Generation of Named Entities

Marisa Jiménez

Microsoft Research
One Microsoft Way
Redmond, WA 98052
USA
marialj@microsoft.com

Abstract

In this paper we present an overview of an approach developed at Microsoft Research to generate strings for named entities such as places and dates. This approach uses abstract representations as input. We first provide an overview of our system to identify named entities in text. Next we present our approach to generate these entities from abstract representations, known as "logical forms" in our system. We then focus on the generation of place names in Spanish. We discuss our technique to generate Spanish place names from a logical form where language-specific features, such as word order, or capitalization conventions do not exist. We finally present the details of a study that we carried out to help us make sound linguistic decisions in the generation of place names in Spanish.

Keywords

Generation, MT, factoids, named entities, logical form

1. Introduction

Proper identification of multi-word expressions that refer to proper names, dates, company names, places, and other entities is a need for most natural language processing (NLP) applications. Information retrieval is probably one of the most popular NLP applications to make full use of name identification techniques (Mani et al, 1993; Cowie and Lehnert, 1996; Paik et al, 1993, among others).

Named entity identification is also important for Machine Translation (MT) systems. MT systems should have some mechanism for identifying person names such as *Mr. John Little.* They should also know that *Mr.* is a title word that should be translated, and that they should avoid translating *John Little* into Spanish as *Juan Pequeño* (Wacholder et al, 1997).

In this paper we first present an overview of the system developed by the NLP group at Microsoft Research to identify named entities. Our system uses what we call factoid rules to identify multi-word expressions that are productive and not in our current dictionaries. We then describe our approach to the generation of named entities. The factoid generation module makes use of an abstract representation to generate different named entities. This module is still under development. At the moment we have rules that generate dates, units, and places.

2. Identification of Named Entities in the Microsoft NLP System

The Microsoft NLP system uses rules to identify multi-word expressions that are productive and not part of our monolingual dictionaries. These expressions are known as *factoids* and the rules we used to identify them are known as *factoid rules*. Factoid rules look like simple grammar rules that apply before the grammar rules do. They are written in G, a Microsoft-internal high-level linguistic formalism used as the development tool by our group.[1] They can apply recursively, and can also be the input to other factoid rules.

Factoid rules were developed to help the analysis component of our system deal with multiple word expressions that were not defined in our dictionaries. The rules can be divided into two groups: those that identify named entities such as person names, dates, company names, place names, and so on, and those that identify other entities of a less clear semantic nature, such as English hyphenated compounds.

Factoid rules use different techniques to identify different named entities. Among these techniques, we make extensive use of a full range of features coded in the words of our monolingual dictionaries. We also use different algorithms to detect dates, proper names that are not in our dictionaries, years, phone numbers, and so on.

In figure 1 we provide an example of an English construction that contains a date factoid:

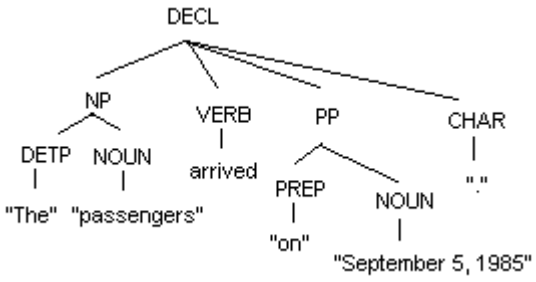

Figure 1: Example of an English sentence containing a date factoid

Factoid rules identify the components of a potential named entity and combine them into a single flat record. The different parts of a particular factoid type within that record are labeled with a unique name (see, e.g., MONTH, DATE, and YEAR in fig. 2). All languages in our system (currently, English, Spanish, French, German, Korean,

[1] See Heidorn (1972) for details about this linguistic formalism.

Japanese, and Chinese) share the same basic internal structure, no matter how different from each other they are. Another important piece of information that is encoded in the factoid record is the semantic type or class a factoid belongs to. Some examples of semantic classes are PERSON, ADDRESS, PHONE, DATE, and TIME, among others. These classes are used to identify the factoid type across different languages. Our factoid generation component also makes extensive use of these semantic class labels.

