Summarising Legal Texts: Sentential Tense and Argumentative Roles

Claire Grover, Ben Hachey, & Chris Korycinski
School of Informatics
University of Edinburgh
{grover,bhachey,ck}@inf.ed.ac.uk

Abstract

We report on the SUM project which applies automatic summarisation techniques to the legal domain. We pursue a methodology based on Teufel and Moens (2002) where sentences are classified according to their argumentative role. We describe some experiments with judgments of the House of Lords where we have performed automatic linguistic annotation of a small sample set in order to explore correlations between linguistic features and argumentative roles. We use state-of-the-art NLP techniques to perform the linguistic annotation using XML-based tools and a combination of rule-based and statistical methods. We focus here on the predictive capacity of tense and aspect features for a classifier.

1 Introduction

Law reports form the most important part of a lawyer's or law student's reading matter. These reports are records of the proceedings of a court and their importance derives from the role that precedents play in English law. They are used as evidence for or against a particular line of legal reasoning. In order to make judgments accessible and to enable rapid scrutiny of their relevance, they are usually summarised by legal experts. These summaries vary according to target audience (e.g. students, solicitors).

Manual summarisation can be considered as a form of information selection using an unconstrained vocabulary with no artificial linguistic limitations. Automatic summarisation, on the other hand, has postponed the goal of text generation *de novo* and currently focuses largely on the retrieval of relevant sections of the original text. The retrieved sections can then be used as the basis of summaries with the aid of suitable smoothing phrases.

In the SUM project we are investigating methods for generating flexible summaries of documents in the legal domain. Our methodology builds and extends the Teufel and Moens (Teufel and Moens, 2002) approach to automatic summarisation. The work we report on in this paper deals with judgments from the judicial branch of the House of Lords. We have completed a preliminary study using a small sample of judgment documents. We have hand-annotated the sentences in these documents and performed automatic linguistic processing in order to study the link between the argumentative role and linguistic features of a sentence. Our primary focus is on correlations between sentence type and verb group properties (e.g. tense, aspect). To this end, we have used state-of-the-art NLP techniques to distinguish main and subordinate clauses and to find the tense and aspect features of the main verb in each sentence. In this paper we report on our NLP techniques and on the findings of our study. We discuss the implications for the summarisation system that we are in the process of developing.

Section 2 provides a brief background to our work including an overview of the Teufel and Moens approach and a description of the annotation scheme we have developed for the House of Lords judgments. Section 3 provides an overview of the tools and techniques we have used in the automatic linguistic processing of the judgments. Our processing paradigm is XML-based and we use specialist XML-aware tools to perform tasks such as tokenisation, part-of-speech tagging and chunking—these are described in Section 3.1. Our primary interest is tense information about individual sentences and to compute this we need to distinguish main from subordinate clauses in order to identify the main verb group. We report on our statistically-based approach to this task in Section 3.2. In Section 3.3 we present the results of our preliminary evaluations based on the small corpus of hand-annotated judgments. Finally, in Section 4 we draw some conclusions and outline future work.

2 Automatic Summarisation

2.1 Background

Much of the previous NLP work in the legal domain concerns Information Retrieval (IR) and the computation of simple features such as word frequency. In order to perform summarisation, it is necessary to look at other features which may be characteristic of texts in general and legal texts in particular. These can then serve to build a model for the creation of legal summaries (Moens and Busser, 2002). In our project, we are developing an automatic summarisation system based on the approach of Teufel and Moens. The core component of this is a statistical classifier which categorises sentences in order that they might be seen as candidate text excerpts to be used in a summary. Useful features might include standard IR measures such as word frequency but other highly informative features are likely to be ones which reflect linguistic properties of the sentences.

The texts we are currently exploring are judgments of the House of Lords, a domain we refer to here as HOLJ[1]. These texts contain a header providing structured information, followed by a sequence of sometimes lengthy judgments consisting of free-running text. Each Law Lord gives his own opinion, so in later phases of this project we will create a strategy for what is effectively multi-document summarisation. The structured part of the document contains information such as the respondent, appellant and the date of the hearing. While this might constitute some part of a summary, it is also necessary to pick out an appropriate number of relevant informative sentences from the unstructured text in the body of the document. This paper focuses on the mixture of statistical and linguistic techniques which aid the determination of the function or importance of a sentence.

