MUSE: a MUlti-Source Entity recognition system

Diana Maynard, Valentin Tablan,Kalina Bontcheva, Hamish Cunningham, Yorick Wilks
([diana,valyt,kalina,hamish,yorick]@dcs.shef.ac.uk)
Department of Computer Science,
Regent Court, 211 Portobello St,
University of Sheffield, S1 4DP, UK
diana@dcs.shef.ac.uk
tel: +44 114 222 1938
fax: +44 114 222 1810

25 July 2003

Abstract. This paper describes a robust and easily adaptable system for named entity recognition from a variety of different text types. Most information extraction systems need to be customised according to the domain, either by collecting a large set of training data or by rewriting grammar rules, gazetteer lists etc., both of which methods can be costly and time-consuming. The MUSE system incorporates a modular set of resources from which different subsets can be mixed and matched as required. The process of selecting the correct resources depending on the text type is fully automatic. This method could be easily extended to deal with different languages in the same way. Results show figures in the 90th percentile for news texts, and slightly lower for other text types.

Keywords: Information Extraction, Language Engineering, GATE, Named Entity Recognition, genre

1. Introduction

In recent years, challenges in Information Extraction (IE) have moved in two major directions. First, a more semantically based approach is required, whereby information extraction is becoming more a task of **content extraction**, as witnessed by programs such as ACE[1], which deals with the semantic analysis of text rather than the linguistic analysis imposed by the MUC competitions. Second, the need is arising for systems which can be quickly and easily tailored to new domains, languages and applications(Maynard et al., 2002b; Maynard et al., 2003a). The TIDES Surprise Language Exercise is an excellent example of this[2].

MUSE is an IE system to perform named entity recognition on diverse types of text with minimal adaptation. It is based on ANNIE, the default IE system that comes with GATE(Cunningham et al., 2002a). In this paper, we describe the design of the system and details of the resources used (Section

[1] http://www.itl.nist.gov/iaui/894.01/tests/ace/
[2] http://www.darpa.mil/iao/TIDES.htm

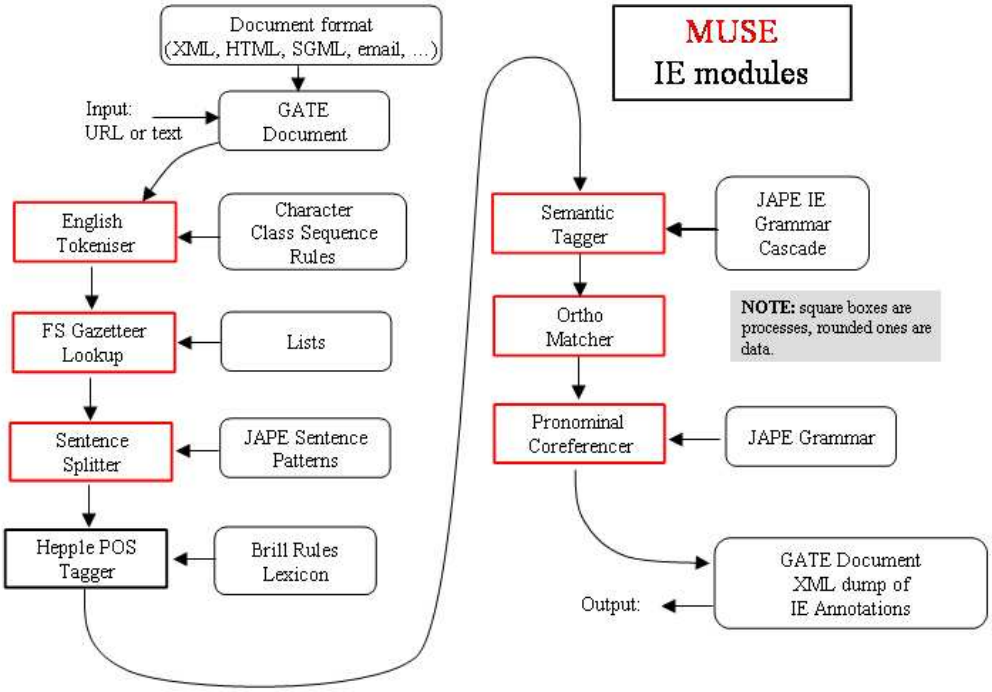


Figure 1. Simple MUSE architecture

2), discuss the problem of named entity recognition from different text types (Section 3), and give details of some evaluations (Section 6).

2. Processing Resources

We first describe the set of processing resources used in the MUSE system. The application consists of a conditional controller operating over a pipeline of processing resources, which run over the language resources (a corpus of documents). Figure 1 shows the basic architecture of the system (but not the complexities of the switching controller mechanism, which will be shown later). The processing resources rely largely on finite-state algorithms and the JAPE language (Cunningham et al., 2002b). More detailed descriptions aof all the processing resources can be found in (Maynard et al., 2003a; Cunningham et al., 2002b).