In figure 2 we provide an example of a date factoid. Inside the record, we display information about the lemma, the components, the linguistic features and the semantic class of this factoid, as well as syntactic information.

```
Lemma     "September_5_,_1985"

Bits      Pers3 Sing PrprN Factoid InitCap
          LexCap OnlyUpr Count Tme Date

Parent    PP1   "on September 5, 1985"

Factrecs  MONTH   September
          DATE    5
          CHAR    ,
          YEAR    1985

FactClass DATE
```

Figure 2: Factoid record

The English system was the first to develop factoid rules. Spanish borrowed from English those factoid rules that were applicable to Spanish. Thirty-seven new Spanish-specific factoid rules were created. For example, we created rules to gather Spanish dates, and person and place names that have the preposition *de* 'of' in the middle of the constituents, e.g. *Juan Colón de Carvajal*, *isla de Pascua* 'Easter Island', and *25 de septiembre de 1985* 'September 25, 1985'.

3. Generation of Factoids

Factoid generation has become important during the development of the Microsoft MT system, since factoids need to be generated in an appropriate language-specific form. Nevertheless, both the generation grammar and the factoid generation modules are application-independent.

The generation component of our system uses as input an abstract representation of a sentence meaning, which we called Logical Form (LF), and produces a sentence tree as output (see (Aikawa et al, 2001), for details). In MT mode, the input to generation is a transferred LF obtained from applying automatically learned transfer mappings to the LF representation of a source sentence (see (Menezes and Richardson, 2001) for details).

Factoid generation rules apply prior to the generation grammar rules, and take as input LF representations of analysis factoids. The transfer component leaves the LF of the source factoid structurally untouched, but translates each one of its parts by checking their translation in our bilingual dictionaries. The factoid generation module takes this representation as input. Using abstract representations to generate factoids allows all our languages to share the same factoid generation rules, with minor tuning for language-specific idiosyncracies.

In figure 3 we provide an example of the LF of a date factoid. This representation contains semantic information about the parts of the factoid, i.e. whether they are months, dates, years, and so on, and also the semantic class that the factoid belongs to, i.e. a date.

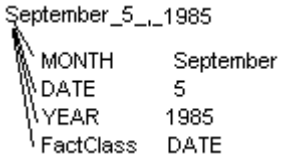

Figure 3: LF of a date factoid

In figure 4 we show an example of the transferred LF of a date factoid. This transferred LF contains the semantic class of the factoid and an unordered list of its constituents.

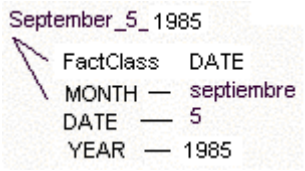

Figure 4: Transferred LF of a date factoid

Not all factoids recognized by our analysis system are translated. We would want to translate dates and place entities such as *March 23, 1976* and *Mount Rainier*, but we would not want to translate an English person name such as *John Little* as *Juan Pequeño* in Spanish. The named entities recognized by our factoid rules that should not be translated are marked with a feature to block their translation during transfer.

All seven languages in our system share the same factoid generation rules. A typical factoid generation rule has a language-independent part where the parts of the factoid are generated in an unordered fashion. There is also a language-specific part in the rule where specific conditions are defined for each language; word order within the parts of the factoid and insertion of different prepositions and other elements would be examples of these language-specific conditions.

Language-specific conditions play a key role in factoid generation rules. For example, when generating a Spanish date, features such as word order, capitalization, preposition insertion, and numbering conventions are taken into account. In Spanish, months always follow the date, and days of the week are not capitalized, which is not the case in other languages (e.g. English). Another characteristic of Spanish has to do with the insertion of the preposition *de* 'of' between dates and months, and

also between months and years. For example, *4 de abril de 1995* would be the Spanish version of *April 4, 1995*. Finally, Spanish uses roman numerals to express centuries, and the word for *century* precedes the numeral. For example, *siglo XXI* would be the Spanish equivalent of *21st century*.

These peculiarities of Spanish date expressions are taken into consideration in our factoid generation rule for dates. In the Spanish-specific part of this rule, we give the correct Spanish canonical order to the different parts of the date. We make the month to follow the date, and we insert the preposition *de* 'of' between dates, months and years. We also make sure that centuries appear in Roman numerals.

4. Generation of Place Names

4.1. Place Names in Spanish

A typical place entity consists of a place type, such as *lake* or *sea,* and a place name, such as *Ontario,* and *Bering*. Some examples of place types in Spanish are *lago* 'lake', *ciudad* 'city', and *monte* 'mount'. Place entities have three main characteristics in Spanish. First, place types do not require as strict capitalization as they do in other languages. Place names, on the other hand, always appear capitalized.

The second characteristic is that the place type always precedes the place name in Spanish (e.g. *monte Rainier, mar de Bering*). In other languages, the place type may appear before or after the place name.

