Learning to Recognize Names Across Languages

Anthony F. Gallippi
University of Southern California
University Park, EEB 234
Los Angeles, CA 90089
USA
gallippi@aludra.usc.edu

Abstract

The development of natural language processing (NLP) systems that perform machine translation (MT) and information retrieval (IR) has highlighted the need for the automatic recognition of proper names. While various name recognizers have been developed, they suffer from being too limited; some only recognize one name class, and all are language specific. This work develops an approach to multilingual name recognition that allows a system optimized for one language to be ported to another with little additional effort and resources. An initial core set of linguistic features, useful for name recognition in most languages, is identified. When porting to a new language, these features need to be converted (partly by hand, partly by on-line lists), after which point machine learning (ML) techniques build decision trees that map features to name classes. A system initially optimized for English has been successfully ported to Spanish and Japanese. Only a few days of human effort for each new language results in performance levels comparable to that of the best current English systems.

1 Introduction

Proper names represent a unique challenge for MT and IR systems. They are not found in dictionaries, are very large in number, come and go every day, and appear in many alias forms. For these reasons, list based matching schemes do not achieve desired performance levels. Hand coded heuristics can be developed to achieve high accuracy, however this approach lacks portability. Much human effort is needed to port the system to a new domain.

A desirable approach is one that maximizes *reuse* and minimizes human effort. This paper presents an approach to proper name recognition that uses machine learning and a language independent framework. Knowledge incorporated into the framework is based on a set of measurable linguistic characteristics, or *features*. Some of this knowledge is constant across languages. The rest can be generated automatically through machine learning techniques.

The problem being considered is that of segmenting natural language text into lexical units, and of tagging those units with various syntactic and semantic features. A lexical unit may be a word (e.g., "started") or a phrase (e.g., "The Washington Post"). The particular lexical units of interest here are proper names. Segmenting and tagging proper names is very important for natural language processing, particularly IR and MT.

Whether a phrase is a proper name, and what type of proper name it is (company name, location name, person name, date, other) depends on (1) the internal structure of the phrase, and (2) the surrounding context.

Internal: "Mr. Brandon"
Context: "The new company, Safetek, will make air bags."

The person title "Mr." reliably shows "Mr. Brandon" to be a person name. "Safetek" can be recognized as a company name by utilizing the preceding contextual phrase and appositive "The new company,".

The recognition task can be broken down into *delimitation* and *classification*. Delimitation is the determination of the boundaries of the proper name, while classification serves to provide a more specific category.

Original: John Smith , chairman of Safetek , announced his resignation yesterday.
Delimit: <PN> John Smith </PN> , chairman of <PN> Safetek </PN> , announced his resignation yesterday.
Classify: <person> John Smith </person> , chairman of <company> Safetek </company> , announced his resignation yesterday.

During the delimit step, the boundaries of all proper names are identified. Next, the delimited proper names are classified into more specific categories.

How can a system developed in one language be ported to another language with minimal additional effort and comparable performance results? How much additional effort will be required, and what degradation in performance, if any, is to be expected? These questions are addressed in the following sections.

2 Method

The approach taken here is to utilize a data-driven knowledge acquisition strategy based on decision trees which uses contextual information. This differs from other approaches which attempt to achieve this task by: (1) hand-coded heuristics, (2) list-based matching schemes, (3) human-generated knowledge bases, and (4) combinations thereof. Delimitation occurs through the application of phrasal templates. These templates, built by hand, use logical operators (AND, OR, etc.) to combine features strongly associated with proper names, including: proper noun, ampersand, hyphen, and comma. In addition, ambiguities with delimitation are handled by including other predictive features within the templates.

To acquire the knowledge required for classification, each word is tagged with all of its associated features. These features are obtained through automated and manual techniques. A decision tree is built (for each name class) from the initial feature set using a recursive partitioning algorithm (Quinlan, 1986; Breiman *et al.*, 1984) that uses the following function as its selection (splitting) criterion:

$$-p*\log2(p) - (1-p)*\log2(1-p) \qquad (1)$$

where p represents the proportion of names belonging to the class for which the tree is built. The feature which minimizes the weighted sum of this function across both child nodes resulting from the split is chosen. A multitree approach was chosen over learning a single tree for all name classes because it allows for the straightforward association of features within the tree with specific name classes, and facilitates troubleshooting.

