# When Triples Are Not Enough:
# Exploring Context Issues Within Soft Data

**Kellyn Rein**
Fraunhofer FKIE
Fraunhofer Str. 20
53343 Wachtberg
GERMANY

kellyn.rein@fkie.fraunhofer.de

## *ABSTRACT*

*Making sense of the vast amounts of new information being created daily is a major challenge for the intelligence community. Political unrest, religious and ethnic factionalization as well as criminal activities in support of terrorism in unstable areas cannot be fought by traditional warfare methods. As terrorist groups are increasingly using the Internet for communication, recruiting and other purposes such as intimidation of the populace, there is an ever increasing need to deal with text-based information. The sheer volume of this information means that automatic processing of text-based information is vital. The unstructured nature of text makes automatic processing difficult, so much of the existing text analytics technology attempts to identify structures for sense-making. However, there are limitations in the technology, as often the extracted information is taken out of context, thereby subtle relationships become lost. Additionally, attempts to structure text-based information via databases or ontologies require a priori decisions about what is important and about relationships between data and entities is important. However, the enemy does not stand still; coping with the chameleon-like shifting of behaviors of ever-learning foes requires flexible management of information acquires, as we do not always know in advance what information will someday prove to be important. We will look at a possible solution for this: Battle Management Language.*

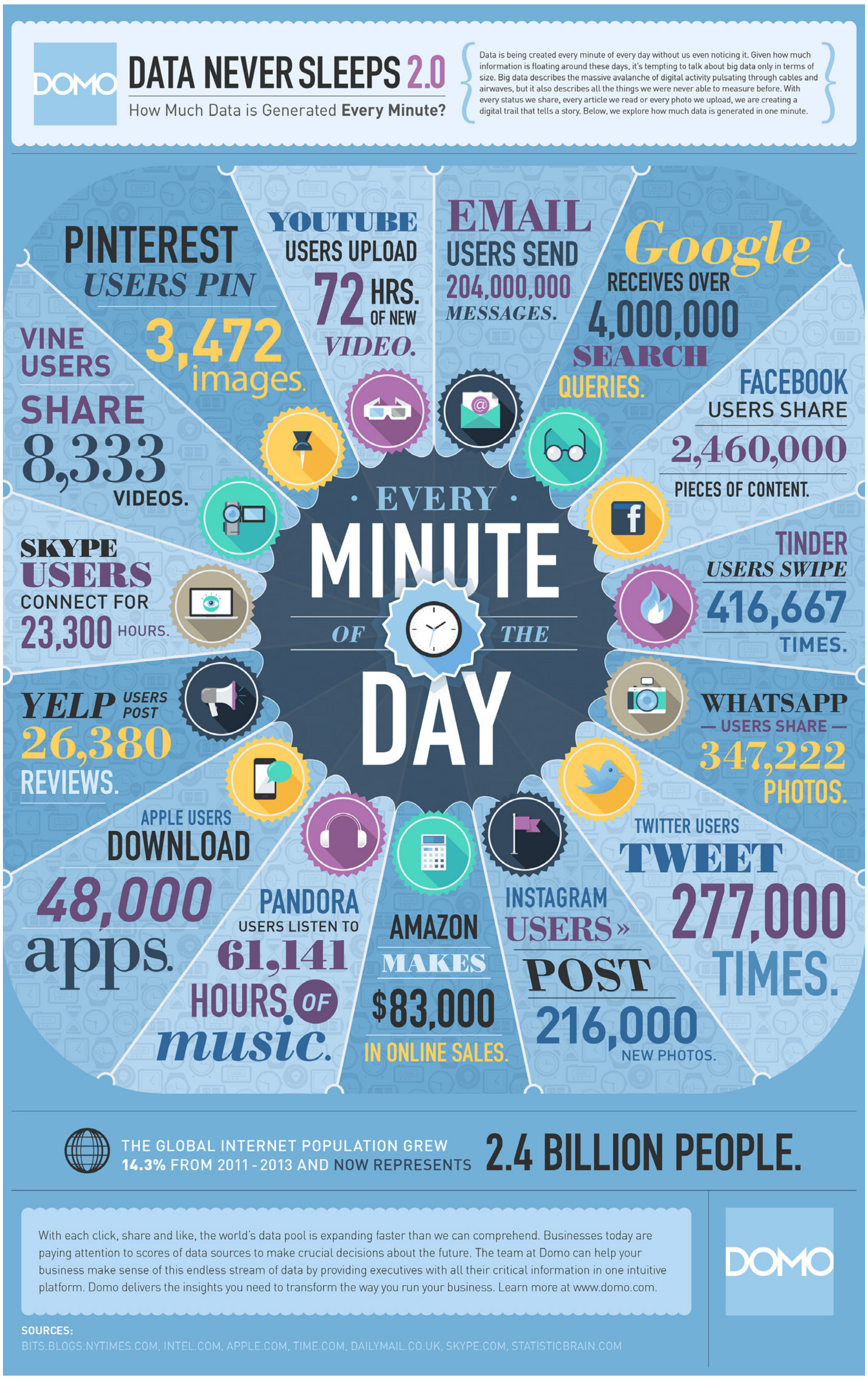
## 1.0 INTRODUCTION

Every day vast quantities of new information are being generated –far too much for any single individual, or indeed any team of analysts, to process. And it is growing dramatically year by year, indeed day by day.

