NE recognition without training data on a language you don't speak

Diana Maynard, Valentin Tablan, Hamish Cunningham
Dept of Computer Science
University of Sheffield
Sheffield, S1 4DP, UK
diana@dcs.shef.ac.uk

Abstract

In this paper we describe an experiment to adapt a named entity recognition system from English to Cebuano as part of the TIDES surprise language program. With 4 person-days of effort, and with no previous knowledge of which language would be involved, no knowledge of the language in question once it was announced, and no training data available, we adapted the ANNIE system for Cebuano and achieved an F-measure of 77.5%.

1 Introduction

The TIDES Surprise Language Exercise is a collaborative effort between a number of sites to develop resources and tools for various language engineering tasks on a surprise language. Within a month of the language being announced, resources must be collected and tools developed for tasks such as Information Extraction (IE), Machine Translation (MT), Summarisation and Cross-Language Information Retrieval (CLIR). The aim is to establish how quickly the NLP community could build such tools in the event of a national emergency such as a terrorist attack.

A dry run for the exercise took place for 10 days during March 2003, in order to see how feasible such tasks would be, how quickly the necessary data could be collected, and to test out the best working practices for communication and collaboration between participating sites. The language chosen for the dry run was Cebuano, which is spoken by 24% of the population in the Philippines, and is the lingua franca of the South Philippines. Twenty four hours before the language was announced, a bomb had exploded in Davao City (the second largest city in the Philippines), and the event classified by the President of the Philippines as a terrorist attack.

1.1 The Cebuano Language

The Linguistic Data Consortium (LDC) at the University of Pennsylvania had previously conducted a survey of the largest 300 languages (by population), in order to establish what resources were available for each language and which languages would be potentially feasible. Their categorisation[1] includes factors such as whether they could find dictionaries, newspaper texts, a copy of the Bible, etc. on the Internet, and whether the language has its words separated in writing, simple punctuation, orthography and morphology, and so on. According to this categorisation, Cebuano was classed as a language which would be of medium difficulty to process - the main problems being that no large-scale translation dictionaries, parallel corpora or morphological analyser could be found. However, the language has a Latin script, is written with spaces between words, and has capitalisation similar to Western languages, all of which make processing a much easier task than for, say, Chinese or Arabic. The important points are therefore that little work has been done on the language, and few resources exist, but that the language is not intrinsically hard to process.

1.2 Named Entity Recognition

We concentrated our efforts on the task of resource development for named entity recognition, since correct entity recognition is a vital precursor to many other applications such as Machine Translation and CLIR. Robust tools for multilingual information extraction are becoming increasingly sought after now that we

[1] available at http://www.ldc.upenn.edu/Projects/TIDES/language-summary-table.html

have capabilities for processing texts in different languages and scripts (for example, in GATE which is fully Unicode-compliant). Following previous efforts to adapt the ANNIE information extraction system (the default IE system that comes with the GATE architecture) to different languages and applications (e.g. (Maynard and Cunningham, 2003; Maynard et al., 2002), we decided to investigate whether we could adapt ANNIE to an unknown language within a very limited period of time.

There are two particularly important points to note about our proposed approach. First, ANNIE is a rule-based system, which means that it does not require large amounts of training data, unlike most NE systems which rely at least partially on machine learning algorithms, such as (Bikel et al., 1999). This is a benefit since we were not likely to find suitable pre-existing training data. Second, we could not guarantee that we would have a native speaker available to help us develop rules for the system. This appears initially to counteract the benefit of using a rule-based system, since how could we expect to develop rules for a language of which we had no knowledge, if we had no training data? Perhaps surprisingly, this turned out not to be a major problem, as will be explained in more detail in the following sections.

2 Resources

A collaborative effort was made by all participants to collect and make available tools and resources which might be useful. These were divided into general tools (not necessarily for Cebuano), monolingual text resources, bilingual text resources, and lexical resources. Other useful information, such as details of Cebuano native speakers who were willing to help, was also made available, where appropriate.

2.1 Text Resources

Clearly, monolingual (Cebuano) texts were necessary in order to have clean data to work on. Various websites were found containing news texts, though these had mostly to be downloaded daily because there were no archives. In particular we found two good sources: Superbalita[2] and iliganon.com[3] (local news from Iligan City and the surrounding area).

