A Synopsis of 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 uses machine learning and a portable framework to simplify the porting task by maximizing reuse and automation.

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.

Whether a phrase (or word) 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, proper name boundaries are identified. Next, the delimited names are categorized.

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 (Farwell *et al.*, 1994; Kitani & Mitamura, 1994; McDonald, 1993; Rau, 1992) 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. Various types of features indicate the type of name: parts of speech (POS), designators,

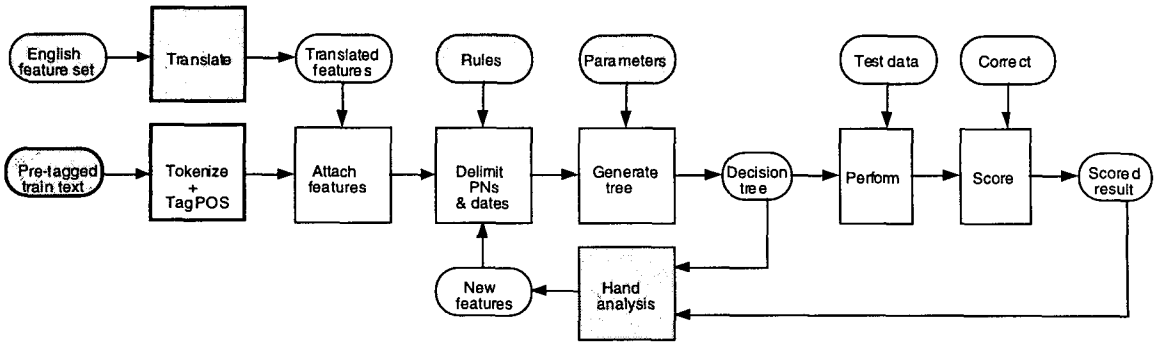

Figure 1. Multilingual development system.

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).

Features are derived through automated and manual techniques. On-line lists can quickly provide useful features such as cities, family names, nationalities, etc. Proven POS taggers (Farwell *et al.*, 1994; Brill, 1992; Matsumoto *et al.*, 1992) predetermine POS features. Other features are derived through statistical measures and hand analysis.

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 splitting criterion:

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

where p represents the proportion of names within a tree node 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 a 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. Once built, the trees are all applied individually, and then the results are merged. Trees typically contained 100 or more nodes.

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.

Figure 1 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 POS tagger (Brill, 1992; Farwell *et. al.*, 1994; Matsumoto *et al*, 1992) attaches parts of speech. The set of derived features is attached. Names are delimited using a set of POS based hand-coded templates. A decision tree is built 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. Hand analysis of results leads to the discovery of new features. The new features are added to the tokenized training text, and the process repeats.

Language-specific modules are highlighted 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, and an optimally derived feature set for English. Parts of the development system that are executed by hand appear 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 companies, persons, locations, dates, and "other". The test set consisted of 10 new messages from the same corpus.

Experimental results were obtained by applying the generated trees to test texts. Proper names which are voted into more than one class are handled by choosing the highest priority class. Priorities are determined based on the independent accuracy of each tree. 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 P&R for companies, persons, locations, and dates is 94.0% (see Table 2).

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.

Three experiments were conducted for Spanish. First, 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 test 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 additional features derived for S-S-S are shown in Table 1 (FN/LN=given/family name, NNP=proper noun, DE="de"). Only a few new features allows for significant performance improvement.

Table 1. Spanish specific features for S-S-S.

Type	Feature	Instances	How many
List	Companies	"IBM", "AT&T", ...	100
	Keyword	"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

The same three experiments are being conducted for Japanese. The first two, E-E-J and J-E-J, have been completed; J-J-J is in progress. Table 2 summarizes performance results and compares them to other work.

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.

Table 2. Performance comparison to other work.

System	Language	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

References

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.

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.

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.