Previous work on summarisation has concentrated on the domain of scientific papers. This has lent itself to automatic text summarisation because documents of this genre tend to be structured in predictable ways and to contain formalised language which can aid the summarisation process (e.g. cue phrases such as 'the importance of', 'to summarise', 'we disagree') (Teufel and Moens, 2002), (Teufel and Moens, 2000). Although there is a significant distance in style between scientific articles and legal texts, we have found it useful to build upon the work of Teufel and Moens (Teufel and Moens, 2002; Teufel and Moens, 1997) and to pursue the methodology of investigating the usefulness of a range of features in determining the argumentative role of a sentence.

Spärck Jones (1999) has argued that most practically oriented work on automated summarisation can be classified as either based on *text extraction* or *fact extraction*. When automated summarisation is based on *text extraction*, an abstract will typically consist of sentences selected from the source text, possibly with some smoothing to increase the coherence between the sentences. The advantage of this method is that it is a very general technique, which will work without the system needing to be told beforehand what might be interesting or relevant information. But general methods for identifying abstract-worthy sentences are not very reliable when used in specific domains, and can easily result in important information being overlooked. When summarisation is based on *fact extraction*, on the other hand, the starting point is a predefined template of slots and possible fillers. These systems extract information from a given text and fill out the agreed template. These templates can then be used to generate shorter texts: material in the source text not of relevance to the template will have been discarded, and the resulting template can be rendered as a much more succinct version of the original text. The disadvantage of this methodology is that the summary only reflects what is in the template.

For long scientific texts, it does not seem feasible to define templates with a wide enough range, however sentence selection does not offer much scope for re-generating the text into different types of abstracts. For these reasons, Teufel and Moens experimented with ways of combining the best aspects of both approaches by combining sentence selection with information about *why* a certain sentence is extracted—e.g. is it a description of the main result, or an important criticism of someone else's work? This approach can be thought of as a more complex variant of template filling, where the slots in the template are high-level structural or rhetorical roles (in the case of scientific texts, these slots express argumentative roles like *main goal* and *type of solution*) and the fillers are sentences extracted from the source text using a variety of statistical and linguistic techniques exploiting indicators such as cue phrases. With this combined approach the closed nature of the fact extraction approach is avoided without giving up its flexibility: summaries can be generated from this kind of template without the need to reproduce extracted sentences out of context. Sentences can be reordered, since they have rhetorical roles associated with them; some can be suppressed if a user is not interested in certain types of rhetorical roles.

The argumentative roles which Teufel and Moens settled upon for the scientific domain (Teufel and Moens, 1999) consist of three main categories:

BACKGROUND: sentences which describe some (generally accepted) background knowledge.

OTHER: sentences which describe aspects of some specific other research in a neutral way.

OWN: sentences which describe any aspect of the work presented in the current paper.

[1] Accessible on the House of Lords website, `http://www.parliament.uk/judicial_work/judicial_work.cfm`

2.2 Summarisation of HOLJ Texts

Judgments of the House of Lords are based on facts that have already been settled in the lower courts so they constitute a genre given over to largely unadulterated legal reasoning. Furthermore, being products of the highest court in England[2], they are of major importance for determining the future interpretation of English law. The meat of a decision is given in the opinions of the Law Lords, at least one of which is a substantial speech. This often starts with a statement of how the case came before the court. Sometimes it will move to a recapitulation of the facts, moving on to discuss one or more points of law, and then offer a ruling.

The methodology we implement is based on the approach used for the summarisation of scientific papers as described above, the first two steps of which can be summarised as follows:

Task 1. Decide which argumentative roles are important in the source text and are of use in the abstract.
Task 2. In a collection of relevant texts, decide for every sentence which argumentative role best describes it; this process is called "argumentative zoning".

