The Design and Implementation of A Part of Speech Tagger for English

Jinxi Xu, John Broglio, Bruce Croft

June 20, 1994

1 Introduction

A part of speech tagger is a program that assigns parts of speech to English words based on the context in which they appear. This report is about Jtag, a part of speech tagger developed in CIIR at UMASS.

Part speech assignment is hard because of the ambiguity of English words and the flexibility of English grammar. Let us look at this simple sentence:

The fly can fly.

Except for the first word, every word in the sentence is ambiguous. 'Fly' can be either a noun or a verb, and 'can' can be either a modal, a noun or a verb.

Automatic part of speech assignment plays a major role in many important applications, such as Information Retrieval, Intelligent User Interface and Speech Recognition/Synthesis. At the Center for Intelligent Information Retrieval (CIIR) at University of Massachusetts at Amherst we use a tagger for query processing.

We can categorize tagging methodology into two broad categories: the Qualitative approach, exemplified by Marcus' Fidditch parser, and the Quantitative approach, exemplified by Church's Parts program. The Qualitative approach treats grammatical and lexical knowledge as a set of rules whose conditions are to be matched against the actual context to deterministically decide the part of speech of a word. The Quantitative approach treats the problem as a set of random variables with associated probabilities and uses overall likelihood to determine the part of speech of a word.

The Quantitative approach has some especially desirable properties for Information Retrieval:

- Robustness. Quantitative approach can be easily modeled by the so-called n-gram model, and the inference calcaulation is simple arithmetics.

- Ease in obtaining the required knowledge (a large set of probabilities, which can be estimated using large on-line tagged corpora).

- Efficiency. Algorithms for the n-gram model are linear with respect to input length.

- Demonstrated effectiveness. Church's *parts* program, for example, has achieved an accuracy in the high 90 percent.

- Compatibility with the idea behind our probabilistic information retrieval model. This compatibility makes it possible to take advantage of our past experience in the development of the tagger.

2 Tagging algorithm and complexity analysis

We assume that English has a fixed tag set T, which contains all parts of speech of English and a fixed vocabulary W, which contains all possible English words (this is of course only a convenient abstraction). We let:

$P(t|u), u, t \in T$ = the probability of observing part of speech t given the preceding part of speech u.

$P(t|w), t \in T, w \in W$ = the probability that word w has part of speech t.

Let sentence $S = w_1w_2w_3...w_l$, $w_i \in W$. We define an assignment of parts of speech I to sentence S as a sequence of tags $t_1t_2...t_l$, $t_i \in T$, and tag t_i is assigned to word w_i. We define the score of I as a function ,

$$\begin{aligned} f(I) &= P(t_1|w_1) \\ &\quad \times P(t_2|t_1) \times P(t_2|w_2) \\ &\quad \times P(t_3|t_2) \times P(t_3|w_3) \\ &\quad \ldots \\ &\quad \times P(t_l|t_{l-1}) \times P(t_l|w_l) \end{aligned}$$

The best assignment maximizes $f(I)$. Conceptually, the basic tagging algorithm finds the best assignment of the sentence and assigns parts of speech to the words in the sentence accordingly. Note there are n^l possible assignments to sentence S, but most of them have zero-valued scores. By checking only the assignments with nonzero scores, a straightforward algorithm needs to check $r_1 \times r_2 \times ... \times r_l$ assignments, where r_i is the number of possible parts of speech of word w_i. A experiment with a text of moderate length shows the average value of r_i is about 1.8. This means the straightforward algorithm takes exponential time in the length of the sentence.

With a little more thought, we can come up with a nice dynamic programming algorithm which takes linear time ($O(l)$).

