Using Human Language Technology for Automatic Annotation and Indexing of Digital Library Content

Kalina Bontcheva, Diana Maynard, Hamish Cunningham, and Horacio Saggion

Dept of Computer Science, University of Sheffield,
211 Portobello St, Sheffield, UK S1 4DP
{kalina,diana,hamish,saggion}@dcs.shef.ac.uk

Abstract. In this paper we show how we used robust human language technology, such as our domain-independent and customisable named entity recogniser, for automatic content annotation and indexing in two digital library applications. Each of these applications posed a unique challenge: one required adapting the language processing components to the non-standard written conventions of 18th century English, while the other presented the challenge of processing material in multiple modalities. This reusable technology could also form the basis for the creation of computational tools for the study of cultural heritage languages, such as Ancient Greek and Latin.

1 Introduction

As digital libraries grow in size and coverage, so does the need for automatic content annotation and indexing. Recent advances in human language technologies like named entity recognition, information extraction, and summarisation have made it possible to create automatically metadata (e.g., extract authors, titles) and document summaries, as well as annotate and index documents with information about persons, locations, dates, etc. These advances have been seen both in the quality of the results and in the robustness of the software solutions available. An increased acceptance of the importance of engineering to the successful application of HLT has led to more predictable systems that can realistically be technology providers for Digital Library systems (which have high reusability and portability requirements). In the digital library context, especially cultural digital libraries (e.g., [8]), language technology can offer new ways of accessing the collections (e.g., through indexes of events), as well as lowering the costs of annotating documents with metadata and other relevant information. While fully-automatic solutions might not be always possible or practical, HLT can frequently be used to bootstrap these laborious tasks.

In this paper we show how we used such technologies for automatic content annotation and indexing in two digital library applications: eighteenth century court trials (OldBaileyIE) and a multilingual and multimodal collection on the Euro2000 football tournament (MUMIS). Each of these applications posed a unique challenge: the court trials required adapting the language processing components to the non-standard written conventions of 18th century English, while the football collection presented the challenge of indexing material in multiple modalities - video, audio, semi-structured documents, and free text (newspaper reports). In the OldBailey application, the digitised content had to be annotated with various kinds of information, some of which was done automatically with minimum adaptation of our general-purpose named entity recogniser, while the rest was performed manually by human annotators. In MUMIS, the purpose was to construct automatically an index of all entities and events, which would then allow users to search the collection in novel ways, e.g., queries like "show me all goals by David Beckham in Euro2000".

In both cases, we used as a basis our robust and customisable named entity recogniser, which comes as part of ANNIE- A Nearly New Information Extraction system, distributed with the GATE language technology architecture and development environment [1] [10, 13]. In OldBaileyIE we also used the graphical environment of GATE, which allows manual annotation verification and correction to be carried out on the processed texts. In MUMIS, we used a full-blown information extraction system, which contained the named entity recogniser as one of its components. This was needed because the application needed to index more complex data, than just entities.

2 Domain-Independent Named Entity Recognition

There is an increasing need for robust and general purpose tools capable of automatically annotating named entities in large volumes of text. The goal is to offer good performance without any tuning for the particular domain and/or text type and at the same time, to design the modules for easy customisation, if the user decides to do so. In response to this challenge, we developed a domain-independent Named Entity (NE) recognition system. It is capable of annotating entities such as persons, organisations (e.g., companies, government bodies), locations, dates, percents, money amounts, addresses.

Most Information Extraction (IE) systems, e.g. [7, 1, 9] are designed to extract fixed types of information from documents in a specific language and domain. To increase suitability for end-user applications, IE systems need to be easily customisable to new domains [17]. Driven by the MUC competitions (e.g. [18, 19]), work on IE, and in particular on named entity recognition (NE), has largely focused on narrow subdomains, such as newswires about terrorist attacks (MUC-3 and MUC-4), and reports on air vehicle launches (MUC-7). In many applications, however, the type of document and domain may be unknown, or a system may be required which will process different types of documents without the need for tuning.

[1] GATE and ANNIE are available freely for download from `http://gate.ac.uk`. A demo of the named entity recogniser is available online at `http://gate.ac.uk/annie/index.jsp`.

Many existing IE systems have been successfully tuned to new domains and applications - either manually or semi-automatically – but there have been few advances in tackling the problem of making a single system robust enough to forego this need. 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.