The third characteristic is that Spanish often inserts the preposition *de* 'of' between a place type and a place name. There are certain place types that are more likely to be followed by *de,* such as *ciudad* 'city', and *municipio* 'municipality', while others are more likely not to be followed by *de*, such as *edificio* 'building' and *lago* 'lake'. There are also other place types that can be either followed by *de* or not. For example, *canal* 'channel' seems to appear almost equally with and without *de* in the Spanish version of the Encarta encyclopedia.

Insertion of the preposition *de* after certain place types can be a challenge when generating place entities from an abstract representation. The LF representation that we use as input to factoid generation rules contains no specific information about preposition insertion, as this is a rather language-specific issue. Furthermore, Spanish is not always clear about which place types require to be followed by *de*. Given these challenges, and in order to make a sound linguistic decision, we decided to study the frequency of *de* in real text prior to the implementation of a generation rule for dates.

4.2. Study of *De* Frequency in Encarta

We conducted a study using the Spanish version of the Encarta encyclopedia to determine the frequency of insertion of the Spanish preposition *de* after a place type. We intended to use the results of the study in the development of our factoid generation rule for places.

We took a text version of Spanish Encarta and, using tools developed in our group, extracted all sequences of place type/place name combinations in the whole encyclopedia. We also accounted for the possibility of *de* insertion between the place type and the place name. Instances of the same place type/place name combination were counted only once. We used our monolingual Spanish dictionary to identify place types by using their dictionary features. We used capitalization to identify place names appearing after the preposition *de* or a place type.

In figures 5 and 6 we provide two graphs with the frequency of appearance/non appearance, respectively, of *de* by place type. In the left axis, we measure frequency by actual number of occurrences in the text, and, in the right axis, we measure frequency by percentage.

