Natural Language Processing for Information Assurance and Security: An Overview and Implementations

Mikhail J. Atallah, Craig J. McDonough, Victor Raskin
Center for Education and Research in Information Assurance and Security
(CERIAS: www.cerias.purdue.edu)
Purdue University
W. Lafayette, IN 47907
mja@cs.purdue.edu, mcdonoug@omni.cc.purdue.edu,
vraskin@purdue.edu

Sergei Nirenburg
Computing Research Laboratory, New Mexico State University
Las Cruces, NM 88003
sergei@crl.nmsu.edu

See a version of the paper at
http://omni.cc.purdue.edu/~raskin/NLP-IAS.pdf

Abstract

This paper explores a promising interface between natural language processing (NLP) and information assurance and security (IAS). More specifically, it is devoted to possible applications of the accumulated considerable resources in NLP to IAS. The paper is of a mixed theoretical and empirical nature. Of the four possible venues of applications,

- (i) memorizing randomly generated passwords with the help of automatically generated funny jingles,

- (ii) natural language watermarking,

- (iii) using the available machine translation (MT) systems for (additional) encryption of text messages, and

- (iv) downgrading, or sanitizing classified information in networks, two venues,

(i) and (iv), have been at least partially implemented and the remaining two (ii) and (iii) are being implemented to the proof-of-concept level.

Natural Language Processing = Computational Linguistics = Natural Language Artificial Intelligence

- NLP is an application of linguistics to the study of human-computer interaction in natural language
- NLP designs and implements automatic systems that, typically, take text in a natural language as input, process it according to the predefined tasks, and then generate output, which may be in the same or another natural language or in some other stipulated format, such as, for instance, a database report or a chart.
- NLP's first task was machine translation (MT) between pairs of such best-known and –described languages as English and Russian or English and French.
  - semantic barier
  - knowledge-based comeback
- NLP can be seen, in security terms, as the effort of decoding the meaning of a text from its surface form

Natural Language and Code-Breaking

- The Navajo Episode
- Non-Mathematical Complexity: Need for a Complete NLP System to Decode
- Explore Ways to Utilize This for IAS

Natural Language and Humor Generation for Memorizing Random Strings

• Let S be a random string representing something that a human is supposed to remember, e.g., a password, a PIN, etc.

• How does one construct a mnemonic that helps the human remember S?

• Let us make a jingle out of it!

NLP for Watermarking.

• Let *T* be a natural language text, and let *W* be a string that is much shorter than *T*. We wish to generate natural language text *T′* such that:

• *T′* has essentially the same meaning as *T*;

• *T′* contains *W* as a secret watermark, and the presence of *W* would hold up in court if revealed (e.g., *W* could say, ``This is the Property of X, and was licensed to Y on date Z");

• the watermark *W* is not readable from *T′* without knowledge of the secret key that was used to introduce *W*;

• for someone who knows the secret key, *W* can be obtained from *T′* without knowledge of *T* (so there is no need to permanently store the original, non-watermarked copy of copyrighted material);

• unless someone knows the secret key, *W* is impossible to remove from *T′* without drastically changing the meaning of *T′*;

• the process by which *W* is introduced into *T* to obtain *T′* is not secret, rather, it is the secret key that gives the scheme its security;

• there is built-in resistance to collusion by two people who have differently watermarked versions of the same text, that is, suppose watermarked versions of *T* are sold to *A* and to *B*: if buyer *A* has *T_A′*, where *W_A′* is hidden using a key that is not known to *A*, and buyer *B* has *T_B′* where *W_B′* is hidden using a key that is not known to *B*, then even if *A* and *B* were to share all the information they have they would not be able to either read or delete the watermark (from either *T_A′* or *T_B′*).

Machine Translation (MT) Techniques for Information Security

MT Resources:

- semi-automatically acquired ontology, both general and domain-specific, for over 60,000 nodes and properties;
- semi-automatically acquired lexicons for a growing number of natural languages (already over a dozen at this writing) for over 40,000 word senses;
- an analyzer which translates a text in a natural language into an text-meaning representation (TMR, a language-independent interlingua which represents the meaning of the text);
- a generator which translates a statement in TMR into a text in a given natural language.

In MT, the analyzer goes first and the generator follows. In IAS, the order is reversed. Otherwise, the processes are identical, and the same resources are usable for both purposes.