2.1. UNICODE TOKENISER

The tokeniser splits the text into very simple tokens such as numbers, punctuation and words of different types. For example, we distinguish between words in uppercase and lowercase, and between certain types of punctuation. The aim is to limit the work of the tokeniser to maximise efficiency, and enable greater flexibility by placing the burden on the grammar rules, which are more adaptable. The default tokeniser is both domain- and language-independent, though minor modifications may be useful for specific languages.

2.2. ENGLISH TOKENISER

The MUSE English Tokeniser is a processing resource that comprises a default Unicode tokeniser and a JAPE transducer. The transducer has the role of adapting the generic output of the tokeniser to the requirements of the English part-of-speech tagger. One such adaptation is to join together into one token constructs like “ ’30s”, “ ’Cause”, “ ’em”, “ ’N”, “ ’S”, “ ’s”, “ ’T”, “ ’d”, “ ’ll”, “ ’m”, “ ’re”, “ ’til”, “ ’ve”, etc. Another task of the JAPE transducer is to convert negative constructs like “don’t” from three tokens (“don”, “ ’ ” and “t”) into two tokens (“do” and “n’t”).

The English Tokeniser should always be used on English texts that need to be processed afterwards by the POS Tagger.

2.3. SENTENCE SPLITTER

The **sentence splitter** is a cascade of finite-state transducers which segments the text into sentences. This module is required for the tagger.

Each sentence is annotated with the type Sentence. Each sentence break (such as a full stop) is also given a “Split” annotation. This has several possible types: “.”, “punctuation”, “CR” (a line break) or “multi” (a series of punctuation marks such as “?!?!”.

The sentence splitter is domain- and application-independent, and to a certain extent language-independent, though it relies (for English) on a small lexicon of common abbreviations to distinguish between full stops marking these from full stops marking ends of sentences.

2.4. POS TAGGER

The **tagger** (Hepple, 2000) is a modified version of the Brill tagger, which produces a part-of-speech tag as an annotation on each word or symbol. The tags used are Penn Treebank style; the list can be found in the Gate User Guide (Cunningham et al., 2002b). The tagger uses a default lexicon and ruleset (the result of training on a large corpus taken from the Wall Street

Journal). Both of these can be modified manually if necessary. Three additional lexicons exist - one for texts in all uppercase (lexicon_cap), one for texts in all lowercase (lexicon_lower), and one for texts which combine all uppercase with normal case words (lexicon_all). To use these, the default lexicon should be replaced with the appropriate lexicon(s) at load time. The default ruleset should still be used in all cases.

While the POS tagger is clearly language-dependent, experiments with the Cebuano language(Maynard et al., 2003b) have shown that by simply replacing the English lexicon with an appropriate lexicon for the language in question, reasonable results can be obtained, at least for Western languages with similar word order and case marking to that of English, with no further adaptation.

2.5. GAZETTEERS

The gazetteer lists used are plain text files, with one entry per line. Each list represents a set of names, such as names of cities, organisations, days of the week, etc. Gazetteer list can be set at runtime to be either case sensitive or case insensitive (by default they are case sensitive).

An index file (lists.def) is used to access these lists; for each list, a major type is specified and, optionally, a minor type. It is also possible to include a language in the same way, where lists for different languages are used. In the example below, the first column refers to the list name, the second column to the major type, and the third to the minor type. These lists are compiled into finite state machines. Any text matched by these machines will be annotated with features specifying the major and minor types.

```
currency_prefix.lst:currency_unit:pre_amount
currency_unit.lst:currency_unit:post_amount
date.lst:date:specific
day.lst:date:day
```

2.6. GAZETTEER FOR DEGRADED TEXTS

An alternative version of the gazetteer lists has been created for degraded texts where case information is not always present. If this gazetteer is used, it should have the runtime parameter set to “case insensitive”. The reason for this version is that if the parameter of the default gazetteer is set to case insensitive, there are many more ambiguities between proper nouns and common nouns, particularly with first names, such as “may”, “will” etc. This version of the gazetteer has been processed automatically to remove common ambiguities such as these, or to place them in specific lists. This was done by comparing the lists with WordNet to find the ambiguous cases.

2.7. PARTIAL MATCHING

The gazetteer can also be set to run in partial matching mode[3]. This means that instead of matching only full words (indicated by a white space or punctuation boundary) it can be set to match partial words. For example, if there were an entry in a gazetteer list "Unit", this would match against part of the word "United" in the text. Generally it is not desirable to break words in this way for English, but in restricted circumstances it can be useful, for example in agglutinative languages or languages with a rich morphology.

2.8. SEMANTIC TAGGER

The MUSE semantic tagger consists of a set of grammars based on the JAPE language. They contain rules which act on annotations assigned previously, in order to produce outputs of annotated entities. Each grammar set contains a series of JAPE grammars run sequentially, such that annotations created by one grammar may be used by a following grammar. This is very important for ambiguity resolution between entity types.