Assume w_i has the possible parts of speech t_{i1}, t_{i2}, ..., t_{ir_i}. Let $g(i, j)$ be the score of the best assignment of part of speech to the subsequence $w_1w_2...w_i$ of sentence S, with the condition w_i is assigned the part of speech t_{ij}. It is easy to see that:

$$
\begin{aligned}
g(1,j) &= P(t_{1j}|w_1) \\
g(i,j) &= max\{g(i-1,k) \times P(t_{ij}|t_{i-1k}) \times P(t_{ij}|w_i)|1 \leq k \leq r_i\}, i > 1 \\
maxf(I) &= max\{g(l,j)|1 \leq j \leq r_l\}
\end{aligned}
$$

The construction of the best assignment accompanies the computation of its score.

Complexity analysis:

We begin with the computation of $g(1,1)$, $g(1,2)$, ..., $g(1,r_1)$. Then we compute $g(2,1)$, $g(2,2)$, ..., $g(2,r_2)$, using $g(1,1)$, $g(1,2)$, ... $g(1,r_1)$. This process continues until we get $g(l,1)$, $g(l,2)$, .., $g(l,r_l)$. Finally, we compute the score of the best assignment. The cost to compute $g(i,j)$ is r_{i-1}, which is independent of j. For each i, the cost to compute the set $\{g(i,1), g(i,2), ..., g(i,r_i)\}$ is $r_{i-1} \times r_i$. It follows that the total cost to compute the score of the best assignment is

$$T(S) = \sum_{i=1}^{l-1}(r_i \times r_{i+1}) + r_1 + r_l$$

Since $r_i \leq n$, and n is a constant, $T(S) \leq n + n + (l-1) \times n^2 = O(l)$.

In fact, r_i is usually much smaller than n. For *Jtag* , $n = 80$, but the largest number of parts of speech for a single word in the *Jtag* lexicon is 10. Assume all words uniformly have r possible tags, then $T(S) = O(r^2 \times l)$.

3 Acquisition of probabilities

Since there are no ready values for the lexical and bigram probabilities, we must estimate them somehow. Fortunately, the large hand tagged on-line corpora (we used Brown Corpus and Treebank) now available enable us to do this by counting.

1. Lexicon Acquisition

   The probability $P(t|w)$ is estimated by dividing the number of occurrences of word w as tag t by the total number of occurrences of word w in the training collections (Brown Corpus and Treebank in our case). For example, 'fly' appears 61 times, 44 times as VB (verb) and 17 times as NN (noun). Therefore,

   $$
   \begin{aligned}
   P(\text{VB}|\text{ 'fly'}) &= 44/61 = 0.721 \\
   P(\text{NN}|\text{ 'fly'}) &= 17/61 = 0.279
   \end{aligned}
   $$

   The process is straightforward enough. However, there are still some subtle points to consider:

- The size and collection bias of the training collection affect the completeness and the accuracy of the lexicon. The larger the size and the wider the coverage of the training collection, the better. The Brown Corpus contains about 1 million words and the Treebank contains about 0.7 million words. (The actual Treebank may be much larger, but we only used the part of it available to us). Since the two corpora have quite different formats and tags, our initial experiments were to use only Treebank or Brown Corpus. Neither was large enough. So we used both Brown Corpus and Treebank. The lexicon used all words in Brown Corpus but only the open class words in Treebank. The reason to ignore closed class words in Treebank is that it is very hard to map closed class tags in Treebank to those in Brown Corpus. Since closed class words are usually high frequency words, igoring them has little affect. The resulting lexicon is much better.
- For our specific application, we want the tagger to recognize and categorize some important classes of proper nouns. In our current implementation, such proper nouns are first names, last names, countries, states of United States. Neither Treebank nor Brown Corpus has enough proper nouns and enough information about the proper nouns. (Proper nouns are only labeled with a generic tag in both collections.)

  For this purpose, we included in the lexicon the person name lists of our Inquery information retrieval system. We also add a list of common English titles, all the countries in the world, and all the states of United States. In case of ambiguities, (e.g., 'Jordan' can be a last name or a country), probabilities were assigned by hand. Fortunately, the tagger relies more on context than on probabilities in processing proper nouns.