> According to computer giant IBM, 2.5 exabytes - that's 2.5 billion gigabytes (GB) - of data was generated every day in 2012.[1]
>
> Think of it this way—five exabytes of content were created between the birth of the world and 2003. ***In 2013, 5 exabytes of content were created each day***. [2] (italics added)

Figure 1 below shows a graphic produced by DOMO in 2014 showing a snapshot of the amount of data on social media that was being generated via the Internet every day at that time. Today, the daily volume certainly exceeds the numbers being shown in the graphic. Note that this does not include other sources of internet information such as on-line news sources, blogs, or government information sites, which are significant sources of intelligence information.

It is immediately clear that making use of this information is only possible using automatic data processing based upon the sheer volume. Powerful algorithms are needed to sift and sort through this ocean of information, in order to detect patterns or links. Sense-making requires that this information be "understood" on a certain level, so that associations between pieces of data can be drawn and the resultant information may be exploited for decision making and threat recognition.


DOMO
DATA NEVER SLEEPS 2.0
How Much Data is Generated Every Minute?
PINTEREST USERS PIN 3,472 images.
YOUTUBE USERS UPLOAD 72 HRS. OF NEW VIDEO.
EMAIL USERS SEND 204,000,000 MESSAGES.
Google RECEIVES OVER 4,000,000 SEARCH QUERIES.
FACEBOOK USERS SHARE 2,460,000 PIECES OF CONTENT.
VINE USERS SHARE 8,333 VIDEOS.
SKYPE USERS CONNECT FOR 23,300 HOURS.
TINDER USERS SWIPE 416,667 TIMES.
YELP USERS POST 26,380 REVIEWS.
WHATSAPP USERS SHARE 347,222 PHOTOS.
• EVERY • MINUTE OF THE DAY
APPLE USERS DOWNLOAD 48,000 apps.
PANDORA USERS LISTEN TO 61,141 HOURS OF music.
AMAZON MAKES $83,000 IN ONLINE SALES.
INSTAGRAM USERS» POST 216,000 NEW PHOTOS.
TWITTER USERS TWEET 277,000 TIMES.
THE GLOBAL INTERNET POPULATION GREW 14.3% FROM 2011 - 2013 AND NOW REPRESENTS 2.4 BILLION PEOPLE.
DOMO

Data is being created every minute of every day without us even noticing it. Given how much information is floating around these days, it's tempting to talk about big data only in terms of size. Big data describes the massive avalanche of digital activity pulsating through cables and airwaves, but it also describes all the things we were never able to measure before. With every status we share, every article we read or every photo we upload, we are creating a digital trail that tells a story. Below, we explore how much data is generated in one minute.

With each click, share and like, the world's data pool is expanding faster than we can comprehend. Businesses today are paying attention to scores of data sources to make crucial decisions about the future. The team at Domo can help your business make sense of this endless stream of data by providing executives with all their critical information in one intuitive platform. Domo delivers the insights you need to transform the way you run your business. Learn more at www.domo.com.

SOURCES:
BITS.BLOGS.NYTIMES.COM, INTEL.COM, APPLE.COM, TIME.COM, DAILYMAIL.CO.UK, SKYPE.COM, STATISTICBRAIN.COM

**Figure 1: The per-minute explosion of new content – in 2014.[3]**

One problem is that much of that information is unstructured: in an interview with BBC News, Laurie Miles, head of analytics for big data specialist SAS, indicated that majority ("about 75%") of this new data is unstructured, coming from sources such as text, voice and video."[1] Unstructured data is data which is not stored in a structured format such as a database or ontology. It includes various forms such as emails, word processing documents, multimedia, video, PDF files, spreadsheets, text messaging content, digital pictures and graphics, mobile phone GPS records, and social media content.

Unstructured data is problematic for computer processing and sense-making. Exploitation of unstructured data is dependent on the type of unstructured data. Some types require specialized algorithms, e.g., image processing. Others require pre-processing before structuring for example, PDF files are often converted to normal text files in order to run more standard text analytic algorithms which use various techniques to analyze and structure text.

Natural language text as a significant source of intelligence information has become increasingly important as the use of on-line information for recruitment, propaganda and criminal activities by terrorists and enemy forces is increasing. Natural language text contains factual information about events or individuals that may be useful for situation awareness and management, but also information which reflects speculation, inference, attitude, mood, belief, etc., that may be of use for intelligence purposes, but which affect decision-making in other ways by providing a context for the data contained in the text.

However, before we go further, definitions of the terminology are in order. This is particularly important for the concept ***context*** which means quite different things for the hard fusion and for hard fusion communities.

## 1.1 Hard vs. Soft Data

The Cambridge Business English Dictionary defines soft data as “information about things that are difficult to measure such as people’s opinions or feelings” [4] and hard data as “information such as numbers or facts that can be proved.”[5]

Objectivity differentiates between hard and soft data thus:

> Hard Data is defined as data in the form of numbers or graphs, as opposed to qualitative information[1]. In the world of Big Data and the Internet of Things (IoT), Hard Data describes the types of data that are generated from devices and applications, such as phones, computers, sensors, smart meters, traffic monitoring systems, call detail records, bank transaction records, among others. This information can be measured, traced, and validated.… Soft Data [is defined] as human intelligence, data that is full of opinions, suggestions, interpretations, contradictions and uncertainties.[6]

Within the fusion community soft data is usually described as data collected from humans in the form of text (natural language, whether written or spoken) rather than the “hard” data from devices (sensors). Soft data may contain “hard” data, that is, factual information may be contained in a natural language utterance. However, determination of what is fact and what is opinion requires some complex linguistic processing.