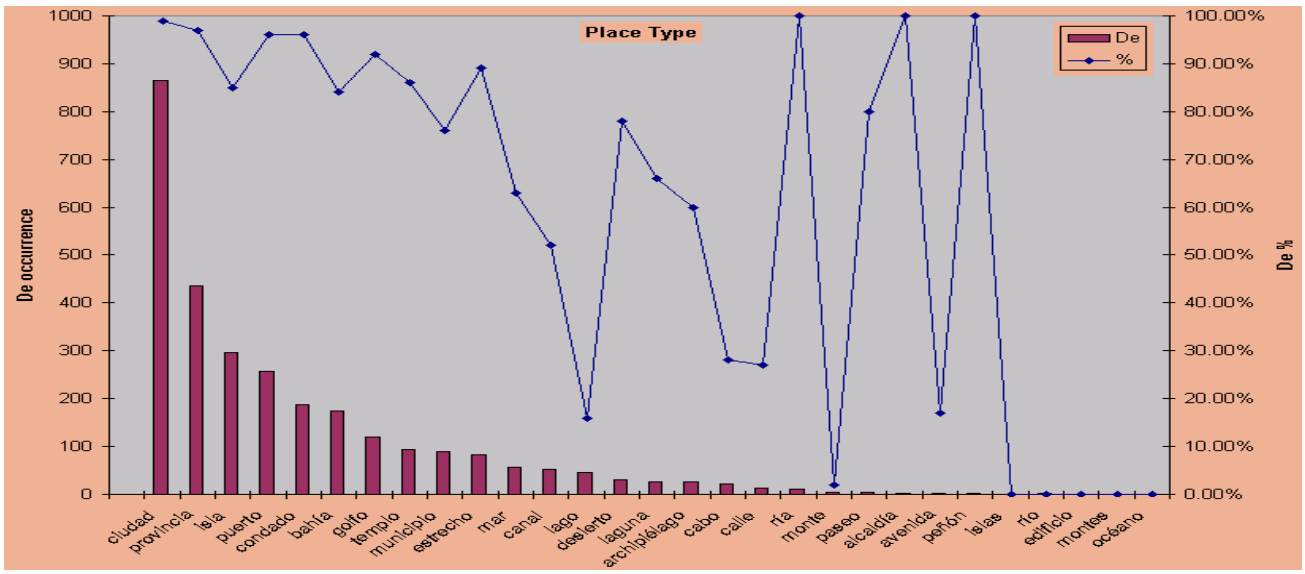

Figure 5: Frequency of *De* after place type in Spanish Encarta

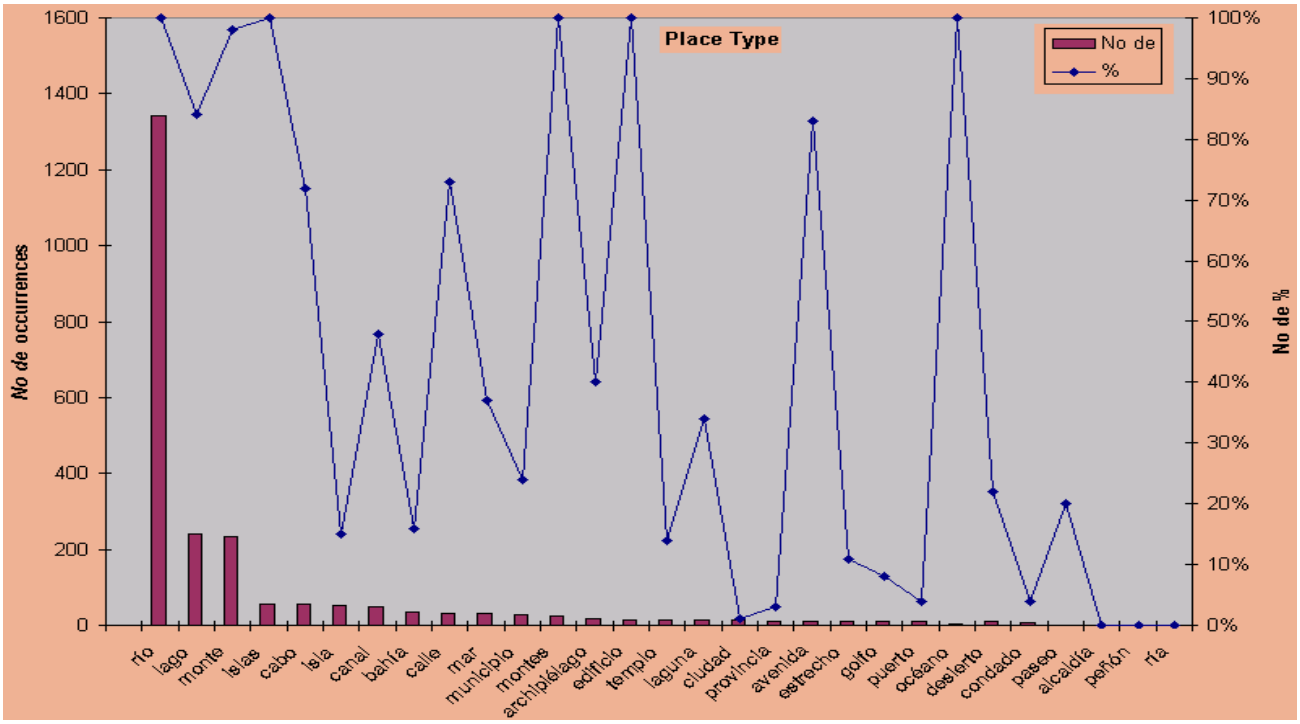

Figure 6: Frequency of *No De* after place type in Spanish Encarta

Our study showed that the place types most likely to be followed by the preposition *de* were *ciudad* 'city' (99%), *provincia* 'province' (97%), *isla* 'island' (85%), *puerto* 'port' (96%), *condado* 'county' (96%), *bahía* 'bay' (84%), and *golfo* 'golf' (92%). *De* was shown to always follow place types such as *ría* 'estuary', *alcaldía* 'jurisdiction of a mayor', and *peñón* 'rocky mountain'. Unfortunately, their occurrence in Spanish Encarta was not very frequent, so we feel that we need more data before reaching a conclusion. *Isla* 'island', and *islas* 'islands' is an interesting pair as they showed very different patterns, depending on whether they appear in singular or plural. The singular form tends to be followed by *de* 85% of the time, while the plural form appears without *de* 100% of the time. *Río* 'river' was overwhelmingly the place type most likely not to be followed by *de* (100%), together with *lago* 'lake' (85%) and *monte* 'mount' (97%). Other place types, such as *islas*, *montes*, and *edificio,* showed a strong preference for not taking *de* although they had a lesser number of actual occurrences in the text.

4.3 Generation of Place Names in Spanish

We used the results of our study as feedback to develop the Spanish-specific part of the factoid rule that generates place entities. There we provide the canonical word order for Spanish places, which is always place type followed by place name. We also insert the preposition *de* after a place type whenever necessary. We insert *de* only after those place types that are most commonly followed by *de*, as shown in our study. In those cases where we feel that there is not enough data to make a sensible decision, we choose not to insert the preposition.