2. Bigram Acquisition

   The bigram data we currently use is trained from Brown Corpus. By definition,

$$P(t|u) = \frac{number\ of\ sequences\ ut\ in\ Brown\ Corpus}{number\ of\ occurrences\ of\ u\ in\ Brown\ Corpus}$$

   For example, the part of speech AT (article) appears in Brown Corpus for 92020 times, and the sequence of AT NN (article noun) appears for 21271 times. Then

$$P(\mathrm{NN}|\mathrm{AT}) = 21271/92020 = 0.2311563$$

   For bigram probabilities concerning person names, countries and states, which can not be estimated by direct counting, we assign values to them according to common sense:

- $P(\mathrm{FN}|\mathrm{TT})$, $P(\mathrm{LN}|\mathrm{TT})$, $P(\mathrm{LN}|\mathrm{FN})$, $P(\mathrm{LN}|\mathrm{MN})$, and $P(\mathrm{MN}|\mathrm{FN})$. Since such combinations are very common in English, we assign very large values to them. (TT = title, e.g., 'Mr.'; FN = first name; MN = middlename; LN = last name).
- $P(\mathrm{LN}|\mathrm{LN})$, $P(\mathrm{FN}|\mathrm{FN})$, $P(\mathrm{FN}|\mathrm{LN})$, etc. Since such combinations are very rare if at all, we assign very small values to them.
- Other bigram probabilities concerning these proper noun tags inherit the values of the corresponding ones associated with the generic proper noun tag NP, i.e.,

  $$\begin{aligned} P(u|t) &= P(\mathrm{NP}|t) \\ P(t|u) &= P(t|\mathrm{NP}) \\ P(u|v) &= P(\mathrm{NP}|\mathrm{NP}) \end{aligned}$$

  for all u and v in { TT, FN, MN, LN, CN, ST }, and all t not in it. (CN = country, ST = state).

4 Fine tuning of the tagger

1. The choice of tag set

   The basic assumption behind the n-gram model is that contextual probabilities are largely independent of other linguistic properties (e.g., semantics, morphology, speciality of the words concerned, etc). That is not always true. For example, $P(\mathrm{NN}|\mathrm{TOIN})$ (a single noun after preposition 'to') is only 0.63 times as large as $P(\mathrm{NN}|\mathrm{IN})$ (a single noun after a preposition in general). Since both 'to' + *noun* and 'to' + *verb* are very common in English, the tagger would unfairly favor noun as the part of speech of a word of *noun-verb* ambiguity after 'to' if we include 'to' in the generic preposition category. For this reason, we add the tag TOIN in the tag set. In general, we need to split 'large' tags into 'smaller' ones until every tag is statistically homogeneous with relation to each other.

   On the other hand, we wish to minimize the number of tags. Since the number of bigram probabilities are the square of the number of tags, too large a tag set would render the training corpus (Brown Corpus) inadequate. Therefore, some tags in Brown Corpus with similar syntactic functions and similar bigram probabilities are merged. For example, RBR, RBT, RN, QLP are conflated to RB (adverb).

2. Manual modifications of bigram probabilities

   Morphological inflections can hamper the statistical homogeneity of those tags categorized solely by syntactic functions. For concreteness, consider the following data from Brown Corpus,

$$\begin{aligned} P(\mathrm{VBG}|\mathrm{BEZ}) &= 0.040 \\ P(\mathrm{JJ}|\mathrm{BEZ}) &= 0.133 \end{aligned}$$

and

$$\begin{aligned} P(\ \mathrm{VBG}|\ \text{`threatening'}\ ) &= 0.65 \\ P(\ \mathrm{JJ}|\ \text{`threatening'}\ ) &= 0.19 \\ P(\ \mathrm{NN}|\ \text{`threatening'}\ ) &= 0.16 \end{aligned}$$

According to the above data, the product of lexical and bigram probabilities will unfairly favor JJ as the part of speech of the word 'threatening' in the sequence 'is threatening' regardless the strong suggestion of present continuous tense.

In fact, according to Brown Corpus, under the condition that the second word ends with *-ing*,

$$\begin{aligned} P(\mathrm{VBG}|\mathrm{BEZ}) &= 0.37 \\ P(\mathrm{JJ}|\mathrm{BEZ}) &= 0.084 \end{aligned}$$

which are dramatically different from the unconditional ones.