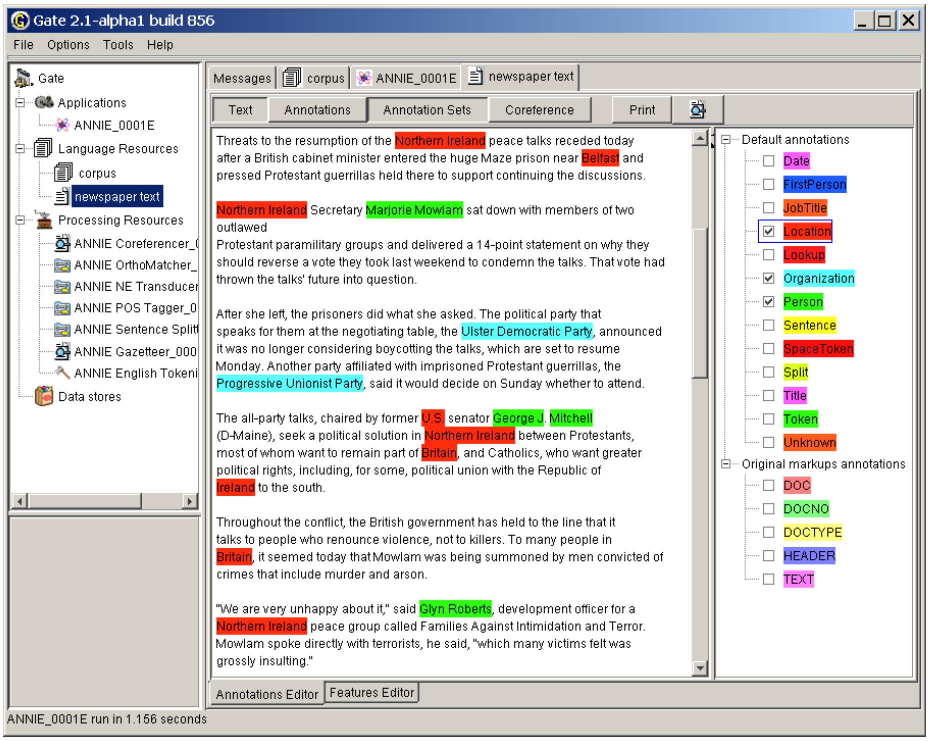
For systems to deal successfully with unknown or multiple types of source material, they must not only be able to cope with changes of domain, but also with changes of genre. By this we mean different forms of media (e.g. emails, transcribed spoken text, written text, web pages, output of OCR recognition), text type (e.g. reports, letters, books, lists), and structure (e.g. layout options). 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. Our NE system aims to identify the parameters relevant to the creation of a name recognition system robust across these types of variability [13].

Using pattern matching for NE recognition requires the development of patterns over multi-faceted structures that consider many different token properties (e.g orthography, morphology, part of speech information etc.). Traditional pattern-matching languages such as PERL get “hopelessly long-winded and error prone” [5], when used for such complex tasks. Therefore, attribute-value notations are normally used, that allow for conditions to refer to token attributes arising from multiple analysis levels. Examples of such systems include the NEA (Named Entity Analysis) rule-based system developed within the FACILE and CONCERTO projects [6, 4, 15], and CPSL (Common Pattern Specification Language) [2], from which the JAPE language was developed.

The named entity recognition modules are easily customisable, because they are based on GATE’s open architecture and consist of manually created sets of pattern-matching rules that can easily be extended to add new entity types, or modified for new domains. The rule patterns are based on information produced by earlier modules, which are responsible for segmenting the text into words (**tokenisation**) and sentences (**sentence splitting**), assignment of part-of-speech information to words (**POS tagging**), and annotations of specific named entity indicators (e.g., ‘Ltd.’, ‘Mr.’) (**gazetteer lists**). For example, the following rule specifies that one or more words, starting with an uppercase letter, followed by a company designator, should be annotated as an organisation.

```
Rule: OrgXKey (
  ({Token.kind == word, Token.orth == upperInitial})+
  {Lookup.type == cdg}
) :orgName -->
  :orgName.Organization
```

The Lookup annotations are created by the gazetteer lookup module, which assigns pre-specified types to the given lists of strings (e.g., a company designator list, person title list). These can be edited within GATE’s visual environment as



Threats to the resumption of the Northern Ireland peace talks receded today
after a British cabinet minister entered the huge Maze prison near Belfast and
pressed Protestant guerrillas held there to support continuing the discussions.

Northern Ireland Secretary Marjorie Mowlam sat down with members of two
outlawed
Protestant paramilitary groups and delivered a 14-point statement on why they
should reverse a vote they took last weekend to condemn the talks. That vote had
thrown the talks' future into question.

