SLINERC: The Sydney Language-Independent Named Entity Recogniser and Classifier

Jon Patrick, Casey Whitelaw, and Robert Munro
Language Technology Research Group
Capital Markets Co-operative Research Centre
University of Sydney
{jonpat, casey, rmunro}@it.usyd.edu.au

Abstract

The Sydney Language Independent Named Entity Recogniser and Classifier (SLINERC) is a multi-stage system for the recognition and classification of named entities. Each stage uses a decision graph learner to combine statistical features with results from prior stages. Earlier stages are focused upon entity recognition, the division of non-entity terms from entities. Later stages concentrate on the classification of these entities into the desired classes. The best overall f-values are 73.92 and 71.36 for the Spanish and Dutch datasets, respectively.

1 Introduction

Identification of named entities is an increasingly important task with applications in many areas of human language technology, including information extraction and machine translation. There has been a move away from hand-coded systems toward machine learning systems that can be trained to recognise named entities in any target language. The linguistic features available to these language-independent systems are obviously more restricted than for language-specific systems. It becomes necessary to work at a meta-linguistic level, and develop techniques to automatically learn the peculiarities of a target language. Techniques including hidden Markov models (Bikel et al., 1997) and maximum entropy theory (Borthwick et al., 1998) have been successful in supervised classification; there have been various approaches based on learning from seed lists and unannotated data (Buchholz and van den Bosch, 2000) (Cucerzan and Yarowsky, 1999).

2 SLINERC Developments

SLINERC focuses on deeper statistical properties of languages, rather than traditional surface-level linguistic features.

Surface-level features include the token itself, capitalisation, sentence position and punctuation. The extraction and use of these features as attributes in a machine learner is straightforward, and requires minimal processing. There are a large number of possible surface-level features, but it is unrealistic to provide more than a handful to machine learners. However, feature selection is most often language-dependent.

Statistically derived attributes can overcome the limitations of surface-level features. Although the extraction of the attributes is more involved, a smaller set of statistical features can capture a wide range of implicit linguistic phenomena. No surface feature selection is necessary, allowing the attributes to stay largely language-independent, whilst giving the machine learner a rich source of data.

As an example, SLINERC contains no capitalisation attributes, yet the capitalisation of a word could contribute to many or all of the statistical attributes for a given stage.

2.1 Recognition and Classification

It is important to understand that what has been referred to as "named entity recognition" is in fact two separate tasks. The first, which we call named entity *recognition*, is the division of text into entities and non-entities. An entity is divided into a headword (B-ENT) and zero or more continuation words (I-ENT). The second, named entity *classification*, is the task of determining what the type of an entity is - person, location, organisation, or other categories as required. Obviously, a system's recognition performance provides an upper bound to the performance of its classification, but recognition is important in its own right, and should be considered in the performance evaluation of any named entity classification system.

2.2 Orthographic Context Tries

SLINERC uses orthographic context tries to capture statistical differences in orthography at both the recognition and classification stages. A trie stores cumulative frequencies of the occurrence of a string of characters in each category. These frequencies can be used to give the relative probability of a given string occurring in each category for each context.

Tries have previously been used in bootstrapping algorithms on unannotated data (Cucerzan and Yarowsky, 1999). The tries used in SLINERC were trained only upon the training corpus provided for each language. No provision was made for an unassigned probability mass, or smoothing across categories.

In SLINERC four different tries were used to capture both word-internal (prefix, left-to-right, and suffix, right-to-left) and contextual (suffix of preceding token, prefix of following token) information. In producing a score for a given string, the probabilities at each trie level are combined using a weighting function; the scores for each trie are then combined. Using a genetic algorithm approach, the weighting function for each trie, and the combination function, were learned using cross-validation on the training sets.

For a given token and context, a set of tries produces a score for each category; traditionally, the highest of these is used directly as a classification. SLINERC also uses the individual scores as attributes, allowing DGRAPH-GP to exploit any systematic misclassifications.

2.3 Character N-Gram Probabilities

As well as tries, SLINERC uses statistical information about the distribution of character n-grams as features in both recognition and classification stages. A list of n-grams and their frequencies in each category is compiled from the training data. As with tries, there is currently no provision for unassigned probability mass or estimate smoothing. When classifying a token from the test set, two values are used for each category:

• The average probability of the token belonging to the category, taken across all character n-grams in the token.