In the simple example below, the pattern described will be awarded an annotation of type “Location”. This annotation will have the attribute “kind”, with value “unknown”, and the attribute “rule”, with value “GazLocation”. (The purpose of the “rule” attribute is simply for debugging).

```
Rule: GazLocation
(
{Lookup.majorType == location}
)
:loc -->
 :loc.Location = {kind="unknown", rule=GazLocation}
```

Most grammar rules use a combination of gazetteer lookup and POS tags, though they may include any kind of annotation such as token type, orthographic information, token length or previous entity annotations found in an earlier phase. Feature information can also be passed from a matched annotation using a more complex kind of rule involving Java code on the RHS of the rule. For example, the minor type from a gazetteer lookup can be percolated into the new annotation, such that we can retain information in the final entity annotation about whether a person is male or female, or classification information about locations.

There are several co-existing grammar sets in operation. A grammar set is selected automatically according to the text type, as explained in Section 5.3.

The different grammar sets mostly contain (pointers to) the same core set of grammars, but differ in some small details such as the definitions of

[3] Note that this is a new feature and may not be present in older versions of GATE.

space and control characters, and in the prioritisation of certain rules. Some examples of the differences are detailed below:

- Email texts have line breaks and headers processed differently, and some special rules. For example, a name in angled brackets (such as <john>) is generally tagged as an email address rather than as a Person.
- Spoken texts use a POS tagger trained on single case text, but have no grammar differences from written texts.
- Scientific texts have some special rules for numbers and abbreviations.
- Sports texts have special rules which annotate certain locations (such as names of teams) as organisations.

Section 6 discusses how these different grammar sets perform and how much effect the text type has on the performance of the system as a whole.

2.9. ORTHOMATCHER

The orthomatcher module detects orthographic coreference between named entities in the text, e.g., *James Somebody* and *Mr. Somebody*. It has a set of hand-crafted rules, some of which apply for all types of entities, while others apply only for specific types, such as persons or organisations. The majority of these rules were originally developed in the LaSIE system (Humphreys et al., 2000), but several new ones were added for MUSE. The orthomatcher module is described more fully in (Dimitrov et al., 2002; Maynard et al., 2003a).

2.9.1. *Classifying Unknown Proper Names via the Orthomatcher*

The orthomatcher is also used to classify unknown proper names and thereby improve the name recognition process. During the named entity recognition phase, some proper nouns are identified but are simply annotated as Unknown, because it is not clear from the information available whether they should be classified as an entity, and if so, what type of entity they represent. A good example of this is a surname appearing on its own without a title or first name, or any other kind of indicator (such as conjunction with another name, or context such as a jobtitle).

The orthomatcher tries to match Unknown annotations with existing annotations, according to the same rules as before. If a match is found, the annotation type is changed from Unknown to the type of the matching annotation, and any relevant features (such as gender of a Person) are also added to match. Two Unknown annotations cannot be matched with each other. Also, no annotation apart from an Unknown one can be matched with an existing annotation of a different type, e.g. a Person can never be matched with an

Organisation, even if the two strings are identical, and its annotation type cannot be changed by the orthomatcher. So, for example, "Smith" occurring on its own in the text might be annotated by the JAPE transducer as Unknown, but if "Mr Smith" is also found in the text (and annotated as a Person), the orthomatcher will find a match between these two strings, and will change the Unknown annotation into a Person one.

3. Data

The corpus used for training and testing the system is diverse in terms of style, domain and genre, in order to provide examples of different text types. The aim is that although the system is tuned towards these types of texts, future use of the system is not limited to a particular style, domain or genre, unlike the design of most current systems. Furthermore, it should be the case that any resulting adaptation necessary on account of this should be minimal and simple to perform.

We therefore needed to have a multi-genre corpus composed of different types of text, both for training and testing purposes, including different genres, subject fields, degrees of formality, styles, etc. The only restriction imposed was that the set of named entities should be the same for all texts. Wew therefore used a subsection of the British National Corpus (BNC) (Burnard, 1995), since it has a detailed yet practical taxonomy of text types.

This subcorpus was divided into various broad areas, such as spoken, written, books, periodicals, etc., according to the different types of text we wish to make distinctions about. Very specific distinctions do not need to be drawn, because it is unlikely to be important for the NE task whether, for example, the author is male or female, or the geographical region in which the speaker lives.

3.1. The Written Subcorpus

Written material in the BNC is classified according to medium, domain, author and audience types. We aimed to make only some broad distinctions for the creation of the subcorpus. For instance, we combine miscellaneous published and unpublished material together, since in practice there is little difference between these two. For example, charity leaflets, local guides and holiday brochures belong to the former, and church magazines and leaflets, newsletters and (unpublished) academic theses belong to the latter.

The subcorpus was selected according to the two most important criteria: domain and medium. Since there are 3 types of medium (combining the miscellaneous types into one), and 9 types of domain, this produces 27 categories. From these, we selected material from each medium, and from a selection of domains, totalling about 3 million words, as follows:

– books from the domain of natural and pure sciences;

– periodicals from the domain of commerce and finance;

– miscellaneous material from the domains of imagination, social sciences, world affairs and arts.

