Ontological Semantic Technology for Detecting Insider Threat and Social Engineering

Victor Raskin
CERIAS/Purdue University
& RiverGlass, Inc
686 Oval Drive
W. Lafayette, IN 47907-2086
1 765 494 3782
vraskin@purdue.edu

Julia M. Taylor
CERIAS/Purdue University
& RiverGlass, Inc
686 Oval Drive
W. Lafayette, IN 47907-2086
1 765 496 5766
jtaylor1@purdue.edu

Christian F. Hempelmann
Linguistics/Purdue University
& RiverGlass, Inc
100 N. University St
W. Lafayette, IN 47907
1 765 494 3782
chempelm@purdue.edu

ABSTRACT
This paper describes a computational system for detecting unintentional inferences in casual unsolicited and unrestricted verbal output of individuals, potentially responsible for leaked classified information to people with unauthorized access. Uses of the system for cases of insider threat and/or social engineering are discussed. Brief introductions to Ontological Semantic Technology and Natural Language Information Assurance and Security are included.

Categories and Subject Descriptors
I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods -- *representation languages, representations*, I.2.7 [**Artificial Intelligence**]: Natural Language Processing -- *language parsing and understanding, text analysis,* K.6.5 [**Management of Computing and Information Systems**]: Security and Protection – unauthorized access.

General Terms
Algorithms, Security, Human Factors.

Keywords
Insider threat, social engineering, unintended inference, default override, ontological semantic technology, natural language information assurance and security.

1. INTRODUCTION
This paper introduces a computational system for automatic extraction of hidden semantic information from the casual and unsolicited verbal output of a "person of interest" (POI), both written (blogs, Facebook, Twitter, etc.) and oral (taped conversations), over any period of time. The resource, under development, is enabled by the Ontological Semantic Technology (OST), an advanced and implemented version of Ontological Semantics (Nirenburg and Raskin 2004).

NSPW'10, September 21–23, 2010, Concord, Massachusetts, USA.

The system can aid applications in Information Assurance and Security (IAS), where the current approaches are proving to be insufficient. These include, but are not limited to, insider threat, social engineering, and related issues.

The system emulates what a good human investigator infers from observations, conversations, and interrogations of a suspect. The easiest case is to find contradicting details about a specific event, and potentially put the person on the suspect list. For example, the POI could say in one conversation that he went to Florida on vacation, and in another that "The Birth of Venus" was worth seeing. The detection of contradiction in this example requires understanding of natural language and access to encyclopedic knowledge about paintings (and, in particular, that "The Birth of Venus" is in Florence, Italy, not Florida).

A harder case is one without an obvious contradiction. To stay with painting, let us suppose that the POI utters this sentence *I've been to Florence recently, and I have to admit, Botticelli does grow on you: you see "The Birth of Venus," "Primavera," and "The Wedding Banquet" and it just hits you.* Again, with the help of encyclopedic knowledge, it can be revealed that while the first two paintings are in the Galleria degli Uffizi, the third one is in a private collection, the access to which usually deserves a special mention. The fact that the POI did not consider it necessary to make such a mention makes his ability to access the private collection his own default. Let us consider the alternative: access to a private collection is not the default for the POI, and he didn't mention the collection on purpose. It is reasonable to think, however, that a smart POI would not then mention the painting either. The default access to a private collection may be not compatible with what is known about the POI to the employer, and thus raise a flag for further investigation.

The system discussed in this paper uses OST to reach human-level understanding of natural language (NL) text and to calculate and extract the information that the POI gives away unintentionally, as demonstrated by the previous examples. Unlike a human investigator, whose time, availability, alertness, knowledge, and skill are subject to natural limitations of fatigue, lapses of memory, and failures to integrate pertinent information, the system addresses the self-generated, voluntary, unstressed, casual output of any volume by the POI, and it can do this automatically, 24/7.

Work on Ontological Semantics was presented at this workshop early in the decade/century. Raskin *et al.* (2001, 2002a, along with 2002b, 2004, published elsewhere) introduced the theory/ methodology of Ontological Semantics and outlined some directions in which it can serve IAS. As Section 2 will outline, several of these directions have been implemented and mainstreamed. Even more significantly, OST (Raskin *et al.* 2010)

can now be applied in more sophisticated ways, as this paper will show. If early on, the emphasis was on the adequate representation of the meaning of NL text in Text Meaning Representations (TMRs), trying to emulate human language understanding as closely as possible, now with that accomplished, the current emphasis is on inferencing and reasoning, neither of which has been addressed in the previous NL IAS OST-based work.

The central notion of this paper is the unintended inference, which OST can detect, calculate, and flag, if necessary, as a security risk[1]. Accordingly, Section 2 contains an up-to-date brief introduction to OST, which is essential for understanding Sections 5 and, especially, 6. Section 3 documents the previous OST applications to IAS throughout the decade and provides a gist of an earlier venture into semantic forensics, the most advanced of those applications. Unlike this paper, none of the earlier applications ventured beyond what was explicitly said in the analyzed texts. Section 4 complements the earlier semantic forensics technologies by introducing the notion of unintended inference, which addresses what is not explicitly said in the text and nevertheless gathered from the text by the human despite not being intended. The section illustrates, on a simple example, the situations in which the unintended inference—revealing the information that the speaker may not intend to reveal nor to be aware that they have—may be utilized for the detection of insider threat and social engineering. Section 5 presents the OST algorithms for the application of the unintended inference mechanism to detect risk in these situations. It also expands the technology to the much more ambitious application of maintaining a close watch on the verbal and other activities of a POI or, for that matter, of any individual or group, on the basis of the automatically calculated similarity between those activities and any predetermined set of "bad" statements. Section 6 illustrates on a seemingly inconspicuous example, undetectable by a non-semantic technology, how the proposed OST-based technology discovers and interprets a revealing unintended inference.

2. OST

At the core of OST are repositories of world and linguistic knowledge, acquired semi-automatically within the approach and used to disambiguate the different meanings of words and sentences and to represent them. These repositories, also known as the static knowledge resources, consist of the ontology, containing language-independent concepts and relationships between them; one lexicon per supported language, containing word senses anchored in the language-independent ontology which is used to represent their meaning; and the Proper Name Dictionary (PND), which contains names of people, countries, organizations, etc., and their description anchoring them in ontological concepts and interlinking them with other PND entries.

[1] A crucial difference between this unintended inference on the one hand and the kind of inferences standardly studied in linguistic pragmatics (Levinson 1983) should be noted: in conversational implicatures, the speaker deliberately enables the hearer to make a supervised inference, thus, reinterpreting the utterance from its literal meaning to the one intended by the speaker; in figuring out presuppositions, pragmatics focuses only on general, commonly-shared presuppositions. In other words, there has been no known effort to research *unintended* inference theoretically nor to implement it computationally, let alone applying it to IAS.

The lexicon and ontology are used by the Semantic Text Analyzer (STAn), a software that produces Text Meaning Representations (TMRs) from the text that it reads. The format of TMRs conforms to the format and interpretation of the ontology. The processed TMRs are entered into InfoStore, a dynamic knowledge resource of OST, from which information is used for further processing and reasoning.

Formally, the OST ontology is a lattice of concepts, formed according to the following rules:

C	→	B |	(atomic concept)
	→	ALL|	(top of the lattice)
	→	ε |	(nothing)
	→	C(R(F1(D)))	(concept with a relation)
	→	C(A(F2(V)))	(concept with an attribute)
D	→	C D |	(disjunction of concepts)
	→	**and** C D	(conjunction of concepts)
V	→	V W	(disjunction of literals)
W	→	W |	(literal)
	→	ε	(nothing)
F1	→	**sem** |	(semantic facet for concepts)
	→	**default** |	(default facet for concepts)
	→	**hier** |	(hierarchy facet for concepts)
	→	**not**	(negation of concepts)
F2	→	**value** |	(facet for strings)
	→	**greater** |	(facet for real numbers)
	→	**less** |	(facet for real numbers)
	→	**equal** |	(facet for real numbers)
	→	**greater-than** |	(facet for real numbers)
	→	**less-than** (facet for real numbers),	

where B, C, D are concepts, R is a relation, A is an attribute, V, D are literals, and F is a facet.