There are several ways to deal with this problem within the framework of n-gram modele:

- Introduce additional tags associated with morphological inflections, e.g., JJ_ING for adjective with ending 'ing', NN_ING for noun ending with 'ing', etc. The bigram data thus obtained will reflect the impact of morphologic inflections. But the large number of additional tags is undesirable for the reason mentioned in the previous section.
- Change the score function dynamically in actual tagging. But this complicates and slows down the tagging process.
- Our solution is to change the original bigram probabilities appropriately. For example, the possible parts of speech for a *v-ing* formed word are JJ, NN, and VBG. Increasing $P(\mathrm{VBG}|\mathrm{BEZ})$ without changing $P(t|\mathrm{BEZ})$, (t is any tag other than VBG), will achieve the same purpose to favor VBG when *v-ing* is after 'is' . Cases such as *has (had, have)* + *v-ed* , *are (be, was, etc.)* + *v-ing* , and *are (be, was, etc.)* + *v-ed* are dealt with similarly. Note we only care about the relative magnitudes of the bigram probabilities.

3. Inconsistency handling

Our basic tagging algorithm above mentioned fails in either of the following cases:

- If a word does not appear in the lexicon.
- For two consecutive words, all appropriate bigram probabilities for the possible parts of speeches of the two words are zero.

In either case, scores of all assignments to the sentence are zero, and the algorithm fails to make a choice. Such cases are not very rare. Jtag has 80 tags, i.e. the number of bigram probabilities are $80 \times 80 = 6400$. According to Brown Corpus, only 3550 of them have non zero estimated values. In actual tagging of a text of 100000 words from Wall Street Journal 1987, on average the lexicon lookup fails once in every thirteen words.

More training data will mitigate such problems, but not solve them. We need some mechanisms to solve them.

In case of lexicon failures, we may use the following strategies to make a reasonable guess on the parts of speeches for the word:

- Assume all parts of speech (80 of them) are equally possible. The tagging algorithm will use bigram probabilities to make the most likely choice.

  One problem with this strategy is that it results in much slower speed. Another problem is its overall low effectiveness due to the grammatical flexibility of English (and possibly natural language in general). We can quantify this flexibility by measuring the entropies associated with the bigram probabilities. For example, the entropy of the possible part of speech of a word immediately after an article is (according to Brown Corpus):

  $$H = -\sum_{i=1}^{80} (P(T_i|\mathrm{AT}) \times log(P(T_i|\mathrm{AT}))) = 1.474$$

  An entropy of 1.47 means a very large uncertainty (has the same uncertainty of $10^{1.47} \doteq 30$ equally possible outcomes).

  Our experiment using this strategy bore this out.

  However, in case of some patterns, (e.g., the word is just before a verb and just after an article), this strategy can be very effective. Our future plan is to use these 'high-resolution' patterns to dynamically increment the lexicon by 'discovering' new words and new parts of speech for existent words. The conjectures can be used and discarded or added to the lexicon for future use.
- Use morphologic clues. Our experiment shows this is a very effective method. Except in proper noun processing, which we will discuss later, Jtag relies solely on morphology in case of lexicon lookup failures.

The most important clue is the suffix. We collected over 100 common English suffixes and their associated parts of speech. If a suffix is associated with more than one part of speech, we assign probabilities at our discretion. For example, *-ly* → *RB* 0.7, *-ly* → *JJ* 0.3. In case of multiple suffixes for a word, the longest one is used. Another important morphological clue is hyphenation. Part of speech of a hyphenated word is guessed based on parts of speech and other properties of its components. Examples are '3-year' (*number-single noun* → *JJ*), and 'law-abiding' (*noun-ving* → *JJ*).

