



Event Data Coding with TABARI

Philip A. Schrodtt
Pennsylvania State University

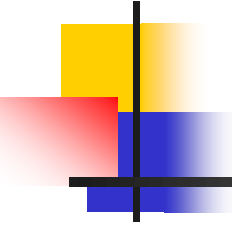
schrodtt@psu.edu

Tools for Text Workshop
University of Washington/Seattle
15 June 2010



Why use Unix?

- Stable
- Free (Linux)
 - Runs identically on servers and laptops
 - Most functions are identical in Mac and Linux
- Used in most cluster computers
- Impressive set of high-level utilities
 - curl from Will's demo
- A great deal of NLP research software is developed first in Unix
- Command line is more efficient than the mouse in advanced applications



Why do we have to learn all this technical crap?!?!

Subtext: I just want to study politics!!!

- Hey, dude, we're "scientists"...
- Could be worse...you could be a medical student...
 - Or a medieval historian
- The past fifty years have, in fact, marked a transition from where political sciences "research tools" consisted of a comfortable chair and a snifter of brandy to where we can make effective use of highly complex machines
 - Burt Monroe's legislative debate data sets at Penn State are second in size to the Sloan Sky Survey of the entire universe
- None of this is likely to change any time soon

TABARI Basics: Resources on the web site



- <http://eventdata.psu.edu>
- TABARI
- Other programs
- Data Sets
- Articles and documentation
 - *Analyzing International Event Data*
 - *Patterns, Rules and Learning*

Reading the Friendly [TABARI] Manual

- Very important
 - “Ode to Coding”
 - Running TABARI
 - Parsing
 - General guides to the data generation process
- Somewhat important (you’ll quickly learn this anyway)
 - Dictionaries
 - Special codes
- Less important
 - *.options* file (unless setting up a new project)
 - Special features (unless doing content analysis)
 - Comparisons with KEDS
 - Diatribes against Microsoft (unless you also hate Microsoft...)



TABARI Basics: Resources on liparidata

liparidata@129.237.60.130; pw Roger!!1105

- TABARI.0.7.3b3
 - TABARI.0.7.Windows
 - TABARI.Source.0.7
 - tabari.manual.0.7.3b3.pdf
- AIED: semi-textbook on event data analysis
- CAMEO.CDB.09b5.pdf: codebook
- Filtering and coding folders
 - Levant.coding.example.dir
 - NexisformatFolder.0906.dir
 - Factiva.Reuters.Levant.2008JFM.dir
 - TABARI.Coding:

TABARI Basics: Data Resources on liparidata

liparidata@129.237.60.130; pw Roger!!1105

- Levant.AFP.CAMEO.0906, Levant.Reuters.CAMEO.0906 : Levant data set, 1979 to June-09, coded from AFP, Reuters
- PITF Worldwide Atrocities Data Set
 - pitf.world.geocode.19950101-20090331.txt
 - Atrocities.codebook.0.9B2.pdf,
- IAEPusersmanual.pdf: Regan political institutions data; 128 variables so useful for machine learning exercises
- Schrodt.PRL.2.0.pdf: unpublishable book on pattern recognition methods for political analysis
 - Bitter? Moi?
- VRA Docs: some hard to find documentation for the VRA data set

TABARI Basics: #1 Problem people have running TABARI



Are your files in Unix format, not Windows or the old Macintosh format?

- If you saved them in Excel, they are not
- If you processed them on a Windows system, they are not

There are a variety of ways to solve this—I use BBEdit—but you have to address it

- Or send me the code to address it

TABARI Basics: #2 Problem people have running TABARI



See problem #1



TABARI Basics: Project File

LEVANT file for TABARI/CAMEO, November 11, 2005

<actorsfile> Levant.actors

<verbsfile> verbs.09b5

<optionsfile> Levant.options

<textfile> Levant.AFPlads.20051101-20060731.leads

<textfile> Levant.AFPlads.20060801-20061031.leads

<textfile> Levant.AFPlads.20061101-20070131.leads

<textfile> Levant.AFPlads.20070201-20070228.leads

<textfile> Levant.AFPlads.20070301-20070331.leads

<textfile> Levant.AFPlads.20070401-20070430.leads

<textfile> Levant.AFPlads.20070501-20070531.leads

<textfile> Levant.AFPlads.20050401-20051031.leads

<problemfile> Levant.problems



TABARI Basics: .options file

COMPLEX: VERBS[6] EXPLAIN

VALID:SOURCE TARGET

SET: MATCH = TRUE

//

//CAMEO Codelist Version: 0.7b3 (Nov 10, 2003)

