island. Next is the testing at instrumented range. Lastly is testing in a controlled but live exercise or test environment.

6.3 Clusters

The “Other” cluster should be researched. Although it is necessary for this cluster that any document can gravitate to it, it is also necessary that this cluster be weak. It may be that a random collection of popular (hence less semantically massive) terms would work.

6.4 Complex Object (Document) Distance

Whether to process in Mahout and the distance algorithm sentence-sized or larger document-sized utterances needs to be researched further. First, we find the distances of sentences in the news story to the activity model sentences. Next we compute the distance of the news story to the activity model. As Figure 7 illustrates, there are many different distances between complex objects. Our approach is analogous to center of mass or mean probability event using what is similar to the Hausdorff distance. In other words, we take the average of the minimum distances between utterances. It is not necessary to weight the average because the distances already account for improbability of the grams within the utterances. The minimums to use are selected using a variation of the happy marriages assignment algorithm with the rules that no singles must remain. (In general this implies polygamy.)



$$\frac{\sum_{\max(m,n)} \min_{i,j} \|U_i, U_j\|}{\max(m,n)}$$

(16)

where :

m is the number of utterances in a news story

n is the number of utterances in the activity model

6.5 Content Acquisition and Conditioning

Content acquisition could be available in both real-time and batch modes in the following way. Real-time processing will use the popular Storm architecture with bigger and longer batch runs going to Hadoop map/reduce jobs. A pluggable connector system will parse text of varying levels of structure from many different feeds like web pages, tweets, databases, and emails with potentially geospatial, strategic or tactical sources. This data staging environment will include tools such as Tika and Boilerope to extract knowledge from this heterogeneous data set and could also be the place for any necessary pre-processing such as normalization, compartmenting, source tagging or storage of the original feeds. With Hadoop, both the storage and processing aspects of content acquisition will be distributed and scalable.

Analogous to initial sensor data processing in the data fusion process, some source-specific conditioning should be researched. For example, the CNN news stories were preceded with metadata that perhaps should have been segregated and processed differently. There are also questions about punctuation, proper names, etc. that could be researched.

6.6 Data store with new possible relationships

The output of the clustering process may be stored in a Hadoop NoSQL database like Apache HBase. HBase is a columnar database able to handle structured and unstructured data by using key value pair that can also be grouped into column families. It is a scalable, highly-available and distributed application making it easy to handle many text sources in a timely manner and at different locations. At the end of every clustering run whether full or canopy, simple functions like ClusterDump can be executed to output and format a block of information, including the centroid vector and topweighted terms in the cluster. These results would then be timestamped and saved. The Hadoop architecture includes many analysis tools including Hive that allows SQL-like querying on the cluster data. Other than storing results, HBase could also be used to cache incoming text and hold the original data for further processing or hold intermediate steps like NLP for checking and cross referencing.

6.7 Operator Utilization

The resulting relationships may be stored in the system and when FMV/imagery or a text document is accessed by the operator, the system would indicate possible related FMV/imagery or text documents. They would be displayed according to the strength of the relationship. The operator can also use the system by querying for AoI for which an activity model was defined, as described in paragraph 3.1.9. In this mode of operation, all the related documents and FMV/imagery related to all variations of the activity model would be provided to the operator, again in likelihood order.

6.8 Lemmas and Synonyms

Tools like WordNet could take the terms of the model statements and provide synonyms for each term to create linguistic variations of the same model. There are two areas that would need further work. One is the combinatoric explosion, e.g., if the statement had 7 terms and each term had 4 synonyms there would be 16,000 combinations. Another is that many combinations may not make sense, even syntactically.

As mentioned before, the Web1T database was not lemmatized and this turned out to be an advantage because it implicitly contained the synonyms. But this had the bad effect of underestimating the frequencies for a concept. It needs to be researched what the performance gain would be to lemmatize the Web1T and keep track of the linguistic variants that pertain to a lemma form. That is, lemmatizing would reduce some grams to a common form. The frequencies for these could be combined but the linguistic variants might be maintained for gram matching. Part of this work would have to address the computational challenges of manipulating the Web1T database. The lemmatizing algorithm and computing plant we used in these experiments would have taken days to weeks to perform lemmatization.

6.9 Null Hypothesis

Better ways to estimate H0 need to be researched. It probably should be, as in the track correlation analog, the expected best overlap, not the expected random overlap as was done in these experiments. A way could be to use the Web1T database to compute average minimum overlaps between different utterance sizes, with the utterances formed from random combinations of grams. However, this could be a time consuming computation, on the order of days or weeks.