Other sites found bilingual text resources online, such as the Bible, but these were not particularly helpful to us since they did not contain the kinds of entities we were interested in. Had we found any such texts, it could have been very useful as a method of mining the English texts for gazetteer entries and grammar rules.

2.2 Lexical Resources

Various lexical resources were located and made available by the participating sites, such as a list of surnames, and some bilingual dictionaries. However, many of these resources were not available until after we had already built our system.

2.3 Other Resources

Due to the limited amount of time available, it was unfeasible to find Cebuano speakers who had computational and linguistic skills, and to train them to use GATE and learn to write grammar rules etc. An extensive search via email and the Internet revealed several native speakers who were prepared to help, however. We made use of one local native speakers to annotate some texts with Named Entities manually (on paper), so that we could evaluate our system. We also made use of a native speaker in the US found by another site, who evaluated some preliminary results for us (again, on paper). The results of our search for speakers was encouraging in that we found many more contacts who could have been used had we had the time and money available.

One particular gem was the discovery of a Yahoo groups email discussion list for Cebuano speakers. Members of this list were able to provide us with some resources such as electronic dictionaries not available on the Internet, and (had we had the time and money) could have again been a very useful source of further information.

3 Adapting ANNIE to Cebuano

GATE is one of the few architectures to support multilingual processing, using Unicode as its default text encoding. While the default IE system is English-specific, some of the modules can be reused directly (e.g. the Unicode-based tokeniser can handle Indo-European lan-

[2]http://www.sunstar.com.ph/superbalita/

[3]http://www.iliganon.com/newsroom/bisaya.html

guages), and/or easily customised for new languages (Pastra et al., 2002).

The system consists of the following resources taken from ANNIE: tokeniser, sentence splitter, POS tagger, gazetteer, NE grammar, and orthomatcher. Some of these were used without modification, the others were modified for Cebuano as described below. Figure 1 shows a diagram of the architecture of the system.

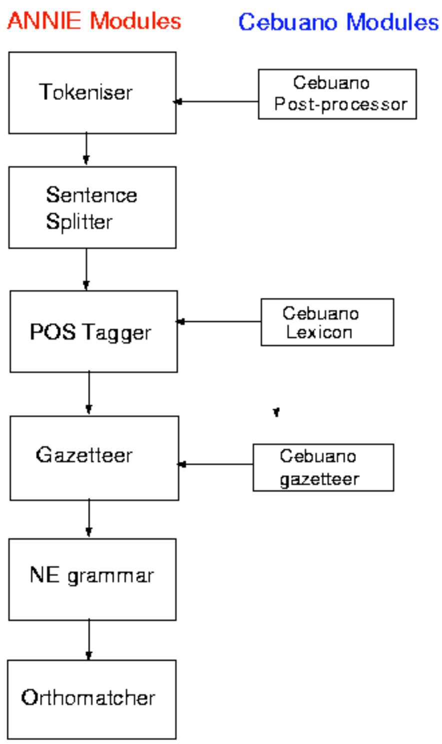

Figure 1: Architecture of Cebuano NE system

3.1 POS Tagger

The Hepple POS tagger, which is freely available in GATE as part of ANNIE, is similar to the Brill's transformation-based tagger (Brill, 1992), but differs mainly in that it uses a decision list variant of Brill's algorithm. This means that in classifying any instance, only one transformation can apply. It is also written in Java.

Having acquired a bilingual Cebuano-English lexicon containing also POS information, we decided to test whether we could adapt the Hepple tagger to Cebuano. On first appearances it seemed that Cebuano word order and morphology is similar to English, and it also has similar orthography. The rules for English (derived from training on the Wall Street Journal) would clearly not be applicable for Cebuano, so we used an empty ruleset, but we decided that many of the default heuristics might still be appropriate. The heuristics are essentially as follows:

1. look up the word in the lexicon
2. if no lexicon entry found:
    - if capitalised return NNP
    - if word containes "-" return JJ
    - if word contains a digit return CD
    - if word ends in "ed", "us", "ic", "ble", "ive", "ish", "ary", "ful", "ical", "less" return JJ
    - if word ends in "s" return NNS
    - if word ends in "ly" return RB
    - if word ends in "ing" return VBG
    - if none of the above matched return NN
3. apply the trained rules to make changes to the assigned categories based on the context

