Evaluating Natural Language Generated Database Records

Rita McCardell

Department of Defense
Fort Meade, Maryland 20755

ABSTRACT

With the onslaught of various natural language processing (NLP) systems and their respective applications comes the inevitable task of determining a way in which to compare and thus *evaluate* the output of these systems. This paper focuses on one such evaluation technique that originated from the text understanding system called Project MURASAKI. This evaluation technique quantitatively and qualitatively measures the match (or distance) from the output of one text understanding system to the expected output of another.

Introduction

Project MURASAKI

The purpose of Project MURASAKI is to develop a foreign language text understanding system that will demonstrate the extensibility of message understanding technology.[1] In its current design, Project MURASAKI will process Spanish and Japanese text and extract information in order to generate records in both natural language databases, respectively. The fields within these database records will contain a natural language phrase or expression in that respective language.

The domain of Project MURASAKI is the disease AIDS. The associated software system will include a general domain model of AIDS in the knowledge base. Within this model, there will be five subdomains:

incidence reports records the occurrence of AIDS and HIV infection in countries and regions, among various populations,

testing policies covers measures to test groups for AIDS,

campaigns describes measures adopted to combat AIDS,

new technologies lists new equipment and material used in detecting and preventing AIDS, and

AIDS research details the various vaccines and treatments that are being developed to prevent AIDS.

The subdomains of **incidence reports**, **testing policies** and **campaigns** are found in the Spanish text while the topics of **incidence reports**, **new technologies** and **AIDS research** are covered in the Japanese text.

Project MURASAKI will demonstrate a sufficient level of full text understanding to be able to identify the existence of factual information within either a given Spanish or Japanese text that belongs within a particular Spanish or Japanese language database. Then, it will determine what information in that text constitutes a single record in the selected database.

The balance of this paper will focus on the evaluation technique: why it was chosen, some basic assumptions underlying it, as well as the design and application of this technique. To illustrate various technical points of this technique, examples will be given using text excerpted from the Spanish AIDS corpus and its associated (generated) Spanish database records. Appendix A contains a sample Spanish AIDS text (Text #124) and its English translation.[2] Appendix B contains a record from the Incidence Reporting database that was generated from Text #124. Similarly, Appendix C contains a record from the Testing Policies database that was also generated from Text #124.

The Need for a Black Box

Given the overall design of this foreign language text understanding program, there arose the need for developing a general purpose evaluation technique[1]. This technique would compare the *actual, computer generated* output of one such system to the *expected, human generated* output of another. That is to say, given some sample piece of (foreign language) text as input, some predefined system output (namely, for project MURASAKI, the generation of a finite number of database records) could be manually generated so that a determination as to the correct performance of the computer system was made. Given this type of "correct" output, it could

[1] Thus, it is not to be confused as a message understanding project, but rather a multi-paragraph (i.e., text) understanding project[5].

[2] In the MURASAKI text corpus, there do not exist any English translations for any of the text.

therefore be possible to measure the performance of an automated system based on this type of well-defined input/output pairs. It was precisely this type of rationale that led to the development of a **black box evaluation** — evaluation primarily focused on *what a system produces externally* rather than *what a system does internally*. In direct contrast to this type of evaluation is **glass box evaluation** — "looking inside the system and finding ways of measuring how well it does something, rather than whether or not it does it" [5].

With the development of the MURASAKI evaluation technique, comes the notion of two types of measures: a quantitative measure and a qualitative measure. The **quantitative measure** determines the number of correct (and/or incorrect) records that have been generated in any one database while the **qualitative measure** evaluates the "correctness" of any database record field.

Background

Some Assumptions

Given the overall design of Project MURASAKI, there are a few assumptions, or rather, some groundwork that needs to be laid, in order to proceed in the development of this evaluation technique. These assumptions are explained as follows:

- Given the nature of the AIDS text corpus, any one text could possibly generate one or more records in one or more databases. This fact is loosely referred to as *domain complexity*. (Furthermore, for any record, all fields may not be filled.)
- Given the structure of the AIDS domain model, it is just as easy (or hard) to distinguish one subdomain from another. That is, each database is as likely to have a record generated in it as another. This hypothesis is known as *subdomain differentiation*.
- Upon the determination of what the *expected* output of Project MURASAKI should resemble, a correct record (in any database) is uniquely identified by the contents of its *key fields* plus the contents of one or more *non-key fields*. This statement constitutes the *definition of a correct record*.[3]

Generated Output: What Could Go Wrong?