At present, we have the following capabilities:

- we have Level 2 MT systems for a number of uncommonly known languages along with a semi-automatic system for rapid developments of such systems for other low-density (i.e., not widely used) natural languages, and we can ensure Web access to such systems;
- we can automatically translate the text *T* of a message in English that needs to be transmitted into text *T′* in a low-density language before encrypting it in any other way; we can complicate it further by translating T′ into T″ in yet another low-density language, and so on; and we can vary those languages within our inventory from one transmission to another;
- we can automatically translate messages in a deliberately distorted way while still preserving the appearance of a meaningful text; the distortion may range from the primitive substitution of (selected) words with antonyms to much more sophisticated manipulations on the lexicon;
- we can cause even more complex distortions of texts, still keeping them meaningful and cohesive, by manipulating the ontological nodes evoked by the words in *T*, and only access to the specific ontology will help figure out what *T* is;
- we can also manipulate the analyzer and generator for the same purpose.

Declassification/Downgrading/Sanitizing of Texts

Situation:

- Executive Order 12958, Classified National Security Information of April 17, 1995

- Billions of pages

- hundreds of pages of instructions for human declassifiers (700 pp. in DOE)

- Average productivity: one 20-page document takes abou 15 person-days

Types of Operation:

- weak declassification: dividing a set of documents into definitely open ones and others with a reasonable degree of accuracy;
- strong declassification: determining the status of each document as unclassified or classified without any margin of error;
- downgrading/sanitizing: strong declassification coupled with a seamless modification of each classified document to an unclassified version;
- on-the-fly downgrading/sanitizing: filtering out electronic communication in real time;

Humorous Mnemonics

For any random string S, to generate a meaningful natural language text *T* that is a good mnemonic for S. The requirements for *T* are:

- it should be easy to extract S from *T*: there are many ways of achieving this, including the naive way of using the first letter of every word in *T*;
- *T* itself should be easy to remember: we achieve this by automatically constructing from S a *T* that has meaning, eventually of the humorous kind, because funny things are particularly easy to remember, and we are using the results of pioneering research in computational humor (see Raskin 1985, 1996; Raskin and Attardo 1994).

Differences from existing humor-generation efforts and software:

- a factor that tends to make our problem more difficult is the requirement that *T*, in addition to being ``memorable'', also corresponds to S.
- another complicating factor is that our generation has to use a little more intelligence than, for instance, what extremely little of it is necessary to generate a light bulb joke (Raskin and Attardo 1994) or a cross joke (Binstead and Ritchie 1997) from a standard template.
- a factor that tends to make our problem easier is that the humor generated does not have to be particularly good; a particularly bad joke can be easy to remember precisely because it is so bad (not that the cited toy systems could generate particularly good jokes either!);
- speaking of toy systems, this particular system has a gratifyingly meaningful, non-toy goal.

Below humor generation, there lies a specific natural language generation task, for which there are available abundant resources in NLP, ready for use or well-defined tweaking if necessary.

Initial Implementation

For the initial stage of the implementation, we limited the problem and the output in the following helpful ways:

- the accepted input is a random-generated password which is only alphabetical (not numerical and consisting of exactly eight Latin characters, e.g., *shbvwwlo*;
- the generated output corresponds to one primitive jingle tune only;
- the generated text follows the same meter;
- the generated text follows the same grammatical template; and, of course,
- the generated text consists of 8 words beginning, respectively, with the letters in the random string.

The tune goes TA-ta-TA-ta-TA-ta-TA/TA-ta-TA-ta-TA-ta-TA.

Accordingly, the meter in each of the two identical lines is 4-foot trochaic, with the 4th foot incomplete.

Template

The grammatical template in each identical line is Name, Verb+Past, Name+Poss, Noun.

With W_n, where 1 £ n £ 8, corresponding, obviously, to the *n*th word in the text,

$W_1 = W_3 = W_5 = W_7 =$ Name (= Noun+Proper)

$W_2 = W_6 =$ V+Past

$W_4 = W_8 =$ N+Common

$W_{1\text{-}3}$ and $W_{5\text{-}7}$ are all bisyllabic and trochaic, i.e., stressed on the first syllable

$W_{4,8}$ are monosyllabic.

The jingle for the random string above will be, for instance:

Sandra handled Byron's vault.

William wasted Lana's ore.

Downgrading: Example of a Task

Instruction:

- Allow mention of:

- nuclear submarine

- Disallow mention of:

- their specific deployment
- reactor capacity
- mode of refuelling

Ontological Node for submarine

submarine

(isa	warship)
(theme-of	build, commission, decommission, deploy, destroy, attack)
(instrument-of	attack, support, transport, threaten)
(manned-by	naval crew)
(propel-mode	surface, sub-surface)
(engine-type	nuclear-engine)
(range	N < x < M)
(speed	K < y < L)
(current-location	body-of-water and/or geographic point and/or coordinates and/or relative, time-range)

(prior-location	body-of-water and/or geographic point and/or coordinates and/or relative, time-range)
(next-location	body-of-water and/or geographic point and/or coordinates and/or relative, time-range)
(current-mission	*Z*)