Our annotation scheme, like our general approach, is motivated by successful incorporation of rhetorical information in the domain of scientific articles. Teufel et al. (1999) argue that regularities in the argumentative structure of a research article follow from the authors' primary communicative goal. In scientific texts, the author's goal is to convince their audience that they have provided a contribution to science. From this goal follow highly predictable sub-goals, the basic scheme of which was introduced in section 2.1 For the legal domain, the communicative goal is slightly different; the author's primary communicative goal is to convince his/her peers that their position is legally sound, having considered the case with regards to all relevant points of law. A different set of sub-goals follows (refer to Table 1).[3]

BACKGROUND	Generally accepted background knowledge: sentences containing law, summary of law, history of law, and legal precedents.
CASE	Description of the case including the events leading up to legal proceedings and any summary of the proceedings and decisions of the lower courts.
OWN	Statements that can be attributed to the Lord speaking about the case. These include interpretation of BACKGROUND and CASE, argument, and any explicit judgment as to whether the appeal should be allowed

Table 1: Description of the basic rhetorical scheme distinguished in our preliminary annotation experiments.

We annotated five randomly selected appeals cases for the purpose of preliminary analysis of our linguistic features. These were marked-up by a single annotator, who assigned a rhetorical label to each sentence. As well as providing a top-level OTHER, we asked the annotator to consider a number of sub-moves for our initial study of the HOLJ domain. These form a hierarchy of rhetorical content allowing the annotator to 'fall-back' to the basic scheme if they cannot place a sentence in a particular sub-move. The following describes the sub-categories we posit in the HOLJ domain and believe will be of use in flexible abstracting:

BACKGROUND

• PRECEDENT – Does the sentence describe a previous case or judgment apart from the proceedings for the current appeal?
E.g. "This was recognised in Lord Binning, Petitioner 1984 SLT 18 when the First Division held that for the purposes of section 47, the date of the relevant trust disposition or settlement or other deed of trust was the date of its execution...."

• LAW – Does the sentence contain public statutes? Does the sentence contain a summary or speak to the history of statutes?
E.g. "Section 12 (3A) begins with the words: "In determining for the purposes of this section whether to provide assistance by way of residential accommodation to a person...."

CASE

• EVENT – Does the sentence describe the events that led up to the beginning of the legal proceedings?
E.g. "The appellant lived at 87 Main Street, Newmills until about April 1998."

• LOWER COURT DECISION – Does the sentence describe or summarise decisions or proceedings from the lower courts?
E.g. "Immediately following Mr Fitzgerald's dismissal IMP brought proceedings and obtained a Mareva injunction against him."

OWN

• JUDGMENT – Does the sentence give an opinion or ruling as to whether the appeal should be allowed?
E.g. "For the reasons already given I would hold that VAT is payable in the sum of £1.63 in respect of postage and I would allow the appeal."

• INTERPRETATION – Does the sentence contain an interpretation of BACKGROUND or CASE items?
E.g. "The expression 'aids' in section 33(1) is a familiar word in everyday use and it bears no technical or special meaning in this context."

• ARGUMENT – Does the sentence state the question at hand, apply points of law to the current case, or otherwise present ar-

[2] To be more specific, the House of Lords hears civil cases from all of the United Kingdom and criminal cases from England, Wales and Northern Ireland.

[3] The basic scheme of the argumentative structure we define turns out to be similar to one which was conceived of for work on legal summarisation of Chinese judgment texts (Cheung et al., 2001).

gument which is to form the basis of a ruling?
E.g. "The question is whether the direction which it contains applies where the local authority are considering whether to provide a person with residential accommodation with nursing under section 13A."

3 Linguistic Analysis

3.1 Processing with XML-Based Tools

As described in Section 2.2, the sentences in our small pilot corpus were hand annotated with labels reflecting their rhetorical type. This annotation was performed on XML versions of the original HTML texts downloaded from the House of Lords website. In this section we describe the use of XML tools in the conversion from HTML and in the linguistic annotation of the documents.

A wide range of XML-based tools for NLP applications lend themselves to a modular, pipelined approach to processing whereby linguistic knowledge is computed and added as XML annotations in an incremental fashion. In processing the HOLJ documents we have built a pipeline using as key components the programs distributed with the LT TTT and LT XML toolsets (Grover et al., 2000), (Thompson et al., 1997) and the *xmlperl* program (McKelvie, 1999). The overall processing stages contained in our pipeline are shown in Figure 1.

In the first stage of processing we convert from the source HTML to an XML format defined in a DTD, *hol.dtd*, which we refer to as HOLXML in Figure 1. The DTD defines a House of Lords Judgment as a J element whose BODY element is composed of a number of LORD elements. Each LORD element contains the judgment of one individual lord and is composed of a sequence of paragraphs (P elements) inherited from the original HTML.