We used this factoid generation rule in the translation process of the two place entities that we show in figure 7. In the first example we have the place name *Sussex* followed by *county*. The LF representation of this factoid correctly identifies *Sussex* as a place name and *county* as a place type. The Spanish translation of *Sussex County* is *el Condado de Sussex*. To obtain this translation, we gave the place type/place name canonical order to the Spanish place generated, and also inserted the preposition *de* after *condado*.

The definite article *el* was also inserted in front of the place type *condado*. Contrary to English, which does not require the obligatory use of a definite article in front of place types, Spanish does require this article, which is inflected for gender and number. We determine the appropriate gender and number of the article by checking the gender and number of the Spanish place type in our monolingual dictionary.

The second example shows the translation process of the place *Manasus River* into Spanish. Its LF representation correctly interprets *Manasus* as a place name, despite the fact that it does not appear in our English dictionary. *Manasus River* is translated as *Río Manasus*. Contrary to the first example, *de* is not inserted after the place type in this case, but we do insert the article *el*. Although the English source sentence, *We sailed the Manasus River*, does include the article *the*, we do not include this information in the LF representation that we use as input. We believe that the use of the article is another language-specific characteristic that should not be included in an abstract linguistic representation.

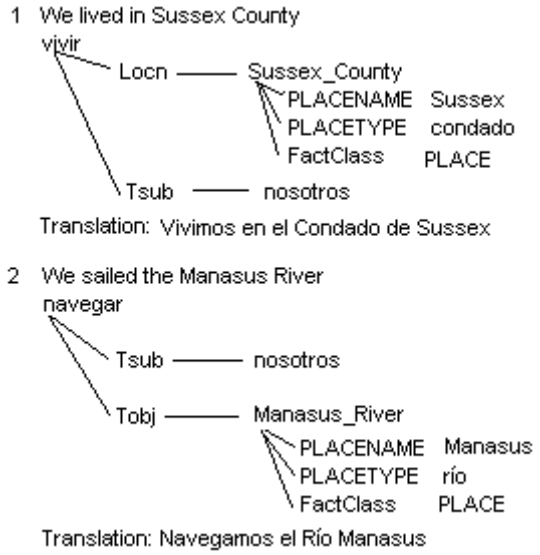

Figure 7: Generation of a Place Name

5. Conclusions and Future Work

In this paper we presented an overview of our system to identify and generate named entities such as dates, places, person names, and so on. This system uses factoid rules to identify named entities. We also described our approach to the generation of named entities. To generate these entities, the factoid generation module makes use of an abstract representation where language-specific features, such as word order, or capitalization conventions do not exist. We finally presented the results of a study we conducted to help us make sound linguistic decisions in the generation of place names in Spanish.

As part of our future work, we intend to create new factoid generation rules to generate other named entities, such as time expressions (e.g. *five o'clock*) and addresses (e.g. *425 Sunset Boulevard*). We also intend to expand our rule to generate place names to handle place entities that do not contain a place type (e.g. *East Spokane*).

Acknowledgments

We would like to acknowledge our colleagues in the NLP group at Microsoft Research for their useful comments. Special thanks go to Joseph Pentheroudakis, Deborah Coughlin, and Martine Pettenaro for their insights and help during the development of this paper.

References

Aikawa, T. et al (2001). Multilingual Natural Language Generation. Paper to be presented at MT Summit VIII, Santiago de Compostela, Spain.

Cowie, J. and Lehnert, W. (1996). Information Extraction. In Communications of the ACM, Vol.39, pp 83-92.

Heidorn, G. (1972). Natural language inputs to a simulation programming system. Ph.D. diss., Yale University (Also published as Technical Report NPS-55HD72101A. Naval Postgraduate School. Monterey, CA.)

Mani, I. T.R. Macmillan, S. Luperfoy, E.P. Lusher, and S.J. Laskowski (1993). Identifying unknown proper names in newswire text. In B. Boguraev and J. Pustejovsky, eds, Corpus Processing for Lexical Acquisition, pp.41-54, MIT Press, Cambridge, Mass.

Menezes A. and Richardson S. (2001). A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. Paper to be presented at ACL 2001, Toulouse, France.

Paik, W., E.D. Liddy, E. Yu, and M. McKenna (1993). Categorizing and standardizing proper nouns for efficient information retrieval, In B. Boguraev and J. Pustejovsky, eds, Corpus Processing for Lexical Acquisition, pp.44-54, MIT Press, Cambridge, Mass.

Wacholder, N., Y. Ravin and R.J. Byrd (1994). Retrieving information from full text using linguistic knowledge, In Proceedings of the Fifteenth National Online Meeting, pp.441-447, New York, May.