After a thorough analysis of the system flow for Project MURASAKI and given a typical AIDS text as system input, the following list represents all possible *undesirable* situations that could arise:

1. Generate one or more records in the **wrong** database.
2. **Not** generate one or more records in the correct database.
3. Generate **too many** records in the correct database, i.e., *over-generate*.
4. Generate **too few** records in the correct database, i.e., *under-generate*.
5. Generate **too many** fields in the correct record.
6. Generate **too few** fields in the correct record.
7. Generate the **wrong** answer in the fields.

Situations 1 and 2 illustrate what could go wrong at the *database* level while scenarios 3 and 4 depict possible problems arising at the database *record* level. The remaining criteria (namely 5, 6 and 7) shows what could happen at the database record *field* level. However, the more crucial way of viewing these problems is not so much in **where** (i.e., at what level) these events occur, but rather in **how** these problems can be detected and thus measured for evaluation purposes. It is with this motivation that the following categorization was derived: a *quantitative measure* could be designed to account for the problems that could arise at both the database and database record levels while a *qualitative measure* could comparably be designed for evaluation at the database record field level.

In the next section, two examples are given depicting how the quantitative measure accounts for problems arising at the first two levels. (Note: 'rec.' is the abbreviation for record in these examples.)

A Quantitative Measure

Background

A *scoring function* is used for the quantitative measure to calculate an aggregate score for the number of *correct records* (as defined previously) generated ('gen.' in the following examples) for a given MURASAKI text. This scoring function assigns one point for the generation of a correct record ('cor.') and $-p$ points, where $0 < p < 1$, for the generation of an incorrect record ('inc.').

Some Questions

Given the two examples in Table 1, the following questions come to mind:

- What should be the value of p? $\frac{1}{2}$? $\frac{1}{3}$? $\frac{1}{4}$? Does bounding it between 0 and 1 imply any linguistic restrictions on focus or coverage of the text? Or rather, should these bounds become parameters of this measure?

[3] All appropriate information should be extracted from the text and placed in the correct database. A change in any of the key fields will result in the generation of a new record. For example, if data from a different time period is presented in the text, a key field change is required, and a new record is generated. If data from a new region is presented, a new record is generated. Examples of key and non-key fields are found in Appendices B and C. Key fields, which are found in the thick, darkened boxes, are the same throughout each database.

Ex. #1:	DB #1	DB #2	DB #3	TOTAL	Ex. #2:	DB #1	DB #2	DB #3	TOTAL
Text *xxx* what if, where	3 rec. 2 gen. 1 cor. 1 inc. (1 inc.)	2 rec. 2 gen. 2 cor.	1 rec. 2 gen. 2 inc.	6	Text 124 what if, where	3 rec. 4 gen. 3 cor. 1 inc.	1 rec. 0 gen. 1 inc.	0 rec. 1 gen. 1 inc.	4
	1-2p	2	-2p	$\frac{3-4p}{6}$		3-p	-p	-p	$\frac{3-3p}{4}$

Table 1: Examples of How the Quantitative Measure Works

- Which is worse: to over-generate or under-generate? That is, should we have one penalty for one and another penalty for the other? (In Example #1 of Table 1, the extra, or *over-generated*, record is also penalized by $-p$ points.)
- What happens if the numerator is negative? Or equal to 0? Should the score in these cases be 0?
- If the score for a single text is $Text_i$, then should the scoring algorithm for the overall (average) Quantitative Score be $\frac{\Sigma(Text_i)}{N}$, where $i = 1, 2, \ldots, N$ and N is the total number of text?

