Statistical and Learning Methods in Natural Language Processing

Ido Dagan
dagan@macs.biu.ac.il

Shuly Wintner
shuly@cs.haifa.ac.il

Spring 2004

Hebrew part-of-speech tagging

Problem: Given a text in Hebrew in which all words are morphologically analyzed, choose the correct analysis/POS for each word.

Hebrew part-of-speech tagging

Example text:

```
&$RWT AN$IM MGI&IM MTAILND LI$RAL K$HM NR$MIM KMTNDBIM ,
AK LM&$H M$M$IM  &WBDIM $KIRIM ZWLIM .
TWP&H ZW HTBRRH ATMWL BWW&DT H&BWDH WHRWWXH $L HKNST ,
$DNH BNW$A H&SQT &WBDIM ZRIM .
```


Hebrew part-of-speech tagging

Example analysis:

```
2
&$R*M**LZRF*** #&$RWT (&$R, MSPR LA~MIWD& ZKR RBIM NPRD )
&$R*M**LZRS*** #&$RWT- (&$R, MSPR LA~MIWD& ZKR RBIM NSMK )
2
AI$*E**LZRF*** AI$-IM (AI$, &CM LA~MIWD& ZKR RBIM NPRD )
HN$IM*P*BY1T ANI-AN$IM (HN$IM, PW&L &TID ZW"N IXID MDBR )
4
LM&$H*t* LM&$Ht (LM&$H, TWAR~PW&L )
M&$*E*1LZYFNY3 L-H-M&$-$LH (M&$, &CM LA~MIWD& ZKR IXID NPRD  (SIOMT:
M&$H*E*1KZYF*** LH-M&$H (M&$H, &CM MIWD& ZKR IXID NPRD )
M&$H*E*1LZYF*** L-M&$H (M&$H, &CM LA~MIWD& ZKR IXID NPRD )
```

Hebrew part-of-speech tagging

Example analysis:

```
3
M$M$*E**LZRF*** M$M$-IM (M$M$, &CM LA~MIWD& ZKR RBIM NPRD )
$IM$*P*ZRAH HM-&K$W-M$M$IM ($IM$, PW&L HWWH ZKR RBIM  )
$M$*E*mLZRF*** M-$M$-IM ($M$, &CM LA~MIWD& ZKR RBIM NPRD )
2
&WBD*E**LZRF*** &WBD-IM (&WBD, &CM LA~MIWD& ZKR RBIM NPRD )
&BD*P*ZRAH HM-&K$W-&WBDIM (&BD, PW&L HWWH ZKR RBIM  )
```


The challenge

- Segmentation: a single token can actually be a sequence of more than one POS:

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td></tr>
<tr><td colspan="3">b$wrh</td></tr>
<tr><td>b</td><td colspan="2">$wrh</td></tr>
<tr><td>b</td><td>h</td><td>$wrh</td></tr>
<tr><td>b</td><td>$wr</td><td>h</td></tr>
</table>

The challenge

- High degree of ambiguity due to the rich morphology and the problems of the orthography. In a particular corpus of 40,000 word tokens, the average number of analyses per word token was found to be 2.1, while 55% of the tokens were ambiguous.
  - In many cases two or more alternative analyses share the same POS.
  - There are cases in which two or more analyses are completely identical, except for their lexeme: xlw.
- Anchors, which are often function words, are almost always morphological ambiguous in Hebrew: $lw, at. Many of them are prefix particles: h,w
- Word order is relatively free.

Existing approaches

- Levinger, Moshe, Uzzi Ornan and Alon Itai (1995). "Learning Morph-Lexical Probabilities from an Untagged Corpus with an Application to Hebrew." *Computational Linguistics* 21(3), 383-404.
- Carmel, David and Yoelle Maarek (1999). "Morphological disambiguation for Hebrew search systems." *Proceedings of the 4th international workshop, NGITS-99*, number 1649 in Lecture notes in computer science, Springer. Pages 312-325.
- Segal, Erel (1999). "Hebrew Morphological Analyzer for Hebrew undotted texts." Masters thesis, Technion.
- Adler, Meni (2001). "Hidden Markov Model for Hebrew part-of-speech tagging." Masters thesis, Ben Gurion University.