After she left, the prisoners did what she asked. The political party that
speaks for them at the negotiating table, the Ulster Democratic Party, announced
it was no longer considering boycotting the talks, which are set to resume
Monday. Another party affiliated with imprisoned Protestant guerrillas, the
Progressive Unionist Party, said it would decide on Sunday whether to attend.

The all-party talks, chaired by former U.S. senator George J. Mitchell
(D-Maine), seek a political solution in Northern Ireland between Protestants,
most of whom want to remain part of Britain, and Catholics, who want greater
political rights, including, for some, political union with the Republic of
Ireland to the south.

Throughout the conflict, the British government has held to the line that it
talks to people who renounce violence, not to killers. To many people in
Britain, it seemed today that Mowlam was being summoned by men convicted of
crimes that include murder and arson.

"We are very unhappy about it," said Glyn Roberts, development officer for a
Northern Ireland peace group called Families Against Intimidation and Terror.
Mowlam spoke directly with terrorists, he said, "which many victims felt was
grossly insulting."

Default annotations
- ☐ Date
- ☐ FirstPerson
- ☐ JobTitle
- ☑ Location
- ☐ Lookup
- ☑ Organization
- ☑ Person
- ☐ Sentence
- ☐ SpaceToken
- ☐ Split
- ☐ Title
- ☐ Token
- ☐ Unknown

Original markups annotations
- ☐ DOC
- ☐ DOCNO
- ☐ DOCTYPE
- ☐ HEADER
- ☐ TEXT

ANNIE_0001E run in 1.156 seconds

Fig. 1. Named Entities recognised by ANNIE

part of the customisation process. Modifying these lists is an easy task and was successfully carried out by non-expert users.

Rule writing requires some knowledge of the JAPE pattern-matching language [11] and ANNIE annotations. The pattern-matching language is based on regular expressions over the annotations; when a sequence of annotations is matched by the left-hand side pattern, then the right-hand side defines the type of annotation to be added (Organization in the example case above).

Our experiences with different systems and users indicate that writing rules to extract useful data from text is not conceptually difficult and can be learnt after some training. GATE's graphical environment allows visual inspection of the annotations and their features, which makes rule writing easier. Also, the types, attributes and values of the annotations produced by each module (e.g., tokeniser, part-of-speech tagger) are documented in GATE's User Guide [11].

The application for which the NE task is designed clearly has an impact on the development of Named Entity Recognition grammars. It can involve considerable effort simply to adapt a grammar to a new domain or task, particularly if different entity types are needed, or if the same entities have different structures and syntactic behaviour in their new context. However, adding rules for new entity types and changing some other rules for the needs of a new domain or text type/genre, is less effort and time consumming than building everything from scratch. This has been the focus of the MUSE system [14, 13], and the default IE system (ANNIE) developed within GATE, which use general-purpose grammars as a basis for developing application-specific ones.

Though the general-purpose grammars in these systems have been developed with reusability in mind, they have originated from specific NE recognition applications. There is always some subset of a purpose-built NE recognition grammar set, that is application independent; this part can be used as a basis for creating a new grammar set for a new application, no matter how different one application is from another. This set of "core" rules corresponds to the named entities (person, organisation, location names) and fixed data structures (date, time and monetary expressions), traditionally identified by any NE recognition system, which are largely domain independent.

Other rule-based NE recognition systems such as Proteus [12] and FASTUS [3] do not seem to have this flexibility of design, and therefore are much harder to adapt to new domains and applications. Current performance of the NE recognition system is around 90-95% Precision and Recall, which is similar to other current systems.

In the following section we will discuss how the named-entity recogniser was easily adapted to deal with the different capitalisation and language conventions of 18*th* century English.

3 OldBailey - Semi-Automatic Annotation of a 18*th* Century English Collection

The application required the following entity types to be recognised: Person, Location, Occupation and Status.

3.1 Adapting the technology

There were two main ways in which the technology needed to be adapted. First, the application required some new entity types to be identified, and some modifications to the existing guidelines for annotation (for example, there was a new entity type "social status"). Second, the resources needed to be modified to take into account differences caused by the language used and the text type, such as different spelling, capitalisation, punctuation, and some noisy input.

First, new gazetteer lists had to be added for status information (e.g. "wife", "spinster", "Lord", "Governor"), since this entity type is not recognised in the default gazetteer.

Second, because the texts dealt with some very specific locations in London, such as "Addington Basin", "Aldermanbury Postern", "Cripplegate", additions had to be made to the lists of locations, and in particular, to the location keywords. For example, words such as "Fink", "Chain", "Ax", "Key", "Workhouse", "Rents" etc. would not normally be associated with locations, but these were commonly used in the 18*th* century (e.g. “Bennet Fink”. We also needed to recognise facilities such as pubs as Locations, so new lists had to be added to take these into consideration.