3.2. THE SPOKEN SUBCORPUS

Spoken material in the BNC has the advantage of being pre-homogenised, in that the layout and markup is identical to that of written text. Transcriptions of spoken data which are not marked up in this way need to be pre-processed in order to avoid loss of accuracy in NE recognition, since capitalisation and punctuation facilitate this task. By using marked-up spoken texts, we eliminate this problem to some extent, although there are still many transcription errors, and minimal punctuation. We are not interested here in the problem of named entity recognition from spoken corpora *per se*, and therefore it is unnecessary to use raw transcriptions. The reason we use spoken texts is simply to provide us with further examples of different styles and genres of texts.

The most important distinctions in the spoken texts are between monologues, such as demonstrations and speeches, and dialogues, such as transcriptions of meetings, general recorded conversation (e.g. at work, at home); and between "context-governed" and "demographic" material. Context-governed texts are divided into 4 domains, while demographic texts are divided according to the details of the respondents, such as age, gender and social class. Strangely, these two categories are mutually exclusive. As with the written subcorpus, we created the spoken subcorpus using the two most important distinctions. We therefore have 8 categories for the spoken text: 2 for speech type (monologue/dialogue) and 4 for domain. From these, we selected texts from the following 3 categories, totalling about 1 million words:

– monologues from the domains of education and business;

– dialogues from the domain of public and institutional matters.

3.3. THE EMAIL CORPUS

The email corpus consists of messages taken from a computer support mailing list (approximately 530,000 words), and messages taken from a medical mailing list (approximately 200,000 words). These are not further categorised in any way.

3.4. CORPUS FORMAT

The original BNC texts were transformed from SGML to XML representations. This is because although GATE can handle certain SGML representations, the document format analysis module could not deal with the particular entities used in the BNC texts. When a document is loaded into GATE, the document format is automatically analysed and the annotations are processed, such that only the text appears in the document viewer of the GUI, but the annotations can be accessed by other modules or visualised on screen by means of the annotations table.

The email texts are represented as EML format. In a similar way, GATE's document format analysis produces annotations for the email headers, which can then be accessed by other modules.

Figure 2 shows a screenshot of an HTML document showing the original markups annotation set (i.e. the original HTML tags which have been transformed into GATE annotations).