These rules make sense for Cebuano because it is unusual for Cebuano words to have endings such as "ic","ly", "ing" etc. This means that in most cases, the tag returned will be NNP (proper noun) if capitalised, or NN (common noun) if not, which is appropriate.

```
muse|n.|batahala sa arte
muse|v.|paghandum, paghanduraw
museum|n.|musiyo
mushroom|n.|libgos, uhong, kaupas
music|n.|honi, musika
musical|adj.|mahitungod sa honi
```

Figure 2: Extract from Cebuano-English lexicon

Adapting the tagger did have a number of problems, mostly associated with the fact that while the English lexicon (used for the tagger) consists only of single-word entries, the Cebuano lexicon contained many multi-word entries (such as *mahitungod sa honi* (musical), as shown in Figure 2). The tagger expects lexicon entries to have a single word entry per line, followed by one or more POS tags, each separated by a single space.

We therefore modified the lexicon so that the delimiter between the lexical entry and the POS tag(s) was a "#" rather than a space, and adapted the tagging mechanism to recognise this. This enabled us to use multi-word lexical entries. As shown in Figure 2, there was also the problem that Cebuano synonyms were placed all on one line, rather than as separate entries, and that, conversely, where a Cebuano entry had more than one POS category associated with it, these had been included as separate entries. This, along with reordering the entries, adjusting the format to fit with the English lexicon and converting the POS tags to Penn Treebank-style tags, was a fairly trivial problem fixed automatically using a series of scripts.

3.2 Tokeniser

Once the lexicon had been reformatted, a final problem remained. The POS tagger is implemented in GATE such that it assigns a POS category as a feature and value to a Token (as identified by the Tokeniser). Many of the Cebuano lexical entries are multi-word, and therefore multi-token (since the tokeniser delimits tokens according to white space), and therefore the entries would not match tokens found in the text and tags could not be correctly assigned. To solve this, we used a similar mechanism to that used for the English tokeniser in GATE (as opposed to the default Unicode tokeniser), which incorporates an extra processing component that joins together various tokens into one in order to deal with the problem of tagging the possessive "'s" as a single unit rather than as two separate ones. This is detailed in the GATE User Guide (Cunningham et al., 2002).

We therefore created two additional Cebuano-specific processing resources to complement the default Unicode tokeniser, in the form of a gazetteer list and JAPE (Java Annotations Pattern Language) grammar. The gazetteer list consisted of all the multi-word entries from the Cebuano lexicon. The JAPE grammar was used to match any of these multi-word entries found in the text and combine the Token annotations (created by the tokeniser) into a single annotation in each case. This was run before the POS tagger, so that the tagger would then have as input one Token annotation per lexical entry, and would be able to generate a single POS tag as a feature on each entry.

We currently have no means of evaluating the POS tagger, but initial results based on the manual annotation of Named Entities look promising (for example, proper nouns are correctly tagged). The creation of the tagger took approximately 2 person-days, and we were able to make it available to other sites within 4 days of the language being announced (we did not start work on it on day 1). This was useful to sites working in a variety of different areas. For example, one site were planning some annotation projection experiments to develop taggers, and wanted output from our tagger to provide a useful reference point. Another site working on date/time tagging needed POS annotations to help them identify numbers, while those working on Machine Translation (MT) and Cross-Language Information Retrieval (CLIR) could also clearly benefit from such information.

The POS tagger can only be used within GATE (which currently has thousands of users at hundreds of sites worldwide), but we were also able to offer a tagging service to the other program participants, whereby they could email us a set of texts and we would return the results of tagging as XML or HTML files within a matter of minutes - either as inline annotations or as TIPSTER-compliant standoff markup (the default GATE method), according to their preference.

Figure 3 shows a small sample of text "Moabot ngadton sa 1," marked with inline POS annotations. Figure 4 shows the same text marked with standoff POS annotations.