As far as people’s names were concerned, fewer changes needed to be made, but one particular feature of the reporting style was that many first names were abbreviated, e.g. “Benj.” for “Benjamin”, “Edw.” for “Edward”, etc. New entries had to be made in the gazetteer lists for people’s first names to take this into account.

Although we already had a list of occupations, many additions had to be made to this, because there were some typical occupations of the 18*th* century which are rare these days, such as “Black-shoe-boy”, “axle tree maker”, “bottled porter dealer”.

Finally, we had to deal with orthographic variations, since the punctuation, spelling and capitalisation were very inconsistent. For example, many proper nouns were spelt both with initial capital letters and without, and sometimes a hyphen was followed by a capital and sometimes by a lowercase letter. For example, “Bread Street” was also found in the text as “Bread-Street”, “Bread-street” and “Bread street”. These non-standard written conventions were dealt with both in the gazetteer lists (e.g. by adding variations of words with and without capital letters) and also in the grammar rules (for example, by looking for strings ending in particular patterns such as “-maker”, “-draper”, “-keeper”, “-broker”, with and without capital letters and hyphens.

Changing the gazetteer lists also meant making some changes to the grammar rules to accommodate these. For example, new rules had to be written for the new entity type “status”, and further rules had to be written to solve conflicts with existing entity types. For example, “Baker” could be either an occupation or a person’s name, so rules had to be written incorporating contextual information in order to disambiguate the entities.

Below we show an example of a typical rule added to the grammar to take account of the new gazetteer lists, which creates an annotation of type “Occupation”. The rule matches a pattern consisting of any kind of word (recognised by the tokeniser), followed by one of the entries in the gazetteer list for job_keys (words which typically indicate jobs, such as “-seller”). It then annotates this pattern with the entity type “Occupation”, and gives it a feature “rule” with value “OccupationKey”. The rule feature is simply used for debugging purposes, so it is clear which particular rule has fired to create the annotation. This rule would annotate a string such as “book-seller” or “potato-merchant”.

```
Rule: OccupationKey
Priority: 50
(
 {Token.kind == word}
 {Lookup.type == job_key}
):jobtitle
-->
:jobtitle.Occupation =
{rule = "OccupationKey"}
```

Most of the adaptation to the system lay in updating the gazetteer lists and checking for conflicts between the new rules and lists, and existing ones. This latter was mainly carried out by running the system on test texts and correcting errors where they arose. The total adaptation time for the grammars and gazetteer lists (by an experienced system developer) was approximately one person-week. Further minor changes to the system were then made by the non-expert system users from the Sheffield University's Humanities Research Institute, who are creating the digital collection.

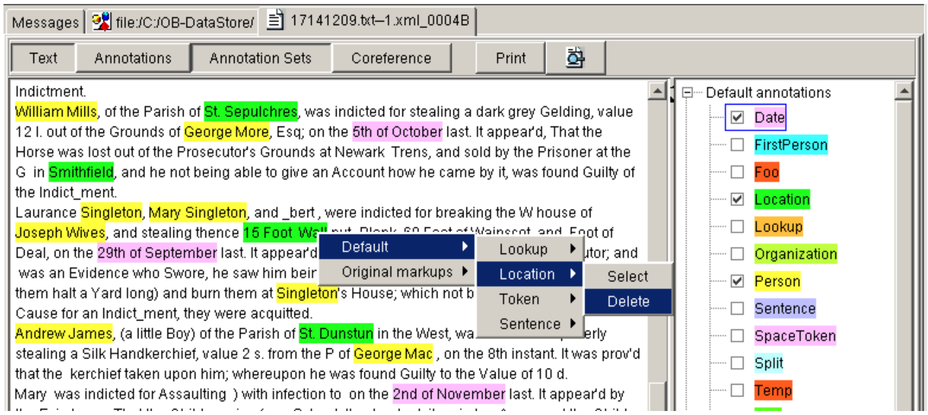
3.2 The Annotation Correction Environment

Automatic named entity annotation systems are typically capable of capturing between 85% and 95% of the entities in the texts (although these numbers are sometimes closer to 100% for entities like dates and money amounts). Nevertheless, in certain applications, as was the case with the Old Bailey collection, it is important to have all entities annotated correctly. This process involves two tasks: deleting wrong annotations and adding new annotations for the entities that have been missed. Still, because the named entity recogniser has already annotated at least 85% of all entities, this task is much less time-consuming than fully manual annotation.