//

//01: MAKE PUBLIC STATEMENT

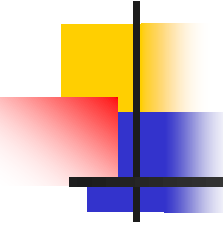
LABEL: 010= Make statement

LABEL: 011= Decline comment

LABEL: 012= Make pessimistic comment

LABEL: 013= Make optimistic comment

LABEL: 014= Consider policy option [continues]



A Review of Basic Grammar (or "Teacher, why do we have to learn to diagram sentences??")

- Subject-verb-object structure of English sentences
 - Sources are subjects
 - Events are transitive verbs, often modified by the object
 - Targets are direct or indirect objects
- Compound phrases and compound sentences
- Subordinate (nonrestrictive) clauses
- The problem of pronoun references
- The problem of disambiguation



Sparse Parsing in TABARI

How TABARI looks at a sentence

Word classes

Common parsing problems

Sparse/Shallow Parsing



- Sparse [shallow] parsing looks only at the syntactic elements of a sentence required for event data coding
 - Generally considered superior to full parsing for domain-specific applications since full parsing is slow and still incomplete
- Subject-verb-object structure
- Compound phrases within a sentence: an event is coded from each phrase
- Compound subjects and objects
- Subordinate phrases—these are ignored
- Pronoun co-references



Automatic text filtering prior to parsing

- All characters are converted to upper-case and any diacriticals are removed (i.e. Ö becomes O, Á becomes A, É becomes E and so forth).
- Periods following lower-case letters are replaced with blanks; this means that the punctuation at the end of a sentence is eliminated but periods in abbreviations (e.g. U.N.) are retained.
- `?' and `!' are replaced with blanks.
- Commas inside numbers are eliminated, so ``1,500" becomes ``1500".
- Semicolons (`;) are replaced with commas.
- Commas and double-quotes (``) are delimited with blanks



TABARI Parsing Operations:

Parsing operations occur in the following order

- Filter the text for punctuation and diacriticals
- Assign types to all words that are in the dictionary; if a word cannot be found it is classified null
- Identify compound actors
- Identify compound noun and adjective phrases
- Dereference pronouns
- Eliminate comma-delimited nonrestrictive clauses



TABARI Parsing Operations, continued:

- Identify the clauses in a compound sentence
- Check the phrases for each verb in each clause of the sentence
- When a verb phrase is found and does not have designated actors, identify source and target:
 - Source is the first actor in the sentence
 - Target is the first actor after the verb that has a code distinct from that of the source; if this does not exist, it is the first actor before the verb



TABARI Word Types

TABARI's parser assigns a number of different types to words in a sentence. Some of these tags appear when the P)arse option is used; others are purely internal.

- Null Word has not been classified (not displayed)
- Litr Literal: word that occurs as part of an actor, verb or pattern
- Actor Actor
- Verb Verb
- Time Time-shifting word (not operationally debugged)
- Attrib Attribution word (not used in CAMEO)
- Determ Determiner: A_, AN_, THE_
- Noun Explicit noun; verb shifted to noun by determiner or capitalization; null-coded actor



TABARI Word Types, continued

- Adjctv Adjective; used to construct compound noun phrases
- Auxil Auxiliary verb—WAS_, WERE_, BEEN_—used in passive voice detection
- Byword BY_, used in passive voice detection
- Comma Comma ",", used in subordinate clause detection and compounds
- Pronoun HE_, SHE_, HIM_, HER_, IT_, THEY_, THEM_, THEIR_
- Conj Conjunction: AND_, BUT_, OR_, NOR_
- Number Strings that begin with a digit except when part of a phrase, as in “Group of 77”
- Issue Word in ISSUES set