In cases where all bigram probabilities for two consecutive words are zero, the default rule is to use the part of speech of maximum lexical probability. This rule can be nicely incorporated in the framework of bigram model by slightly changing the score function. We redefine the score function as:

$$f(I) = \prod_{i=1}^{l} P(t_i|w_i) \times \prod_{i=1}^{l-1}(P(t_{i+1}|t_i) + \delta)$$

where δ is a very small number. The effect of this is that when bigram probabilities are very small, lexical probabilities will dominate the score function. Since the training collection is limited, our estimation method is more prone to error in case of small probabilities. This means the above method has the additional advantage of reducing 'brittleness' of bigram probabilities. Experiments show a δ of 0.005 gives good results.

4. Proper nouns

Jtag is able to recognize classes of proper nouns for our specific application. They are title, first name, last name, country and state. Unlike other words, proper nouns need special processing.

- Countries and states. Many countries and states are of multiple words. Instead of looking up them in the lexicon, we rely on a lexical analyzer (flex program) to recognize them. To minimize the code of the lexical analyzer, countries and states of single word length are stored in the lexicon.
- Person names. Common titles, e.g., 'Mr.', 'Miss' and 'President' are stored in the lexicon. Also included in the lexicon are a list of candidate first names and a list of candidate last names.

  Words like 'John', 'Jack' and 'Smith' in our lists of candidate person names are almost always person names. But candidate person names like 'East', 'Light', and 'Bird', are more ambiguous. Furthermore, if 'Mr. XYZ' appears in the text, XYZ should be tagged as person name, even though it is not in our person name lists.

Our general strategy is to recognize sequences of capitalized words, and make the decision based on the typical English name patterns and the lexical lookup of the separate words. For example, 'President John F. Kennedy' is tagged as 'President/TT John/FN F./MN Kennedy/LN', even though 'F.' is not listed as a person name.

5. Abnormal text types, tokenization

   Normally, English has conventions for using capitalized words in the beginning of sentences, titles or headlines of text, or proper nouns. But some texts contain long strings of capitalized text which fall outside the rules. The NPL collection is even more 'abnormal', since it contains only solid capitalized text. Without special information, the tagger will make many errors in tagging such abnormal text types. To prevent this, a program preprocesses the input to identify abnormal capitalized text types and pass the information on to the tagger. In processing such text types, the tagger will use a special lexicon lookup routine and proper noun recognization heuristics instead of the standard ones.

   Good performance requires a good tokenizer to correctly separate input into a stream of words. English tokenization is rather hard. Contractions, punctuations, special characters in proper nouns, etc, must taken into considerations.

5 Comparison with related work

Jtag is inspired by Church's work, and it is similar to his *parts* program in many ways.

Parts uses trigram probabilities (the probability of observing part of speech X given the next two parts of speech Y and Z) instead of bigram ones. Presumably, it uses more context than Jtag, but this is offset by the amount of training material currently available since trigram model requires much more contextual probabilities (the cube of the number of tags. If the number of tags is 80, then the number of trigram probabilities would be 512,000. The Brown Corpus, however, contains only about one million words). Extensive comparison between Jtag and *parts* shows the same level of accuracy.

Church's tagging algorithm takes more time due to the trigram model used. Assume the average number of parts of speech of an English word is r. A rough estimation of his algorithm is $O(r^4 \times l)$ time to tag a sentence of l words while our algorithm takes $O(r^2 \times l)$ time to do the same job. Actual tagging of a large text (Wall Street Journal 1987) on a SUN 4 work station by both programs show that *parts takes* 98 seconds per million words, while Jtag takes only 36.5 seconds per million words.

6 Aknowledgement

Jtag is developed on the basis of a tagger implemented by David D. Lewis and Michelle Lamar. Our tagging algorithm is very similar to the one they used.

Bob Krovetz, Jeff Jing and Tom Kalt give much help in the form of valuable proposals and feedback.

7 APPENDIX

We list the tagging of a sample text (some Tipster topics) by both our Jtag and Church's *parts* program. Tagging errors are indicated by '**'.