The result is a hierarchical collection of co-occurring features which predict inclusion to or exclusion from a particular proper name class. Since a tree is built for each name class of interest, the trees are all applied individually, and then the results are merged.

2.1 Features

Various types of features indicate the type of name: parts of speech, designators, morphology, syntax, semantics, and more. Designators are features which alone provide strong evidence for or against a particular name type. Examples include "Co." (company), "Dr." (person), and "County" (location). For example, of all the company names in the English training text, 28% are associated with a corporate designator.

Other features are predetermined, obtained via on-line lists, or are selected automatically based on statistical measures. Parts of speech features are predetermined based on the part of speech tagger employed. On-line lists provide lists of cities, person names, nationalities, regions, etc. The initial set of lexical features is selected by choosing those that appear most frequently (above some threshold) throughout the training data, and those that appear most frequently near the positive instances in the training data.

Some features, such as morphological, keyword, and key phrase features, are determined by hand analysis of the text. Capitalization is one obvious

Table 1. Features summary.

Type	Feature	Example	How many
Part of Speech	Proper Noun	"Aristotle"	NA
	Common Noun	"philosophy"	NA
Designator	Company	"Corp.", "Ltd."	100 E, 110 S, 60 J
	Person	"Mr.", "President"	70 E, 70 S, 43 J
	Location	Country, State, City	520 E, 900 S, 570 J
	Date	Month, Day of week	56 E, 19 S, 19 J
Morphology	Capitalization	"A-", "B-"	1 E, 1 S, 0 J
	Company Suffix	"-corp", "-tec"	5 E, 0 S, 30 J
	Word Length	WL>8, WL<3	4 E, 4 S, 2 J
List	Companies	"IBM", "AT&T"	0 E, 100 S, 7K J
	Persons	"Smith", "Michael"	21K E, 21K S, 185K J
	Locations	"Gulf of Mexico"	20 E, 20 S, 2K J
	Nationalities	"Japanese"	220 E, 0 S, 0 J
	Keyword(s)	"based in", "said he"	44 E, 49 S, 54 J
Template	Company	< NNP CN_desig >	210 E, 210 S, 210 J
	Person	< P_Desig NNP >	90 E, 95 S, 90 J
	Location	< NNP L_desig >	190 E, 190 S, 190 J
	Date	< MM Num , Num >	17 E, 18 S, 70 J
	Proper Name	< NNP NNP >	140 E, 140 S, 140 J
Special Purpose	Lngst Cm Sbstr	"VW" <- Volkswagen	1 E, 1 S, 1 J
	Duplicated PNs	DUP_2+, DUP_5+	5 E, 5 S, 2 J

morphological feature of importance. Determining keyword and key phrase features amounts to selecting prudent subject categories. These categories are associated with lists of lexical items or already existing features. For example, many of the statistically derived lexical features may fall under common subject categories. The words "build", "make", "manufacture", and "produce" can be associated with the subject category "make-type verbs". Analysis of the immediate context surrounding company names may lead to the discovery of key phrases like "said it", "entered a venture", and "is located in". Table 1 shows a summary of various types of features used in system development. The *longest common substring* (LCS) feature (Jacobs *et al.*, 1993) is useful for finding proper name aliases.

2.2 Feature Trees

The ID3 algorithm (Quinlan, 1986) selects and organizes features into a discrimination tree, one tree for each type of name (person, company, etc.). The tree, once built, typically contains 100+ nodes, each one inquiring about one feature in the text, within the locality of the current proper name of interest.

An example of a tree which was generated for companies is shown in Figure 1. The context level for this example is 3, meaning that the feature in question must occur within the region starting 3 words to the left of and ending 3 words to the right of the proper name's left boundary. A "(L)" or "(R)" following the feature name indicates that the feature must occur to the left of or to the right of the proper name's left boundary respectively. The numbers directly beneath a node of the tree represent the number of negative and positive examples present from the training set. These numbers are useful for associating a confidence level with each classification. Definitions for the features in Figure 1 (and other abbreviations) can be found in the appendix.

The training set used for this example contains 1084 negative and 669 positive examples. To obtain the best initial split of the training set, the feature "CN_alias" is chosen. Recursively visiting and optimally splitting each concurrent subset results in the generation of 97 nodes (not including leaf nodes).