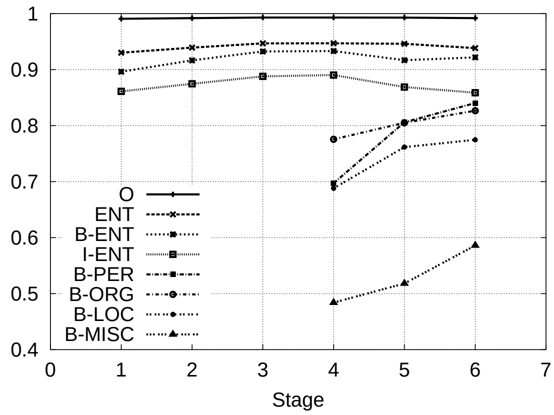

Figure 1: f-value for each stage, Spanish (combined tests)

• The n-gram in the token that has the highest probability of occurring in the category. This captures individual n-grams that are highly indicative of a particular category.

2.4 Probabilistic Learned Lists

The use of external lists, or gazetteers, in named entity recognition is problematic; it is unclear whether the benefit is worth the associated maintenance costs (Mikheev et al., 1999). In SLINERC, we compile learned lists only from the training data provided. For each token, we calculate a set of probabilities:

• Probability of the token occurring as part of a particular category

• Probability of the token occurring directly before a particular category

• Probability of the token occurring directly after a particular category

These form the basis of continuous-valued attributes that are used in both the recognition and classification stages. The data is only useful when the test corpus contains many of the same words as the training set; as the two corpora diverge, the less benefit will be gained from learned lists.

3 Multistage Recognition and Classification

At the heart of the SLINERC system is DGRAPH-GP, a decision-graph based machine learner and classifier (Patrick and Goyal, 2001). which has been shown to be superior to C4.5.

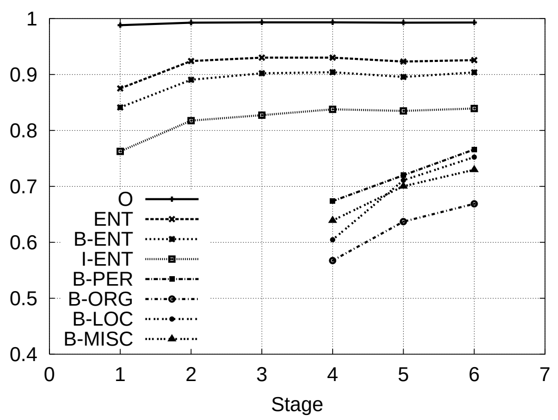

Figure 2: f-value for each stage, Dutch (combined tests)

Spanish	SLINERC seen	SLINERC unseen	learned lists
PER	92.52%	51.00%	91.31%
LOC	77.48%	59.70%	81.95%
ORG	89.06%	44.58%	86.09%
MISC	49.26%	13.16%	58.62%

Dutch	SLINERC seen	SLINERC unseen	learned lists
PER	93.44%	61.43%	95.25%
LOC	94.85%	46.99%	91.79%
ORG	82.71%	35.07%	62.15%
MISC	81.74%	39.37%	79.10%

Table 1: Recall values for each category on seen and unseen data

Following is a brief description of each stage, and a description of the attributes given to DGRAPH-GP. Results are reported for each stage in Figures 1 and 2.

3.1 Recognition Stages

The first three stages are focused upon named entity recognition; each token is classified as either a non-entity (O), entity headword (B-ENT) or entity phrase continuation (I-ENT).

3.1.1 Stage 1

Recognition is based on character bigram distribution. This stage also uses a multistate punctuation attribute (specifying if a token is punctuation, and if so, what type), and a binary attribute of whether or not the token is sentence-initial.

3.1.2 Stage 2

Stage 2 uses the same punctuation and sentence-initial attributes as Stage 1, but instead of character bigrams, it uses the scores from orthographic tries and a tentative trie-based classification.

3.1.3 Stage 3

Stage 3 combines the results from Stages 1 and 2, and augments them using probabilistic learned lists. This is the final recognition stage, and is used as the basis for the classification (division into entity categories) stages.