Our users from the Humanities Institute used the visual annotation environment that comes with the GATE system. This environment makes the annotation process as simple and quick as possible and yet allows the flexibility to add new annotation types.

Wrong annotations are detected by the annotator visually, by inspecting the highlighted text strings; the GATE environment uses different colours for the different types of annotations (see Figure 2). To delete a wrong annotation, the user right-clicks on the highlighted text and selects the type of annotation they want deleted (there could be more than one annotations of different kinds associated with the same string). For example, in Figure 2, the system wrongly recognised '15 Foot Wall' as a location, because of a space in the digitised text, which changed the original phrase '15 Foot Wallnut Plank' to become '15 Foot Wall nut Plank'[2]. GATE also offers a facility to delete all occurences of wrong annotations, when they refer to the same entity. For example, if the House has wrongly been identified as a location, all occurrences of this entity in the text can be viewed and deleted by pressing the Delete key, when the entity is selected in the Coreference data panel on the right (see Figure 3). In this way, the user removes all occurrences with one action, rather than having to delete all these annotations individually.

[2] In 18*th* century English nouns were frequently capitalised. This example also demonstrates that the digitised content contained noisy data, such as underscores, spaces, etc.



Indictment.
William Mills, of the Parish of St. Sepulchres, was indicted for stealing a dark grey Gelding, value
12 l. out of the Grounds of George More, Esq; on the 5th of October last. It appear'd, That the
Horse was lost out of the Prosecutor's Grounds at Newark Trens, and sold by the Prisoner at the
G in Smithfield, and he not being able to give an Account how he came by it, was found Guilty of
the Indict_ment.
Laurance Singleton, Mary Singleton, and _bert , were indicted for breaking the W house of
Joseph Wives, and stealing thence 15 Foot Wa
Deal, on the 29th of September last. It appear'd
was an Evidence who Swore, he saw him beir
them halt a Yard long) and burn them at Singleton's House; which not b
Cause for an Indict_ment, they were acquitted.
Andrew James, (a little Boy) of the Parish of St. Dunstun in the West, wa
stealing a Silk Handkerchief, value 2 s. from the P of George Mac , on the 8th instant. It was prov'd
that the kerchief taken upon him; whereupon he was found Guilty to the Value of 10 d.
Mary was indicted for Assaulting) with infection to on the 2nd of November last. It appear'd by

Fig. 2. An Old Bailey example text from 1714

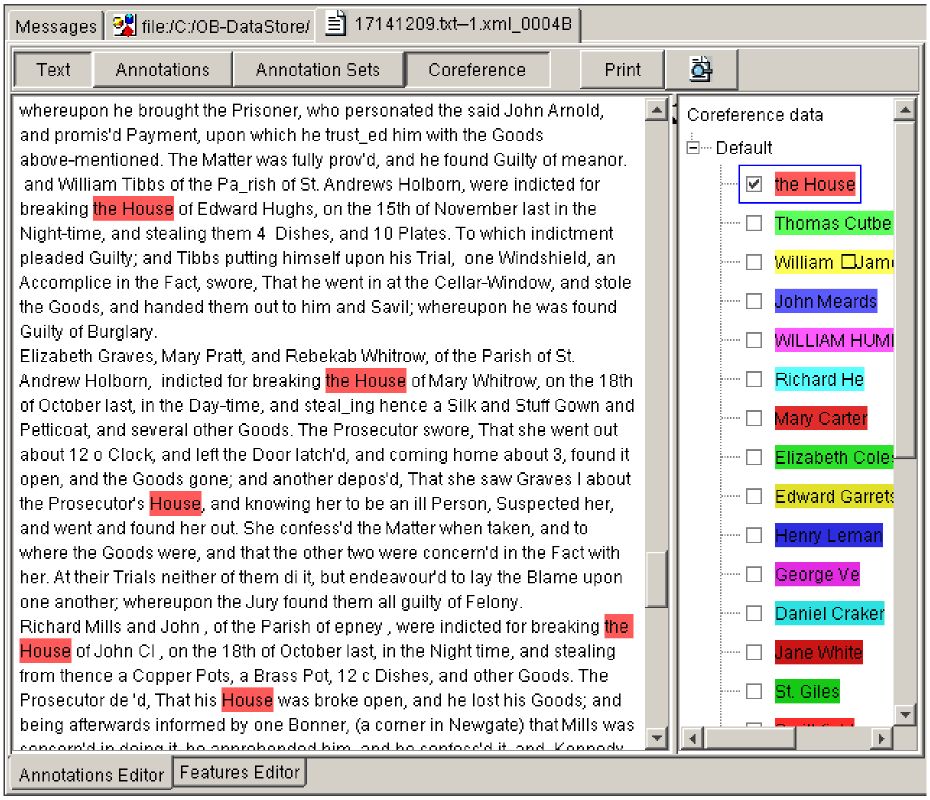
New annotations are added by selecting the text with the mouse (e.g., "William Mills") and then clicking on the desired annotation type (e.g., Person), which is shown in the list of types on the right-hand-side (see Figure 2).