Learning morpho-lexical probabilities

Levinger, Moshe, Uzzi Ornan and Alon Itai (1995). "Learning Morph-Lexical Probabilities from an Untagged Corpus with an Application to Hebrew." *Computational Linguistics* 21(3), 383-404.

Learning morpho-lexical probabilities

Given a text T with n words $w_1, \ldots, w_n$, for each morphologically ambiguous word w_i whose analyses are $A_1, \ldots, A_k$ there is one analysis, $A_r \in \{A_1, \ldots A_k\}$, which is the correct analysis in context.

Morphological disambiguation is the task of selecting, for each ambiguous word in a text, its correct analysis.

The morpho-lexical probability of an analysis A_i for some word w is the estimate of the conditional probability $P(A_i|w)$ from a given corpus:

$$P(A_i|w) = \frac{\textit{number of times } A_i \textit{ is the correct analysis of } w}{\textit{number of occurrences of } w}$$

Note that this probability is independent of context.

Learning morpho-lexical probabilities

Conjecture:

> In many cases a native speaker of Hebrew can accurately "guess" the right analysis of a word, without even being exposed to the concrete context in which it appears.

Strategy:

> For each ambiguous word, find the morpho-lexical probabilities of each possible analysis. If any of these analyses is significantly more frequent than the others, select it.

Of course, the probabilities can be used as part of a more elaborate tagging scheme.

Learning morpho-lexical probabilities

The problem: in order to estimate morpho-lexical probabilities, a tagged corpus is needed. How can these probabilities be estimated from an untagged corpus?

Main idea: to estimate the probabilities, use not only the ambiguous word itself, but also all the members of the set of its similar words.

A similar word for some word w is another word form sharing the same lexical entry as w, but differing in at least one morphological feature. The rules for defining the set of similar words for each word w are pre-defined and are manually constructed.

Using similar words

Consider the ambiguous word hqph. Its three analyses and each analysis' similar words are:

- hqph "round"
  $sw_1 = \{$hhqph$\}$
- h+qph "the coffee"
  $sw_2 = \{$qph$\}$
- hqp+h "her perimeter"
  $sw_3 = \{$hqp+w, hqp+m, hqp+n$\}$

Using similar words

Here, the set of similar words of a definite noun is assumed to include its indefinite counterpart, and vice versa; and the similar words of a noun with a possessive suffix include other inflections of the same noun with different possessive suffixes, in the same person but different numbers and genders.

In practice, rules can be as specific as those; or as general as the following:

> The set of similar words of some word w is the set of all words whose lexical entry is the same as the lexical entry of w.

Using similar words

Example: suppose that the word hqph occurs 200 times in the corpus. Its similar words distribution can be:

- $sw_1 : \{\text{hhqph} = 18\}$
- $sw_2 : \{\text{qph} = 180\}$
- $sw_3 : \{\text{hqpw} = 2, \text{hqpm} = 2, \text{hqpn} = 2\}$

For this example, we would want to assign the following probabilities to each analyses: 0.09, 0.90 and 0.01, respectively.

Using similar words

Two problems with this approach:

- The set of similar words might be empty (at)
- One word may occur in more than one set of similar words (spri)

Using similar words

This calls for the following representation of similar words:

- sw_1 : $\{$hqph $= 200,$ hhqph $= 18\}$
- sw_2 : $\{$hqph $= 200,$ qph $= 180\}$
- sw_3 : $\{$hqph $= 200,$ hqpw $= 2,$ hqpm $= 2,$ hqpn $= 2\}$

Note that the ambiguous word form is considered an element of the set of similar words!

The algorithm

- Do until tired:

1. Assume that the proportions of each of the different analyses are equal.
2. For each analysis, compute its average number of occurrences by summing all the counters for each word in the set of similar words, and dividing by the size of this set. The ambiguous word is included in every set of similar words.
3. If a word occurs in more than one set, its contribution to each set is determined by the proportion of each set, as was determined in the previous iteration.
4. Compute the new proportions of the sets as the proportions of the average number of occurrences of each analysis.

- Normalize the proportions to obtain probabilities.

The algorithm

Input:

w — A word with k analyses, $A_1, \ldots, A_k$.

$sw_1, \ldots, sw_k$ — The sets of similar words of $A_i, \ldots, A_k$.

$C(sw)$ — The number of times each sw, a member of some sw_i set, occurs in the corpus.