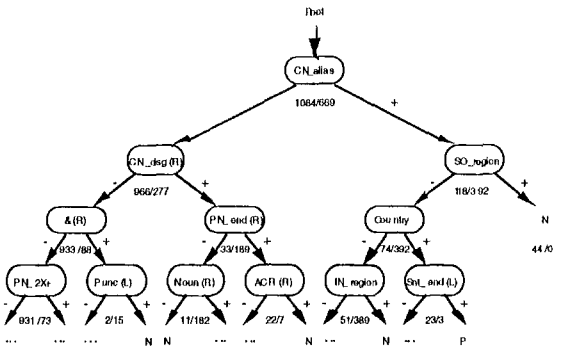

Figure 1. Company tree example (context is +/- 3).

2.3 Architecture

Figure 2 shows the working development system. The starting point is training text which has been pre-tagged with the locations of all proper names. The tokenizer separates punctuation from words. For non-token languages (no spaces between words), it also separates contiguous characters into constituent words. The part of speech (POS) tagger (Brill, 1992; Farwell *et. al.*, 1994; Matsumoto *et al.*, 1992) attaches parts of speech. The set of derived features is attached. During the delimitation phase, proper names are delimited using a set of POS-based hand-coded templates. Using ID3, a decision tree is generated based on the existing feature set and the specified level of context to be considered. The generated tree is applied to test data and scored. Manual analysis of the tree and scored result leads to the discovery of new features. The new features are added to the tokenized training text, and the process repeats.

2.4 Cross Language Porting

In order to work with another language, the following resources are needed: (1) pre-tagged training text in the new language using same tags as before, (2) a tokenizer for non-token languages, (3) a POS tagger (plus translation of the tags to a standard POS convention), and (4) translation of designators and lexical (list-based) features.

These language-specific modules are highlighted in Figure 2 with bold borders. Feature translation occurs through the utilization of: on-line resources, dictionaries, atlases, bilingual speakers, etc. The remainder is constant across languages: a language independent core development system, and an optimally derived feature set for English.

Also worth noting are the parts of development system that are executed by hand. These are shown shaded. Everything else is automatic.

3 Experiment

The system was first built for English and then ported to Spanish and Japanese. For English, the training text consisted of 50 messages obtained from the English Joint Ventures (EJV) domain MUC-5 corpus of the US Advanced Research Projects Agency (ARPA). This data was hand-tagged with the locations of company names, person names, locations names, and dates. The test set consisted of 10 new messages.

Experimental results were obtained by applying the generated trees to test texts. The initial raw text is tokenized and tagged with parts of speech. All features necessary to apply rules and trees are attached. Phrasal template rules are applied in order to delimit proper names. Then trees for each proper name type are applied individually to the proper names in the featurized text. Proper names which are

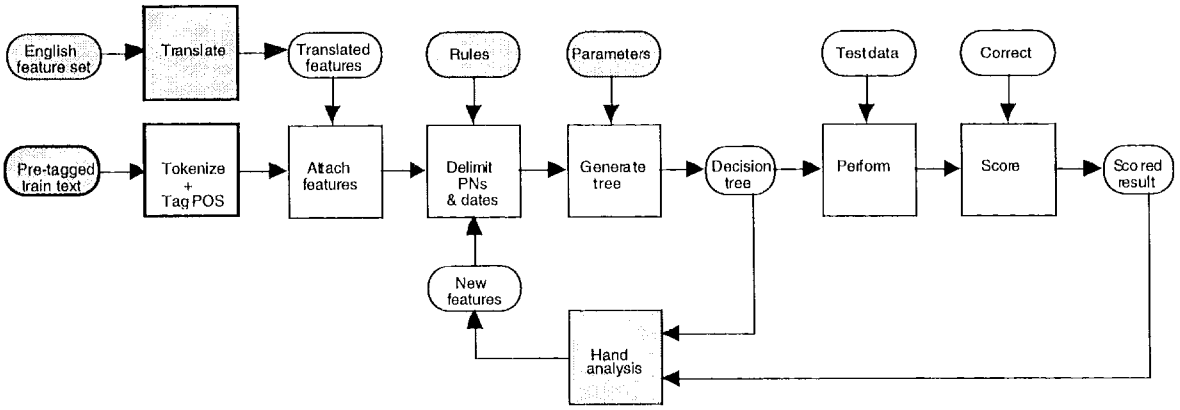

Figure 2. Multilingual development system.

voted into more than one class are handled by choosing the highest priority class. Priorities are determined based on the independent performance of each tree. For example, if person trees perform better independently than location trees, then a person classification will be chosen over a location classification. Also, designators have a large impact on resolving conflicts.

