SISTEMA DE DESAMBIGUACION MORFOLOGICA PARA ORDENADORES PERSONALES

Agustin Reinaldos Meca
Ignacio Moreno-Torres Sánchez

Jean Piero Zacci
Giovanna Turrini

Depto de Lenguajes y CC
Facultad de Informatica
Universidad de Malaga
España

Giuseppe Cappelli
ILC Pisa

RESUMEN

Presentamos a continuación una herramienta cuyo objetivo, junto con otros programas que están siendo desarrollados actualmente, posibilitar la creación de corpora lematizados, una de cuyas aplicaciones será la obtención de diccionarios de frecuencia (del español e italiano).

Entendemos por corpus lematizado un texto en el que cada palabra tiene asociada una etiqueta de información lingüística en principio su lema (que suele coincidir con la raiz), y información morfológica sobre la clase de palabras, el número, género, persona...

Para obtener un corpus lematizado seguimos los siguientes pasos. En primer lugar un analizador morfológico o diccionario de formas debe asignar a cada palabra tantas etiquetas como soluciones el analizador morfológico ofrezca (e.g. a la palabra «vino» le podríamos asignar etiquetas «nombre» y «verbo»).

En segundo lugar, dada una palabra con más de una etiqueta, debemos seleccionar la correcta en cada caso. El programa que aquí presentamos es el que resuelve este problema utilizando información estocástica y reglas de contexto. La herrmienta es, en nuestra opinión, de gran valor por diversas posibilidades que ofrece tales como la modificación del conjunto de etiquetas (a partir de un conjunto predefinido), la ayuda a la creación de reglas de contexto, o la posibilidad de modificación de la fórmula de transición (a partir de la cual obtenemos las estadísticas) y de mantener diferentes estadísticas para diferentes tipos de textos.

SOFTWARE TOOLS FOR CORPORA ON PC

INTRODUCTION

At the ILC of Pisa and the University of Malaga we are preparing a set of programs which should serve to create a grammatically- tagged corpora of Spanish/Italian and a frecuency dictionary of each language. The system will work under a PC-environment and it has been divided in two independent modules: dictionary management and disambiguation. Both modules are language independent.

Our main objective has been to create an easily transportable tool that could be distributed to different research groups. For that it needed a «packed» dictionary and a language independent, theory-neutral disambiguation system.

1. THE DICTIONARY MANAGER

The morphological analyzer (M.A.) the former of two modules which act in cascade and associate each word in a text with one lemma and morphological code. Given a word, the M.A. associate it with the lemma, or lemmas, which it comes from, each with the corresponding

morphological code (or codes). Thus it does not resolve any lexical ambiguity; that is, in fact, the purpose of the disambiguating module.

M.A. is indeed a dictionary manager in that it uses or organizes one or more dictionaries, represented as lists of forms rather than by means of inflexion paradigms. Each form appears only once and is the key to achieve the associated information: lemmas and morphological codes.

Searching a word follows a pattern matching criterion, hence the accepted lexicon is constituted by the collection of words explicitely inserted, mostly, in one dictionary.

The format of the dictionary may be defined in terms of a BNF grammar as follows:

<Dict>::<Line>|<line>sep1<Dict>
<Line>::<Form>sep2<Info>
<Info>::<Lemmainfo>|<Lemmainfo>sep3<Info>
<Lemmainfo>::<Lemma>sep4<PoSInfo>sep5<MorphoInfo>
<MorphoInfo>::<MorphoCode>|<MorphoCode>sep6<MorphoInfo>

<Form>,<Lemma>,<PoSInfo>,<MorphoCode> are secuences of asciicodes that do not contain special symbols: sep1, sep2, sep3,sep4, sep5, sep6.

Here are some examples from the Italian dictionary:

a#a\E[$a\SN[NN
abate#abate\SM[MS
abbagli#abbagliare\VTIP[S2IP,S1CP,S2CP,S3CP$abbaglio\SM[MP

Actually, one may also provide the presence of use counters, which may be useful for later adaptations upon dictionary contents, with the further aim of building frecuency lexicons. The use of counters should be adjusted as a consecuence of text analysis; in this phase there also exists the possibility of finding new words and thereby increasing the dictionary.

The M.A. module includes some useful tools that allow the user to examine and/or manage both dictionaries and other kinds of data such as the lists of part of speech and morphological codes or the like. Besides it is possible to extract collection of words depending on their counter values.