The annotation environment comes as standard with a number of pre-defined annotation types (e.g., Person, Organization). New types can be defined and made available in the annotation environment by specifying them in the XML Schema language supported by W3C. For example, the schema for defining an Occupation annotation type looks like:

```
// Occupation schema
<?xml version="1.0"?>
<schema
xmlns="http://www.w3.org/2000/10/XMLSchema">
 <!-- XSchema deffinition for Occupation-->
  <element name="Occupation">
 </element>
</schema>
```

New schemas need to be created once for each application and shared by all users who do the actual annotation. Typically the schemas are created by the software support personnel, then they automatically appear in the visual environment, so users only need to be trained to use them.

The GATE visual annotation environment can also be used independently from the language processing tools. It offers the advantage of making the annotation process independent from the particular document format used, e.g., XML. Our experience with users from the humanities has shown that they find the annotation task easier when it involves selecting text and associating types in a visual environment, than when it requires writing XML markup. The colour coding scheme which associates different types of annotations with different colours



whereupon he brought the Prisoner, who personated the said John Arnold, and promis'd Payment, upon which he trust_ed him with the Goods above-mentioned. The Matter was fully prov'd, and he found Guilty of meanor. and William Tibbs of the Pa_rish of St. Andrews Holborn, were indicted for breaking the House of Edward Hughs, on the 15th of November last in the Night-time, and stealing them 4 Dishes, and 10 Plates. To which indictment pleaded Guilty; and Tibbs putting himself upon his Trial, one Windshield, an Accomplice in the Fact, swore, That he went in at the Cellar-Window, and stole the Goods, and handed them out to him and Savil; whereupon he was found Guilty of Burglary.

Elizabeth Graves, Mary Pratt, and Rebekab Whitrow, of the Parish of St. Andrew Holborn, indicted for breaking the House of Mary Whitrow, on the 18th of October last, in the Day-time, and steal_ing hence a Silk and Stuff Gown and Petticoat, and several other Goods. The Prosecutor swore, That she went out about 12 o Clock, and left the Door latch'd, and coming home about 3, found it open, and the Goods gone; and another depos'd, That she saw Graves I about the Prosecutor's House, and knowing her to be an ill Person, Suspected her, and went and found her out. She confess'd the Matter when taken, and to where the Goods were, and that the other two were concern'd in the Fact with her. At their Trials neither of them di it, but endeavour'd to lay the Blame upon one another; whereupon the Jury found them all guilty of Felony.

Richard Mills and John , of the Parish of epney , were indicted for breaking the House of John Cl , on the 18th of October last, in the Night time, and stealing from thence a Copper Pots, a Brass Pot, 12 c Dishes, and other Goods. The Prosecutor de 'd, That his House was broke open, and he lost his Goods; and being afterwards informed by one Bonner, (a corner in Newgate) that Mills was

Coreference data
Default
☑ the House
☐ Thomas Cutbe
☐ William ☐Jam
☐ John Meards
☐ WILLIAM HUMI
☐ Richard He
☐ Mary Carter
☐ Elizabeth Cole
☐ Edward Garrets
☐ Henry Leman
☐ George Ve
☐ Daniel Craker
☐ Jane White
☐ St. Giles