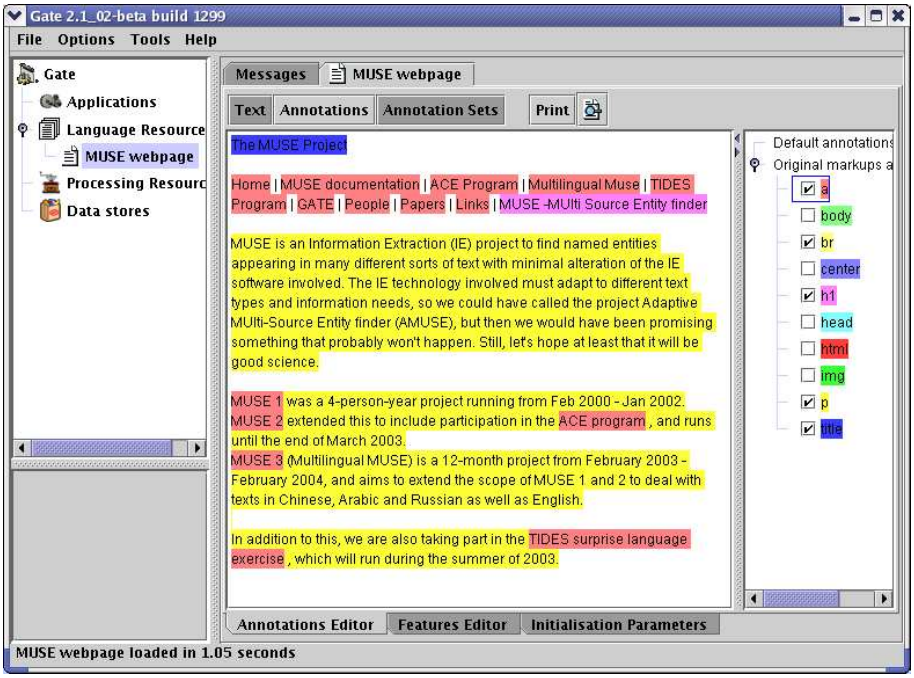


Figure 2. HTML document with Original Markups annotation set

Table I. Distribution of entity types in different corpora

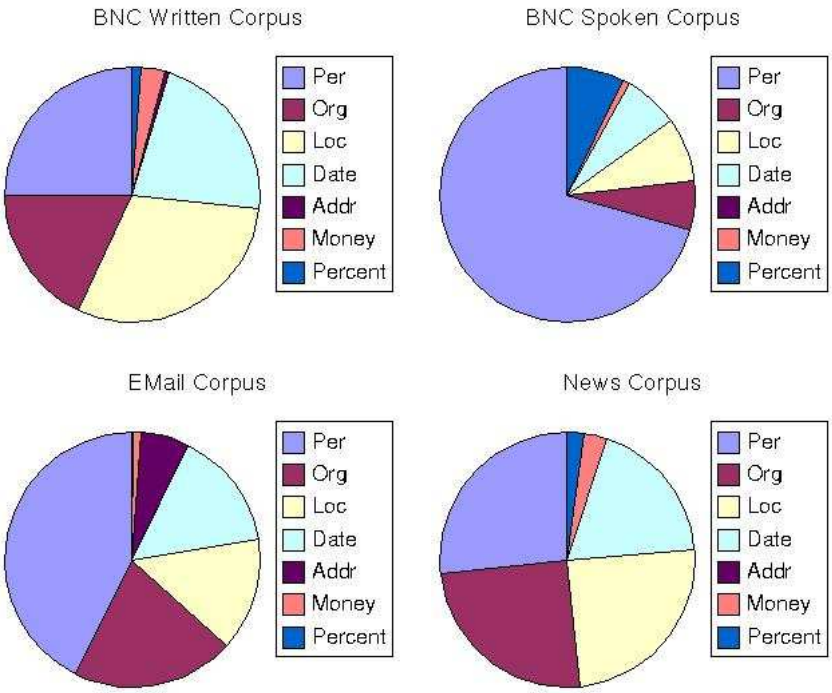
Corpus	Per	Org	Loc	Date	Address	Money	Percent	Total
Written	178	131	216	156	3	20	7	711
Spoken	312	26	34	33	0	5	33	443
Email	144	69	49	52	23	3	0	340
News	891	809	806	634	5	94	54	3293

4. Analysis of the Corpus

Analysis of a sample section of the corpus also showed a difference in the proportion of entities of each type, according to the text. Table I shows the number of entities of each type occurring in our test corpus used for evaluation. The important factor is the relative proportions of each entity type occurring, for each text type. This is more clearly depicted in Table II. We also investigated a corpus of news texts, as a comparison, since news texts are most frequently used for evaluation of named entity recognition. It is interesting to note that there is a much higher proportion of Persons in the spoken corpus than in the email corpus, and a much higher proportion of Persons in the email corpus than in the written corpus. Dates and Organisations seem to occur far less frequently in spoken texts, whereas Persons and Locations occur more frequently. There is a much more marked difference between the BNC texts and the email texts, however. Some entity types are almost exclusively to be found in emails, such as IP, internet and email Addresses, and identifiers. Dates occur much more frequently in emails, while Locations occur much less frequently. These are all reasonably intuitive, given the nature of email, and the fact that the written texts are all at least several years old (and less likely to contain references to the internet, for example).

The relative proportions of Date, Location and Organisation are roughly the same in each of the 3 corpora, though the total proportion of these differs greatly, because of the widely varying proportions of Person. The proportion of Location to Date and Organisation is slightly higher in the written corpus, however. This fits with the general theme that email corpora and spoken corpora are more similar to each other than either of them are to written corpora. It also fits with the findings from ACE that Persons are more prevalent than other entities. The news corpus is mostly similar in constitution to the BNC written corpus, except for a slightly higher prevalence of Persons over Locations in the news corpus.

Table II. Distribution of entity types in different corpora



We identified a number of features of different text types which require adaptation to the processing resources. Although changes may be necessary to both the grammars and gazetteer lists, the adaptation is only required in the grammars themselves, because the gazetteer lists are designed in such a way that they can be manipulated in different ways from the grammar. When calling for entries found in a gazetteer, we can specify a broader or narrower set, depending on our requirements (e.g. we can specify that military titles are to be included or excluded as part of a set of general titles). When adapting a core set of resources to a new domain, new gazetteer lists must be defined. If, however, we have a set of resources for different domains, and simply switch between them according to the application, we do not need to make changes to the gazetteer list, because we can simply reference the correct gazetteer lists for a domain according to the grammar we have chosen.

Table III. Features of different text formats

	Written	Spoken	Email
Line Breaks	control char replaces space	control char replaces space	control char and space
Spacing	no extra spaces	some extra spaces	some extra spaces
Other spacing	none	none	reply separators
Spelling	few errors	some errors with names; stumbles etc. mid-word/entity	errors with all words
Punctuation	mostly correct	some missing	frequent spurious and missing
Capitalisation	mostly correct	some missing capitals	missing and extra capitals
Numbers	as figures	as words	as figures
Abbreviations		interspersed with spaces	

5. The MUSE Approach

The aim of the MUSE system is to perform named entity recognition (NE) on many different types of texts with minimal alteration to the IE software. This essentially means that the system should be as robust as possible. In practice, though, it is hard to generate this kind of robustness without sacrificing specificity and therefore at least precision, if not recall as well. To prevent this, we introduce an element of adaptivity into the system, such that the system can perform differently according to some criteria concerning the text being used. This combination of robustness and adaptivity suggests that a clear distinction should be made between the two types of knowledge required by the system, which we call *foreground* and *background* knowledge. Background knowledge represents more general information which is unlikely to change, whereas foreground information represents more specific knowledge tailored to the text or text type (according to the way in which different text types

are grouped). The aim is to make as much use of background information as possible (in keeping with the preference of robustness over adaptivity) and to restrict foreground information to a minimum. This not only encourages faster and easier processing, but also enables the scope for change to be kept to a minimum, and makes such changes easier to carry out.