A Qualitative Measure

Background

Before proceeding into the design of the qualitative measure, some background is needed in order to motivate this measure. For Project MURASAKI, a database field is defined to be logically equivalent to that of a **SLOT** while the contents of that field is equivalent to its **FILLER**.[4] The slots define three types of **DOMAINS**: (1) unordered, e.g., OCCUPATIONS, (2) ordered, e.g., MONTHS-OF-THE-YEAR and (3) continuous, e.g., HEIGHT. The slot fillers have three types of **ATTRIBUTES**: (1) symbolic, e.g., (temperature(value tepid)), (2) numeric, e.g., (weight(value 141.3)) and (3) hybrid, e.g., (test_results(value(1,000 people were deported))). Also, the slot fillers have three types of **CARDINALITY**: (1) single, e.g., (sex(value male)), (2) enumerated, e.g., (subjects(value(math physics art))) and (3) range, e.g., (age(value(0 100))).

The notion of **IMPORTANCE VALUES** (IVs) are introduced here and are used to numerically describe how easy/hard it was (is) to extract a particular field's (or slot's) information from the text. These importance values are assigned to both the key and the non-key fields of a database record for each of the five databases.[5] Importance values are integers from 1 to 10, inclusive, and are interpreted as follows:

IV	Interpretation
10	very **easy** to extract
⋮	⋮
5	moderately **easy/hard** to extract
⋮	⋮
1	very **hard** to extract

With this view of importance values[6], the extraction process for Project MURASAKI may now be considered as two subprocesses; that is, *extraction* plus *deduction*. For example, the key field *fuente* (meaning "source") may be filled with *OMS* or any one of the other periodicals and technical papers that are listed in the header line of each text (reference Appendix A, where the *fuente* is *El País*). Since the *fuente* field is constrained to only a few possible fillers, an importance value of 9 has been assigned to it.[7]

Scoring Functions & Algorithm

Scoring functions are also used for the qualitative measure to calculate an aggregate penalty for the fields (both key and non-key) in a database record. There are three types of scoring functions based upon the cardinality of the slot fillers: (1) single, (2) enumerated and (3) range.[8] An example of an ordered domain with single fillers is that of TEMPERATURE:

```
(make-frame TEMPERATURE
      (instance-of (value field))
      (database-in (value x))
      (element-type (value symbol))
      (domain-type (value ordered))
      (cardinality (value single))
      (elements (value cold cool tepid
                  lukewarm warm hot scalding)))
```

[4] The origination of this knowledge representation scheme (KRS) was taken from [4]. The application of this KRS to Project MURASAKI was taken from[1].

[5] Recall that each database, for both Spanish and Japanese, corresponds to one of the five different subdomains within the AIDS domain model.

[6] Informal feedback thus far has indicated that these values are geared to having more emphasis placed on the records that contain *easier* fields and less on the *harder* ones, thus not rewarding those who perform well on the harder fields.

[7] An importance value of 10 would have been assigned had it not been for the fact that in some instances, the "deduction" portion of the extraction process for this field specifies the conversion of some sources to their respective acronym, e.g., *OMS* is *Organización Mundial de la Salud (WHO)*.

[8] In Project MURASAKI, only slots that contain *single* fillers have been identified thus far.

(The filler *x* in the *database-in* slot represents the single character identification value for a particular AIDS database.) Continuing with this example, if the following actual output (AO) were to be matched against what was expected (EO, expected output),

AO: (temperature (value cool))
EO: (temperature (value lukewarm))

the penalty assigned to this mismatch would depend on two variables: (1) D, the distance between the fillers in the *ordered* set of values and (2) C, the size of the domain. The scoring function that relates these two variables is

$$P = \frac{W \times D}{\mathcal{F}(C)} \tag{1}$$

where W is the numerical weight on the distance between the fillers and $\mathcal{F}$ is a damping function on the size of the domain.