Fig. 3. Viewing and deleting all annotations for an entity

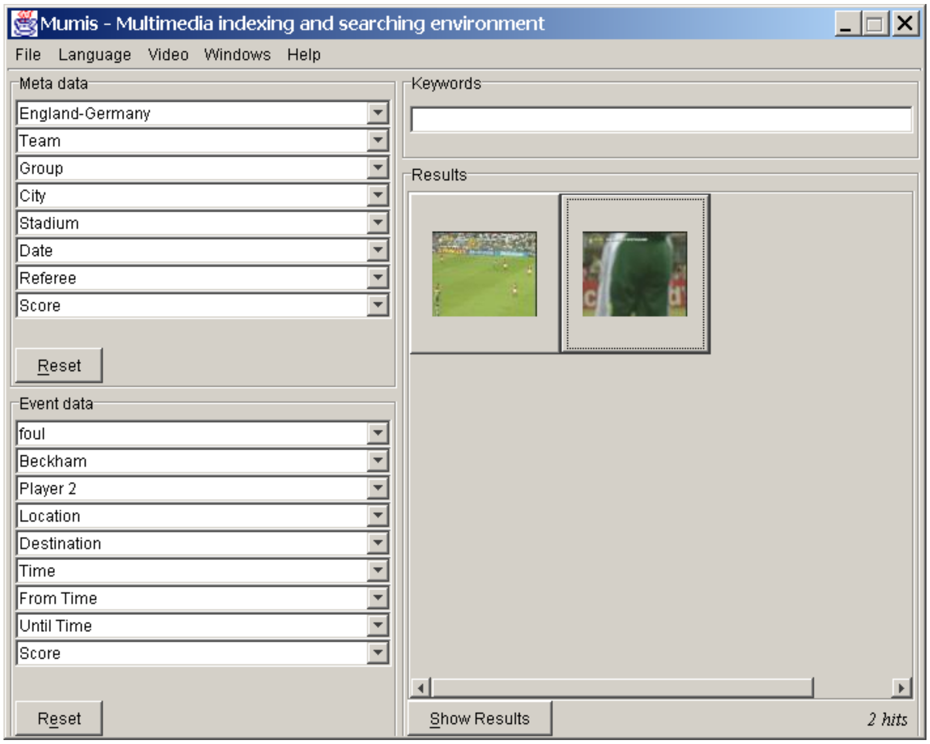


Fig. 4. The MUMIS interface

makes it easier for users to find and correct the annotations. Once the annotation process is completed, the user only needs to save the document, which automatically inserts the newly added markup in the appropriate format (e.g., HTML, XML).

4 MUMIS - Indexing and Search in Multiple Media

The MUMIS (MUltiMedia Indexing and Searching environment) system uses Information Extraction (IE) components developed within GATE to produce formal annotations about essential events in football video programme material.[3] The textual sources used for this project are taken from reports of the Euro2000 Championships: semi-structured ticker reports that give a minute by minute objective account of the match; match reports that also give a full account of the match but may be subjective; and comments that give general

[3] `http://mumis.vda.nl/`, funded by the EC's 5th Framework HLT programme under grant number IST-1999-10651. Project partners: Universities of Twente, Nijmegen and Sheffield, Esteam AB, VDA Ltd., DFKI.

information such as player profiles. These reports are drawn from a variety of online media sources (BBC-online, Press Association, The Guardian, etc.). The video material was indexed by running the IE system on the output of an automatic speech recognition system. Information about the same event in multiple sources is merged into a common index, which also contains references to the video material, thus allowing direct access to the relevant part of the video.

The indexing task in this particular application is focused on the entities and the events that occur in this domain, i.e., the aim is to enable users to access only the relevant parts of the documents, which show a particular entity and/or event of interest. Therefore, such information is collected for each document, regardless of its modality, and used to create an index of all entities and events that have occurred in the collection (e.g., red cards, goals). This kind of "conceptual" indexing is complementary to the more general indexing problem in digital libraries, where documents are typically indexed for access by author, title, etc. One of the issues here is to determine whether or not a name has been used to refer to the same entity in two different documents. For example, Cambridge in one document might refer to Cambridge in England, while in the other document it might refer to Cambridge, Massachusetts. These issues have already been recognised in previous digital library research, not only in research on human language technology, e.g., [8].

The indexed multimedia information is accessed via a menu-based user interface (in Dutch, English, and German) (see Figure 4) which allows users to query the indexed information and play video fragments matching the query (e.g., "all fouls on Beckham").

The domain-independent named entity recogniser was extended to recognise domain-specific entities such as teams, players, and referee. Our philosophy of clean separation between linguistic data and the algorithms that process it resulted in a substantial saving of effort, since only new domain-specific data (i.e., names and grammar rules) had to be created, while the algorithms using them remained unchanged. Further details on the adaptation process and on the use of information extraction for multimedia indexing and search can found in [16].

5 Conclusion

In this paper we presented briefly the domain-independent portable named entity recogniser developed as part of the ANNIE Information Extraction system. ANNIE is intended to be useable in many different applications, on many different kinds of text and for many different purposes. Here we showed how it can be applied for annotating (semi)-automatically digital library content and also for indexing such content by entities and events. The system was built for portability and offers a comprehensive support for documents in many different formats, from badly-spelled lower case email messages to structured XML or HTML pages to newswires. It is capable of dealing with noisy data and processing of large data volumes (for further details on that see [10]). Finally, ANNIE is also capable of processing and visualising multilingual documents, based on

Unicode [20], which makes it particularly suitable for digital library applications which need to deal with content in languages other than English. For example, it could be a basis for creation of computational tools for the study of cultural heritage languages, such as Ancient Greek and Latin.