Given a set of facets $\mathcal{F}$, a set of objects $\mathcal{D}$, where $\mathcal{D}$ is the disjoint union of $\mathcal{D}c$ and $\mathcal{D}d$, and its interpretation function $\mathcal{I}$, for every atomic concept B, $\mathcal{I}[B] \subseteq \mathcal{D}c$; for every literal V, $\mathcal{I}[V] \subseteq \mathcal{D}d$; for every relation R, $\mathcal{I}[R] \subseteq \mathcal{D}c \times \mathcal{F} \times \mathcal{D}c$; for every attribute A; $\mathcal{I}[A] \subseteq \mathcal{D}c \times \mathcal{F} \times \mathcal{D}d$. Moreover, the following is true:

$\mathcal{I}[ALL] = \mathcal{D}$
$\mathcal{I}[\varepsilon] = \emptyset$
$\mathcal{I}[C\ D] = \mathcal{I}[C] \cup \mathcal{I}[D]$
$\mathcal{I}[and\ C\ D] = \mathcal{I}[C] \cap \mathcal{I}[D]$
$\mathcal{I}[(R(F1(D)))] = \{x \in \mathcal{I}[C] \mid y \in \mathcal{I}[D], f \in F1, \langle x, f, y\rangle \in \mathcal{I}[R]\}$
$\mathcal{I}[(A(F2(V)))] = \{x \in \mathcal{I}[C] \mid y \in \mathcal{I}[V], f \in F2, \langle x, f, y\rangle \in \mathcal{I}[A]\}$
$\mathcal{I}[(C(R(F1(D))))] \subseteq \mathcal{I}[C]$

Finally, concept C is a descendant of D if $\mathcal{I}[C] \subseteq \mathcal{I}[D]$.

Each OST lexicon contains senses of words (and phrasals) of a natural language that the lexicon describes, mainly in terms of their syntax and semantics. The meaning of each entry is described in the semantic structure (sem-struc) using concepts from the ontology. The syntactic structure (syn-struc) of the entry is governed by the syntactic rules of the natural language described, using a simplified generic notation for lexical functional grammar (LFG—Bresnan 1983). Ambiguous (polysemous or homonymous, if one needs to make the distinction) words are defined with as many senses as needed in the following form:

(word
    (word-sense1)

…
(word-senseN)
)

Each word sense WS is formed according to the following rules:

(WS-PosNo
(**cat**(Pos))
(**synonyms** "WS-PosNo"))
(**anno**(**def** "Str")(**ex** "Str")(**comments** "Str"))
(**syn-struc**((M)(**root**(**$var0**))(**cat**(Con))(M))
(**sem-struc**(Sem))
)

M	→	(Srole((**root**(Var))(**cat**(Cpos)))	
	→	(Srole((opt(+))(**root**(Var))(**cat**(Cpos)))	
	→	(M(M))	
Pos	→	N |	(noun)
	→	V |	(verb)
	→	Adj |	(adjective)
	→	Adv |	(adverb)
	→	Prep |	(preposition)
	→	Det	(determiner)
Con	→	NP |	(defined by rules omitted
	→	VP |	due to space limitation)
	→	Con Con |	
	→	Pos	
SRole	→	**subject** |	(syntactic roles of which
	→	**directobject** |	only some are shown
	→	**pp-adjunct**	due to space restriction)
	→	**…**	
No	→	[1-9]	(any digit)
Str	→	[A-Z|a-z| |,|.]	(any string)
Var	→	**$var**No |	
	→	Str	
Sem	→	C |	(any ontology concept)
	→	^Var(R(F1(C))) |	(R, F1, C from ontology)
	→	C(R(F1(^Var)))	(C, R, F1 from ontology)

The Proper Name Dictionary (PND) follows a format similar to that of the lexicon. This paper will not address the PND, thus a detailed description is omitted.

The following example of simple disambiguation of *the dog ate a mouse* (cf. Hempelmann et al. 2010) shows entries from each resource and the basic processing mechanism. Our lexicon contains the following, partially condensed, senses of the polysemous/homonymous words *eat* and *mouse*. The entries for *dog*, *a*, *the,* which only have one sense, are not shown here.

```
(eat
  (eat-v1
      (cat(v))
      (anno(def "to ingest for nourishment
                        through digestion")
           (ex "he ate the apple"))
      (synonyms "consume-v1, feed-v2")
      (syn-struc(
           (subject((root($var1))(cat(np))))
           (root($var0))(cat(v))
           (directobject((root($var2))
               (opt(+))(cat(np))))))
      (sem-struc(eat
           (agent(value(^$var1)))
           (theme(value(^$var2))))))
  (eat-v2
      (cat(v))
      (anno(def "eat outside the home")
           (ex "they eat out once a week."))
      (syn-struc(
           (subject((root($var1))(cat(np))))
           (root($var0))(cat(v))
           (phr((root(out))(cat(phr))))))
      (sem-struc(meal
           (agent(value(^$var1)))
           (location(sem(restaurant))))))
  (eat-v3
      (cat(v))
      (anno(def "gradually destroy, erode")
           (ex "the heavy rains ate away at
                        the sandrock cliffs."))
      ...
      (sem-struc(dissolve
           (precondition(value(^$var1)))
           (theme(value(^$var2))))))
  (eat-v4
      (cat(v))
      (anno(def "to swallow other than for
                                nourishment")
           (ex "he ate the pill"))
      (syn-struc(
           (subject((root($var1))(cat(np))))
           (root($var0))(cat(v))
           (directobject((root($var2))
               (opt(+))(cat(np))))))
      (sem-struc(swallow
           (agent(value(^$var1)))
           (theme(value(^$var2))))))
  (eat-v5
      (anno(ex "what's eating Gilbert?"))
      ...
      (sem-struc(fear
           (precondition(value(^$var1)))
           (experiencer(value(^$var2))))))
)
(mouse
  (mouse-n1
      (cat(n))
      (anno(def "a computer mouse")
            (ex "he bought a new mouse"))
      (synonyms "")
      (syn-struc((root($var0))(cat(n))))
      (sem-struc(computer-mouse)))
  (mouse-n2
      ...
      (sem-struc(rodentia)))
)
```

It is easy to see from the senses that only *eat-v1* and *eat-v4* are meaningful in the example: a dog can swallow a mouse (by mistake?) or eat it for its nourishment value. When STAn reads the lexicon, it looks for syntactic as well as semantic clues in order to find entries that can be discarded right away. For example, *eat-v2* will be discarded because according to its syn-struc, it needs the word *out* to be next to the word *eat*. Similarly, *eat-v3* is discarded. We are thus left with senses *eat-v1*, *eat-v4,* and *eat-v5*, and using

the combination of the appropriate syn-strucs and sem-strucs, we have the following hypotheses for the simplified interpretations of the sentence:

- eat
  (agent(dog))
  (theme(rodentia))
- eat
  (agent(dog))
  (theme(computer-mouse))
- swallow
  (agent(dog))
  (theme(rodentia))
- swallow
  (agent(dog))
  (theme(computer-mouse))
- fear
  (precondition(dog))
  (experiencer(rodentia))
- fear
  (precondition(dog))
  (experiencer(computer-mouse))

Next, STAn accesses the ontological knowledge to confirm or deny the possibility of such interpretations. The definitions of the concepts EAT and SWALLOW are shown below. The concept FEAR is omitted as, according to the ontology, DOG is not an acceptable PRECONDITION of FEAR. In other words <fear1, sem, dog1> $\notin$ I[PRECONDITION].

```
(eat
   (definition(value("to eat and drink")))
   (is-a(hier(survival-event)))
   (effect(sem(defecate)))
   (theme(default(food))(sem(animal plant)))
   (has-event-as-part(sem(bite chew digest
   swallow)))
)
(swallow
   (definition(value("to swallow")))
   (is-a(hier(immerse)))
   (end-location(sem(stomach)))
   (path(sem(esophagus)))
   (part-of-event(sem(eat)))
   (agent(sem(animal)))
   (start-location(sem(mouth)))
)
```

Note that the concept EAT lacks an AGENT in this definition. The AGENT is inherited from its parent.