## 1.2 Your Context is Not My Context

As it turns out, in the worlds of hard and soft data, the meaning of the word “context” depends upon, well, the context.

Oxford Dictionaries[7] offers the following two-pronged definition:

1. the circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood.
2. the parts of something written or spoken that immediately precede and follow a word or passage and clarify its meaning.

Likewise, dictionary.com[8] gives two variations on context:

1. the parts of a written or spoken statement that precede or follow a specific word or passage, usually influencing its meaning or effect.

2. The set of circumstances or facts that surround a particular event, situation, etc.

Within the sensor data fusion community, “context” indicates knowledge about the immediate environment of the sensors. For example, Schilit and Theimer [9] use context to indicate the location, identities of nearby people and objects, and changes to those objects. Thus the meaning of context for the hard data community aligns with definition 1 from the Oxford Dictionary and definition 2 from dictionary.com.

In contrast to sensors, for which “context” is generally accepted to be the physical environment in which they are located, the physical environment, indeed the actual location, of the human “sensor” may be completely irrelevant. The intelligence analyst providing white paper about power struggles in the Middle East may be writing from a physical location a great distance from where these struggles are taking place. Thus “context” has a quite different meaning for soft data.

At the data level, the soft data community is generally concerned with context within the text itself, in particular at the sentence level, which provides meaning to the words and phrases being used. For example, the presence of lexical markers of uncertainty (e.g., “possible”, “probable”, “unlikely”) provide a context as to the writer’s stance on the truth of the proposition made within a given statement. Likewise, indicators of opinion or hearsay (e.g., “believe”, “think”, “our sources say”) provide clues as to the source of the information in the sentence, thereby also affecting the quality of the information.

Additionally, “context” of soft data may lie in a variety of other factors which may be difficult to measure: political orientation of the writer, overt or covert intent in the writing, social and cultural background. Misunderstanding of natural language communications can often result when the writer’s context is ignored or not taken into account. Similarly, the meaning of the information may vary depending upon the perspective of the consumer, that is, depending upon the expectations, knowledge, prejudices, experience of the information recipient, there may be different spin from analyst to analyst.

Finally, because natural language utterances seldom contain a single “signal” but rather tend to combine multiple pieces of information, we often can only make sense of those when we preserve the complete complex message rather than individual components, as the context of the individual components may later provide deeper meaning or may be re-interpreted due to information which is acquired later.

## 2.0 SITUATION AWARENESS VS INTELLIGENCE

To a certain extent, the timeline for consumption of data plays a role in its interpretation, therefore a brief discussion of the context dimension “timeline” is worthwhile.

### 2.1 Situation Awareness – Shorter Timelines, Smaller Footprint

According to Endsley, [10] situation awareness is "the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the *projection of their status in the near future*," (italics added). Situation awareness is a continuous updating of important environmental elements in the area of interest which includes the current locations of military units, both friendly and hostile, tracking movements of personnel and equipment, as well locations and conditions of facilities, status of important structures such as bridges and roads. Additionally, this may also include background information on non-military or paramilitary activities such as refugees, political climate, tribal coalitions, etc. Situation awareness information is often captured and displayed visually on maps in the command and control (“C2”) systems being used to give decision-makers an overview of the current state of affairs. In the case of trackable changes, such as the movements of individuals, vehicles or military units, there may be some projections as to a future state (e.g., “where that column may be in one hour”) to further assist commanders.

Thus the timeline is generally relatively limited, the geographical area likewise usually restricted, and the possible threats relatively well understood or defined by experience. Situational awareness is generally restricted to a relatively short timeline (current state plus projections that may forecast seconds, minutes or perhaps hours) and decision support focuses on such things as use of resources, blocking hostile activity and the protection of life and property. Thus situation awareness depends on knowing or predicting the state of the elements of interest in the (complex) environment under consideration. Very often a significant percentage of the information underlying the situation awareness picture comes from devices such as video and still cameras, motion detection sensors, acoustic sensors, radar, etc. Algorithms to make sense of the data produced by the devices are improving continuously.

Natural language information for situation awareness often concerns movements or changes within the area of interest, and text analytic processing to update the situation awareness may be relatively lightweight. Thus new information which is not concerned with the immediate situation and current location of interest will not play a role.

### 2.2 Intelligence – Long(er) Timelines, Larger Footprint

In contrast to situation awareness, sense-making for intelligence purposes often involves timelines which are much longer, covering weeks, months or years rather microseconds, minutes or hours. Furthermore, the geographical area covered may be very extensive. For example, the current fight against ISIS involves information-gathering on several continents, and that information is to a very great extent text-based. Additionally, in such situations, intelligence work is carried out over longer periods of time, during which assets may be acquired and set in motion and often requires the cooperation of multiple agencies. The data collected may include focused reports from intelligence assets and analyses from various agencies, but also may include many types of open sources including news sources, government documents and research results. Additionally, in such wide-ranging activities as the battle against ISIS, there are multiple natural languages involved, as well as cultural and social backgrounds which influence the acquisition of and the quality of information. Thus, the environmental scanning may be subtle and complex, involving political and cultural changes, economic shifts, and other trends which may be indicative of activities which pose threats. In such cases, open sources such as newspapers, television, government reports, blogs, social media, etc. as well as reports from intelligence assets and analysts are useful sources of information. However, the information in these sources must first be understood and then collated and examined for patterns of behavior that are indicative of developing threats.