3.3 Gazetteers

Perhaps surprisingly, there seemed to be little information available on the Internet that could be used to compile gazetteer lists. Some lists of Philippine cities were donated to us, but little else seemed to be readily available. We therefore investigated the news corpora collected by various sites, and discovered a corpus of Cebuano local news texts, of which the majority were in English, but some were in Cebuano. We mined the English texts for names of organisations, locations, people's first names, etc. and created

```
<Token gate:gateId="69" orth="upperInitial" category="NNP" length="6"
  kind="word" string="Moabot">Moabot</Token>
<Token gate:gateId="1184" orth="multi" category="NN" kind="word"
  string="ngadto sa">ngadto sa</Token>
<Token gate:gateId="75" length="1" category="CD" kind="number"
  string="1">1</Token><Token  gate:gateId="76" length="1" category=","
  kind="punctuation" string=",">,</Token>
```

Figure 3: Example of inline POS annotations

```
<Node id="106"/>Moabot<Node id="112"/>
<Node id="113"/>ngadto<Node id="119"/>
<Node id="120"/>sa<Node id="122"/>
<Node id="123"/>1<Node id="124"/>,<Node id="125"/>

<Annotation Type="Token" StartNode="106" EndNode="112">
<Feature>
  <Name className="java.lang.String">orth</Name>
  <Value className="java.lang.String">upperInitial</Value>
</Feature>
<Feature>
  <Name className="java.lang.String">category</Name>
  <Value className="java.lang.String">NNP</Value>
</Feature>
<Feature>
  <Name className="java.lang.String">length</Name>
  <Value className="java.lang.String">6</Value>
</Feature>
<Feature>
  <Name className="java.lang.String">kind</Name>
  <Value className="java.lang.String">word</Value>
</Feature>
<Feature>
  <Name className="java.lang.String">string</Name>
  <Value className="java.lang.String">Moabot</Value>
</Feature>
</Annotation>
```

Figure 4: Example of standoff POS annotations

some new gazetteer lists. We also created lists of expressions such as days of the week, months of the year, numbers etc. from online dictionaries and phrasebooks. Furthermore, we found some typical clue words in Cebuano news texts such as jobtitles which were recognisable due to their similarity with either English or Spanish. For example "Presidente" followed by a proper noun clearly could be translated as President, which enabled us to deduce that the following proper noun was a person's name. These clue words were also compiled into gazetteer lists.

The GATE gazetteer processing resource enables gazetteer lists to be described in 3 ways: majorType, minorType and language. The major and minor types enable entries to be classified according to two dimensions or at 2 levels of granularity – for example a list of cities might have a majorType "location" and minorType "city". Using the language classification enabled us to keep the same structure for the Cebuano lists as for their English counterparts, and simply alter the language label, enabling us a method of differentiation. Because some names of English entities were found in the Cebuano texts (e.g. "Cebu City Police Office"), we required both the English gazetteer (to recognise "Office") and the Cebuano gazetteer (to recognise "Cebu City", which is not in the English gazetteer). Using both gazetteers improved recall and did not appear to affect precision, since English entities did not seem to be ambiguous with Cebuano entities or proper nouns. We did not perform extensive evaluation on this though, for reasons of time.

3.4 Named Entity Grammars

Most of the JAPE rules for NE recognition in English are based on POS tags and gazetteer lookup of candidate and context words (more detail is given in e.g. (Cunningham et al., 2002). Assuming similar morphological structure and word order, the default grammars are therefore not highly language-specific, as was discovered when they were adapted for Romanian (Hamza et al., 2002; Pastra et al., 2002). We did not have time to make a detailed linguistic study of Cebuano, though for the full experiment in the summer we would do this.

3.5 Orthomatcher

We used the orthographic coreference module (orthomatcher) to boost recognition of unknown words. This works by matching entities tagged as Unknown with other types of entities (Person, Location etc.) if they match according to the coreference rules. For example, "Smith" on its own might be tagged as Unknown, but if "John Smith" is tagged as a Person, the orthomatcher will match the two entities and retag "Smith" as a Person. We predicted that the rules would not be particularly language-specific, given a language with similar morphology and word order, so we used the orthomatcher directly, without modification. Manual inspection of texts showed that the orthomatcher was helpful in improving recall. For example, it recognised "Pairat" as a Person due to coreference with "Leo Pairat" which was correctly recognised as a Person by the first grammar. Although we were not focusing on coreference per se, we noticed that many coreferences were correctly identified, which proves indeed that the rules used are not particularly language-specific.