$Inc(sw)$ — A set of indices representing the analyses which sw is a similar word of.

ϵ — A threshold determining the convergence of the algorithm.

The algorithm

Internal variables:

P^i_j — The approximated morpho-lexical probability of A_j after iteration i.

$SumAnal_j$ — The sum over the contribution of all the words in sw_j.

$AvgAnal_j$ — The average contribution of a single word in sw_j to $SumAnal_j$.

The algorithm

$P_1^0 = P_2^0 = \cdots = P_k^0 = \frac{1}{k}$
$i = 0$
repeat
  $i = i + 1$
  for j between 1 and k do
    $SumAnal_j = \Sigma_{sw \in sw_j} C(sw) \times \frac{P_j^{i-1}}{\Sigma_{l \in Inc(sw)} P_l^{i-1}}$
    $AvgAnal_j = \frac{SumAnal_j}{|sw_j|}$
  for j between 1 and k do
    $P_j^i = \frac{AvgAnal_j}{\Sigma_{l=1,\ldots,k} AvgAnal_l}$
until $(max_j |P_j^i - P_j^{i-1}| < \epsilon)$

Learning morpho-lexical probabilities: problems

- In any give set of similar words, some of the words might themselves be ambiguous, and their counters might reflect the wrong analyses.
- In some cases two sets of similar words, corresponding to two different analyses, are identical (spr, m, sbl). Two such analyses cannot be disambiguated.

Learning morpho-lexical probabilities: results

Word	Approximated prob.	Corpus prob.
awlm	0.968	0.983
	0.032	0.017
at	0.995	1.000
	0.001	0.000
	0.004	0.000
xwd$	0.976	0.962
	0.024	0.038
lpni	0.725	1.000
	0.274	0.000
	0.001	0.000
alh	0.141	0.667
	0.005	0.000
	0.001	0.000
	0.849	0.333
	0.001	0.000

Morphological disambiguation for Hebrew search systems

Carmel, David and Yoelle Maarek (1999). "Morphological disambiguation for Hebrew search systems." *Proceedings of the 4th international workshop, NGITS-99*, number 1649 in Lecture notes in computer science, Springer. Pages 312-325.

Objective: reducing the degree of morphological ambiguity using statistical data automatically derived from large Hebrew corpora, in order to improve the recall of Hebrew search engines.

Hemed

Hemed is a disambiguator which receives the output of a Hebrew morphological analyszer and prunes the candidate analyses, reducing their number.

Main idea: instead of dealing with words, deal with morphological patterns as the basic elements for disambiguation. Pruning is done by evaluating the likelihood of each analysis pattern, using statistical data which reflect the relative frequency of the morphological patterns in a typical Hebrew text.

Statistical data are collected from a large non-annotated Hebrew corpus, using only unambiguous words.

The number of retained valid analyses can be controlled via a threshold parameter, so the precision/recall tradeoff can be controlled.

Morphological patterns

A morphological pattern is defined according to the information returned by the morphological analyzer. Assumed output:

Feature	Size	Values
POS	12	Noun, Verb, Adj, Numeral, Prep, Pron, Que, Conj, Particle, Adv, Abbrev, PropN
Prefix	7	m, $, h, w, k, l, b (only last one)
Number	2	sg, pl
Gender	3	m, f, m/f
Person	4	1, 2, 3, all
Tense	5	Past, Present, Future, Imperative, Infinitive
Binyan	7	1, ... 7
Status	2	absolute, construct
Suffix	$2\times 3\times 4$	Number $\times$ Gender $\times$ Person

Morphological patterns

For non-verbs, the pattern consists of:

$\langle$POS, prefix, number, gender, person, status, suffix $\langle$num, gen, pers$\rangle\rangle$

For verbs, tense and binyan replace the status feature in the pattern.

Morphological patterns

In a corpus of 10,000,000 Hebrew word tokens, 25,000 Hebrew words were observed, but only 2,300 unique morphological patterns.

Pattern statistics are therefore more reliable (do not suffer from data sparseness) and easier to maintain than word statistics.

Pattern statistics are collected from the corpus using only unambiguous words. Since 45% of the tokens are unambiguous, the sample size is approximately 4,500,000 tokens.

Alternative: count *all* patterns, including those of ambiguous ones.

Morphological disambiguation

Given a morphologically ambiguous word, compute the morphological patterns of each of its analyses and rank them by frequency.