Once the document has been converted to this basic XML structure, we start the linguistic analysis by passing the data through a pipeline composed of calls to a variety of XML-based tools from the LT TTT and LT XML toolsets. The core program in our pipelines is the LT TTT program *fsgmatch*, a general purpose transducer which processes an input stream and rewrites it using rules provided in a hand-written grammar file, where the rewrite usually takes the form of the addition of XML mark-up. Typically, *fsgmatch* rules specify patterns over sequences of XML elements or use a regular expression language to identify patterns inside the character strings (PCDATA) which are the content of elements. The other main LT TTT program is *ltpos*, a statistical combined part-of-speech (POS) tagger and sentence identifier (Mikheev, 1997).

The first step in the linguistic annotation process uses *fsgmatch* to segment the contents of the paragraphs into word tokens encoded in the XML as W elements. Once the word tokens have been identified, the next step uses *ltpos* to mark up the sentences as SENT elements and

	TENS	ASP	VOIC	MOD
proposes	PRES	SIMP	ACT	NO
was brought	PAST	SIMP	PASS	NO
would supersede	PRES	SIMP	ACT	YES
to grant	INF	SIMP	ACT	NO
might have occurred	PRES	PERF	ACT	YES
had been cancelled	PAST	PERF	PASS	NO

Table 2: Tense, Aspect, Voice and Modality Features

to add part of speech attributes to word tokens (e.g. `<W C='NN'>opinion</W>` is a word of category noun). Note that the tagset used by *ltpos* is the Penn Treebank tagset (Marcu et al., 1994).

The following step performs a level of shallow syntactic processing known as "chunking". This is a method of partially identifying constituent structure which stops short of the fully connected parse trees which are typically produced by traditional syntactic parsers/grammars. The output of a chunker contains "noun groups" which are similar to the syntactician's "noun phrases" except that post-head modifiers are not included. It also includes "verb groups" which consist of contiguous verbal elements such as modals, auxiliaries and main verbs. To illustrate, the sentence "I would allow the appeal and make the order he proposes" is chunked in this way:[4]

```
<NG>I</NG> <VG>would allow</VG> <NG>the appeal</NG>
and <VG>make</VG> <NG>the order</NG> <NG>he</NG>
<VG>proposes</VG>
```

The method we use for chunking is another use of *fsgmatch*, utilising a specialised hand-written rule set for noun and verb groups.

Once verb groups have been identified we use another *fsgmatch* grammar to analyse the verb groups and encode information about tense, aspect, voice and modality in attributes on the VG elements. Table 2 gives some examples of verb groups and their analysis.

The final stage in the process is the step described in detail in Section 3.2, namely the process of identifying which verb group is the main verb group in the sentence. We call this process from our pipeline using *xmlperl* to pass each sentence in turn to the main verb identifier and to receive its verdict back and encode it in the XML as the value of the MV attribute on sentence elements. Figure 2 shows a small part of one of our documents after it has been fully processed by the pipeline.[5]

[4]Judgments - In re Kanaris (Respondent)(application for a writ of Habeas Corpus)(on appeal from the Administrative Court of the Queen's Bench Division of Her Majesty's High Court of Justice), heard on 30 January 2003, paragraph 2