TABARI Clause Types

- TABARI's parser also determines several types of clauses.
- Clause Clause in compound sentence
- Compound Compound noun phrase
- Reference Pronoun reference
- Subord Subordinate (nonrestrictive) clause
- Replace Used to standardize actor names in XML input
- NullTag Deactivates tags in subordinate clauses (not displayed)
- Halt End of text (not displayed)



Noun identification

TABARI changes verbs into “nouns” under the following circumstances:

- a. word was preceded by a determiner THE, A or AN
- b. word was capitalized in the middle of a sentence

This is used to deal with words such as ATTACK and FORCE that could be either nouns or verbs.

TABARI also changes actors into “nouns” (which are then ignored as potential actors) when these are null-coded. This prevents the null-coded actors from being included in the complexity count. Date-restrictions are resolved in determining the null coding.



Passive Voice

Passive voice (e.g. “Bosnia was attacked by Serbia”) detection is automatic and does not require specific patterns for each verb. The following rules are used to detect passive voice:

- 1. Auxiliary verb WAS_, WERE_ or BEEN_ occurs one or two words prior to verb
- 2. BY_ immediately follows verb;
- 3. Source and target have not been set by “\$”, “+” or “%” in verb patterns

Passive voice detection forces the subject of any later verbs to be re-evaluated if multiple verbs are coded in a sentence.



Developing TABARI Dictionaries

- General principles of TABARI dictionaries
- Patterns
- Special codes and restrictions
- Common problems



Actors Dictionary

- The "standard dictionary"
- Adding regionally-specific actors and sub-state actors
 - **Individual codes**
 - **Generic agents**
- The *Actor_Filter* program
- Special Codes
 - Date-restricted codes
 - Compound actor codes



<Noun> and <Adjective> lists

- These are optional and go in the *.actors* file.
- Many “null-coded” words in TABARI dictionaries were nouns (and occasionally adjectives) that had the same stem as a verb. These can be moved into the appropriate lists
- TABARI uses both nouns and adjectives in the identification of compound *noun* phrases, which by process of elimination increases the effectiveness of compound *phrase* identification



Modifying the Dictionaries: Verbs

- Stemming — use this with caution
- Null codes—eliminate ambiguous phrases, e.g. "Israeli-occupied", "Iranian-backed"
- Paired codes—allow symmetric events:
X VISITS Y Y HOSTS X



Verbs Dictionary

The *.Verbs* file contains a combination of simple verbs (e.g. PROMISED) and verbs plus associated words (e.g. PROMISED FUNDS). For example

- **ACCEPT**
 - * DEMAND [080]
 - REFUSE TO_ * [120]
 - * CEASEFIRE [0871]
 - * INVITATION TO_ [036:036]
 - * RESPONSIBILITY [015]
 - NOT * MEDIATION [126]

The root verb is ACCEPT and it will match ACCEPT, ACCEPTS, and ACCEPTED.



Verbs Dictionary, continued

- **ACCEPT**
 - * **DEMAND** [080]
 - **REFUSED TO *** [120]
 - * **INVITATION TO_** [036:036]

The phrases that start with “-” are the associated phrases along with their codes, e.g.

- **ACCEPTED DEMAND**
 - will be coded 080 (“yield”) while
- **REFUSED TO ACCEPT**
 - will be coded 120 (“reject”). The phrases tend to involve the direct object of the verb, though this is not always the case.