5.1. Genre and Named Entity Recognition

Previous research in genre identification has demonstrated its usefulness in a number of potential applications, such as IR (Karlgren, 1998) and Digital Libraries (Rauber and Muller-Kogler, 2001).There are certain grammatical constructions and word senses which are to some extent dependent on text genre, and clearly this makes identifying genre useful for applications such as parsing, word sense disambiguation and information retrieval, to name but a few. What has been less widely recognised, however, is the influence of genre on the effectiveness of NLP tools such as IE systems.

IE systems mostly extract fixed information from documents in a particular language and domain. For the technology to be suitable for real-world applications, IE systems need to be easily customisable to new domains (Karkaletsis et al., 1999). Due in no small part to the MUC competitions (e.g. (Sundheim, 1995; Sundheim, 1998)), work on IE, and in particular on NE recognition, has largely focused on narrow subdomains. For example. MUC 3 and MUC 4 focused on newswires about terrorist attacks, while MUC 7 was concerned with reports on air vehicle launches. Some work has been carried out on adapting existing systems to new domains, but there have been few advances in tackling the problem of making a single system robust enough to deal with different domains. The adaptation of existing systems to new domains is hindered by both ontology and rule bottlenecks. A substantial amount of knowledge is needed, and its acquisition and application are non-trivial tasks. For IE systems, the complexity of the domain may be particularly influential (Bagga, 1998).

An independent, though related, issue concerns the adaptation of existing systems to different text *genres*. By this we mean not just changes in domain, but different media (e.g. email, spoken text, written text, web pages), text type (e.g. reports, letters, books), and structure (e.g. layout). The genre of a text may therefore be influenced by a number of factors, such as author, intended audience and degree of formality. For example, less formal texts may not follow standard capitalisation, punctuation or even spelling formats, all of which can be problematic for NLP.

5.2. Dealing with Structural Problems

When designing a system to deal with different text types, we need to consider style and structure problems that may occur. One example of this is the use of

control characters and line breaks. In the BNC spoken corpus, texts have very short lines which are truncated with control characters. These control characters replace the space character which would otherwise have been present. In the email corpus, however, the control character is used in addition to the space character to indicate a line break. Although this difference seems trivial, it can make quite a crucial difference if control and space characters are not correctly interpreted. Consider the following example from an email:

```
Manchester Metropolitan University
Manchester
United Kingdom
```

If a single control character is treated here as equivalent to a space character, the importance of the line break is missed, and we can easily end up with the incorrect annotation [Manchester United].

```
and I didn't just see that
in, in the picture of Van
Gogh but I could see that
in the sunflower.
```

In this example from the BNC spoken corpus, the control character *must* be treated in the same way as a space character, in order to ensure that Van Gogh is regarded as two elements of one entity, and that the whole entity is tagged as a name. If the control character is treated as a line break, at best each half of the name is tagged separately, and at worst, neither half is tagged at all.

The spoken texts also have some peculiarities which cause problems for processing. Although the BNC spoken texts have been partially snorified, they are not equivalent to the written texts in format, because punctuation and capitalisation are not always present, and words are not always spelled correctly. For example, all numbers are transcribed as words. Recognising times and years (e.g. "twenty seven minutes past two" and "two thousand and one") requires extra mechanisms. Other problems are caused by abbreviations being interspersed with spaces (e.g. "B B C" for "BBC"), and stumbles, hesitations and mispronunciations being transcribed precisely, e.g. "British erm Broadcasting Corporation".

Emails have particular structural anomalies such as the reply separator, which can occur at any point in the sentence, including between two words which, when combined, form a named entity. So this requires special consideration. As with the spoken texts, emails also tend to have limited use of punctuation and capitalisation, so global strategies need to be adopted to deal with these. However, in order to preserve accuracy, in formal written texts where punctuation and spelling are assumed to be correct, these types of strategies should not be used.

Ambiguities and use of priority may also differ according to the text type. For example, the month "May" is only recognised as a month when it begins with a capital letter, due to its ambiguity with the modal verb. In texts where capitalisation is not necessarily correct, however, this rule may need to be relaxed, and/or further criteria stipulated. The use of co-reference plays an important role in situations like these, because we can delay making a decision about the entity type until we have checked to see if we can co-refer it with another annotation of the same type (we can apply the "one-sense-per-discourse" approach to proper nouns, and thereby make the assumption that "May" will not occur in the same text as both a person's name and the month, for example).

5.3. SWITCHING CONTROLLER

The MUSE system has been designed in such a way as to take these problems into account. The main mechanism for dealing with different text types is the switching controller. This enables an application to be made conditional, such that instead of having a fixed chain of processing resources, each resource can be made to fire only if a particular feature is present on the document. We developed a very simple text categorisation module, using a combination of gazetteer lists and JAPE grammars, that automatically determines certain textual characteristics such as domain, style and source. This adds features to the document which are then used to determine which other processing resources should be used on the document. This means that we can use different resources for emails and formal written texts, for example, in order to combat some of the difficulties outlined in the previous section.