[5]Judgments - Robertson (AP) v Fife Council, heard on 25 July 2002, paragraph 1

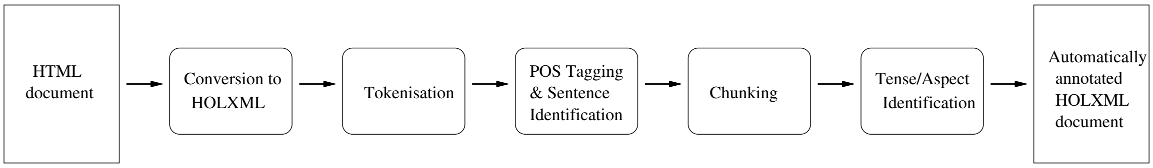

Figure 1: Processing Stages

```
<LORD>
<P>
<SENT MV='0' sid='1'><NG><W C='NNP'>LORD</W>
<W C='NNP'>SLYNN</W></NG> <W C='IN'>OF</W> <NG>
<W C='NNP'>HADLEY</W></NG></SENT>
</P>
<P>
<SENT MV='0' sid='2'><NG><W C='PRP$'>My</W>
<W C='NNS'>Lords</W></NG><W C=','>,</W></SENT>
</P>
<P no='1'>
<SENT MV='1' sid='3'><NG><W C='PRP'>I</W></NG>
<VG ASP='PERF' MODAL='NO' TENSE='PRES' VOICE='ACT'
vgid='1'><W C='VBP'>have</W> <W C='VBN'>had</W></VG>
<NG><W C='DT'>the</W> <W C='NN'>advantage</W></NG>
<W C='IN'>of</W> <W C='VBG'>reading</W> <NG>
<W C='DT'>the</W> <W C='NN'>draft</W></NG>
<W C='IN'>of</W> <NG><W C='DT'>the</W> <W C='NN'>
opinion</W></NG> <VG ASP='SIMPLE' MODAL='NO'
TENSE='INF' VOICE='PASS' vgid='2'><W C='TO'>to</W>
<W C='VB'>be</W> <W C='VBN'>given</W></VG>
<W C='IN'>by</W> <NG><W C='PRP$'>my</W>
<W C='JJ'>noble</W> <W C='CC'>and</W> <W C='JJ'>
learned</W> <W C='NN'>friend</W> <W C='NNP'>Lord</W>
<W C='NNP'>Hope</W></NG> <W C='IN'>of</W> <NG>
<W C='NNP'>Craighead</W></NG><W C='.'>.</W>
</SENT> .....
</P>
....
</LORD>
```

Figure 2: A Sample of Annotated HOLJ

3.2 Clause and Main Verb Identification

The primary method for identifying the main verb and thus the tense of a sentence is through the clause structure. We employ a probabilistic clause identifier to do this. This section gives an overview of the clause identification system and then describes how this information is incorporated into the main verb identification algorithm.

The clause identifier was built as part of a post-conference study (Hachey, 2002) of the CoNLL-2001 shared task (Sang and Déjean, 2001). CoNLL (Conference on Natural Language Learning) is a yearly meeting of researchers interested in using machine learning to solve problems in natural language processing. Each year an outstanding issue in NLP is the focus of the shared task portion of the conference. The organisers make some data set available to all participants and specify how they are to be evaluated. This allows a direct comparison of a number of different learning approaches to a specific problem. As we will report, the system we have built ranks among the top designed for 2001 shared task of clause identification.

The clause identification task is divided into three phases. The first two are classification problems similar to POS tagging where a label is assigned to each word depending on the sentential context. In phase one, we predict for each word whether it is likely that a clause starts at that position in the sentence. In phase two, we predict clause ends. In the final step, phase three, an embedded clause structure is inferred from these start and end predictions.

The first two phases are approached as straightforward classification in a maximum entropy framework (Berger et al., 1996). The maximum entropy algorithm produces a distribution $p_*(\vec{x},c)$ based on a set of labelled training examples, where $\vec{x}$ is the vector of active features. In evaluation mode, we select the class label c that maximises p_*.

The features we use include words, part-of-speech tags, and chunk tags within a set window. The classifier also incorporates features that generalise about long distance dependencies such as sequential patterns of individual attributes. Consider the task of predicting whether a clause starts at the word *which* in the following sentence:[6]

> Part IV ... is of obvious importance if the Act is to have the teeth which Parliament doubtless intended it should.

The fact that there is this subordinating conjunction at the current position followed by a verb group (*intended*) to the right gives much stronger evidence than if we only looked at the word and its immediate context.

The more difficult part of the task is inferring clause segmentation from the predicted starts and ends. This does not translate to a straightforward classification task as the resulting structure must be a properly embedded and more than one actual clause may begin (or terminate) at a start (or end) position predicted in the previous two phases. Because of the limited amount of labelled train-

[6]Judgments - Anyanwu and Other v. South Bank Student Union and Another And Commission For Racial Equality, heard on 22 March 2001, paragraph 4

SYSTEM	PRECISION	RECALL	$F_{\beta=1}$
CoNLL 1st	84.82	73.28	78.63
Our system	*83.74*	*71.25*	*76.99*
CoNLL Ave	72.46	60.00	65.64

Table 3: Scores for our clause identification system on the Penn Treebank compared to the best and average CoNLL-2001 scores.