3.2 Classification Stages

Once a reasonable level of recognition has occurred, it is simpler to perform the entity classification step. Classification occurs in three stages that are organised in the same fashion as the recognition stages.

3.2.1 Stage 4

Character bigram distribution forms the basis for Stage 4, as in Stage 1. In compiling the bigram distributions, only tokens that formed entities were considered. Attributes follow the pattern of Stage 1, with two attributes (average, maximum) per category.

3.2.2 Stage 5

Orthographic tries were trained only on entity headwords, and weighting and combination functions were obtained through the use of genetic algorithms. The scores used as features in this stage assume that the token is an entity-initial token, and classifies accordingly. Attributes are as per Stage 2; a tentative classification, and a score for each category.

3.2.3 Stage 6

Results from Stages 4 and 5 are augmented with learned lists, as in Stage 3. Probabilities of a word occuring as, or before, each category are used as attributes.

4 Results

The multi-stage process, separating recognition from classification, has lead to results competitive with previously reported systems. Figures 1 and 2 show the incremental improvement of results stage by stage; Table 2 shows the official results, as reported by the evaluation script.

The results in the figures show the f-values for each category. These are calculated on a word-by-word basis, and no phrase grouping is performed. The discrepancy between these and the CoNLL results is due to errors in matching entire phrases.

SLINERC performs particularly well at named entity recognition, with f-values of 93.1 and 93.8 for Dutch and Spanish, respectively. The result at Stage 3 provides a solid foundation for classifications in later stages.

Table 1 shows performance on seen and unseen tokens. The performance on unseen tokens shows the benefits of statistical and contextual features; gazetteer-based approaches perform very badly on unseen data. SLINERC outperforms a simple learned-list based classifier, even for seen data; this shows the importance of contextual information in correct classification. The relative difficulty of identifying MISC entities is apparent in the low performance on unseen data in Spanish; in Dutch, MISC entities performed similarly to other categories.

5 Conclusions

SLINERC is based entirely on statistical properties of the training data. It uses no external data sources (gazetteers), nor does it make any assumptions about the target language. It has proven to be robustly language-independent, with consistently competitive performance. The techniques used could easily form the basis of a more informed named entity recognition system, through the use of either domain-specific gazetteers, or language-specific linguistic features.

After making our initial submission to the CoNNL shared task, we reorganised our processing system significantly. This caused a reduction in the f-value for Spanish development data from 74.9 to 70.3. It is clear the interaction effects of our multiple stages has significant effects on the final results.

Spanish dev	precision	recall	$F_{\beta=1}$
LOC	67.05%	76.32%	71.39
MISC	54.19%	43.60%	48.32
ORG	71.13%	68.71%	69.90
PER	82.91%	72.67%	77.45
overall	71.61%	68.97%	70.26

Spanish test	precision	recall	$F_{\beta=1}$
LOC	78.51%	72.79%	75.54
MISC	50.97%	38.64%	43.96
ORG	72.56%	79.14%	75.71
PER	80.44%	80.00%	80.22
overall	74.32%	73.52%	73.92

Dutch dev	precision	recall	$F_{\beta=1}$
LOC	66.47%	70.38%	68.37
MISC	68.97%	62.87%	65.78
ORG	72.80%	51.25%	60.16
PER	67.86%	69.49%	68.67
overall	68.87%	63.01%	65.81

Dutch test	precision	recall	$F_{\beta=1}$
LOC	76.12%	77.30%	76.71
MISC	71.04%	60.15%	65.15
ORG	70.42%	61.35%	65.57
PER	77.67%	78.44%	78.05
overall	74.01%	68.90%	71.36

Table 2: CoNNL results for Spanish and Dutch, both test sets.

References

Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of ANLP-97*, pages 194–201.

A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman. 1998. Exploiting diverse knowledge sources via maximum entropy in named entity recognition.

S. Buchholz and A. van den Bosch. 2000. Integrating seed names and n-grams for a named entity list and classifier.

S. Cucerzan and D. Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence.

A. Mikheev, M. Moens, and C. Grover. 1999. Named entity recognition without gazetteers.

Jon D. Patrick and Ishaan Goyal. 2001. Boosted decision graphs for nlp learning tasks. In Walter Daelemans and Rémi Zajac, editors, *Proceedings of CoNLL-2001*, pages 58–60. Toulouse, France.