To test the first hypothesis, we need to check if EAT allows RODENTIA as a THEME and DOG as an AGENT. It can be seen from the properties of the concept EAT, that ANIMAL is a legitimate THEME with a facet sem, and since RODENTIA is a child of ANIMAL, it is also a legitimate theme. Therefore, ANIMAL is inherited as an AGENT, and DOG a legitimate filler for that property. Thus, the first hypothesis holds. To test the second hypothesis, we need to check if EAT also allows COMPUTER-MOUSE as a THEME. According to its properties, only FOOD, ANIMAL, or PLANT can be its THEME and COMPUTER-MOUSE is none of them. Thus, the second hypothesis returns false, and this interpretation is not valid. The third and fourth hypotheses return true, using similar tests as can be verified from the properties above. Three interpretations out of six are in principle acceptable.

Below is the actual output of the analyzer, with the numbered instances of concepts replacing the concepts themselves: thus, instead of DOG we see dog1.

```
DEBUG 13 Apr 2010 17:20:08 [main.SemTextAnalyzer]
The dog ate a mouse
List of TMRs:
TMR 1:
Event: eat-v4, swallow1
  agent(value (dog-n1, dog1 ))
  theme(value (mouse-n2, rodentia1 ))
TMR 2:
Event: eat-v4, swallow2
  agent(value (dog-n1, dog2 ))
  theme(value (mouse-n1, computer-mouse1 ))
TMR 3:
eat-v1, eat1
  agent(value (dog-n1, dog3 ))
  theme(value (mouse-n2, rodentia2 ))
```

A clear disadvantage and the subject of frequent criticism of similar meaning-based approaches (cf. Sowa 2000) is the considerable upfront investment in the acquisition of the single language-independent ontology and a lexicon for each natural language. Ontological Engineering, meaning the creation and maintenance of all resources of an ontological system like OST, is known to be hard (e.g., Fridman-Noy and Hafner 1997, Devedzic 2002) and existing guidelines are preliminary, often pertain to controlled vocabularies, not ontologies (Obrst 2007), or are merely concerned with formal and logical consistency (Guarino 2004), rather than with descriptive adequacy.

Application-oriented OST is logically and formally consistent, with its plane of operation being NL meaning. As such, OST generally agrees that its grain size needs to be at the mesoscopic level (Smith 1995) and not have primary commitment to the representation of scientific knowledge. On the other hand, OST resources easily accommodate do-main-specific information, thus allowing meaning representation and analysis at a finer grain size, as illustrated in Section 4 and, especially, Section 5 below. The units for capturing the meaning of language in ontology-based resources cannot be statements of commonsense knowledge in formal logic (Lenat 1990), but, rather, imputed concepts as they exist for human-like linguistic meaning-encoding tasks and as describable by linguistic semantics.

Formal and logical consistency in OST is ensured by checking all acquired and maintained entries against the grammars of the relevant resources and flagging inconsistencies for cleaning or removal. But in addition, the tasks of applications, rather than artificial evaluation criteria used at TREC-style competitions (see Hempelmann 2007, Raskin *et al.* 2010), are guiding the depth and breadth of the resources and tools.

As a result, STAn can adequately interpret the text, sentence by sentence, disambiguating it according to the lexical senses of the words from the lexicon, fully informed by the knowledge of the world captured in the ontology. This section has demonstrated how the explicit information in each sentence is interpreted by STAn, excluding so far the implicit, elided information. The applications described in the following sections integrate information from several sentences.

Table 1: NL IAS Applications

Application	Function	Reference
Mnemonic String Generator	Generates jingles corresponding to random-generated passwords	Raskin *et al.* 2001a
Syntactic NL Watermarking	Embeds the watermark in the syntactic tree of a sentence	Atallah *et al.* 2001
Semantic NL Watermarking	Embeds the watermark in the TMR tree of a sentence	Atallah *et al.* 2002
NL Tamperproofing	Embeds a brittle watermark to detect any changes to the text	Atallah *et al.* 2002
Automatic Terminology Standardizer	Translates different terminological dialects in IAS into TMRs	Raskin *et al.* 2002a
NL Streaming Processor	Interprets incoming information before it is complete	Raskin *et al.* 2002b
NL Steganalysis	Detects the presence of a hidden message	Raskin *et al.* 2002b
Web Crawler for Planned Attacks	Crawls the web in search of credible information on computer attacks	Raskin *et al.* 2002b

3. IAS AND SEMANTIC FORENSICS

Early in the decade, OST was used to improve IAS with regard to NL files. The result has been a number of applications, some of them NL counterparts of pre-existing applications, others NL extensions and developments of known applications, and still others unique to NL IAS, like the technique described in this paper. In the most implemented one, NL watermarking (see Atallah *et al.* 2002), a procedure based on a secret large prime number (Wagstaff and Atallah 1999) selects certain sentences in a text for water-mark bearing and transforms their TMRs into bit strings that contribute up to 4 bits per sentence to the watermark. The goal of the software is to embed a robust watermark in the hidden semantic meaning of NL text, represented as its TMR in tree structure. The NLP role is to "torture" the TMR tree of the sentence, whose contributing bits do not fit the watermark, so that they do. The tool for that is a number of minuscule TMR tree transformations, resulting in such surface changes as *The coalition forces bombed Kabul* → *The coalition forces bombed the capital of Afghanistan*. The main applications are summarized in Table 1.

As a direct predecessor to the techniques proposed in this paper, Raskin *et al.* (2004) describe a semantic forensic[2] system based on an earlier incarnation of OST. While other disciplines within cyber forensics explore largely non-textual materials—and those which look at texts, with the above-mentioned exceptions, do not do so linguistically—semantic forensics, as defined here, uses NLP to identify the clues of deception in NL texts in order to reconstruct the described events as they actually occurred.

Like all NLP systems, a semantic forensic NLP system models a human faculty. In this case, it is the human ability to detect deception, i.e., to know when we are being lied to and to attempt to reconstruct the truth. The former ability is a highly desirable but, interestingly, unnecessary precondition for deception detection. The latter functionality is the ultimate goal of semantic forensic NLP but, like all full automation in NLP, it may not be easily attainable.

Humans detect lying by analyzing the meaning of what they hear or read and comparing that meaning to other parts of the same discourse, to their previously set expectations, and to their knowledge of the world. Perhaps the easiest lie to detect is a direct contradiction: if one hears first that John is in Boston today and then that he is not, one should suspect that one of the two statements is incorrect and to investigate, if one is interested, a crucial point. The harder type of deception to perceive is by omission.

Thus, reading a detailed profile of Howard Dean, one time a leading contender for the Democratic nomination in the US 2004 presidential election, one could notice that the occupation of every single mentioned adult was indicated with the exception of the candidate's father (who had been a stockbroker).

Yet more complicated a lie is glossing over, such as saying that one has not had much opportunity to talk to John lately, which may be technically true, while covering up a major fallout with John. And perhaps topping the hierarchy is lying by telling the truth: when, for instance, a loyal secretary tells the boss' jealous wife that her husband is not in the office because he is running errands downtown, she may well be telling the truth but what she wants to accomplish is for the wife to infer, incorrectly, that this is all the boss is doing downtown. Jumping slightly ahead, this inference is intended.