3.1 English

Various parameterizations were used for system development, including: (1) context depth, (2) feature set size, (3) training set size, and (4) incorporation of hand-coded phrasal templates.

Figure 3 shows the performance results for English. The metrics used were recall (R), precision (P), and an averaging measure, P&R, defined as:

$$P\&R = 2*P*R/(P+R) \quad (2)$$

Obtained results for English compare to the English results of Rau (1992) and McDonald (1993). The weighted average of the P&R for companies, persons, locations, and dates is 94.0%.

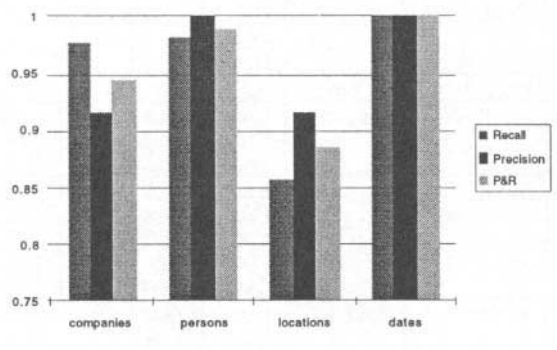

Figure 3. English performance results.

The date grammar is rather small in comparison to other name classes, hence the performance for dates was perfect. Locations, by contrast, exhibited the lowest performance. This can be attributed mainly to: (1) locations are commonly associated with commas, which can create ambiguities with delimitation, and (2) locations made up a small percentage of all names in the training set, which could have resulted in overfitting of the built tree to the training data.

Features strengths were measured for companies, persons, and locations. This experiment involved removing one feature at a time from the text used for testing and then reapplying the same tree. Figure 4 and Table 2 show performance results (P&R) when the three most powerful features are removed, one at a time, for companies, persons, and locations respectively. This experiment demonstrates the power of designator features across all proper name types, and the importance of the alias feature for companies.

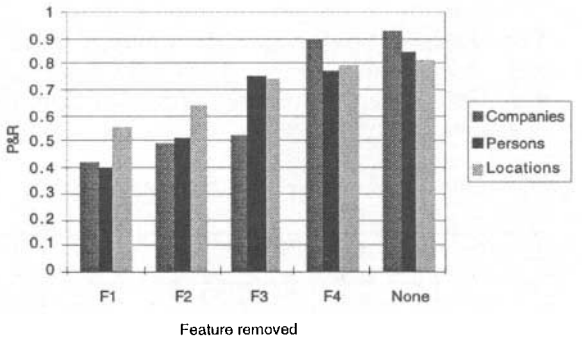

Figure 4. Feature strengths for English.

Table 2. Strongest features for English.

Feature	Companies	Persons	Locations
F1	CAP	P_desig	CAP
F2	CN_desig	CAP	L_desig
F3	CN_alias	ATH_reg	In
F4	Hyphen	F_I_L	Region

3.2 Spanish

Three experiments have been conducted for Spanish. In the first experiment, the *English* trees, generated

from the feature set optimized for *English*, are applied to the *Spanish* text (E-E-S). In the second experiment, new Spanish-specific trees are generated from the feature set optimized for English and applied to the Spanish text (S-E-S). The third experiment proceeds like the second, except that minor adjustments and additions are made to the feature set with the goal of improving performance (S-S-S).

The additional resources required for the first Spanish experiment (E-E-S) are a Spanish POS-tagger (Farwell *et al.*, 1994) and also the translated feature set (including POS) optimally derived for English. The second and third Spanish experiments (S-E-S, S-S-S) require in addition pre-tagged Spanish training text using the same tags as for English.

The obtained Spanish scores as compared to the scores from the initial English experiment (E-E-E) are shown in figure 5.