		PRECISION	RECALL	$F_{\beta=1}$
1.	*(past)*	97.78	88.00	92.63
	(pres)	81.58	93.93	87.32
2.		90.80	84.04	87.29

Table 4: Performance results on a sample from the HOLJ corpus for (1) tense identification and (2) main verb group identification.

ing material, we run into data sparsity problems if we try to predict 3 or more starts at a position.

To deal with this, we created a maximum entropy model whose sole purpose was to provide confidence values for potential clauses. This model uses features similar to those described above to assign a probability to each clause candidate (defined as all ordered combinations of phase one start points and phase two end points). The actual segmentation algorithm then chooses clause candidates one-by-one in order of confidence. Remaining candidates that have crossed brackets with the chosen clause are removed from consideration at each iteration.

We obtained a further improvement (our F score increased from 73.94 to 76.99) by training on hand-annotated POS and chunk data from the Treebank. Table 3 compares precision, recall, and F scores for our system with CoNLL-2001 results training on sections 15-18 of the Penn Treebank and testing on section 21 (Marcus et al., 1993). The F score is more than 10 points above the average scores, failing to surpass only the best performing CoNLL system.

Once clause boundaries have been determined, they are used to identify a sentence's main verb group. A verb group that is at the top level according to the clause segmentation is considered a stronger candidate than any embedded verb group (i.e. a verb group that is part of a subordinate clause). In addition, there are several other heuristics encoded in the algorithm. These sanity checks watch for cases in which the complex clause segmenting algorithm described above misses certain strong formal indicators of subordination. First, we consider whether or not a verb group is preceded by a subordinating conjunction (e.g. *that*, *which*) and there is no other verb group between the subordinator and the current verb group. Second, we consider whether a verb group starts with a participle or infinitive *to* (e.g. *provided* in "accommodation provided for the purpose of restricting liberty", *to* in "counted as a relevant period to be deducted"). These heuristics are in the following ranked order (those closer to the beginning of the list being more likely characteristics of a main verb group):

1. *Does not* occur within an embedded clause, *is not* preceded by a subordinating conjunction, *does not* start with a participial or infinitival verb form.
2. *Does* occur within an embedded clause, *is not* preceded by a subordinating conjunction, *does not* start with a participial or infinitival verb form.
3. *Does not* occur within an embedded clause, *is* preceded by a subordinating conjunction.
4. *Does not* occur within an embedded clause, *does* start with a participial or infinitival verb form.
5. *Does* occur within an embedded clause, *is* preceded by a subordinator.
6. *Does* occur within an embedded clause, *does* start with a participial or infinitival verb form.

We also observed in the corpus that verb groups closer to the beginning of a sentence are more likely to be the main verb group. Therefore we weight verb groups slightly according to their sentence position in order to prefer those closer to the beginning of a sentence within a given category. Scores for main verb group identification are presented below in the results section below.

3.3 Results

As mentioned above, the current work has concentrated on identifying the rhetorical structure of the HOLJ domain. In studying this structure, we have begun looking for formal indicators of rhetorical categories. The linguistic analysis described in the previous sections is motivated by an observation that tense may be a useful feature. Specifically, it was observed in the corpus that sentences belonging to the CASE rhetorical role are nearly always in the past tense while sentences belonging to the other rhetorical categories are very seldom in the past tense. Here, we report a preliminary analysis of this relationship. An empirical study of the annotated files reported in section 2.2 provides the starting point for these tasks.

Our identification of the inflection for a sentence depends on the tools described in sections 3.1 and 3.2 above. These consist of (1) identifying the tense of verb groups, and (2) identifying the main verb group. Results for these two steps of automatic linguistic analysis calculated from a sample of 100 sentences from the HOLJ corpus are summarised in Table 4.[7]

[7]For main verb group identification, we report scores that take points away for missing coordinated main verbs. This is

For the evaluation of verb group tense identification, we report scores for identifying past and present, defined by the tense, aspect, and modality features on verb groups as follows:

past: TENSE=PAST, ASPECT=SIMPLE, MOD=NO
pres: TENSE=PRES, ASPECT=SIMPLE, MOD=NO

The source of errors for tense identification is mainly due to errors in the POS and chunking phases. In the case of past tense, the POS tagger has difficulty identifying past participles because of their similarity to simple past tense verbs. Performance for present tense verbs is lower because they are more easily mistaken for, say, nouns with the same spelling. For example, there were two errors in our sample where the verb *falls* was tagged as a noun and assigned to a noun group chunk instead of a verb group.