A new TMR contradicting a previously processed one should lead to an InfoStore flag. The InfoStore component of OST, based on the earlier concept of a fact repository (see Nirenburg and Raskin 2004: 350-1), records the remembered TMR instances. A contradiction will be flagged when some two or more TMR (fragments) are discovered and compared, and a contradiction, a gap, or some incongruence is discovered. In the case of the senior Dean's occupation, apparently too shameful for the reporter to mention, InfoStore will detect the gap by presenting this information, as given in simplified form in Figure 1.

```
exist1
    human1
        has-family-name   "Dean"
        has-suffix        "III"
        has-given-name    "Howard"
        has-social-role   physician1
        has-spouse
            human2
```

[2] This should not be confused with the unrelated, mostly British-based school of Linguistic Forensics (see, for instance, McMenamin 20032, Gibbons 2003, Olsson 2004), which focused on bringing (informal, non-computational) linguistic knowledge to the attention of forensic specialists and legal experts. While some of it was based on the seminal works by R. Shuy (1993, 1998, 2005), who pioneered linguistic trial expertise in the 1980s, Linguistic Forensics was not interested in methodological, let alone technological implementations.

```
            has-family-name     "Dean"
            has-given-name      "Judy"
            has-social-role     physician2
    has-parent
        human3
            has-family-name     "Dean"
            has-suffix "Jr"
            has-given-name      "Howard"
            has-social-role     *unknown*
```

Figure 1: Example of an InfoStore Result

To detect a gloss-over, it is not quite enough to receive a new TMR which contains an event involving a different interaction between these two individuals at the same time. The co-reference module of the analyzer (cf. Nirenburg and Raskin 2004: 301-5) will have to be able to determine or at least to suspect that these events are indeed one and the same event rather than two consecutive or even parallel events. Even the time parameters are not a trivial task to equate, as in the case of *I have not much opportunity to talk to John lately* and *John insulted me last May*. It would be trivial, of course, if the temporal adverbials were *since that night at Maude's* and *that night at Maude's*, respectively. But a human sleuth does not get such incredibly easy clues most of the time and has to operate on crude proximity and hypothesizing. Also helping him or her is a powerful inferencing module, a must for an NLP system of any reasonable complexity, reinforced by a microtheory of euphemisms, which must contain representative sets of event types that people lie about, and of fossilized, cliché-like ways of lying about them, as in *How is this paper?—Well… it's different!*

Finally, Raskin *et al.* (2004) outlines the expansion of the ontology by acquiring scripts of complex events, already found necessary for other higher-end NLP tasks (see Raskin *et al.* 2003), still a desideratum in OST. The main mechanism described is the instantiation of a script, the example used being BANKRUPTCY. Simplified to a few subevents, the two main semantic forensic mechanisms on the basis of scripts can be summarized as follows: 1. If a necessary element of a script is missing it is likely to be intentionally omitted. 2. If an element that commonly occurs as part of a script is found in a text, but no other element of it, that is, the script is underinstantiated, then the script is likely to be intentionally omitted (see Figure 2).

```
script1
    has-event-as part
        and
            event1          found in text
            event2          not found in text
            event3          found in text
script1
    has-event-as-part
        and
            event1          not found in text
            event2          found in text
            event3          not found in text
```

Figure 2: Simplified Script Structure

Semantic Forensics was an innovative and reasonably sophisticated application of the OST technology deception detection. At that time, however, the technology was still rather limited in its inference abilities, let alone accessing unintended inferences described in the next sections. Using the semantic forensic methods, one could catch lines 11-12, 15-18, and 21-22 of Table 2. It is the remaining 16 lines of the table that the following sections will focus on. To summarize, semantic forensics could only process what was actually said in the analyzed text, while the unintended inference functionality focuses on the unsaid. Needless to say, both are perfectly useful and should work in unison.

4. WD-INFERENCE

This section demonstrates how the unintended inference that is required for catching compromising non-lies is detected and interpreted. Taylor *et al.* (2010) introduced the analysis of a female Facebook user's update, describing her bar experience as, "A white dude was hitting on me all night." It occurred to the authors that, without knowing the race of the writer, the update strongly suggests that she is non-white, which was confirmed by an informal poll. What seems to be at work here is that the mention of the race of the *dude* indicates the unexpected and previously unannounced significance of his race. If the writer were white and were typically being hit on by white guys, it is unlikely that she would be motivated or interested in posting an update that informs her friends on an unremarkable, frequent, and expected occurrence. The race indication, especially standing alone without any further description, appears to indicate perfectly clearly to her readers that his being white is somehow unusual for her.

There are two unequally likely interpretations: either the author usually dates people of her own race, and is therefore non-white, or the author does not date white people and her race is, therefore, unknown to us. Given the still prevalent societal stereotype of people dating within their own race, the former interpretation appears to be more likely. Also for the latter interpretation, some personal knowledge about the writer has to be available to the reader, while the former does not require it, and thus is much more accessible. One can refer to the societal stereotype involved in the former interpretation as general knowledge captured in the ontology, and the personal information about a writer/speaker as the personal profile.

An unrelated conversation between two female adult cousins, both professionals, contained a similar example, albeit with a different property involved: *My manager wants me to fly coach to Germany*. Information that is relevant for the inference is that most people fly coach; it is the default for them, and therefore, they are unlikely to make a special mention of it. The fact that the speaker did reveals that her expectations are different. The easiest interpretation is: she usually does not fly coach to Germany when traveling on business.

To generalize, we assume, throughout the paper, that information is revealed by the speaker to the hearer for one of two purposes. One[3] is to add information to the shared knowledge, be it to mention a totally new event or one of its aspects or to add details to a previously introduced or generally known situation. The other is to contradict or override a default of a known situation or aspect of this situation. In other words, "assuming that Grice's [(1975)] Maxims of Quantity and Manner—and possibly of Relevance—are correct, any word in the sentence should either add to the information that the hearer has or adjust the information when necessary" (Taylor *et al.*, 2010; cf. Prince 1981). It is when what is adjusted is an underlying commonly assumed default that we can figure out what the speaker's default is.

[3] This statement may appear to contradict what happens in Malinowski's (1923) 'phatic' mode of communication, where people talk for the sake of talking, as for instance at cocktail parties, without any intention to convey information unknown to the interlocutor. In fact, however, a default may be overridden in that mode as well.

It is easier to calculate an overridden default when it can be done on the basis of universally available information. Thus, it is known that most people fly coach. It is an immediately related fact that people don't mention the cabin class when talking about flying, and it is generally assumed that it is coach. In the cousins' conversation above, the speaker verbalizes the typical default, and that acquires significance.

The most reasonable way of interpreting it is to override a typical default and infer that normally she flies business/first class, at least to Germany, and this is, then, her personal default. It is possible that the speaker intended to convey that information to the hearer. For forensic purposes, however, it is much more important to expose cases when the default information is given away unintentionally. We will address the question of differentiating between the intentional and unintentional cases of default overriding directly below.

As Taylor *et al.* (2010) states, "[t]he computational choice between overwriting and simply adding information depends on the knowledge of what information is implicitly salient for the speaker or whether enough priming has been achieved by the explicitly communicated text, respectively." The salience is either marked in the appropriate ontological concept(s) and/or is calculated by OST. The priming is determined by the prior adjacent occurrence of a conceptually related word. Thus, in *Because we always fly to Germany business class, can you imagine my manager dare to ask me to fly to Frankfurt in coach*, the priming completely supersedes and thus eliminates the unintended inference. Now, let us consider a modification of this example, *The company is in such a bad financial state, that my manager asked me to fly coach to Germany*. Here, the default override may still be activated, but it loses its non-intentionality because the reason for the departure from the default is primed and, therefore, it is not anything that the speaker would rather not reveal.

The simple informal algorithm for calculating the unintended inference of the underlying default is as follows:

> IF in a sentence a non-evaluative property P of a concept C is filled AND P is not primed, i.e. not appearing in the preceding predetermined low number of TMRs,
>
> THEN the speaker's filler of P for C is set to the disjunction of all acceptable fillers of P for C, on the DEFAULT facet, except for the filler appearing in the TMR.

We will enhance and adapt this basic algorithm to several IAS applications in the next section.

5. UNINTENDED INFERENCE AND IAS PROBLEMS

5.1 Insider Threat

Insider threat has been actively addressed by the IAS community for at least a decade (Wood 2000, Brackney and Anderson 2004, Stamper and Masterson 2002)[4]. The issue has effectively challenged the maturity of the field, making it clear that cybersecurity faces an increasing slew of issues that require a much more multidisciplinary perspective and an occasional non-technical solution (part of which, incidentally, our approach proposes to algorithmize and implement).