## 3.0 PROCESSING NATURAL LANGUAGE DATA

As previously mentioned, natural language data is unstructured and therefore difficult to process automatically. In this section, we take a look at some of the technologies used for processing natural language information: information extraction and text analytics. Some of the text analytics processes are rather shallow – for example, sentiment analysis basically skims texts looking for certain words which describe emotion (e.g., “angry”, “furious”, “disappointed”) while running a tally of how often they appear, which then becomes a basis for measuring prevailing sentiment among a group of people. Other processes require more in-depth analysis of the grammatical structures contained within a sentence to produce usable data. We will begin by looking at information extraction, the more complex process performed on a complete sentence, and follow up with a discussion of text analytics, a variety of different processes that focus on either shallow processing of some lexical and grammatical forms, or which utilize the results of the more complex information extraction process.

### 3.1 Information Extraction

Human beings convey information using natural language formulations which may represent a wide spectrum of information about past present or future activities, events, beliefs, interpretation of all of the preceding and much more. Information extraction uses natural language processing algorithms to automatically extract useful information from unstructured or semi-structured text documents. There are several layers of processing needed for information extraction. Jenge et al. [11] describe a standard IE processing pipeline consists of the following elements:

1. A tokenizer that determines individual tokens of the text, i.e., individual words, numbers, abbreviations and punctuation marks.
2. A gazetteer that compares these tokens to elements of lists containing the names of various types such as person names, organizations, towns, countries, landmarks, etc.. Tokens which match one or more elements in the list will be annotated with the corresponding type, e.g. *male forename*. This element of the pipeline may be very domain-specific; for example, the list containing towns and landmarks may be limited to a specific region or country.
3. The sentence splitter determines the boundaries of sentences, which is actually more difficult than it may first seem. Rules about splitting are required to prevent the sentence splitter from, say, considering every period as the end of a sentence; such rules prevent the splitter from mis-identifying the end of a sentence after a “Mr.” or “Dr.” or an “e.g.”.
4. The part-of-speech-tagger determines the part-of-speech of the word tokens (e.g., “noun”, “verb”, “preposition”) based upon the definition of the word, as well as the context in which it appears (“age”, for example, can be either a noun or a verb, depending upon how it is used in the sentence, which is determined by the words which precede or follow it).
5. A named-entities transducer combines elements annotated by the gazetteers in step 2 above. For example, for the sequence “Dr. Mohammed el-Baradei”, the gazetteer will provide the annotations *title* for “Dr.”, *male forename* for “Mohammed” and *surname* for “el-Baradei” whereas a named-entity transducer uses these annotations to calculate the annotation *person* for the whole sequence. It should be noted that gazetteer lists are often domain specific for a given situational context to reduce underlying processing resources.

Once the above in the data processing pipeline are completed, the next step is to look deeper and determine the actions, events and situations reported in the text and assign semantic roles to their participants. The expression of actions, events and situations is the domain of the verbal vocabulary, i.e. of verbs and to some degree also of deverbal nouns (nouns derived from verbs).The first step is to parse the sequences of tokens according to an underlying (usually context-free) grammar. Using parsing methodologies based upon grammatical rules, the relationships between the tokens of the sentence can be determined. (For a fuller description of parsing, both shallow and deep, cf.11, 12, 13.)

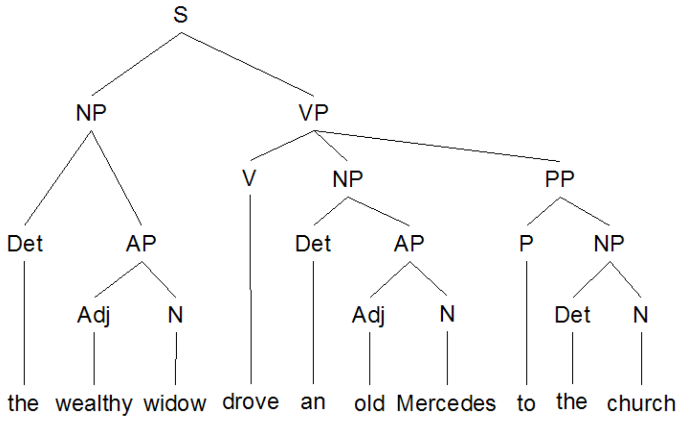
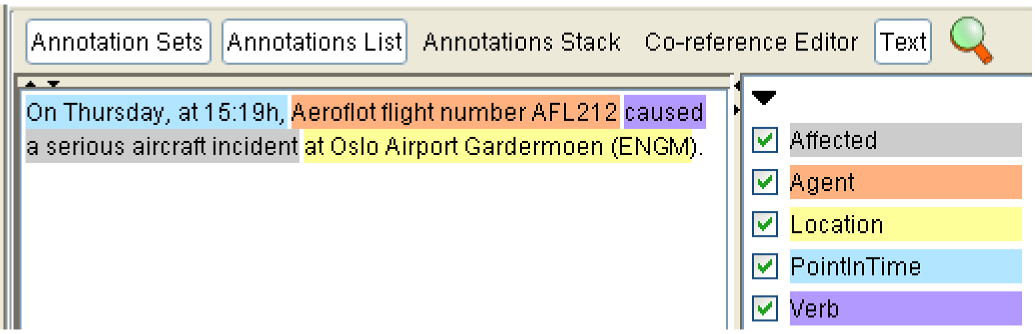