Verbs Dictionary, continued

- **ACCEPT**
 - * **DEMAND** [080]
 - **REFUSED TO** * [120]
 - * **INVITATION TO**_ [036:036]

The “*” indicates where the verb itself should appear, so the first phrase would match

- **DEMAND WAS ACCEPTED**

If a verb by itself uniquely identifies a code it is on a line by itself along with the code. A verb can also have a default code followed by exception phrases:

- **SAID** [010]
 - * **WILL COOPERATE** [030]
 - * **DETAINED SUSPECT** [173]
 - * **WOULD PROVIDE FOOD** [0333]



Stemming

Stemming can be used to match different forms of the same word using a single string called a “stem”. For example

- **ACCEPT: ACCEPTS ACCEPTED ACCEPTING**
SYRIA: SYRIA'S SYRIAN SYRIANS

This is lemmatization on the cheap. TABARI handles stemming by matching from the beginning of the word. A word is considered to match a stem provided every character in the stem matches. In other words, **SYRI** will match all four forms of **SYRIA** but it will not match **SYRACUSE**.



Stemming, continued

When phrases have the same initial letters, long phrases are checked before shorter ones: For example **SIGNALED**, **SIGNED** and **SIGN** have the search order:

- **SIGNALED**
- **SIGNED**
- **SIGN**

To prevent a character string from being used as a stem, put an underscore (`_`) after the string: this means that the string will match only if it is followed by a space. The phrase **OF_** will only match “**OF**” whereas the stem **OF** would match “**OF**”, “**OFFER**” and “**OFFICIAL.**”

ALWAYS(!) underscore short words!



Stemming, continued

Considerable dictionary tweaking is required to determine which stems can be used without causing problems. For example **TOUR** looks like a useful stem for TOUR, TOURED and TOURING, but it also matches TOURIST. The phrase:

- **HEAD**
- * **FOR**

handles the verb forms HEADING FOR and HEADED FOR but also matches HEADQUARTERS FOR.

To force an entire word to be used in a match, end it with an underscore; alternatively, problematic words such as HEADQUARTERS can be eliminated using null codes (see below).



\$ and + in patterns

Patterns can specify where to look for the source and target; using the characters "\$" and "+". For example, the pattern

- ADVISE
- + WAS * BY \$

would do the correct assignments on the phrases

- EGYPT WAS ADVISED BY THE UNITED STATES

\$ and + are assigned to the first actor that is encountered in the appropriate direction with the search starting at the location of the verb ("*"). In other words in matching

- - + WAS * BY \$

the system will search for the target using the first actor before WAS ... ADVISE and search for the source using the first actor after BY.



% in patterns

The symbol "%" specifies a *compound actor* that should be assigned to both the source and target; it works with either coded or parsed compounds. This is typically used when dealing with consultations:

- REPRESENTATIVES OF SYRIA AND JORDAN WILL MEET IN CAIRO

The pattern

- MEET
- % * IN

would do the correct assignments

- <SYRIA> <MEET> <JORDAN> <JORDAN> <MEET> <SYRIA>

rather than

- <SYRIA> <MEET> <EGYPT> <JORDAN> <MEET> <EGYPT>



^ in patterns

The symbol ^ (caret) can be used in a pattern to skip over an actor without assigning it to a source or target. For example, the sentence

- Russia's president will meet in Switzerland with the U.S. Secretary of State

coded with the pattern

- MEET
- * IN ^ WITH + [044:044]

will generate the event

- RUSGOV USAGOV [044:044]

rather than

- RUSGOV SWZ [044:044]



Paired Codes

Most event coding schemes will generate symmetric events of the form

<Actor₁> <Event₁> <Actor₂>
<Actor₂> <Event₂> <Actor₁>

For example, a visit by a Jordanian official to Syria would generate the pair

JOR 042	SYR	(visit; go to)
SYR 043	JOR	(receive visit; host)

These are coded automatically by using a pair of codes separated by a colon (:)

FLEW
- \$ TO + [042:043]



Paired Codes, continued

There are an assortment of circumstances where the WEIS coding scheme generates symmetric events of the form

<Actor₁> <Event₁> <Actor₂>
<Actor₂> <Event₂> <Actor₁>

For example, negotiation between Israel and Egypt would generate the pair:

ISR 046 EGY (negotiate with)
EGY 046 ISR (negotiate with)

A visit by a Jordanian official to Syria would generate the pair

JORGOV 042 SYR (visit; go to)
SYR 043 JORGOV (receive visit; host)



Paired Codes, continued

Paired codes are coded automatically by using a pair of codes separated by a colon (:); for example

FLEW

- \$ TO + [042:043]

would do the visit-&-receive pair.