Mature work on insider threat has focused on the foundational, conceptual, theoretical aspects of the challenge, hoping to develop better countermeasures, which now take the form of policies of prevention of the conditions that are seen as leading to insider threat (Maloof and Stephens 2007, Greitzer *et al.* 2008, Stolfo *et al.* 2008, Willison and Siponen 2009, Faramond and Spafford 2010). This educational effort has also translated into popular literature (e. g., Cole and Ring 2006).

One of the most useful efforts has focused on the very definition of insider threat, leading to a much more accurate and sophisticated non-binary model (see Bishop 2005, Bishop and Gates 2008). The definition issue was the focus of the Dagstuhl Seminar (08302) on "Countering Insider Threats" in July 2008 (Bishop *et al.* 2008—and these seminars continue). An important collective attempt was made there to define the phenomenon and to outline alternative programs of dealing with it.

This paper can be seen as an attempt at furthering that program of action. We accept their view that "[a]n insider is a person that has been legitimately empowered with the right to access, represent, or decide about one or more assets of the organization's structure." We do, however, differentiate between "accessing" and "deciding," on the one hand, and "representing" on the other, leaving the latter to the distinct threat of social engineering that we are addressing next. According to Bishop *et al.* (2008), "there are masqueraders (individuals pretending to be legitimate insiders but without valid access); traitors (legitimate insiders acting in malicious ways) and naïve insiders (who cause damage without malicious intent)." We will first utilize OST to differentiate between traitors and naïve insiders, and expand on masqueraders in the next scenario.

We assume the most difficult case for exposing an insider traitor, namely when the perpetrator neither attempts to use any systems which they are not authorized to use nor does anything unusual within their authorized access. In other words, they are merely disclosing the very information they normally work with to unauthorized parties. Obviously, the software packages designed to detect unauthorized activities will fail to identify this individual or information that was jeopardized.

Using OST to distinguish between the intentional and unintentional inferences, we are hoping to establish whether the act was "obvious or stealthy", using Bishop *et al.*'s (2008) terminology. In future phases of the development of the technology, we may make further steps towards "the all-embracing knowledge" of "the insider's intent," which Bishop *et al.* assumed to be unattainable.

We will first consider the malicious insider, called Alice or Bob (A/B), who is unlikely to get caught with some sort of access violations or security leaks. In the best-case scenario for A/B, they leave no trace of their activity, other than the fact that the information that only they (and probably several others, all duly authorized) possessed is now known to unauthorized person(s).

[4]Greenwald (2010a) cites the Ware and Anderson reports as proof that the insider threat problem is older than that. Not really so, though: while Anderson (1972, vol. II, 12-16) does make some statements about "the malicious user" that are partially relevant to insider threat, its first volume, The Executive Summary, has no mention of the issue explicitly or implicitly. The closest Ware (1970) comes to insider threat is an authorized user actively infiltrating different areas of the system, other than what they normally use. The most interesting and difficult case of insider threat, and indeed a much more recent concern, involves no infiltration or trespassing, as it were, in computer use. Greenwald makes the further valid point that, back then, before computer networking, all the concerns were about insiders.

We will assume that there may have been changes in A/B's behavior/habits/thinking since before they got engaged in malicious activity and that these changes should be hidden from others. A U.S. Government spy-catching manual (www.hanford.gov/oci/maindocs/ci_i_docs/howspiesarecaught.pdf) states:

> **Changes in Behavior** Espionage usually requires keeping or preparing materials at home, traveling to signal sites or secret meetings at unusual times and places, change in one's financial status with no corresponding change in job income, and periods of high stress that affects behavior. All of these changes in normal pattern of behavior often come to the attention of other people and must be explained.

The easiest example of a change from the list above to address here is the increase of discretionary funds that A/B can spend (for other reasons, see Faramond and Spafford 2010).

With that in mind, let us return to the example of flying coach/business, illustrated by Table 2, where column S indicates the specific flight by A/B that the conversation addresses; column E shows what class A/B was expected to fly; column F indicates the actual cabin class flown by A/B on this occasion; column U indicates what A/B usually flies, which could be A/B's personal default[5]; and column I sums up the inference.

For instance, if A/B says nothing about the cabin class of their last flight in a casual blog entry or conversational context, and we expect them to fly coach (lines 1-4), we assume that they flew coach this time and that they fly coach in general. There are several possibilities, if somehow it becomes known that:

- This is indeed so, as per line 1: there is nothing suspicious about it (column I).
- If, in fact, they usually fly business, including this particular time (lines 2-4), the omission of this information may be suspicious.
  - They may wish to conceal the fact that they usually fly business (lines 2, 4).
    - Line 2 is more suspicious, according to our model, as it could be perceived as the deliberate hiding of their typical but unexpected (by others) behavior.
    - Line 4 is compatible with the default override prediction of the unintended inference for this individual, raising the question why they usually pay the distinctly higher fare.
  - Saying nothing about what appears to be an unusual occurrence is atypical line 3), raising the possibility of a new source of income.

If they say nothing AND we expect them to fly business (lines 5-8), we assume their flying business this time and in general. Again, there are several possibilities—if it becomes known that:

- This is indeed so, as per line 8: there is nothing suspicious.
- If, in fact, they fly coach usually and/or this particular time, the omission of this information may be suspicious, but not nearly as much as in lines 1-4 (they could just be cheap and/or embarrassed about it).
  - Line 5 is compatible with the default override prediction of the unintended inference for this individual, raising the question why they usually pay the much lower fare—however, there may be innocent explanations.
  - Line 6 raises the question of atypical behavior of not commenting on the unusual occurrence, again with innocent enough explanations possible.
  - Line 7 is rather suspicious and may indicate a new circumstance, such as a new source of income, that suddenly allows a match of the group's expectations, yet remains unspecified.

If A/B says *coach* it indicates that either they typically fly business or, less likely, they fly coach and want to reconfirm that. With that in mind, there are several scenarios:

- They are expected to fly coach (lines 9-12):
  - If they fly coach this time and usually fly coach (line 9), it probably indicates a low threshold of triviality (Raskin and Triezenberg 2003, Raskin 2005): A/B states everything explicitly, without allowing for much obvious information to be inferred; if this is not so, the flag should be raised.
  - If they usually fly business, but flew coach this time (line 10), their statement (column S) is compatible with the default override prediction of the unintended inference A/B, raising the question why they usually pay the much higher fare.
  - If they flew business this time, they lied about the flight, regardless of what they usually fly, and that should be investigated. The unintended inference works wonders here: by stating the expected (possibly trying to conform with the general expectation of the group) they trigger the flag that something is wrong. Notice the difference in interpretation with lines 3-4.
- They are expected to fly business (lines 13-16):
  - Line 13 raises two questions: why they restate their default and why they usually pay the lower fare. Both may have sufficiently innocent explanations: their accommodating the group's default (instead of their own) and economic reasons, respectively.
  - Line 14 is compatible with the default override prediction.
  - Lines 15 and 16 raise the same objections as lines 11 and 12. However, the inference does not help flagging these statements.

Lastly, if they say *business* it should indicate that either they typically fly coach, or that they talk to somebody who typically flies coach. With that in mind, there are several scenarios:

- Flying coach this time (lines 17-18, 21-22) and lying about it hardly points to somebody who just got rich by selling secrets, but should be analyzed carefully in the social engineering scenarios (see also below).
- Flying business this time while typically flying coach is compatible with the default override prediction:
  - It does not have to be flagged if the group's expected behavior is flying coach (line 19).
  - It could be perceived as suspicious and may indicate a new circumstance, such as a new source of income that suddenly allows a match of the group's expectations (line 23).
- Flying business this time and typically flying business are atypical for the restatement of business default, unless the goal is to reemphasize the higher expense.

[5] This is usually reflected by a significantly higher frequency of either cabin class usage in relation to the others.