**Figure 2: The complete parse tree of "The wealthy widow drove an old Mercedes to the church."[jenge et al]**

The next step is to assign semantic roles to constituents, a process which is called "semantic role labelling." The process of semantic role labelling links word meanings to sentence meaning by exploiting syntactic, lexical, and semantic information. In English, syntactic information is based upon word order information. For example, in the sentences "dog bites man" and "man bites dog", who is doing the biting and who is being bitten is determined by who appears before the verb and who appears after. Lexical information is provided mostly by verbs and prepositions. For example, a prepositional phrase that starts with the preposition "*at*" normally signals a constituent that will be labelled either *location* (e.g., "*at the townhall*") or *point in time* (e.g., "*at one o'clock*"). Figure 3 shows an example of semantic role labelling using MIETER [ ] which has been developed by Fraunhofer FKIE. Semantic information normally provides constraints that can be used to decide among alternative roles when disambiguation is required.



On Thursday, at 15:19h, Aeroflot flight number AFL212 caused a serious aircraft incident at Oslo Airport Gardermoen (ENGM).

☑ Affected
☑ Agent
☑ Location
☑ PointInTime
☑ Verb

**Figure 3: preliminary labelling of semantic role information as calculated by MIETER developed by Fraunhofer FKIE.**

As can be seen in the example sentences in Figures 2 and 3, a single sentence may contain a myriad of individual pieces of data: the widow is wealthy, she drove a Mercedes, she can drive, the car is old, she went to the church for some reason, the aircraft incident was serious, it happened on a Thursday, it happened at 15:19, the aircraft involved belonged to Aeroflot, its flight number was AFL212, and so on.

Any and all of this information may be of interest in the future. Once we have processed it, we want to store the information which we have derived. Storage of soft data will be discussed in Section 4.

## 3.2 Text Analytics

Text analytics (previously referred to as "text analytics") consists of a variety of techniques for analyzing natural language text and retrieving certain types of information from the documents at hand. Using analysis techniques based upon lexical and grammatical patterns in the language used, sentences can be parsed so that information about documents as well as individual structures within documents and sentences (and to a small extent between sentences) may be discovered. These techniques include (but are not limited to):

*Document classification:* Using a variety of techniques based upon linguistic and statistical analysis, documents may be classified (type of content it contains, which natural language in which it has been written, etc.), summarized (what the document is "about") or clustered (based upon a predefined or learned classification). When working in an international environment, this plays an important role, as the further processing for information extraction is dependent upon the natural language in which the document is written.

*Named entity recognition/pattern recognition:* by using characteristics of the natural language, names of individuals, places, organizations, and such may be retrieved. For example, in English, the capitalization of nouns is generally indicative of a proper noun. That is, "President" refers to a specific individual (e.g., Barack Obama), whereas as "president" is generally non-specific. Useful patterns such as telephone numbers, social security numbers and email addresses may be recognized and extracted by looking for specialized characters such as "@" or numbers which conform to a pattern (e.g.: xxx-xx-xxxx is the pattern used for US social security numbers).

*Coreference identification*: Alternate names for the same object may be determined through correlation analysis: "Barack Obama", "President Obama", "the US president", "the 44th president" and "44" all refer to the current president of the US.

*Sentiment analysis:* uses lexical clues such as specific words or phrases buried within the text to determine prevailing sentiment, emotion or opinion. This has been recently most extensively used in social media analysis of, for example, tweets or Facebook entries. ("Trending" uses same general principle, but with topics rather than lexical items indicating emotion, for example, hash-tags.)

*Relationship and event extraction:* relationships between objects found in the text ("Susan works at ABC Company", "Jane is the sister of Bob", "Mozart died in 1791") may be discovered.

The results of the extraction processes are then available for use in logic models and algorithms which will look for yet more complex and subtle relationship between the entities which have been discovered. Much of the information thus extracted is stored in databases, and, increasingly for large volumes of data, a specialized type of storage called a triple store, designed to efficiently store and retrieve triples. These will be discussed in more depth in the following section. Some of this will serve as background information for context, i.e., to aid in disambiguation such as, for example, determining when "44" refers to Mr. Obama and when it refers to, say, someone's age?). Other algorithms combine the extracted information: *Susan Smith works for ABC Company* and *Sam Brown works for ABC Company* establishes a link between Susan Smith and Sam Brown.

Once we have extracted information from text using any or all of the above techniques, we will wishing to save it for possible use in the future.

## 4.0 STRUCTURING NATURAL LANGUAGE DATA

Extracted text-based information is often stored in structured formats for further processing and simplified access. Currently, the most widely structures for storage of text-based information for automatic processing generally fall into two categories: ontologies, and databases / triple stores, the latter of which are a special kind of database. Each of these has its strengths and weaknesses for sense-making, which we will discuss in this section.

### 4.1 Ontologies and Databases

Ontologies contain information about the characteristics of and relationships between different classes of objects within a specific domain, that is, a definition of a shared concept of the objects in the domain. For example, within a domain containing human beings a "parent" is a (human) object who has at least one instance of an object called "child", a "mother" is a special subclass of parent with the extra characteristic that she also has the gender "female" and so on. Thus, when an object is described as a specific class within the domain of interest, there is knowledge about some aspects of the object ("Mary must be female because she is a mother") and relationships between objects ("if Mary is Susan's mother, then Susan is Mary's child"). Ontologies have the advantage that we have defined in advance exactly what each class of objects is and how it relates to all other objects within our domain of interest.