As mentioned before, an example of an unordered domain with single fillers is OCCUPATIONS. Since the distance, D, is not meaningful for this example, the penalty assigned to the match becomes a function merely of the size of the domain (and hence the probability of the correct filler appearing):

$$P = \frac{W}{\mathcal{F}(C)} \tag{2}$$

Consider the slot CASOS_NOTIFICADOS from the Incidence (I) Reporting database. It is a continuous domain with (single) numeric fillers and its attribute entry is the following:

```
(make-frame CASOS_NOTIFICADOS
        (instance-of (value field))
        (database-in (value I))
        (element-type (value number))
        (domain-type (value continuous))
        (cardinality (value single))
        (unit-size (value 1))
        (elements (value (0 1.000.000))))
```

As before, suppose we are trying to match the CASOS_NOTIFICADOS slots between the actual output and the expected output:

AO: (casos_notificados (value 2.700))
EO: (casos_notificados (value 2.781))

Since only numbers can be represented in a continuous domain, the elements of the domain are defined by giving the endpoints of the domain (or closed interval) and the unit size of representation is used in computing the distance between fillers. When defined in this manner, the same scoring function that was used for an ordered domain with single fillers (namely Equation 1) can be used to compute the penalty for continuous domain sets as well.

The overall Score for a single database record is

$$\frac{\Sigma(IV_i \times P_i)}{\Sigma IV_i} \tag{3}$$

for $i = 1, 2, \ldots,$ (number of fields in that database record). The P_i's are the computed *penalties* between each field of the actual output and the expected output for that particular database record. The IV_i's are the *importance values* for the corresponding fields of that database record.

The Scoring Algorithm that computes the overall qualitative measure for the entire text corpus is given below:

```
for each TEXT
    for each DB RECORD
        for each DB RECORD FIELD
            if EO_field and AO_field are equal
                then no penalty
                else
                    begin
                        compute penalty ;;; based on
                            appropriate scoring function
                        weight penalty ;;; according to
                            the IV of that field
                        add weighted penalty
                            to total record penalty
                    end
```

Some Unresolved Issues

So far, fields that contain either numeric fillers or single word fillers (fillers that are both easily "distanceable") have been discussed. However, one would think that the more linguistically complex fields, i.e., those containing generated natural language phrases, would be more of a true test for the qualitative measure of this evaluation technique. Consider, for example, a non-key field like **población** ("population") (from Appendix C):

AO: **población** inmigrantes
EO: **población**
personas que pretendían entrar en el país ("people who try to enter the country")

How should one extend the current notion of the qualititative measure to include evaluating the *distance* between natural language phrases of this kind? It would appear that **población** would be an unordered domain containing symbolic information. However, what are the elements of this domain? Should they have cardinality *single*? Should they include *only* those phrases that were generated from the expected output or should they *additionally* include all semantically equivalent phrases, i.e., those containing a common set of semantic primitives or attributes, as well? If the latter situation were to prevail, then, in the example listed above, should a penalty be assessed? If so, by how much? Or rather, should one group together all semantically equivalent phrases and then determine the distance between these classes?

Consider another example of an *unordered* domain field from the Testing Policies Database:

AO: **resultados** han deportado a 1000 personas que resultaron

EO: **resultados** desde 1985, han deportado a 1000 personas que resultaron

Should this non-key field be defined as having both a symbolic and numeric, i.e., hybrid, attribute? If so, should a scoring function based on symbolic and numeric text be designed? Given the example above, should a penalty be assigned for lack of a specific time element (in the actual output) or are these phrases semantically equivalent?

A possible algorithmic extension to the current qualitative measure is outlined as follows:

1. for a given database field, **obtain** and **examine** all possible fillers,
2. **group/classify** semantically equivalent phrases (by those that share common semantic primitives/attributes, e.g., theme, agent, actor, time, etc.) and then
3. **calculate** the distance between each group/class (through determining by just how many semantic primitives/attributes they differ from each other).

If this approach were taken, the scoring function of Equation 1 would be applicable where D would be the distance between *classes* of fillers rather than just between the fillers themselves.

Conclusion

It is hoped that this evaluation technique will prove effective for Project MURASAKI and thus become the basis on which to develop a general purpose evaluation tool. Research continues on answering those **quantitative** questions and on resolving those **qualitative** issues.