Output only those analyses whose patterns have frequency greater than the threshold.

Hemed: evaluation

A set of 16,000 words were manually annotated. Accuracy is defined as the number of words for which the output of the system includes the correct analysis.

At a threshold of 0, accuracy is 98% (due to filtering) and the ratio of words with a single analysis is 62%. At a threshold of 0.5, accuracy is 86% (74% for ambiguous words) and 100% of the words are assigned a single analysis.

Discussion

Both systems are not addressing context-dependent morphological disambiguation. They only try to estimate the probability of each of the possible analyses of each word in the text by considering the word itself and properties of its various analyses.

Both works are unsupervised: they only consult a corpus of morphologically analyzed (but not disambiguated) texts.

Discussion

To overcome the problem of data sparseness, Levinger et al. use similar words. In particular, two words are considered similar if they share the same prefixes.

However, it is not clear why the distribution of words in a corpus should obey the rules defined by Levinger et al. In particular:

- Some inflections might be less common than others (e.g., feminine less frequent than masculine)
- The distribution of prefixes is probably independent of the word itself, especially for prefixes such as w or $.

Discussion

Possible improvements: compute morpho-lexical probabilities defining similar words as:

- words with the same lexical entry
- words with the same POS
- various combinations of morphological features

Discussion

Carmel and Maarek compute statistics using the unambiguous words in a given corpus. There is no guarantee that ambiguous words distribute similarly to unambiguous ones.

Also, in defining patterns all the morphological information is used, except the lexical entry and the prefixes prior to the first one. Different combinations of features can be more informative.

Erel Segal's disambiguator

Segal, Erel (1999). "Hebrew Morphological Analyzer for Hebrew undotted texts." Masters thesis, Technion.

This is the first work which uses contextual information for morphological disambiguation in Hebrew.

Erel Segal's disambiguator

Main idea: find the correct morphological analysis by combining probabilistic methods with syntactic analysis.

The solution consists of three consecutive stages:

- The word stage: find all possible morphological analyses of each word and approximate, for each analysis, the probability that it is the correct analysis, independently of the context.
- The pair stage: use transformation rules, which correct the analysis of a word according to its immediate neighbors.
- The sentence stage: use a rudimentary syntactical analyzer to evaluate different alternatives for the analysis of whole sentences.

The word stage

Use a variant of the similar words algorithm.

To overcome the sparseness problem, assume that the occurrences of the morphemes of a word are statistically independent and estimate the probability of each morpheme independently.

The probability of an analysis is derived by multiplying the probabilities of each of its morphemes.

The pair stage

Use transformation rules to improve the analysis.

Rules operate on pair of words (with their analyses).

Example:

> if the current analysis of w_1 is a proper-noun and the current analysis of w_2 is a noun and w_2 has an analysis as a verb that agrees with w_1 on gender and number, then add 0.5 to its morphological score, and normalize the scores.

Transformation rules are acquired automatically using an analyzed training corpus.

By the end of this stage, 93.8% of the words in the corpus are assigned their correct analysis.

The sentence stage

Syntactically parse the sentence (actually the POSs of the sentence).

Syntactic grammaticality, estimated by the syntactic parser, is used as one of two measures for the correctness of the analysis. This is combined with the score that results from the pair phase.

The syntactic parser uses a handcrafted grammar with about 150 rules, defined over approximately 10 nonterminals and 30 terminals.

Finally, the scores of the morphological phases and the syntactic phase are combined using a weighted average.

The reported performance is 96.2% accuracy. More reliable tests reveal accuracy of 85% only.

Hidden Markov Model for Hebrew part-of-speech tagging

Adler, Meni (2001). “Hidden Markov Model for Hebrew part-of-speech tagging.” Masters thesis, Ben Gurion University.

Features for morphological disambiguation

- Word-level features
- Contextual information

Alternative strategies:

- Compute morpho-lexical probabilities separately, then combine them with contextual information
- Define a single classification problem involving both types of features.

Features for morphological disambiguation

- The word form itself (but obvious problem of data sparseness)
- The lemma (citation form, or lexical entry)
- POS
- All the other features returned by the morphological analyzer
- The number of possible analyses/POSs for the word (?)
- How to deal with ambiguity?

Then, use the same features for a window of k words around the target word.