We can also use ontologies to store information about individual concrete entities. For example, as we shall see later in this paper, the definition of BML we define generalities about the verb types in the language production rules, but specifics about individual verbs used.

Ontologies provide a great deal of information about the world in which we are operating. However, the fact that an ontology is designed to describe facts and relationships about "how the world is" means that we can only use it to support us in reasoning about those facts and relationships. We will discuss the limitation of this later in Section 5.

Databases are useful for storing large amounts of often complex information about specific instances of objects within the domain of interest. Relational databases such as SQL are very widely used for this purposes. The information contained within a relational database is stored in a series of files containing objects (records) of similar structures, which can be represented as tables. Within a file, the record structures are (generally) identical for all elements, consisting of named fields with constraints on the type of data which is allowed in each field. In order to retrieve information, one must have exact knowledge about the structures of the individual files (e.g., the names of the fields in the table) as well as the relationships between the various files. The structures for the files are determined prior to filling in information on individual instances – thus ensuring conformity to ease retrieval. However, determining the structure ahead of time means that the analysts have made a priori decisions as to what information is needed and what information belongs together. Later changes to the structures within the database are possible, but not always easy to effect. Because of the complexity of many databases, as well as the need to have very detailed information about the structures, an alternative

A special variant of databases known as a triple store is a potential solution to some of the complexity issues of a relational database. A triple store contains "atomic" information contained in triples rather than as records inside of more complexly structured file. A triple is a three-part data entity in the form subject-predicate-object: "1-800-555-1234 is-a telephone number", "Susan Smith works-at ABC Company", "ABC Company produces widgets", etc. In a triple store, each triple is an autonomous piece of information, which is does not rely on structures such as a database record format to provide some context. There are advantages to this, one of which is that record formats and schema do not need to be modified if there are changes and updates to the type of information being stored. Another advantage is that queries are simplified because one does not need to know the names of files and fields to make a query. Yet another advantage is that the presentation of a query is easily shown in a graph format, facilitating visualization of the query results. Also, using triples supports inference and logical operations on the data.

Some of the information which is the result of text analytics is easily stored in a triple store. For example, coreference identification, and relationship and event extraction are clearly candidates. More complex information, for example, the details of the aircraft incident described by the sentence in Figure 3 would be difficult, if not downright impossible, to store as a triple.

## 4.2 Out of Context, Out of Mind

Intelligence requires careful and systematic collection of information with the goal of detecting patterns of behaviour being used by the enemy in order to disrupt threatening activities. Over time those who threaten the security and well-being of citizens learn from past mistakes and modify their behavior to again escape detection. This means that the threat models and behavioral expectations which are created today may well be outdated tomorrow. This also means that information which we find unimportant today may be highly significant tomorrow. Additionally, patterns of activity may become more nuanced and complex over time; we may not always know in advance what we are looking for. Therefore, to make decisions in advance about what is and what isn't important may prove to be a mistake.

Extracting isolated pieces of information out of the context in which they were stated may result in incorrect information being stored. Consider the following sentences:

1) Elaine flew from London to Stockholm via Amsterdam on 17 November.
2) Wolfgang gave Johanna Petra's book.

From 1) we can, of course, extract triples such as "Elaine flew to Stockholm", "Elaine flew via Amsterdam" and "Elaine flew on 17 November." However, if we are looking for patterns of behavior, it may turn out that the most interesting information is that Elaine flew via Amsterdam on that particular date (perhaps because another person of interest also was at Amsterdam airport on that day) – something which would be hard to reconstruct unless this information remains connected. Thus the context (day, time, from where, to where, etc.,) may be key to understanding the meaning of Elaine's travel. (Strictly speaking, we could, of course, make a query requesting all information involving Elaine's activities. If Elaine travels a lot, we would get a long list of dates, a long list of intermediate stops and a long list of destinations – from which we would have to decide which pieces belong to which others.)

Sentence 2) contains a subject (***Wolfgang***), a verb (***gave***), a direct object (***Petra's book***) and an indirect object (***Johanna***) which means that there are (at least) four major components to this statement, rendering it impossible to represent as a triple as it stands. A database could be suitable to capture the information in 2), as long as we have anticipated in advance that such information will be stored and built the appropriate structures to handle this.

For example, suppose Elaine is a person of interest and modifies her behavior, so that her flight patterns may include multiple intermediate stops, during any one of which she may make contact with another person of interest. It would be important to store information about the multiple intermediate stops in order to look for patterns. However, in order to do that we must anticipate the possibility in advance that this may happen. And, of course, we must anticipate that the intermediate stops are of importance.

### 4.3 An Alternative Representation for Flexibility

Originally designed for commanding simulated units, BML is a standardized language for military communication (orders, requests and reports) which has been developed under the aegis of the NATO MSG-048 "Coalition BML" and has been expanded to communicate not only orders but also requests and reports. BML is based upon the Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM)[10] which is used by all participating NATO partners. As NATO standard, (STANAG 5525), JC3IEDM defines terms for elements of military operations, both wartime and non-war, and thus provides a vocabulary sufficiently expressive to use for both military and non-military communications in a variety of different deployment types. It also provides a basis for standardized reporting among NATO coalition partners. While BML has been predominantly developed for use by the military, the principles underlying the grammar and standardized representation of natural language text can be expanded into any domain. Extensions of BML for other domains such crisis management (CML), police investigation (IML) and e-government (C2LG) already exist or are in development.