Output from Jtag

```
Document/NN will/MD provide/VB information/NN on/IN
the/AT proposed/VBN configuration/NN ,/, components/NNS
,/, and/CC technology/NN of/IN the/AT USA/NP 's/$ ``/OTHERPUNC
star/NN wars/NNS ''/OTHERPUNC anti-missile/JJ defense/NN
system/NN ./. Document/NN will/MD report/VB on/IN laser/NN
research/NN related/VBN ,/, or/CC potentially/RB related/VBN
,/, to/TOIN the/AT USA/NP 's/$ Strategic/NP Defense/NP
Initiative/NP ./. Document/NN will/MD report/VB those/DTS
proposed/VBN or/CC enacted/VBN changes/NNS to/TOIN
USA/NP federal/JJ ,/, state/NN ,/, or/CC local/JJ welfare/NN
laws/NNS and/CC regulations/NNS which/WDT are/BER propounded/VBN
as/CS reforms/NNS ./. Document/NN will/MD enumerate/VB
provisions/NNS of/IN the/AT USA/NP ./. Catastrophic/JJ
Health/NP Insurance/NP Act/NN of/IN 1988/CD ,/, or/CC
the/AT political/JJ //OTHERPUNC legal/JJ fallout/NN
from/IN that/DT legislation/NN ./. Document/NN will/NN**
state/NN** reasons/NNS why/WRB USA/NP stock/NN markets/NNS
crashed/VBD on/IN 19/CD October/NP 1987/CD Document/NP
will/MD report/VB proposed/VBN or/CC enacted/VBN changes/NNS
to/TOIN USA/NP laws/NNS and/CC regulations/NNS designed/VBN
to/TO prevent/VB insider/NN** trading/VBG** ./. Document/NN
will/MD inform/VB on/IN Japan/NP 's/$ regulation/NN
of/IN insider/NN** trading/VBG** ./. Document/NN will/MD
report/VB on/IN Japanese/JJ policies/NNS or/CC practices/NNS
which/WDT help/VB protect/VB Japan/NP 's/$ domestic/JJ
market/NN from/IN foreign/JJ competition/NN ./. Document/NN
must/MD refer/VB to/TOIN one/CD of/IN the/AT following/NN
:/: OTC/NP Ltd./NP ,/, Hi/NP Tech/NP Enterprises/NP
,/, Minnesota/NP Mining/NP and/CC Manufacturing/NP
,/, Integrated/NP Solutions/NP Inc./NP ,/, MIPS/NP
Computer/NP Systems/NP Inc./NP ,/, or/CC Ask/NP Computer/NP
```

```
Systems/NP Inc./NP ./. Document/NN will/MD discuss/VB
efforts/NNS by/IN the/AT black/JJ majority/NN in/IN
South-Africa/NP to/TO overthrow/VB domination/NN by/IN
the/AT white/JJ minority/NN government/NN ./. Document/NN
will/MD discuss/VB efforts/NNS by/IN the/AT UN/NP or/CC
those/DTS nations/NNS currently/RB possessing/VBG nuclear/JJ
weapons/NNS to/TO control/VB the/AT proliferation/NN
of/IN nuclear/JJ weapons/NNS capabilities/NNS to/TOIN
the/AT non-nuclear/JJ weapons/NNS states/NNS ./. Document/NN
will/MD provide/VB financial/JJ data/NNS relative/JJ
to/TOIN answering/VBG the/AT question/NN ,/, how/WRB
much/AP money/NN worldwide/JJ is/BEZ being/BEG invested/VBN
in/IN the/AT biotechnology/NN arena/NN ?/? Document/NN
will/MD report/VB on/IN non-traditional/JJ applications/NNS
of/IN space/NN satellite/NN technology/NN ./. Document/NN
will/MD provide/VB data/NNS on/IN launches/NNS worldwide/JJ
of/IN non-commercial/JJ space/NN satellites/NNS ./.
Document/NN will/MD report/VB specific/JJ consequence/NN
(/( s/NN** )/) of/IN the/AT USA/NP ’s/$ Immigration/NP
Reform/NP and/CC Control/NP Act/NN of/IN 1986/CD ./.
Document/NN will/MD identify/VB a/AT generic/JJ drug/NN
which/WDT can/MD be/BE substituted/VBN for/IN a/AT
brand/NN name/NN drug/NN in/IN the/AT treatment/NN
of/IN at/IN least/AP one/CD medical/JJ condition/NN
./. Document/NN will/MD provide/VB at/IN least/AP one/CD
datum/NN which/WDT helps/VBZ build/VB a/AT description/NN
,/, quantification/NN ,/, and/CC evaluation/NN of/IN
the/AT capacity/NN of/IN the/AT USA/NP cellular/JJ
telephone/NN network/NN ./. Document/NN will/MD provide/VB
background/NN information/NN on/IN international/JJ
terrorist/JJ groups/NNS or/CC individuals/NNS ,/, or/CC
detail/NN** the/AT activities/NNS of/IN such/JJ groups/NNS
or/CC individuals/NNS ./. Document/NN will/MD report/VB
activities/NNS by/IN established/VBN political/JJ authorities/NNS
against/IN international/JJ terrorists/NNS ./. Document/NN
will/MD report/VB on/IN actual/JJ or/CC alleged/JJ
private/JJ sector/NN economic/JJ consequences/NNS of/IN
international/JJ terrorism/NN ./. Document/NN will/MD
discuss/VB the/AT life/NN and/CC death/NN of/IN a/AT
prominent/JJ USA/NP person/NN from/IN a/AT specific/JJ
form/NN of/IN cancer/NN ./.
```