Either of the paired codes can be designated as dominant; both events in the pair will be coded.



ALTERNATIVE PATTERNS

Patterns can contain sets of alternative matches; these are done using the following notation:

$$\{ \text{pattern}_1 \mid \text{pattern}_2 \mid \dots \mid \text{pattern}_n \}$$

This will match a string containing

$$\text{pattern}_1 \text{ OR } \text{pattern}_2 \text{ OR } \dots \text{ OR } \text{pattern}_n.$$

The {, |, and } should be space-delimited.



ALTERNATIVE PATTERNS, cont.

An underscore on a word *inside* an alternative set forces the string to match completely—in other words, it deactivates stemming. To force the element *following* a set to be consecutive, put an underscore after the closing bracket. For example

```
{ WAS | IS | WILL_BE }_HERE
```

is equivalent to

```
{ WAS_HERE | IS_HERE | WILL_BE_HERE }
```

There can be a connector prior to the pattern:

```
MILLIONS_OF_{ DOLLARS | EUROS | YEN }
```

is equivalent to

```
{ MILLIONS_OF_DOLLARS | MILLIONS_OF_EUROS | MILLIONS_OF_YEN }
```




Selecting cases

- Discard codes—gets rid of sports events, natural disasters
- Number of words of a certain type
- Location of words (e.g. verb not near beginning)
- Problematic words, e.g. GEORGIA
- Default codes—codes assigned if nothing else is found



Modifying the Dictionaries: Actors

- ❖ Date-restricted actors—change actor code depending on the data, e.g. Boutros Boutros-Ghali, Kiev
- ❖ Coded compound actors—single phrase generates multiple codes, e.g. NORTH_AND_SOUTH_KOREA



Coded Compound Actors

- A single phrase can generate multiple actor codes by separating the actor codes with a slash ('/').
 - NORTH_AND_SOUTH_KOREA [PRK/KOR]
- This method can also be used to expand the membership of alliances if the alliance is not being coded as a distinct actor:
 - G7 [USA/DEU/FRA/ITA/GBR/JAP/CAN]

Note: This is referred to as a *coded* compound actor, as distinct from a *parsed* compound, which is based on the structure of the sentence itself.



Date Restricted Codes

When individuals will change their role over time, multiple coded can be designated with information on when each code should be applied.

Date restrictions can have any of three formats

- <YYMMDD Assign the code for events prior to and including this date
- >YYMMDD Assign the code for events after and including this date
- YYMMDD-YYMMDD Assign the code for events between the two dates

A code without a date restriction will be used as a default. Default codes should be the *last* code in the list.

Codes used with date restrictions can be complex (compound codes and selection codes) as well as simple codes.



Date Restricted Codes: Example

Boutros Boutros-Ghali was in the foreign ministry of Egypt prior to 1 January 1992, but became Secretary General of the United Nations on that date. To code the period 1990-1994, Boutros-Ghali needs to be assigned the code EGY part of the period, and UNO for the remainder.

```
BOUTROS-GHALI [EGY <911231] [UNO >920101]
```

At the end of 1997, Boutros-Ghali left his UN position but continued as an Egyptian diplomatic, so the following default code could be use:

```
BOUTROS-GHALI [UNO 920101-971231] [EGY]
```

Adding date-restrictions to actors can be done separately from coding (Wikipedia and Goggle are very handy for this), and requires a lot of attention, particularly in parliamentary systems.



Modifying the Dictionaries: Selection

- Null codes—eliminate phrases that would otherwise be confused for actors or verbs
- Discard codes—gets rid of sports events, natural disasters
- Default codes—codes assigned if nothing else is found



Null Code [---]

- The null code "---" is used to eliminate phrases that would otherwise be confused for actors or verbs. Phrases with null codes can be in either the Actors or Verbs file.
- Experience has shown that finding the appropriate phrases to be null coded is a key element in bringing the accuracy of a coding system above the 75% level.
- Many of the null-coded words in older dictionaries are in fact nouns and adjectives, and should be moved to those lists.