BML has been designed as a controlled language [11] based on a formal grammar [12,13]. This grammar has been designed after one of the most prominent grammars from the field of computational linguistics, Lexical Functional Grammar (LFG) [14]. As a result, BML is an unambiguous language which can easily be processed automatically.

As described in [14], a basic report in BML delivers a "statement" about an individual task (action), event or status. A task report is about a military action either observed or undertaken. An event report contains information on non-military, "non-perpetrator" occurrences such as flooding, earthquake, political demonstrations or traffic accidents. Event reports may provide important background information for a particular threat: for example, a traffic accident may be the precursor of an IED detonation. Status reports provide information on personnel, materiel, facilities, etc., whether own, enemy or civilian, such as number of injured, amount of ammunition available, condition of an airfield or a bridge.

At present this data representation is also being used for multilevel fusion, including within a NATO research group (IST-106) as a means to bridge the gap between information generated by devices and by algorithms (which present their results in BML) to fuse both hard and soft data, as well as low-level and high-level fusion results[21,22,23]. Furthermore, the underlying concept may also be used to enable information fusion across multiple natural languages by converting and mapping to the (English-like) BML, thus lowering the barrier of multilingual information[23,24].

Using various natural language processing techniques and text analytics as described previously, natural language statements can be processed and converted to BML[24]. BML has the advantage that the production rules of the underlying grammar capture all of the content information held in context. Clues as to source type (e.g., eyewitness or third party) as well as linguistic clues as to uncertainty of the information (e.g., “possibly”, “probably”, “might be”, etc.) are reduced to information concerning source type and reliability, credibility of the information and a label which among other things establishes provenance as it is generated based upon time/date information.

The statement “*Coalition forces report the detonation of a bomb at the Old Market in XYCity at shortly past 4 p.m. today*” would be represented as a BML string (Figure 1, lower) and but also can be implemented as a feature-value (structured) matrix or other structured form for use.

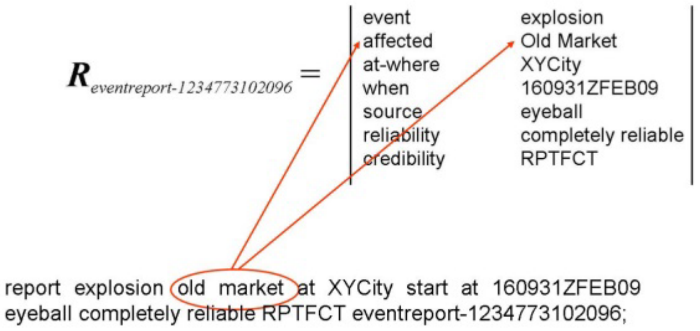

$R_{eventreport\text{-}1234773102096}$ =
event
affected
at-where
when
source
reliability
credibility
explosion
Old Market
XYCity
160931ZFEB09
eyeball
completely reliable
RPTFCT
report explosion old market at XYCity start at 160931ZFEB09
eyeball completely reliable RPTFCT eventreport-1234773102096;

**Figure 3: Representation of the report “Coalition forces report the detonation of a bomb at the Old Market in XYCity at shortly past 4 p.m. today” as a BML string (bottom) and a feature-value (structured) matrix.**

Note that all information remains both intact and in context. Additionally, the simplified representation of the statement as a parsable BML string means that this representation is implementation independent and therefore can be easily mapped into other formats such as XML as needed for further processing.[25]

BML is based upon a series of context-free production rules which consist of terminal and non-terminal symbols which define the grammar and define what types of elements can be used where. For a detailed discussion of BML see [17,18,20].

However, for the purpose of our discussion here we will take a look at a specific example to illustrate the flexibility of BML as a basis to support automatic processing of soft data.

Returning to statement 1) in the previous subsection (Elaine’s flying history), we discussed the problem of storing information about her multiple intermediate stops.

Among the rules for verbs of motion in BML, is the non-terminal “RouteWhere” which describes a route (movement). This non-terminal can be expanded in the following three ways:

a) RouteWhere → ***along*** RouteName
b) RouteWhere → ***towards*** Location | ***towards*** Bearing

c) RouteWhere → (***from*** Location) ***to*** Location (***via*** Location*)

In a) "RouteWhere" can be expanded by the keyword "***along***" followed by the unique name ("RouteName") of a route which is already known (i.e., is stored in the database). In b) only the direction of the movement is known, so "RouteWhere" is expanded by the keyword "***towards***" followed by either a location (such as a city or landmark) or a bearing (i.e., cardinal point such as "north" or degrees between 0 and 360). In c) "RouteWhere" can be expanded by a sequence of three spatial constituents, namely an optional starting point (also called origin) that is preceded by the keyword "***from***", a mandatory destination preceded by the keyword "***to***", and an optional path identified by the keyword "***via***". In the case of the path constituent it is possible to list more than one location following the keyword "***via***", i.e., the path between origin and destination need not be a straight line.