Each dictionary may be compressed and indexed, to increase performance and save space (more than 50% theoretically, it might be reached space savings greater than 75%, for instance by ulteriorly coding the compressed dictionary with the Huffman method). At present, the compression method retains byte alignment (that is why Huffman method is not applied); furthermore dictionaries may be accesed and used in their compressed form. The compression technique prunes the form leading characters that match those in the preceding form, and replaces it with an integer which counts the the number of cut characters; analogously, it prunes the lemma's leading characterswith respect to the form they belong to. A translation table will store the one byte code for part of speech and morphological codes.

2. DISAMBIGUATION

We have found two different types of disambiguating systems for LN:

- rule based systems such as MORFSIN (ILC, Pisa 198?): a complex set of context rules should decide which is the correct analysis for one word.
- stocastic systems such as TAGGIT (Garside,Leech,Sampson 1987): the posibilities that a category A is followed by another category B are studied and represented in a Transition Matrix (TM). In case we have an ambiguous secuence of words:

Henry likes stews
NP NNS NNS
VBZ VBZ

we find different possibilities or secuences of categories; in this case we find the following secuences:

NP NNS NNS
NP VBZ NNS
NP NNS VBZ
NP VBZ VBZ

(NP = proper noun
NNS = singular common noun
VBZ = 3rd person of lexical verb)

If we multiply the values that we have in the TM for:

NP NNS

by the value that we have for:

NNS NNS

we get a vaue for the first secuence.

In this way we may obtain different value for each secuece the higher of which should be the one of the correct secuence.

Garside et al. (1987) describe the ways in which such idea may be improved to obtain quite good results for English real texts.

Aparently a rule based system should be easier to control, since it is the linguist who adds whatever he knows to be correct; but stocastic methodologies have, in fact, proved to work well for such problems. We feel that in both lines one might come to a robust system; but the problem is not so much with the computational methodology as it is with the correct classification of the words of the language that you wish to disambiguate.

That is a problem which we would not want to solve now; but we wanted to avoid the possibility that: in case you want to change the categories or subcategories, you have to change the whole system.

At the same time we needed a system that would transportable to another language (at least Italian, Spanish).

That led us to the idea that we needed a 'disambiguator generator'. That is, a system which would generate a different set of rules or MT for different languages (or even sublanguages). The difficulties to obtain a good set of context rules, for one language alone, made us decide in favour of a stocastic methodology. But, as we knew that statistics could not solve everyting, the possibility to create context rules was added. Moreover, the system suggests possible context rules to the linguists for those cases where statistics are not enough.

Briefly the system offers these possibilities:

1. Given a disambiguated text (it should have a minimun of 5000 words) and a set of grammatical categories it creates a MT for that set of categories. It also extracts «every

ambiguous context» and keep it in a separate ambiguous-context-file. After examining 15.000 words it should be possible for the user to create out of this file a set of context-rules.

2. The set of categories may be changed any moment. Obviously, the transition matrix should be generated again, but it is immediate if you have already tagged texts.

3. The context rules include up to 5 categories. These secuences of 'more than two' categories work like transitions with value = 0 . They can be of the following types:

1- those which say that a secuence is incorrect;

2- those which say that among two possibilities one is correct; for instance, the word «que» after a verb is 'always' a conjuntion and not a relative pronoun.

3- those wich say when there are certain types of ambiguity the system should not decide which one is correct. For instance «que» in Spanish seems in some contexts imposible to disambiguate unless you do syntactic analysis.

The '*' with its usual meaning can be used to define contexts more freely.

With these rules we expect to increase the reliability and robustness of the system; Making it give a solution when it is possible, and making it ask a question when it may not solve the problem.

4. A 'rarity marker' (as described by Garside et al.) has been absolutely necessary to correctly predict that, for instance,

era (WAS/NOM)

is almost always a form of the verb 'ser' (to BE).

Some other cases are:

pero (CONJ/NOM) but/apple tree
como (CONJ/VER) how/I eat
eras (WAS/NOM) was/age
nada (PRON/NOM) nothing/nothingness
para (PREP/VER) for,to../he stops,stop(imperative form)

5. The default transition function is multiplication, but it is possible to define another transition function using common arithmetic operators and some parameters of the system: number of times any category has appeared to obtain the TM total number of words processed to obtain the MT In this way we can avoid the 0s of rare secuences.