Null Code [---]: Example

Using the actors

ISRAEL [ISR]
WEST BANK [PAL]

the phrase

ISRAELI-OCCUPIED WEST BANK AND GAZA

will generate both ISR and PAL as actors. By adding the null code

ISRAELI-OCCUPIED [---]

only PAL is generated as an actor.



Discard Code [###]

The text is likely to contain some events which involve multiple international actors but which are non-political: sports events are the most common; traffic accidents and natural disasters involving tourists a close second. These can be automatically discarded by using the discard code [###].

Excessive number of actors



Syria said today the U.S. veto of a U.N. Security Council motion on Israeli settlements was "the most prominent phenomenon of U.S. hostility to the Arabs and U.S. support for Israeli plans to annex the West Bank"

Excessive number of verbs



The PLO, **raising the stakes** before **renewed** Middle East peace **talks**, has **accused** the U.S. of **cheating** Palestinians by **reneging** on **promises** to **grant** Israel \$10-billion in **loan guarantees** only if it **halted** all settlements in occupied territories.



COMPLEXITY CONDITIONS

This option defines the conditions under which a sentence is considered too complex to code. In autocoding mode, these sentences are diverted to a ".*complex*" file.

- VERBS[n] Source rejected if it contains n or more verbs
- ACTORS[n] Source rejected if it contains n or more actors
- PRONOUNS[n] Source rejected if it contains n or more pronouns
- CONJ[n] Source rejected if it contains n or more conjunctions
- SUBORD[n] Source rejected if it contains n or more comma-delimited subordinate clauses

- LATEVERB[n] Source rejected if there is no verb within n words of the beginning of the sentence. If LATEVERB is active, a sentence will also be rejected if it contains no verb.

COMPLEXITY, continued

- **NOACTPRIOR** Source rejected if no actor occurs prior to the first verb
- **NOACTAFTER** Source rejected if no actor occurs after the first verb
- **NOVERB** Source rejected if there is no verb
- **NOSOURCE** Source rejected if there is no source
- **NOTARGET** Source rejected if there is no target
- **NOEVENT** Source rejected if there is no event
- **EXPLAIN** Inserts an explanation of why the source was rejected into the output file inside `/*...*/` delimiters; also displays this on the screen when not autocoding

Example

- **COMPLEX: VERBS[5] ACTORS[8] LATEVERB[8] NOVERB PRACT**



SHOW PARSED TEXT

The parsed text display shows the text as "seen" by TABARI with words color-coded by type:

- Red Actor
- Blue Verb
- Green Conjunction
- Magenta Noun
- Maroon Determiner
- Lime Pronoun
- Black Adjectives, literals
- Grey Untyped words

Compound actors, noun and adjective phrases are in [...]; compound clauses are in {... }.

The display is in the file `TABARI.Parse.html`, which can be opened in a browser.



Parsing: Common Problems

- Pronoun coreferencing
 - Within sentences—see rules in manual
 - Between sentences—pronoun forwarding
- Compound nouns that are mistaken for compound clauses
 - “Palestinian police and demonstrators threw stones at Israeli...”
- Noun-verb disambiguation
 - TABARI, unlike KEDS, uses determiners and capitalization to tag nouns. It could also use adjectives
- Verb disambiguation
 - This is largely handled through verb phrases, particularly null-coded phrases



Parsing: Common Problems, continued

- Stemming
 - Works but it is risky; a better approach would be to explicitly define allowable verb (tense) and noun (plural, possessive, adjectival) forms. The VRA coder apparently does this.
- Large number of words between parts of a pattern
 - This could be modified by placing a restriction on the pattern matching
- Missing actors
 - Usually not a problem when the actor dictionaries have been augmented with Actor_Filter results and spot-checking. But if the actor dictionaries are incomplete, the target may be incorrectly taken from a subordinate clause



Coding problems found in news wire texts

The following examples are taken from actual Reuters lead sentences and illustrate a number of common problems encountered by automated coding systems. Similar problems will be found in AFP.