The main verb group identification algorithm considers only verb groups assigned by the chunker, whether they are true verb groups or not. Thus, these scores also reflect the algorithm's ability to deal with noise introduced in earlier stages.[8] One obvious problem is that the algorithm is thus not capable of identifying a verb group as being main if the chunker does not identify it at all. The primary source of errors in the remaining sentences are also propagated from earlier stages in the pipeline. The six cases where the algorithm did not identify the main verb group can be attributed to bad part-of-speech tags, bad chunk tags, or poor clause segmentation.

Teufel et al. (1999) do not explicitly use tense information in their heuristic categories. They also point out that their process of identifying indicator phrases is completely manual. Our integration of linguistic analysis techniques allows us to automate the availability of certain linguistic features we think will be useful in sentence extraction and rhetorical classification.

Our analysis not only makes available information about the tense of the main verb, but all the acquired annotation from intermediate steps: part-of-speech tags, chunk tags, clause structure, and tense information for all verb groups. To illustrate the utility of tense information, we will look at the relationship between our main rhetorical categories and simple present and past tense.

probably too strict an evaluation as like constituents tend to be coordinated meaning that the tense of a sentence can normally be identified from just one of the top-level main verb phrases.

[8] For the evaluation of main verb group identification, we ignore sentences that are not properly segmented (i.e. part of a sentence is missing or more material is included in a sentence than there should be). In these cases, the actual main verb group may or may not be present when the main verb identification algorithm is run. Sentence segmentation is an interesting problem in its own right. A state-of-the-art approach is included in our XML pipeline (Mikheev, 2002). Though we may get slightly better performance if we tailor the segmentation algorithm to our domain, in a random sample of 100 sentences, there were only 4 cases of bad segmentation.

	BACKGROUND	CASE	OWN
past	-0.135	0.356	-0.261
pres	0.105	-0.301	0.228

Table 5: Correlation between the categories in our basic rhetorical scheme and sentential tense information.

The correlation coefficient is a statistical measure of 'related-ness'. Values fall in the range $[-1.0, 1.0]$, where -1 means the variables are always different, 0 means the variables are not correlated, and 1 means the variables are always the same. Table 5 presents correlation scores between our basic rhetorical scheme and verb tense.

For illustrative purposes, we will focus on identifying the CASE rhetorical move. There is a moderate positive correlation between sentences determined to be past tense and sentences marked as belonging to the case rhetorical category. Also, present tense and the CASE rhetorical move have a moderate negative correlation. This suggests two features based on our linguistic analysis that will help a statistical classifier identify the CASE rhetorical move: (1) the sentence is past tense, and (2) the sentence is not present tense. Furthermore, comparing rows indicates that these are both good discriminative indicators. In the case of past tense, there is a positive correlation with the CASE rhetorical move while there is a very weak negative correlation with BACKGROUND and a slightly stronger negative correlation with OWN.

These results also illustrate the complexity of tense information. In order to identify simple past tense sentences, we look to see if the TENSE attribute of the main verb group has the value PAST, the ASPECT attribute has the value SIMPLE and the MODAL attribute has the value NO. Feature construction techniques offer a means for automatic discovery of complex features of higher relevance to a concept being learned. Employing machine learning approaches that are capable of modelling dependencies among features (e.g. maximum entropy) is another way to deal with this.

4 Conclusions and Future Work

The work reported forms the initial stages in the development of a automatic text summarisation system for judicial transcripts from the House of Lords. We have presented an initial annotation scheme for the rhetorical structure of the domain, assigning a label indicating the argumentative role of each sentence in a portion of the corpus. A number of sophisticated linguistic tools have been described that identify tense information. Finally, correlation scores were presented illustrating the utility of this information.

The next phase of the project will involve refining our annotation scheme. Once we have done this, we will create formal instructions and complete the annotation of the larger corpus. As part of the process of annotating our corpus, we will continue to examine possible indicators of the rhetorical role for a sentence.