Acknowledgements

I would like to thank Roberta Merchant, Mary Ellen Okurowski and John Prange for their assistance and support with this work. Also, I would like to thank Tom Doerr who was instrumental with the preparation of this document. But most of all, I would like to thank my mom for everything. It is in her memory that this paper will be presented.

References

[1] McCardell, R. 1990. "An Evaluation Technique for STUP Database Records". An unpublished document.

[2] McCardell, R. 1988. "Lexical Selection for Natural Language Generation". Thesis Proposal, Computer Science Department, University of Maryland Baltimore County.

[3] Merchant, R. and M. E. Okurowski. Personal Communciation. January & February, 1990.

[4] Nirenburg, S., R. McCardell, E. Nyberg, P. Werner, S. Huffman, E. Kenschaft and I. Nirenburg. 1988. DIOGENES-88, CMU Technical Report CMU-CMT-88-107, Center for Machine Translation, Carnegie Mellon University.

[5] Palmer, M., T. Finin, and S. M. Walter. 1989. "Workshop on the Evaluation of Natural Language Processing Systems". RADC-TR-89-302, Final Technical Report, Unisys Paoli Research Center.

Appendix A: Sample Spanish AIDS Text and Translation

##124 08jul89 El País Madrid palabras 89[9]

Los Emiratos Arabes Unidos han deportado, desde 1985, a 1.000
The United Arab Emirates has deported, since 1985, 1,000

personas que resultaron positivas en las pruebas de detección del SIDA y
people who tested positive on AIDS screening tests and

que pretendían entrar en el país. Un portavoz de su embajada en
who tried to enter the country. An embassy spokesperson in

España manifestó que "es las solución menos mala", ya que la nación "es
Spain said that "it is the less harmful solution", because the nation "is

muy pequeña, tiene menos de medio millón de habitantes y no puede
very small, it has less than half a million inhabitants, and it cannot

hacer frente a los enfermos". La Organización Mundial de la Salud ha
care for the patients". The World Health Organization

registrado 10.000 nuevos casos de SIDA en el pasado mes de junio,
registered 10,000 new cases of AIDS last June,

ascendiendo el número total a 167.373. España tiene 2.781 casos
raising the total number to 167,373. Spain has 2,781 cases

registrados.
registered.

[9] This is the header line for Text #124. This article was reported in the *El País* newspaper, located in Madrid, on July 8, 1989 and contains 89 words.

Appendix B: An Incidence Reporting Database Record

INCIDENCIA DEL SIDA

artículo 124-02I fecha 00jun89 fuente El País

región todo el mundo

fuente de la información OMS

VIH: varones _____ mujeres _____ niños _____

categoria _____

infectados por VIH (porcentaje) _____

infectados por VIH (estimados) _____

infectados por VIH (notificados) _____

modo de transmisión _____

prevalencia: _____% de población de _____

tasa de progresión al SIDA: _____% para _____años

tasa de progresión al SIDA: _____% para _____años

tasa de progresión al SIDA: _____% para _____años

tasa de progresión al SIDA: _____% para _____años

período de duplicación _____meses

incremento mensual _____%

SIDA: varones _____ mujeres _____ niños _____

casos notificados 10.000 nuevos casos en junio 1989

casos estimados _____ para año(s) _____

prevalencia: _____% de población de _____

tasa de letalidad _____% / casos notificados en _____

tasa de letalidad _____% / casos notificados antes de _____

fallecidos _____ (número)

fallecidos _____% de los casos notificados _____

relación m:f _____

período de duplicación _____meses

Appendix C: A Testing Policies Database Record

PRUEBAS CONTRA EL SIDA

artículo 124-01T fecha 08jul89 fuente El País
región Los Emiratos Árabes Unidos
fuente de la información portavoz de Los Emiratos Árabes Unidos en España

autoridad de acción _____

nivel de acción _____

período _____

población personas que pretendían entrar en el país
población _____
población _____
población _____

local de la prueba _____

tipo de prueba _____
tipo de prueba _____
tipo de prueba _____

resultados desde 1985, han deportado a 1.000 personas que resultaron positivas