Reuters Problems: Attribution

A Palestinian suspected of collaborating with Israel died Saturday after being stabbed by masked Arabs in the occupied Gaza Strip, **hospital officials said**.

Iraq said Saturday it did not intend to breach Kuwait's sovereignty but Iraqi smugglers could be crossing the border to hunt for weapons abandoned in the Gulf War.

An influential U.S. lawmaker said he might block further action on foreign aid this session of Congress, a move that could stall \$10 billion in loan guarantees Israel wants from the United States, the **Washington Post said** Friday.

On the eve of a visit by the U.N. General Assembly's president, Israeli army gunfire killed four Palestinians as the occupied territories erupted in violence for the second time in four days. [**no explicit source**]



Reuters Problems: Unidentified source

A suicide car bomber has killed 12 Israeli soldiers and wounded 14 just across the Lebanese border in the most lethal attack against Israeli forces since the start of their withdrawal from South Lebanon.

An Arab shot and killed by two men in downtown Athens was identified today as a top-ranking member of the Palestine Liberation Organization, the group said today.

Reuters Problems: Feature stories



15 of the 20 leads on 1 January 91 deal with meetings, warnings, political comments, troop and refugee movements and other activities that can be coded as events, but five do not:

New Year's Day moved the countdown to the U.N. deadline for Iraqi forces to leave Kuwait into its final fortnight, with no one optimistic that last-ditch diplomatic efforts would avert a war.

U.S.-led forces are likely to take up to three days, rather than hours, to gain air supremacy over Iraq if war erupts, Western military sources say.

Britain was worried that Iraq might attack Kuwait 30 years ago and drew up plans to dislodge Iraqi troops from the territory, according to cabinet papers.

An African witch doctor who divines the future with magical stones says there will be a short war in the Persian Gulf which will see limited loss of life and the defeat of Iraq.

"Storming" Norman Schwarzkopf, commander of half a million Americans poised for war against Iraq, is a tough guy in public whose idea of relaxation is to listen to wild ducks squawk on tape.



The general issue of passive voice: OVS ordering

An Israeli army patrol **was attacked by** Hamas rocket fire last night, the Jerusalem Post reports.

KEDS coding:

PSEREBHMS ISRMIL 194 (fight with artillery, tanks)

TABARI is supposed to deal with passive voice automatically. There are reports that this does always work; examples would be helpful

General problems: Ambiguous words



Noun Verbs: FORCE and ATTACK

- As nouns: "A guerrilla **force** launched an **attack**"
- As verbs: "Rebel radio said guerrillas would **attack** in order to **force** concessions"

Also: ARMS, BATTLE, FIRE, HELP, ORDER, PLAN, PLEDGE, STRIKE and SUPPORT

TABARI will correctly disambiguate nouns preceded by an article so it would get “attack”, but probably not get “force”, since “guerrilla” is probably an actor.



General problems:Marker words

Short, common words—”marker words”

- BY — 29 definitions in the *Random House College Dictionary*
- IN— 31 definitions
- TO—25 definitions

See also: French "de", Arabic "fi", German “zu”

These don't cause problems for TABARI itself, but sometimes cause problems in patterns when they occur in an unexpected context



TABARI Problems:

Actor name occurring before the actual target

European Union governments agreed in principle on Monday to a **German** proposal for EU financial aid to **Kenya**.

IGOEEC DEU [0331] (Express intent to provide economic aid)

The more general problem here is a triadic—rather than dyadic—event. These are fairly common, but most event data coding schemes are intrinsically dyadic.

TABARI Problems:

Event near the end of the lead

A surprise setback today hit efforts to end the war between Israeli forces and Palestinian guerrillas when Syria rejected the idea of the entire Palestine Liberation Organization moving to its territory

ISRMIL PALREB 120 (Reject)

TABARI Problems:

Incorrect interpretation of conjunctions

An artillery battle between Israeli forces and Palestinian rebels in Beirut broke a 24-hour-old ceasefire today as President Reagan agreed in principle to send U.S. troops to help evacuate Palestinians from the city

ISRMIL USA 196 (Violate ceasefire)
PSEREB USA 196 (Violate ceasefire)