We are also interested in improving the tools we use to identify tense features. One way to do this is retraining the clause identifier. The legal language of the HOLJ domain is considerably different than the expository newspaper text from the Penn Treebank. Furthermore, the Penn Treebank is American English. Ideally, we would like to hand-annotate a portion of the legal judgments with syntactic parse information and train a clause identifier from this. However, this kind of work is very labour intensive and a more realistic approach to ensuring that the training data is slightly more representative might be to retrain the clause identifier on a corpus of British English like the British National Corpus (Burnage and Dunlop, 1992).

Finally, as mentioned above, we are specifically interested in employing feature construction and selection techniques for identifying the relationship between tense features. We are also interested in employing feature mining techniques for automatically identifying cue phrases within sentences. This could be similar to (Lesh et al., 1999), where sequential features are mined from the textual context for a context-sensitive approach to spelling correction.

Acknowledgments

This work is supported by EPSRC grant GR/N35311.

References

Adam Berger, Stephen Della Pietra, and Vincent Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Gavin Burnage and Dominic Dunlop. 1992. Encoding the British National Corpus. In Jan Aarts, Pieter de Haan, and Nelleke Oostdijk, editors, *Design, Analysis and Exploitation, Papers from the 13th International Conference on English Language Research on Computerized Corpora*.

Lawrence Cheung, Tom Lai, Benjamin Tsou, Francis Chik, Robert Luk, and Oi Kwong. 2001. A preliminary study of lexical density for the development of xml-based discourse structure tagger. In *Proceedings of the 1st NLP and XML Workshop*, pages 63–70.

Claire Grover, Colin Matheson, Andrei Mikheev, and Marc Moens. 2000. Lt ttt—a flexible tokenisation tool. In *LREC 2000—Proceedings of the 2nd International Conference on Language Resources and Evaluation*, pages 1147–1154.

Ben Hachey. 2002. Recognising clauses using symbolic and machine learning approaches. Master's thesis, University of Edinburgh.

Neal Lesh, Mohammed Zaki, and Mitsunori Ogihara. 1999. Mining features for sequence classification. In *Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining*, pages 342–346.

M. Marcu, G. Kim, M. A. Marcinkiewicz, and R. MacIntyre. 1994. The penn treebank: annotating predicate argument structure. In *ARPA Human Language Technologies Workshop*.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

David McKelvie. 1999. Xmlperl 1.0.4 xml processing software. `http://www.cogsci.ed.ac.uk/~dmck/xmlperl`.

Andrei Mikheev. 1997. Automatic rule induction for unknown word guessing. *Computational Linguistics*, 23(3):405–423.

Andrei Mikheev. 2002. Periods, capitalized words, etc. *Computational Linguistics*, 28(3):289–318.

Marie-Francine Moens and Rik De Busser. 2002. First steps in building a model for the retrieval of court decisions. *International Journal of Human-Computer Studies*, 57(5):429–446.

Erik Tjong Kim Sang and Hervé Déjean. 2001. Introduction to the CoNLL-2001 shared task: clause identification. In *Proceedings of The 5th Workshop on Computational Language Learning*, pages 53–57.

Karen Spärck-Jones. 1998. Automatic summarising: factors and directions. In *Advances in Automatic Text Summarisation*, pages 1–14. MIT Press.

Simone Teufel and Marc Moens. 1997. Sentence extraction as a classification task. In *Workshop on Intelligent and scalable Text summarization*, pages 58–65. ACL/EACL.

Simone Teufel and Marc Moens. 1999. Argumentative classification of extracted sentences as a first step towards fexible abstracting. In *Advances in Automatic Text Summarization*, pages 137–175, New York. MIT Press.

Simone Teufel and Marc Moens. 2000. What's yours and what's mine: Determining intellectual attribution in scientific text. In *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 84–93.

Simone Teufel and Marc Moens. 2002. Summarising scientific articles- experiments with relevance and rhetorical status. *Computational Linguistics*, 28(4):409–445.

Simone Teufel, Jean Carletta, and Marc Moens. 1999. An annotation scheme for discourse-level argumentation in reserach articles. In *Proceedings of the 9th Conference of the European Chamber of the ACL*, pages 110–117. ACL.

Henry Thompson, Richard Tobin, David McKelvie, and Chris Brew. 1997. Lt xml. software api and toolkit for xml processing. `http://www.ltg.ed.ac.uk/software/`.