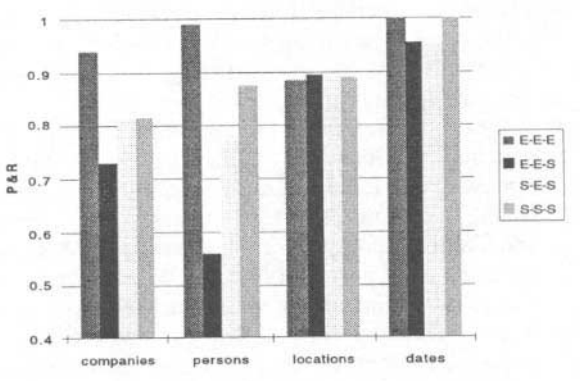

Figure 5. P&R scores for Spanish versus English.

The additional Spanish specific features derived for S-S-S are shown in Table 3. Only a few new features added to the core feature set allows for significant performance improvement.

Table 3. Spanish specific features for S-S-S.

Type	Feature	Instances	How many
List	Companies	"IBM", "AT&T", ...	100
	Keyword(s)	"del" (OF THE)	1
Template	Person	< FN DE LN >	1
	Person	< FN DE NNP >	1
	Date	< Num OF MM >	1
	Date	< Num OF MM OF Num >	1

3.3 Japanese

The same three experiments conducted for Spanish are being conducted for Japanese. The first two, E-E-J and J-E-J, have been completed; J-J-J is in progress.

The additional resources required for the first Japanese experiment (E-E-J) are a Japanese tokenizer and POS-tagger (Matsumoto *et al.*, 1992) and also the translated feature set optimally derived for English. The second and third Japanese experiments (J-E-J, J-J-J) require in addition pre-tagged Japanese training text using the same tags as for English.

The obtained Japanese scores as compared to the scores from the initial English experiment (E-E-E) are shown in Figure 6. The weighted averages of the P&R measures across all languages, for companies, persons, locations, and dates, are shown in Figure 7. Table 4 shows comparisons to other work.

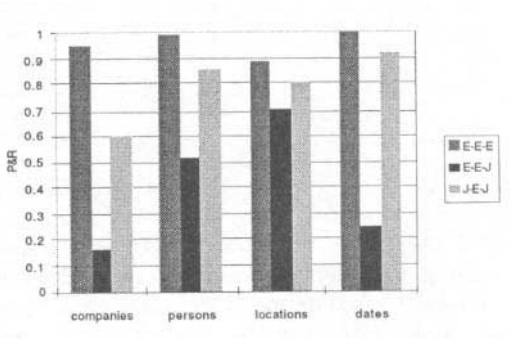

Figure 6. P&R scores for Japanese versus English.

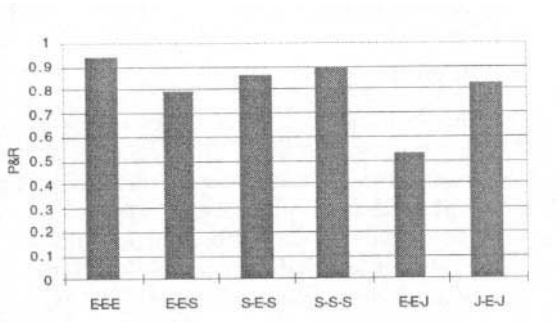

Figure 7. Weighted P&R scores comparison.

Table 4. Performance comparison to other work.

System	Lang.	Class	R	P	P&R
Rau	English	Com	NA	95	NA
PNF (McDonald)	English	Com Pers Loc Date	NA	NA	"Near 100%"
Panglyzer	Spanish	NA	NA	80	NA
MAJESTY	Japanese	Com Pers Loc	84.3 93.1 92.6	81.4 98.6 96.8	82.8 95.8 94.7
MNR (Gallippi)	English	Com Pers Loc Date (Avg)	97.6 98.2 85.7 100	91.6 100 91.7 100	94.5 99.1 88.6 100 94.0
MNR	Spanish	Com Pers Loc Date (Avg)	74.1 97.4 93.1 100	90.9 79.2 87.5 100	81.6 87.4 89.4 100 89.2
MNR	Japanese	Com Pers Loc Date (Avg)	60.0 86.5 80.4 90.0	60.0 84.9 82.1 94.7	60.0 85.7 81.3 92.3 83.1

4 Related Work

Proper name recognition has been addressed by others (Farwell et al., 1994; Kitani & Mitamura, 1994; Rau, 1992), with the goal of incorporating this capability into IR and MT systems. Related problems have been studied which utilize contextual information and learning. Examples include postediting of documents (article selection) (Knight & Chander, 1994), word sense disambiguation (Black, 1988; Siegel & McKeown, 1994), and discourse analysis (Soderland & Lehnert, 1994).