The GATE visual environment was designed originally to help researchers develop HLT applications, and manual annotation facilities were needed as they form an important part of the application-building process, i.e., the performance evaluation. However, our experience with the OldBailey project has shown that this environment could be used in the future as part of a tool specifically aimed at digital libraries, which support the end user with the annotation task. In the future, we hope to be able to undertake such research in collaboration with interested digital library researchers.

References

1. D. Appelt. An Introduction to Information Extraction. *Artificial Intelligence Communications*, 12(3):161–172, 1999.
2. D.E. Appelt. The Common Pattern Specification Language. Technical report, SRI International, Artificial Intelligence Center, 1996.
3. E. R. Appelt, J.R. Hobbs, J. Bear, D. Israel, M. Kameyama, A. Kehler, D. Martin, K. Myers, and M. Tyson. SRI International FASTUS system: MUC-6 Test Results and Analysis. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, California, 1995.
4. W. Black and F. Rinaldi. Facile pre-processor v3.0 - a user guide. Technical report, Department of Language Engineering, UMIST, 2000.
5. W. Black, F. Rinaldi, and D. Mowatt. Facile: Description of the named entity system used for muc-7. In *Proceedings of the 7th MUC*, 1998.
6. F. Ciravegna, A. Lavelli, N. Mana, J. Matiasek, L. Gilardoni, S. Mazza, M. Ferraro, W. Black, F. Rinaldi, and D. Mowatt. Facile: Classifying texts integrating pattern matching and information extraction. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI99)*, 1999.
7. J. Cowie and W. Lehnert. Information Extraction. *Communications of the ACM*, 39(1):80–91, 1996.
8. Gregory Crane, Robert F. Chavez, Anne Mahoney, Thomas L. Milbank, Jeffrey A. Rydberg-Cox, David A. Smith, and Clifford E. Wulfman. Drudgery and deep thought. *Communications of the ACM*, 44(5):34–40, 2001.
9. H. Cunningham. Information Extraction: a User Guide (revised version). Research Memorandum CS–99–07, Department of Computer Science, University of Sheffield, May 1999.
10. H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. 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*, 2002.
11. H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, and C. Ursu. *The GATE User Guide*. `http://gate.ac.uk/`, 2002.
12. R. Grishman. Information Extraction: Techniques and Challenges. In *Information Extraction: a Multidisciplinary Approach to an Emerging Information Technology*, Springer 1997.

13. D. Maynard, V. Tablan, H. Cunningham, C. Ursu, H. Saggion, K. Bontcheva, and Y. Wilks. Architectural elements of language engineering robustness. *Journal of Natural Language Engineering – Special Issue on Robust Methods in Analysis of Natural Language Data*, 2002. forthcoming.
14. D. Maynard, V. Tablan, C. Ursu, H. Cunningham, and Y. Wilks. Named Entity Recognition from Diverse Text Types. In *Recent Advances in Natural Language Processing 2001 Conference*, Tzigov Chark, Bulgaria, 2001.
15. J. McNaught, W. Black, F. Rinaldi, E. Bertino, A. Brasher, D. Deavin, B. Catania, D. Silvestri, B. Armani, A. Persidis, G. Semerano, F. Esposito, V. Candela, G.P. Zarri, and L. Gilardoni. Integrated document and knowledge management for the knowledge-based enterprise. In *Proceedings of the 3rd International Conference on the practical application of Knowledge Management*. The paractical application company, 2000.
16. H. Saggion, H. Cunningham, D. Maynard, K. Bontcheva, O. Hamza, C. Ursu, and Y. Wilks. Extracting Information for Automatic Indexing of Multimedia Material. In *3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 669–676, Las Palmas, Gran Canaria, Spain, 2002.
17. S. Soderland. Learning to extract text-based information from the world wide web. *Proceedings of Third International Conference on Knowledge Discovery and Data Mining (KDD-97)*, 1997.
18. Beth Sundheim, editor. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, Columbia, MD, 1995. ARPA, Morgan Kaufmann.
19. Beth Sundheim, editor. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. ARPA, Morgan Kaufmann, 1998.
20. V. Tablan, C. Ursu, K. Bontcheva, H. Cunningham, D. Maynard, O. Hamza, Tony McEnery, Paul Baker, and Mark Leisher. A unicode-based environment for creation and use of language resources. In *Proceedings of 3rd Language Resources and Evaluation Conference*, 2002.