Output from *parts*

```
Document/NP/NP will/MD provide/VB information/NN on/ONIN
the/AT proposed/VBN configuration/NN ,/, components/NNS
,/, and/CC technology/NN of/IN the/AT U.S./NP/NP 's/$
"/`` star/NN wars/NNS "/`` anti-missile/JJ defense/NN
system/NN ./. Document/NP/NP will/MD report/VB on/ONIN
laser/NN research/NN related/VBN ,/, or/CC potentially/RB
related/VBN ,/, to/TOIN the/AT U.S./NP/NP 's/$ Strategic/NP/NP
Defense/NP/NP Initiative/NP/NP ./. Document/NP/NP will/MD
report/VB those/DTS proposed/VBN or/CC enacted/VBN
changes/NNS to/TOIN U.S/NP./NP federal/JJ ,/, state/NN
,/, or/CC local/JJ welfare/NN laws/NNS and/CC regulations/NNS
which/WDT are/BER propounded/VBN as/CS reforms/NNS
./. Document/NP/NP will/MD enumerate/VB provisions/NNS
of/IN the/AT U.S/NP./NP Catastrophic/NP/NP Health/NP/NP
Insurance/NP/NP Act/NP/NP of/IN 1988/CD ,/, or/CC the/AT
political/legal/JJ fallout/NN from/IN that/DT legislation/NN
./. Document/NP/NP will/MD state/VB reasons/NNS why/WRB
U.S/NP./NP stock/NN markets/NNS crashed/VBN on/ONIN
19/CD October/NP 1987/CD Document/NP/NP will/MD report/VB
proposed/VBN or/CC enacted/VBN changes/NNS to/TOIN
U.S/NP./NP laws/NNS and/CC regulations/NNS designed/VBN
to/TO prevent/VB insider/NN** trading/VBG** ./. Document/NP/NP
will/MD inform/VB on/ONIN Japan/NP/NP 's/$ regulation/NN
of/IN insider/NN** trading/VBG** ./. Document/NP/NP
will/MD report/VB on/ONIN Japanese/NP/NP policies/NNS
or/CC practices/NNS which/WDT help/VB protect/VB Japan/NP/NP
's/$ domestic/JJ market/NN from/IN foreign/JJ competition/NN
./. Document/NP/NP must/MD refer/VB to/TOIN one/PN
of/IN the/AT following/NN** :/: OTC/NP/NP Ltd/NP./NP
,/, Hi/NP/NP Tech/NP/NP Enterprises/NP/NP ,/, Minnesota/NP/NP
Mining/NP/NP and/CC Manufacturing/NP/NP ,/, Integrated/NP/NP
Solutions/NP/NP Inc/NP./NP ,/, MIPS/NP/NP Computer/NP/NP
Systems/NP/NP Inc/NP./NP ,/, or/CC Ask/VB** Computer/NP/NP
Systems/NP/NP Inc/NP/NP ./. Document/NP/NP will/MD
discuss/VB efforts/NNS by/IN the/AT black/JJ majority/NN
in/ININ South/NP/NP Africa/NP/NP to/TO overthrow/VB
domination/NN by/IN the/AT white/JJ minority/NN government/NN
./. Document/NP/NP will/MD discuss/VB efforts/NNS by/IN
the/AT United/NP/NP Nations/NP/NP or/CC those/DTS nations/NNS
currently/RB possessing/VBG nuclear/JJ weapons/NNS
to/TO control/VB the/AT proliferation/NN of/IN nuclear/JJ