Thus, regardless of how many intermediate stops Elaine chooses to make, BML can preserve them all. Descriptions of activities, events and states can be stored with their full details in a machine-readable way, offering the flexibility needed to allow for revisiting the information at a later date with a different view in mind.

## 5.0 CONCLUSIONS

Fusion of hard and soft data remains an area of research in which much work still needs to be done. In part, that is because of the differing "worlds" in which the two communities operate.

In this paper, we have examined the different understanding of the word "context" between the hard data and soft data communities. We have shown that defining "context" for soft data is complex and variable, and have looked at some of the issues involved.

Natural language processing algorithms and text analytics allow much useful information to be gleaned from text. However, the differing needs of the short timeframe situation awareness and of (often very) long intelligence time horizon, as well as the changeability of human activity being observed requires not only the storage of information which at present may seem irrelevant, but also flexibility in the underlying storage structures. We have discussed the use of ontologies, databases and triplestores, and, where appropriate, some of the advantages and disadvantages. We have also briefly discussed an alternative representation of soft data in BML, showing briefly how this may solve issues of context at the data (sentence) level.

## 6.0 REFERENCES

[1] http://www.bbc.com/news/business-26383058(downloaded Sept 2015)

[2] http://aci.info/2014/07/12/the-data-explosion-in-2014-minute-by-minute-infographic/(downloaded Sept 2015)

[3] http://www.domo.com/blog/2014/04/data-never-sleeps-2-0/(downloaded Sept 2015)

[4] http://dictionary.cambridge.org/dictionary/english/soft-data (downloaded Sept 2015)

[5] http://dictionary.cambridge.org/dictionary/english/hard-data (downloaded Sept 2015)

[6] http://www.objectivity.com/hard-data-vs-soft-data/(downloaded Sept 2015)

[7] http://www.oxforddictionaries.com/definition/english/context(downloaded Sept 2015)

[8] http://dictionary.reference.com/browse/context(downloaded Sept 2015)

[9] B. Schilit and M. Theimer. (1994) “Disseminating Active Map Information to Mobile Hosts.” IEEE Network, 8(5) pp.22-32

[10] M.R. Endsley(1995). Toward a theory of situation awareness in dynamic systems. Human Factors 37(1), pp. 32–64.

[11] C. Jenge, S. Kawaletz and U. Schade. (2008) “Combining Different NLP Methods for HUMINT Reports, RTO Information Systems Technology Panel (IST) Symposium, Stockholm, 2008

[12] M. Hecking. Information Extraction from Battlefield Reports. In *Proceedings of the 8th* International Command and Control Research and Technology Symposium (ICCRTS), Washington, DC, 2003.

[13] M. Hecking (2008). System ZENON. Semantic Analysis of Intelligence Reports. In *Proceedings of the LangTech 2008*, Rome, Italy,.

[14] B. Haarmann, L. Sikorski and U. Schade. (2011) Text Analysis beyond Keyword Spotting. MCC 2011, Military Communication and Information Technology: Amsterdam; 17.-18. October 2011

[15] Joint Consultation, Command and Control Information Exchange Data Model, http://www.mip-site.org/publicsite/04-Baseline_3.0/JC3IEDM-Joint_C3_Information_Exchange_Data_Model/JC3IEDM-Annex%20E-Domain%20values-UK-DMWG-Edition_3.0_2005-12-09.pdf

[16] W.-O. Huijsen,(1998) “Controlled Language – An Introduction,” in *Proc. of the Second International Workshop on Controlled Language Applications (CLAW98)*. Pittsburgh, PA: Language Technologies Institute, Carnegie Mellon University, May 1998, pp. 1-15.

[17] U. Schade and M.R. Hieb (2006). “Development of Formal Grammars to Support Coalition Command and Control: A Battle Management Language for Orders, Requests, and Reports.” *11th ICCRTS*. Cambridge, UK, 2006.

[18] U. Schade, M. Hieb, M., Frey and K. Rein (2010). Command and Control Lexical Grammar (C2LG) Specification. Version 1.3. FKIE Technical Report. June 2010.

[19] J. Bresnan.(2001) *Lexical-Functional Syntax*. Malden, MA: Blackwell.

[20] U. Schade and M.R. Hieb (2007). “Battle Management Language: A Grammar for Specifying Reports.” *2007 Spring Simulation Interoperability Workshop* (Paper 07S-SIW-036). Norfolk, VA, Mar. 2007.

[21] J. Biermann, J. Garcia, K. Krenc, V. Nimier, K. Rein and L. Snidaro (2014). Multi-level Fusion of Hard and Soft Information, Proceedings Fusion 2014, Salamanca, July 2014.

[22] Rein, K. Schade, U. & Remmersmann T. (2012). Using Battle Management Language to Support All Source Integration. NATO RTO IST-112 Joint Symposium. Quebec City, Canada, Spring 2012.

[23] Rein, K. & Schade, U. (2012). Battle Management Language as a “Lingua Franca” for Situation Awareness. Proceedings of IEEE CogSiMa 2012, New Orleans, March 2012.

[24] Kawaletz, S. & Rein, K. (2010). Methodology for standardizing content of military reports generated in different natural languages. Proceedings of MCC 2010, Wroclaw,Poland, Sept. 2010.

[25] Rein, K. (2013). Re-Thinking Standardization for Interagency Information Sharing. Chap. 16 in *Strategic Intelligent Management, National Security Imperatives and Information and Communications Technologies Series*, B. Akhgar, ed., Elsevier Scientific.