5 Future Work

An investigation of the causes of performance degradation across languages will be conducted, with the goal of pinpointing and concurrently taking steps to minimize their effects. Other plans include using ML techniques to further reduce the amount of human effort: (1) automate the building of templates for delimitation, (2) automate the discovery of new features from test results, and (3) expand the search space traversed by the tree building algorithm to include splits on feature combinations.

Acknowledgments

The author would like to offer special thanks and gratitude to Eduard Hovy for all of his support, direction, and encouragement from the onset of this work. Thanks also to Kevin Knight for his early suggestions, and to the Information Sciences Institute for use of their facilities and resources.

References

Black, E. 1988. An Experiment in Computational Discrimination of English Word Senses. In *IBM Journal of Research and Development*, 32(2).

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. 1984. *Classification and Regression Trees*. Wadsworth International Group.

Brill, E. 1992. A Simple Rule-Based Part of Speech Tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing, ACL*.

Farwell, D., Helmreich, S., Jin, W., Casper, M., Hargrave, J., Molina-Salgado, H., and Weng, F. 1994. Panglyzer: Spanish Language Analysis System. In *Proceedings of the Conference of the Association of Machine Translation in the Americas (ATMA)*. Columbia, MD.

Jacobs, P.S., Krupta, G., Rau, L., Mauldin, M.L., Mitamura, T., Kitani, T., Sider, I. and Childs, L. 1993. GE-CMU: Description of the SHOGUN System Used for MUC-5. In *Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann, pp. 109-120.

Kitani, T. and Mitamura, T. 1994. An Accurate Morphological Analysis and Proper Name Identification for Japanese Text Processing. In *Transactions of Information Processing Society of Japan*, Vol. 35, No. 3, pp. 404-413.

Knight, K. and Chander, I. 1994. Automated Postediting of Documents. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI)*, pp. 779-784.

Lehnert, W., McCarthy, J., Soderland, S., Riloff, E., Cardie, C., Peterson, J., Feng, F., Dolan, C., and Goldman, S. 1993. UMass/Hughes: Description of the CIRCUS System Used for MUC-5. In *Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann, pp. 277-292.

Matsumoto, Y., Kurohashi, S., Taegi, H. and Nagao, M. 1992. *JUMAN Users' Manual Version 0.8*, Nagao Laboratory, Kyoto University.

McDonald, D. 1993. Internal and External Evidence in the Identification and Semantic Categorization of Proper Names. In *Proceedings of the SINGLEX workshop on "Acquisition of Lexical Knowledge from Text"*, pp. 32-43.

Quinlan, J.R. 1986. Induction of Decision Trees. In *Machine Learning*, pp. 81-106.

Rau, L.F. 1992. Extracting Company Names from Text. In *Proceedings of the Seventh Conference on Artificial Intelligence Applications*, pp. 189-194.

Siegel, E.V. and McKeown, K.R. 1994. Emergent Linguistic Rules from Inducing Decision Trees: Disambiguating Discourse Clue Words. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI)*, pp. 820-826.

Soderland, S. and Lehnert, W. 1994. Corpus-Driven Knowledge Acquisition for Discourse Analysis. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI)*, pp. 827-832.

Appendix A. Abbreviations

Table 5. Definitions for abbreviations.

Abbreviation	Definition
ACR	Acronym
ATH_reg	Occurs in <Author> ... </Author>
CAP	Capitalized
CN_alias	LCS of full company name
CN_dsg	Company name designator
Country	Country name
FN	First (given) name
F_I_L	First name + initial + last name
Hyphen	Hyphen (punctuation)
IN_region	Occurs in <IN> ... </IN> region
In	Lexical "in"
LCS	Longest common substring
LN	Last (family) name
L_desig	Location designator
NNP	Proper noun
Noun	General noun
PN_end	Proper name end delimiter
PN_2X+	Proper name occurs 2+ times
Punc	Punctuation
P_desig	Person designator
Region	Geographical region name
SO_region	Occurs in <SO> ... </SO> region
Snt_end	Sentence end boundary
&	Ampersand character