weapons/NNS capabilities/NNS to/TOIN the/AT non-nuclear/JJ
weapons/NNS states/NNS ./. Document/NP/NP will/MD provide/VB
financial/JJ data/NNS relative/JJ to/TOIN answering/VBG
```

```
the/AT question/NN ,/, how/WRB much/JJ money/NN worldwide/JJ**
is/BEZ being/VBG invested/VBN in/ININ the/AT biotechnology/NN
arena/NN ?/. Document/NP/NP will/MD report/VB on/ONIN
non-traditional/JJ applications/NNS of/IN space/NN
satellite/NN technology/NN ./. Document/NP/NP will/MD
provide/VB data/NNS on/ONIN launches/NNS worldwide/JJ**
of/IN non-commercial/JJ space/NN satellites/NNS ./.
Document/NP/NP will/MD report/VB specific/NN consequence/NN
(/( s/NN** )/) of/IN the/AT U.S./NP/NP 's/$ Immigration/NP/NP
Reform/NP/NP and/CC Control/NP/NP Act/NP/NP of/IN 1986/CD
./. Document/NP/NP will/MD identify/VB a/AT generic/JJ
drug/NN which/WDT can/MD be/BE substituted/VBN for/FORIN
a/AT brand/NN name/NN drug/NN in/ININ the/AT treatment/NN
of/IN at/IN least/JJ one/CD medical/JJ condition/NN
./. Document/NP/NP will/MD provide/VB at/IN least/JJ
one/CD datum/NN which/WDT helps/VBZ build/VB a/AT description/NN
,/, quantification/NN ,/, and/CC evaluation/NN of/IN
the/AT capacity/NN of/IN the/AT U.S/NP./NP cellular/JJ
telephone/NN network/NN ./. Document/NP/NP will/MD
provide/VB background/NN information/NN on/ONIN international/JJ
terrorist/JJ groups/NNS or/CC individuals/NNS ,/, or/CC
detail/VB the/AT activities/NNS of/IN such/JJ groups/NNS
or/CC individuals/NNS ./. Document/NP/NP will/MD report/VB
activities/NNS by/IN established/VBN political/JJ authorities/NNS
against/IN international/JJ terrorists/NNS ./. Document/NP/NP
will/MD report/VB on/ONIN actual/JJ or/CC alleged/VBN
private/JJ sector/NN economic/JJ consequences/NNS of/IN
international/JJ terrorism/NN ./. Document/NP/NP will/MD
discuss/VB the/AT life/NN and/CC death/NN of/IN a/AT
prominent/JJ U.S/NP./NP person/NN from/IN a/AT specific/NN**
form/NN of/IN cancer/NN ./.
```

References

Kenneth Church, “A Stochastic Part of Speech Tagger for English”, Proceedings of Second Conference on Applied Natural Language Processing, 1988.

Mitch Marcus, “A Theory of Syntactic Recognition for Natural Language”, MIT Press, 1980.

W. Francis and H. Kucera, “Frequency Analysis of English Usage”, Houghton Mifflin Comapny, Boston, 1982.