Table 2 : Cabin Class Situations and Speaker Reports

Line#	Says about this time (S)	Expected to usually fly (E)	Flies this time (F)	Usually flies (U)	Indication (I)
1	nothing	coach	coach	coach	OK
2	nothing	coach	coach	business	**cover up?**
3	nothing	coach	business	coach	new income?
4	nothing	coach	business	business	expensive habits?
5	nothing	business	coach	coach	cheap habits?
6	nothing	business	coach	business	setback?
7	nothing	business	business	coach	**cover up?**
8	nothing	business	business	business	OK
9	coach	coach	coach	coach	low th-d triviality?
10	coach	coach	coach	business	expensive habits?
11	coach	coach	business	coach	**cover up?**
12	coach	coach	business	business	**cover up?**
13	coach	business	coach	coach	could be OK?
14	coach	business	coach	business	OK
15	coach	business	business	coach	**cover up?**
16	coach	business	business	business	**cover up?**
17	business	coach	coach	coach	trying to impress?
18	business	coach	coach	business	hiding a setback?
19	business	coach	business	coach	OK
20	business	coach	business	business	expensive habits?
21	business	business	coach	coach	trying to impress?
22	business	business	coach	business	trying to impress?
23	business	business	business	coach	new income?
24	business	business	business	business	could be OK

- Additionally, line 20 raises the question of a much higher fare, with some possibly innocent explanations
- In line 24, the restatement of the default can only be explained by bragging about wealth, and as such, does not have to be suspicious.

It should be noted that while the 8 cases of lying (S differs from F) could be caught without the described inference, as mentioned in Section 3 above, the inference helps not only with the other 16 cases but also to separate the 4 high-threat cases of lying from the (potentially) more innocent lies. By raising the flag, the unintended inference triggers an investigation of specific details. With its help, not only can A/B's motives be explained better when the values of the other 3 columns are equal, but it would also be difficult, if not impossible, to catch the suspicious *nothing* cases without it because there is no lying *per se* in there. This means that the grain size of description of a particular event (or willingness to drill into the finer-grain details) could help identify a malicious insider.

Thus, the inference helps to flag cases that should be investigated further based on what is said in conversation, the expected typical value for that attribute, and, if suspected, the actual value of the instance for a given attribute. It also, in most cases, spares the investigator's effort on reviewing A/B's actual typical behavior. And, of course, it does it automatically, over A/B's casual unsolicited and unconstrained verbal output (see more on this in Section 6).

The general algorithm is as follows:

```
If S==Nothing
    U.exp := E
    If U.exp == F
        We are okay, until we find out that U.exp != U
    Else
        S should have been F, if we guessed the values correctly
        if U.exp == U && (U is not widely known)
            trouble
        else adjust U.exp
            we are okay
Else
    If E == S //possible trouble
        If F == S //at least not lying
            If U ==S
                Low level of triviality,
                but should be checked
            Else
                OK, according to WD-inference
```

```
        Else
            Real trouble //lied, but why?
            If S has lower function value than F
                        //(in this case, cabin class)
                inside threat
            Else
                Pretending / Social engineering?
    Else
        //Should be okay
        If U != S
            Fits the expected behavior
        Else
            Accommodating group
```

A/B's verbal output can have any significant situation as a topic. Each situation will replace the attributes and their values characterizing the flight situation with those appropriate to the topic. We imagine that typical situations and their properties, complete with the incriminating values, are specified in writing for the benefit and training of human counter-espionage experts. It would be no trouble for OST to read, understand, and formalize the salient information from these manuals for use in a computational unintentional inference system of the kind we are describing here. This information will be part of encyclopedic knowledge resource similar to the locations of the paintings in the examples of Section 1. Also, in casual conversations about sensitive professional information, it might be possible for OST, to the extent it is possible for human participants or post-factum investigators, to tell apart a naïve insider as well as a social engineer from a malicious insider, based on the grain size of the information conveyed.

5.2 Social Engineering

Social engineering presents a somewhat less varied case than insider threat in spite of an obvious complication, namely, the brevity of a typical hit and thus the small amount of text to analyze for unintentional inference clues. In line with the issue of 'representing' (Bishop *et al.* 2008), Mitnick (2002) refers to it simply as "getting people to do things they wouldn't ordinarily do for a stranger," while actually being one[6] (see also Mitnick and Simon 2005 and Long *et al.* 2008). In other words, a social engineer called A/B passes themselves for somebody they are not, typically an employee of the targeted company or its contractor, in order to gain access to their computer network, preferably as a user with appropriate privileges.

"If someone calls you and uses all the slangy, insider terms of your business, seems conversant in numbering systems unique to your office, and even mirrors your feelings about management and customers, you are going to think that person as is an 'us,' not a 'them.' " (Long *et al.* 2002). This being the most common and successful approach, typically implemented in no more than two brief conversations, often over the phone, the operation leaves pretty little text on tape. Nevertheless, the unintended inference mechanism can successfully identify an overload of the defaults of a group whose membership A/B claims.

It may be assumed that a real insider will find it unnecessary to overwhelm their victim with proof of their membership, taking it for granted and readily producing factual, documented proofs if challenged. On the contrary, similar to lines 9-12 and 21-24 in Table 2 above, A/B will state explicitly several of the group's defaults (or complete lists of the range of a property) in order to establish their non-existent insider status. Unlike those lines, however, the statement of the default is always suspicious in a potential social engineering situation, i.e., when a company employee is addressed by a stranger.

For the unintended inference mechanism to work, the system needs to have a reasonably representative list of properties with their ranges that can be used in this situation. On top of the information listed in the quotation above, the names of other employees and contacts obtained from a previous encounter are used routinely. The comparison of the conversation with such a property list should raise a flag if the number of explicitly stated defaults exceeds a predefined threshold, which may be as low as two.

The recommended lines of defense in the literature include raising the employees' awareness through seminars or posters. The unintended inference mechanism can actually prompt an employee in real time to introduce a value from the range of a property not used by A/B, such as the name of a non-existing fellow employee or a department name, in order to trick A/B into admitting that they know him or it. An experienced social engineer, however, will try to avoid answering the question and to fill the vacuum with asserting information about other properties for which they have already obtained some validating material. Additional distraction can be also used, preferably tied to the original question.

For example, in a scene from the movie "Inglourious Basterds," an English officer in the uniform of a German captain is asked about the origin of his weird accent. When an attempt to avoid answering the question fails, he has to invent a remote mountain village and immediately dives into a tale about his family appearing on skis in a famous Nazi movie, where the lead role was played by his accomplice, present at the table, who enthusiastically confirms the story. The social engineering sources quoted above relate real-life episodes of successful break-ins that follow the same structural lines, and they are all largely susceptible to the unintended inference treatment.

The algorithm of the previous section can be considerably simplified as, in this case, we are only dealing with columns E and S of Table 2.

```
Given a list L[n] of properties with corresponding values, including
the defaults for this group
Count_defaults=0
While conversation lasts
    If L[i].S == L[i].E
        Count_default ++
    If count_default > allowed_default_threshold
        Choose unused L[k] and ask a casual question with a
false value of a property
        If incorrect response or inconclusive response followed
by a set of (distracting) sentences
                Flag, this is a problem
```

The very simplicity of the algorithm enables the system to analyze the data in real time and to deliver the prompt to the targeted

[6] Greenwald (2010b) accepts this definition but rejects the one implied by the next sentence that formulates a common understanding of social engineering as passing oneself for something one is not. He cites a couple of convincing real-life cases he has actually witnessed, where an outsider, flaunting their outside status, was "adopted" as one of their own by friendly insiders. This kind of situation blends, in a sense, with insider threat.

employee, followed, if appropriate, by the flag, while the conversation is still happening.

6. OST IN ACTION

We will now demonstrate, again on the cabin class example, how OST implements the unintentional inference system to flag suspicion in a potential insider threat situation. We consider the following to be available: text (T) produced by A/B; a list (L) of items to watch for in employees' lifestyle and behavior that the company/agency uses for security purposes; OST ontology, lexicon, InfoStore, and Semantic Text Analyzer (STAn) to translate text into TMRs in machine understandable form that is stored in the InfoStore knowledge base; a database containing information about the company or agency departments, employees (and their salaries, etc.); a database containing information about company business travel, including the record of every past flight of employees. (Additionally, at an advanced stage of investigation, the airlines' records of personal travel for a suspect employee can be obtained on a search warrant.)