A screenshot of the switching controller in MUSE is shown in Figure 3. In this example, we can see that the "complete tagger" is only used when the text has a feature "style" with value "degraded". This ensures that for degraded texts such as emails, where case information is likely to be incorrect or missing (ie some parts of the text are in single case, or where capital letters are not used correctly), a different version of the tagger is used. This version is a combination of the default POS tagger which was trained on text with normal case, and a version which was trained on singlecase text.

When moving between different text types and genres, e.g. from formal to informal text, the first challenge is to determine the extent to which lexical resources need tuning and extraction rules need retraining. The learning of patterns and adaptation of existing resources (e.g. extending lexicons and gazetteers to deal with different domains or text types) is a circular problem, because each is necessary for the other to take place.

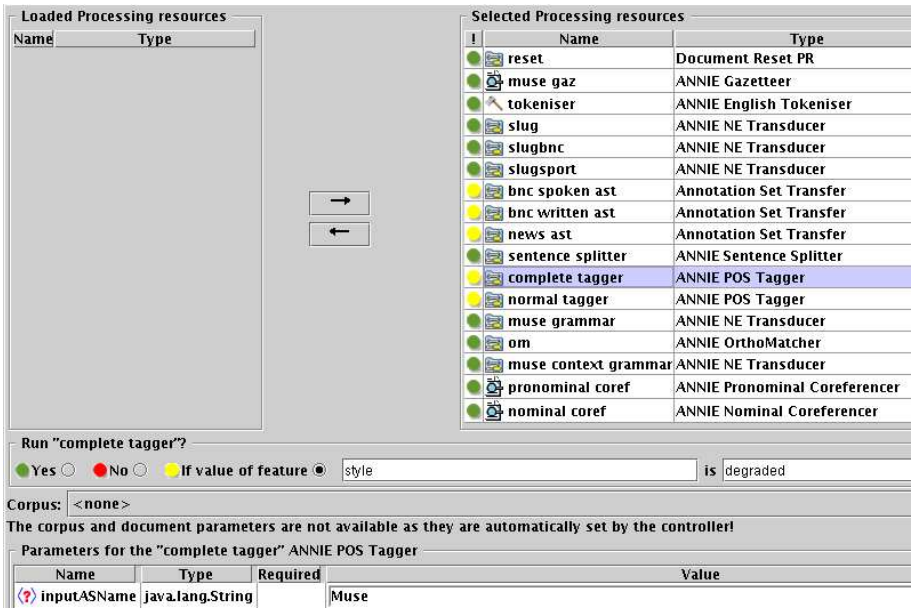


Figure 3. Switching Controller Mechanism

6. Evaluation of MUSE

We evaluated the results in terms of precision, recall and F-measure on a variety of text types: business news texts taken from the Internet, BNC written texts (of varying styles), BNC spoken texts (of varying styles) and emails of varying sources.

Tables IV, V, VI and VII give details of results for each entity type on each text type. It was left to the system to choose the most appropriate processing resources for each text type and style, as described previously. We can see that the results are quite similar for all text types. The system performed best on news texts, which is not surprising since we had most training data for these, and they are generally more regular than other text types. The system performed better on written BNC texts than on spoken ones, which is again unsurprising given that the written texts are somewhat cleaner and more regular. It also performed very well on emails, but this was due to the high performance on Addresses, which were not present in the other text types. Ignoring Addresses brings the figure more into line with the values for spoken texts, which is to be expected, since the text types are quite similar.

We also evaluated the system with and without using the Orthomatcher module, to see what effect matching Unknown entities with coreferring known entities had on the overall performance. Table VIII shows the results on the news texts without the orthomatcher module. Comparing these with the re-

Table IV. Results for BNC Written Corpus

Type	P	R	F
Person	76.9	90.8	83.3
Org	73.0	71.1	72.1
Location	75.1	69.8	71.7
Date	89.5	69.8	78.4
Money	86.8	89.4	88.1
Percent	100	100	100
Total	**83.6**	**82.7**	**83.2**

Table V. Results for BNC Spoken Corpus

Type	P	R	F
Person	93.1	92.8	92.9
Org	90.5	73.1	80.9
Location	95.8	67.6	79.3
Date	67.1	71.2	69.1
Money	100	60	75.0
Percent	90.9	90.9	90.9
Total	**79.4**	**80.4**	**79.9**

sults in Table VII, we can see that using the Orthomatcher module decreases Precision very slightly (from 93.7% to 93.5%) but improves Recall from 91.6% to 92.3%), thereby improving the F-measure from 92.7% to 92.9%. Perhaps surprisingly, it is not Persons but Organisations which gain the greatest benefit from the Orthomatcher (improving Recall by 3%).