4 Evaluation

The team at the University of Maryland offered to get one of their native speakers to evaluate some sample texts annotated by our system. The annotations done by this native speaker were not perfect (we noticed that they had wrongly tagged some generic and common nouns as Locations, for example), but they were the only method of evaluation we had available within our restricted time. We used the system to tag 10 news texts taken from the Superbalita news corpus, and wrote a small JAPE grammar to produce the output in a form whereby each entity type was highlighted in a different colour when saved as an HTML file, so that the result could be viewed in a web browser, without access to the actual annotations. This was because it was too time-consuming to teach the annotator to use GATE. The annotator marked on a paper copy which entities were correct, incorrect, partially correct and missing, and faxed us the copies.

The results were 85.1% Precision, 58.2% recall, and an F measure of 69.1%. Because of the way the marking was done, we do not have figures to hand for the individual entity types.

We also ran a second experiment with a further 12 files from the Superbalita news corpus, and 9 files from the Iliganon news corpus. These texts were annotated by a local Cebuano speaker prior to our experiment, and the automated scoring tools in GATE were used to evaluate the results of the system. The results (in terms of Precision, Recall and F-measure) are shown in Table 1, together with with the results from our baseline system, the default ANNIE system for English, which we ran on the same test set. ANNIE typically scores for Precision and Recall in the 90th percentile for English news texts.

Clearly the results for Recall are much higher for these texts than for the other set. We suspect that there are two reasons for this. First, between running the first and second experiment, we added to the gazetteers using information from the English news corpus. Second, we strongly suspect that there are many super-

Cebuano system	P	R	F	Baseline system	P	R	F
Person	71	65	68	Person	36	36	36
Organization	75	71	73	Organization	31	47	38
Location	73	78	76	Location	65	7	12
Date	83	100	92	Date	42	58	49
Total	**76**	**79**	**77.5**	**Total**	**45**	**41.7**	**43**

Table 1: NE results on Iliganon texts

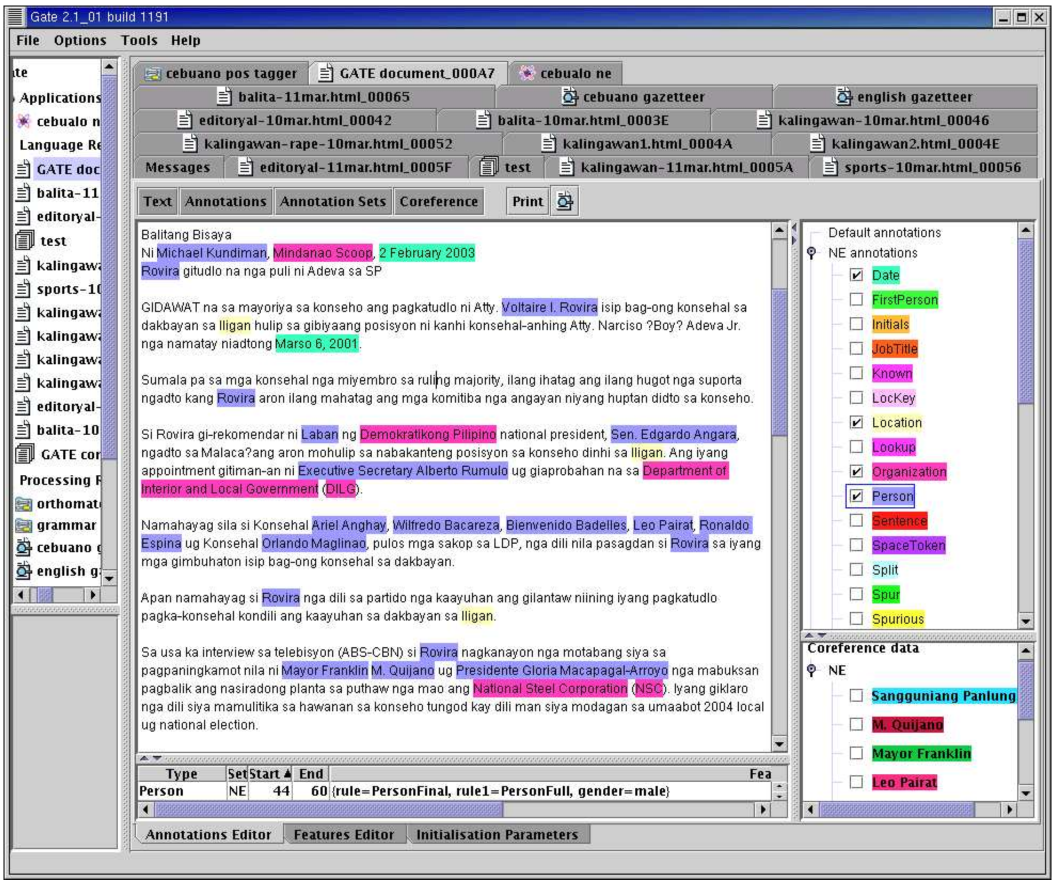

Figure 5: Architecture of Cebuano NE system

fluous key annotations in the data for the first experiment. For example, we noticed that many common nouns, such as "the doctor" and "the committee" had been wrongly tagged as Person and Organization entities.