In the process of reading and interpreting text T, as shown in Section 2 above, the system looks for sentences that pertain to any item on the watch list L. One of these items is related to A/B's expenses in relation to their official income. Among those there are A/B's travel expenses, including the airplane ticket prices that A/B has paid out of pocket.

Suppose the following sentence is detected: *I need to replace my laptop battery—it died one quarter of the way to LA last week, so I couldn't finish the presentation.* The sentence is about a problem with a laptop battery. The information about a flight to Los Angeles, presumably from the East Coast, and therefore the longest it can be domestically, is in the background. (Note that no keyword search for any flight information will detect this sentence.)

First, as per Section 2, OST will produce three TMRs, one for each clause, and connect them. The first TMR, for *I need to replace my laptop battery,* corresponds to the following:

```
(buy1     (agent(value(human1)))
          (theme(value(battery1
                    (part-of-object(value(laptop-computer1)))
          )))
          (iteration(greater-than(1)))
          (free-will(value(low)))
)
```

The second TMR, for *it died one quarter of the way to LA last week*, corresponds to the following:

```
(use-artifact1    (phase(value(end)))
          (instrument(value(battery1
                    (part-of-object(value(laptop-computer1)))
          (during(value(travel1
                    (instrument(sem(airplane1)))
                    (end-location(value(Los-Angeles1)))
                    (start-location(value(unknown)))
                    (time(value(week
                              (number(equal-to(-1)))
                    )))
          )))
          (time-phase(equal-to(.25)))
)
```

Finally, the third TMR, for *I couldn't finish the presentation*, is

```
(create1  (agent(value(human1)))
          (theme(value(computer-file1
                    (purpose-of(sem(present)))
          )))
          (phase(value(end)))
          (success(value(0)))
)
```

The three TMRs are connected by these two relations:

```
TMR1(effect-of(value(TMR2)))
TMR2(cause(value(TMR3)))
```

While, for brevity's sake, we did not expand every concept in the TMRs above into their full ontological form, complete with the properties, facets, and values, from the concept LAPTOP-COMPUTER we know that it can only be powered through a power cord or a laptop battery. The concept AIRPLANE has the property CABIN-CLASS, with values COACH and BUSINESS/FIRST. Each has SEATs in the cabin (inherited from AIRPLANE as a filler for the property HAS-OBJECT-AS-PART). The seats in the BUSINESS/FIRST class have additional property HAS-OBJECT-AS-PART with the filler ELECTRICAL-OUTLET, while the SEATs in the COACH class do not. (This is a simplified representation, without the potential for selected coach seats to have an outlet, as well as other classes of travel).

The speaker was unable to use an electrical outlet during the flight because their only source of power was a laptop battery, so the OST inferencing mechanism will be able to fill the property cabin-class in the concept AIRPLANE of the second TMR with COACH.

This information was unintentionally revealed by the speaker. Depending on the other property values pertaining to air travel in general that the speaker states explicitly or reveals unintentionally in their other conversations, blogs, etc., the information above—along with all such pertinent information synthesized from the OST InfoStore—will fit into the appropriate line of Table 2 in the previous section and trigger the corresponding response.

Let us consider a slightly modified version of the sentence: *I need to replace my laptop battery—it died one quarter of the way to LA last week, good that I didn't check the power cord.* A similar analysis will reveal that the speaker flew business class and trigger the appropriate action, as per Table 2. Again, as this is slightly simplified, a thorough investigation should determine if the speaker was fortunate to get a coach seat with a working outlet. It is very likely, however, that a traveler would mention this as an unusual occurrence.

7. CONCLUSION/ ACKNOWLEDGEMENTS

In this paper, we have demonstrated how an unintentional inference can be used as a line of protection against insider threat and social engineering, given access to an individual's casual and unsolicited verbal output such as blogs, tweets, Facebook updates and conversations with friends, relatives, and colleagues. The unintended inference reaches into text without any apparent contradictions or other visible flags. We have also illustrated the use of OST to automate this process in a computational system, parts of which have already been implemented while others are conceptualized and algorithmized. We believe that OST-based systems like the one described will find further IAS applications, beyond those already implemented.

We are grateful to Steven J. Greenwald for his helpful comments on an earlier draft of the paper (Greenwald 2010a, 2010b). Among other things, cited and commented upon above, we would like to respond to his expression of fear that the system outlined above would greatly strengthen "the police state." We have assuaged the fear of the system's spying on any individual by comparing it with the search warrant that is equally revealing but is tightly limited and controlled constitutionally and severely limited by the courts. We assume that the system will only target an individual with an appropriate search warrant. We certainly agree with Greenwald that the internal resources of the system, especially the ontology (low levels) and the specifically forensic reasoning rules, should be protected to prevent gaming. We are also grateful to the anonymous reviewers for the helpful comments and to Mark Burgess, the NSPW-appointed "Shepherd."

8. REFERENCES

[1] Anderson, J. P. 1972. *Computer Security Technology Planning Study, Vols. I, II*. Bedford, MA: AFSC,

[2] Atallah, M. J., Raskin, V., Crogan, M., Hempelmann, C. F., Kerschbaum, F., Mohamed, D., and Naik, S. 2001. Natural Language Watermarking: Design, Analysis, and a Proof-of-Concept Implementation. In I. S. Moskowitz. Ed. *Information Hiding: 4th International Workshop, IH 2001, Pittsburgh, PA, USA, April 2001 Proceedings.* Berlin: Springer, 185-199.

[3] Atallah, M. J., Raskin, V., Hempelmann, C. F., Karahan, M., Sion, R., Topkara, U., and Triezenberg, K. E. 2002. Natural Language Watermarking and Tamperproofing. In F. A. P. Petitcolas, Ed. *Information Hiding: 5th International Workshop, IH 2002, Proceedings.* Berlin: Springer, 196-210.

[4] Bishop, M. 2005. Position: Insider is relative. In C. F. Hempelmann and V, Raskin, Eds., *Proceedings: New Security Paradigms Workshop 2004. September 20-23, White Beach Resort, Halifax, Canada.* New York: ACM Press.

[5] Bishop, M., and Gates, C. 2008. Defining the Insider Threat. In *Proceedings of the Cyber Security and Information Intelligence Research Workshop,* Paper #15.

[6] Bishop, M., Gollmann, D., Hunker, J., and Probst, C. W. 2008. Countering insider threat. In *Proceedings of the Dagstuhl Seminar 08302*, July 20-25.

[7] Brackney, R., and Anderson, R. 2004. Understanding the insider threat. In: *Proceedings of a March 2004 Workshop*. Technical report, RAND Corporation, Santa Monica, CA.

[8] Bresnan, J. 1983. *The Mental Representation of Grammatical Relations*. Cambridge, MA: MIT Press.

[9] Cole, E., and Ring, S. 2006. *Insider Threat: Protecting the Enterprise from Sabotage, Spying, and Theft.* Rockland, MA: Syngress.

[10] Devedzic, V. 2002. Understanding Ontological Engineering. In *Communications of the ACM*.

[11] Farahmand, F., and Spafford, E. H. 2010. Understanding Insiders: An Analysis of Risk-Taking Behavior. *Information Systems Frontiers (*forthcoming).

[12] Fridman-Noy, N., and Hafner, C. D. 1997. The State of the Art in Ontology Design: A Survey and Comparative Review. *AI Magazine* 18(3): 53-74.

[13] Gibbons, J. 2003. *Forensic Linguistics: An Introduction to Language in the Justice System***,** Malden, MA: Blackwell.

[14] Greenwald, S. J. 2010a**.** Re: Fwd: Partially changed from v12. E-mail of 5:30 p.m. EDT, April 20.