6.10 Ontology and Other JDL Levels

Once the association between utterances and documents is indicated, further content-based analysis could be performed. The activity models are structured into a formal ontology we built for the DoD and that is in widespread use for capabilities development and acquisition. It is a rigorous data model for coalition data exchange and integration and mathematical analysis of exchanged datasets. The foundation is formal, higher-order and 4D, based on four dimensionalism [19,20]. It is extensional (see Extension [metaphysics]), using physical existence as its criterion for identity. In practical terms, this means the ontology is well suited to managing change-over time and identifying elements with a degree of precision that is not possible using names alone. The methodology for defining the ontology is very precise about criteria for identity by grounding reasoning about whether two things are the same using something that can be accurately identified. Thus the clustering could be considered ontology-aided clustering. This could provide opportunities for further data exploitation.

7. SUMMARY

This work used the JDL data fusion process and algorithm patterns to a specific type of next generation analytics -- the detection of Activities of Interest (AoI). Big data technologies (Hadoop, Mahout) were used with a custom semantic distance algorithm and an ontologic model of AoI. It has yielded some positive results, even with limited experiment time, approximations, and a relatively small computing plant. There is much more research to conduct, however, before it will be known if the approach would be robust in real-world situations. On the other hand, there appear to be other potential applications of the JDL data fusion patterns to next generation analytics.

REFERENCES

-
- [1] Hassan Khosravi and Bahareh Bina, "A Survey on Statistical Relational Learning", *Advances in Artificial Intelligence: 23rd Canadian Conference on AI Ed. Atefeh Farzindar, Vlado Keselj, Springer-Verlag*, 2010
 - [2] Heinsohn, Jochen. "Probabilistic description logics", *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, 1994
 - [3] ZhiMing Li, Shanping Li, Zhiyu Peng, "Ontology matching based on Probabilistic Description Logic", 7th WSEAS Int. Conf. on Applied Computer & Applied Computational Science (ACACOS '08), Hangzhou, China, April 6-8, 2008
 - [4] Rudi L. Cilibrasi and Paul M.B. Vitanyi, "The Google Similarity Distance", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, No 3, IEEE, March 2007
 - [5] R. B. Bradford, "Relationship Discovery in Large Text Collections Using Latent Semantic Indexing", *SIAM Conference on Data Mining Workshop on Link Analysis, Counterterrorism and Security*, 2006
 - [6] Pavel Shvaiko and Jerome Euzenat, "Ontology Matching: State of the Art and Future Challenges", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, No. 1, IEEE, January 2013
 - [7] Henrik Leopold, Mathias Niepert, Matthias Weidlich, Jan Mendling, Remco Dijkman, Heiner Stuckenschmidt, "Probabilistic Optimization of Semantic Process Model Matching", *BPM'12 Proceedings of the 10th international conference on Business Process Management*, 2012
 - [8] Gowri Shankar Ramaswamy, F. Sagayaraj Francis, "Latent Table Discovery by Semantic Relationship Extraction Between Unrelated Sets of Entity Sets of Structured Data Sources", *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 2, March 2011
 - [9] Chee Yee Chong, D. Hall, M. Liggins and J. Llinas (editors), Distributed Data Fusion for Network-Centric Operations, CRC Press, Nov 2012
 - [10] Sean Owen, Robin Anil, Ted Dunning, and Ellen Friedman, Mahout in Action, Manning Publications Co., Manning Publications Co. 20 Baldwin Road PO Box 261 Shelter Island, NY 11964, First edition, 201
 - [11] White, Tom, Hadoop: the Definitive Guide, O'Reilly, 2012

[12] <http://catalog ldc.upenn.edu/LDC2006T13>

[13] Functional Description of the Data Fusion Process, Data Fusion Development Strategy, Office of Naval Technology, November, 1991

[14] Steinberg, A. N., Bowman, C. L., White, F. E., "Revisions to the JDL Data Fusion Model", <http://www.dtic.mil/dtic/tr/fulltext/u2/a391479.pdf>

[15] McDaniel, D., "An Information Fusion Framework for Data Integration", Software Technology Conference, 2001, http://www.silverbulletinc.com/solutions/downloads/files/An_Information_Fusion_Framework_for_Data_Integration.pdf

[16] Yaakov Bar-Shalom, Thomas E. Fortmann, Tracking and Data Association, Academic Press, 1988

[17] Abramowitz, M., Stegun, I. A., (ed), [Handbook of Mathematical Functions], U.S. Department of Commerce, National Bureau of Standards, Applied Mathematics Series 55, 1964, pg. 940

[18] <https://sadie.nmci.navy.mil/jafe/default.aspx>

[19] Rea, M. C., "Four Dimensionalism", The Oxford Handbook for Metaphysics. Oxford University Press

[20] Sider, Theodore, Four-Dimensionalism: An Ontology of Persistence and Time, Oxford University Press, 2003