Because our native speaker was not experienced in NE recognition and we had no time for full training, we found some cases where the key annotations were inaccurate. For example many relative dates (of the type "next week", "this year") were missed, and organisations where the abbreviation was given in brackets were treated as the same entity as the full name, e.g. "National Steel Corporation (NSC)" was tagged as one Organization entity and not two.

There were many cases where our system correctly identified entities that our human annotators missed or tagged incorrectly. For example, our human annotator wrongly tagged "Sangguniang Panlungsod" (City Council) as a Location, which our system correctly identified it as an Organization. We identified this mistake by using GATE's AnnotationDiff tool to search for errors. Looking at the text the entity seemed to refer to an organisation, so we searched for it on the Internet using Google, and discovered a website which gave the English translation "City Council" from which we can deduce that it is an Organization.

Figure 5 shows a screenshot of a Cebuano news text tagged by our system.

5 Conclusions and further work

Such errors indicate that the evaluations are by no means conclusive, but they do indicate that our system achieved a very creditable performance. If the manual annotations had been more correct, we believe that the evaluation results would have been higher.

We were extremely pleased with the results, given the time constraints of the work and the fact that we had used no native speaker or training data to produce the system. We have no real comparison against other systems on this kind of work, because we were the only site participating in the program who attempted named entity recognition. For the full exercise in June 2003, however, we imagine that there will be other systems performing NE recognition, against which we can form some performance comparison. It is interesting to compare this work with that of (Palmer and Day, 1997), who demonstrated the large differences in languages for the NE task, but who also concluded that much of the NE recognition task can be performed with a very simple analysis of NE strings and

Clearly the choice of the Cebuano language brought some important benefits for our system. We needed to make only very small changes to the tokeniser and NE grammar, and needed no modifications at all to the sentence splitter and orthomatcher components. We did not make use of any morphological analysis or parsing components. A language with a different script and/or significantly different morphology or word order would have necessitated many more modifications to the system, and clearly we would have struggled to produce such a system within the time limits without a native speaker. However, for such a language, there might have been more tools and resources already available. For example, many people have already worked on tools for Chinese and Arabic, and there is a lot more data available. Cebuano was very limited in this respect. A significantly different language would therefore have necessitated a totally different approach, for example using machine learning techniques. Luckily GATE offers support for this since it is now integrated with WEKA and has various components for machine learning and Hidden Markov Models, so this could be an option for the full exercise in the summer.

References

D. Bikel, R. Schwartz, and R.M. Weischedel. 1999. An Algorithm that Learns What's in a Name. *Machine Learning, Special Issue on Natural Language Learning*, 34(1-3), Feb.

E. Brill. 1992. A simple rule-based part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, Trento, Italy.

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, and C. Ursu. 2002. *The GATE User Guide*. `http://gate.ac.uk/`.

O. Hamza, D. Maynard V.Tablan, C. Ursu, H. Cunningham, and Y. Wilks. 2002. Named Entity Recognition in Romanian. Technical report, Department of Computer Science, University of Sheffield.

D. Maynard and H. Cunningham. 2003. Multilingual Adaptations of a Reusable Information Extraction Tool. In *Proceedings of the Demo Sessions of EACL'03*, Budapest, Hungary.

D. Maynard, H. Cunningham, K. Bontcheva, and M. Dimitrov. 2002. Adapting A Robust Multi-Genre NE System for Automatic Content Extraction. In *The Tenth International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA 2002)*.

D. Palmer and D. Day. 1997. A statistical profile of the named entity task. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP)*, Washington, D.C.

K. Pastra, D. Maynard, H. Cunningham, O. Hamza, and Y. Wilks. 2002. How feasible is the reuse of grammars for Named Entity Recognition? In *Proceedings of 3rd Language Resources and Evaluation Conference*.