[15] Greenwald, S. J. 2010b**.** Re: Partially changed from v12. E-mail of 8:33 p.m. EDT, April 20

[16] Greitzer, F. L., Moore, A. P., Cappelli, D. M., Andrews, D. H., Carroll, L. A., and Hullet, T. D. 2008. Combating the Insider Cyber Threat. *IEEE Security and Privacy*, 61-64.

[17] Grice, H. P. 1975 Logic and conversation. In: P. Cole and J. L. Morgan, Eds. *Syntax and Semantics. Vol.3. Speech Act***s**. New York: Academic Press, 41-58

[18] Guarino, N. 2004. Toward a Formal Evaluation of Ontology Quality. *IEEE intelligent Systems* 19(4): 78-79.

[19] Hempelmann, C. F. 2007. Beyond proof-of-concept: Implementing ontological semantics as a commercial product. In V. Raskin and J. M. Spartz. Eds. *Proceedings of the 4th Midwest Computational Linguistics Colloquium 2007.* Purdue University, West Lafayette, Indiana. April 28.

[20] Hempelmann, C. F., Taylor, J. M., and Raskin, V. 2010. Application-guided ontological engineering, In H.A. Arabnia, D. de la Fuente, E. B. Kozerenko, and J. A. Olivas. Eds. *Proceedings of International Conference on Artificial Intelligence,* Las Vegas, NE, July 2010.

[21] Lenat, D. B. 1990. CYC: Toward programs with common sense. *Communications of the ACM* 33(8): 30–49.

[22] Levinson, S. C. 1983. *Pragmatics*. Cambridge, UK: Cambridge University Press.

[23] Long, J., Wiles, J., Pinzon, S., and Mitnick, K. D. 2008. *No Tech Hacking: A Guide to Social Engineering, Dumpster Diving, and Shoulder Surfing*. Rockland, MA: Syngress.

[24] Malinowski, B. 1923. The problem of meaning in primitive languages. Supplement to C. K. Ogden and I. A. Richards, *The Meaning of Meaning*. London: Routledge and Kegan Paul, 146-152.

[25] Maloof, M. A., and Stephens, G. D. 2007. ELICIT: A system for detecting insiders who violate need-to-know, In *Proceedings of RAID 2007, Lecture Notes in Computer Science 4637*. New York: Springer, 146–166

[26] McMenamin, G. R. 2002. *Forensic Linguistics: Advances in Forensic Stylistics***.** Boca Raton, LA: CRC Press.

[27] Mitnick, K. D., Simon, W. L., and Wozniak, S. 2002. *The Art of Deception: Controlling the Human Element of Security***.** Indianapolis: Wiley.

[28] Mitnick, K. D., and Simon, W. L. 2005. *The Art of Intrusion: The Real Stories Behind the Exploits of Hackers, Intruders and Deceivers.* Indianapolis: Wiley.

[29] Nirenburg, S., and Raskin, V. 2004. *Ontological Semantics*. Cambridge, MA: MIT Press.

[30] Obrst, L. 2007. Ontology and ontologies: why it and they matter to the intelligence community." In *Proceedings of the Second International Ontology for the Intelligence Community Conference. OIC-2007*. Columbia, MD. November 28-29.

[31] Olsson, J. 2004. *Forensic Linguistics: An Introduction to Language, Crime and the Law***.** New York: Continuum.

[32] Prince, E. 1981. Towards a taxonomy of given-new information. In: Cole P. (ed.), *Radical Pragmatics*. New York, NY: Academic, 223-255.

[33] Raskin, V. 2005. The threshold of triviality in telling tales: Is it inherent in inferences? In: S. Attardo and L. Birden, Eds. *Abstracts of ISHS 2005, the 17th Annual Conference of the International Society of Humor Studies*. Youngstown, OH: Youngstown State University.

[34] Raskin, V., Atallah, M. J., McDonough, C. J., and Nirenburg, S. 2001. Natural language processing for information assurance and security: An overview and implementations. In: M. Schaefer, Ed. *Proceedings. New Security Paradigm Workshop. September 18th-22nd, 2000, Ballycotton, County Cork Ireland.* New York: ACM Press, 51-65.

[35] Raskin, V., Hempelmann, C. F., and Taylor, J. M. 2010. Guessing vs. Knowing: The Two Approaches to Semantics in Natural Language Processing, In A. E. Kibrik. Ed**.** *Proceedings of Annual International Conference Dialogue 2010*, Moscow, Russia, May 2010

[36] Raskin, V., Hempelmann, C. F., and Triezenberg, K. E. 2004. Semantic forensics: NLP Systems for deception detection. In D. Cavar, and P. Rodriguez. Eds. P*roceedings of the First Annual Midwest Colloquium in Computational Linguistics*. Bloomington, IN: Indiana University

[37] Raskin, V., Hempelmann, C. F., Triezenberg, K. E., and Nirenburg, S. 2002a. Ontology in information security: A useful theoretical foundation and methodological tool. In: Raskin, V. and C. F. Hempelmann, Eds. *Proceedings. New Security Paradigms Workshop 2001. September 10th-13th, Cloudcroft, NM, USA*. New York: ACM Press, 53-59.

[38] Raskin, V., Nirenburg, S., Atallah, M. J., Hempelmann, C. F., and Triezenberg, K. E. 2002b. Why NLP should move into IAS. In: S. Krauwer, Ed. *Proceedings of the Workshop on a Roadmap for Computational Linguistics*. Taipei, Taiwan: Academia Sinica, 2002: 1-7

[39] Raskin, V., Nirenburg, S., Hempelmann, C. F., Nirenburg, I., and Triezenberg, K. E. 2003. The genesis of a script for bankruptcy in ontological semantics. In: G. Hirst and S. Nirenburg, Eds. *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning, Edmonton, Canada.* ACL, 30-37.

[40] Shuy, R. W. 1993. *Language Crimes*. Oxford: Blackwell.

[41] Shuy, R. W. 1998. *The Language of Confession, Interrogation and Deception***.** Thousand Oaks, CA: Sage.

[42] Shuy, R. W. 2005. *Creating Language Crimes: How Law Enforcement Uses (and Misuses) Language*i New York: Oxford UP.

[43] Smith, B. 1995. Formal ontology, commonsense and cognitive science. *International Journal of Human Computer Studies* 43(5/6), 626-640.

[44] Sowa, J. F. 2000. *Knowledge Representation: Logical, Philosophical, and Computational Foundation*. Pacific Grove, CA: Brooks/Cole.

[45] Stolfo, S. J., Bellovin, S. M., Hershkop, S., Keromytis, A., Sinclair, S., and Smith, S. W., Eds. 2008. *Insider Attack and Cyber Security: Beyond the Hacker*. New York: Springer.

[46] Stamper, C. L., and Masteson, S. 2002. Insider or outsider? How employee perception of insider status affect their work behavior. *Journal of Organizational Behavior*, 23, 875-894.

[47] Taylor, J. M., Raskin, V,, Hempelmann, C. F., and Attardo, S. 2010. An unintentional inference and ontological property defaults. In *Proceedings of SMC 2010: IEEE International Conference on Systems, Man, and Cybernetics*. Istanbul, Turkey, October 10-13.

[48] Wagstaff, S. S., and Atallah, M. J. 1999. Watermarking with quadratic residues. In *Proceedings of the IS&T/SPIE Conference on Security and Watermarking of Multimedia Contents*. SPIE—The International Society for Optical Engineering, San Jose, CA, 3657, 283-288.

[49] Ware, W. H. 1970. *Security Controls for Computer Systems. (U): Report of Defense Science Board Task Force on Computer Security*. The Rand Corporation for the Office of the Director of Defense Research and Engineering.

[50] Willison, R., and Siponen, M. 2009. Overcoming the insider: Reducing employee computer crime through situational crime prevention, *Communications of the ACM*, 52 (9), 133-137.

[51] Wood, B. 2000. An Insider Threat Model for Adversary Simulation. *SRI International, Research on Mitigating the Insider Threat to Information Systems – #2 Proceedings of a Workshop Held by RAND, August 2000*.