Table VI. Results for Email Corpus

Type	P	R	F
Person	85.9	80.7	83.2
Org	56.4	69.2	62.2
Location	83.7	83.7	83.7
Date	90.4	90.4	90.4
Money	83.3	83.3	83.3
Address	95.2	87.0	90.9
Total	**82.0**	**82.6**	**82.3**

Table VII. Results for News Corpus

Type	P	R	F
Person	81.7	93.7	87.3
Org	92.7	84.2	88.2
Location	96.2	93.5	95.0
Date	89.9	85.5	87.7
Money	97.8	98.2	98.0
Percent	99.4	98.4	98.9
Total	**93.5**	**92.3**	**92.9**

7. Concluding Remarks

The MUSE system is the first rule-based NE system that we know of that aims to deal with multiple text genres inside a single application with no manual intervention, and it appears to succeed in this task. While the performance is not quite as high as that of systems as reported in e.g. the MUC evaluations, we believe that this is a small sacrifice to pay for the advantages gained by having such a flexible and robust system. The switching controller mechanism and the fact that foreground and background knowledge are kept distinct means that new modules can easily be added in order to adapt to

Table VIII. Results for News Corpus without Orthomatcher

Type	P	R	F
Person	81.5	91.6	86.3
Org	93.8	81.0	86.9
Location	96.3	94.1	95.2
Date	90.0	85.3	87.6
Money	97.8	98.2	98.0
Percent	99.4	98.4	98.9
Total	**93.7**	**91.6**	**92.7**

new domains and text types without modification of the core system and with minimal adaptation, since most components can be reused. The ease and speed of adaptation is demonstrated in some of the other applications we have created, such as the system for Romanian (Hamza et al., 2002; Maynard and Cunningham, 2003), the system for Cebuano (developed in a week) (Maynard et al., 2003b), HaSIE (Maynard et al., 2002a), and Multiflora (Wood et al., 2003), amongst others.

Overall, we believe that the system we have developed is in many ways unique. Further work will be based on extending the system to deal with new languages (Chinese and Arabic) and participating in TIDES initiatives such as the Surprise Language Program and future ACE programs.

References

Bagga, A.: 1998, 'Analysing the Complexity of a Domain with respect to an Information Extraction task'. In: *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. http://www.itl.nist.gov/iaui/894.02/-related_projects/muc/index.html.

Burnard, L.: 1995, 'Users Reference Guide for the British National Corpus'. http://info.ox.ac.uk/bnc/.

Cunningham, H., D. Maynard, K. Bontcheva, and V. Tablan: 2002a, 'GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications'. In: *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*.

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

Dimitrov, M., K. Bontcheva, H. Cunningham, and D. Maynard: 2002, 'A Light-weight Approach to Coreference Resolution for Named Entities in Text'. In: *Proceedings of the Fourth Discourse Anaphora and Anaphor Resolution Colloquium (DAARC)*. Lisbon.

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

Hepple, M.: 2000, 'Independence and Commitment: Assumptions for rapid training and execution of rule-based POS taggers'. In: *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000)*. Hong Kong.

Humphreys, K., R. Gaizauskas, and H. Cunningham: 2000, 'LaSIE Technical Specifications'. Technical report, Department of Computer Science. University of Sheffield.

Karkaletsis, V., C. Spyropoulos, and G. Petasis: 1999, 'Named Entity Recognition from Greek texts: the GIE Project'. In: S.Tzafestas (ed.): *Advances in Intelligent Systems: Concepts, Tools and Applications*. pp. 131–142, Kluwer Academic Publishers.

Karlgren, J.: 1998, 'Stylistic Experiments for Information Retrieval'. In: T. Strzalkowski (ed.): *Natural Language Information Retrieval*. Kluwer.

Maynard, D., K. Bontcheva, H. Saggion, H. Cunningham, and O. Hamza: 2002a, 'Using a Text Engineering Framework to Build an Extendable and Portable IE-based Summarisation System'. In: *Proceedings of the ACL Workshop on Text Summarisation*.

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

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

Maynard, D., V. Tablan, K. Bontcheva, H. Cunningham, and Y.Wilks: 2003a, 'MUlti-Source Entity recognition – an Information Extraction System for Diverse Text Types'. Research Memorandum CS–03–02, Department of Computer Science, University of Sheffield.

Maynard, D., V. Tablan, and H. Cunningham: 2003b, 'NE recognition without training data on a language you don't speak'. In: *ACL Workshop on Multilingual and Mixed-language Named Entity Recognition: Combining Statistical and Symbolic Models*. Sapporo, Japan.

Rauber, A. and A. Muller-Kogler: 2001, 'Integrating automatic genre analysis into digital libraries'. In: *Proceedings of the First ACM-IEEE Joint Conf on Digital Libraries*. Roanoke, VA.

Sundheim, B. (ed.): 1995, 'Proceedings of the Sixth Message Understanding Conference (MUC-6)'. Columbia, MD: ARPA, Morgan Kaufmann.

Sundheim, B. (ed.): 1998, 'Proceedings of the Seventh Message Understanding Conference (MUC-7)'. ARPA, Morgan Kaufmann.

Wood, M. M., S. J. Lydon, V. Tablan, D. Maynard, and H. Cunningham: 2003, 'Using parallel texts to improve recall in IE'. In: *Recent Advances in Natural Language Processing*. Bulgaria.