Large-scale, Parallel Automatic Patent Annotation

Milan Agatonovic, Niraj Aswani, Kalina Bontcheva, Hamish Cunningham, Thomas Heitz, Yaoyong Li, Ian Roberts, Valentin Tablan
Department of Computer Science
University of Sheffield
{Initial.Surname}@dcs.shef.ac.uk
[*]

ABSTRACT

When researching new product ideas or filing new patents, inventors need to retrieve all relevant pre-existing know-how and/or to exploit and enforce patents in their technological domain. However, this process is hindered by lack of richer metadata, which if present, would allow more powerful concept-based search to complement the current keyword-based approach. This paper presents our approach to automatic patent enrichment, tested in large-scale, parallel experiments on USPTO and EPO documents. It starts by defining the metadata annotation task and examines its challenges. The text analysis tools are presented next, including details on automatic annotation of sections, references and measurements. The key challenges encountered were dealing with ambiguities and errors in the data; creation and maintenance of large, domain-independent dictionaries; and building an efficient, robust patent analysis pipeline, capable of dealing with terabytes of data. The accuracy of automatically created metadata is evaluated against a human-annotated gold standard, with results of over 90% on most annotation types.

Categories and Subject Descriptors

H.3.1 [**Information Storage And Retrieval**]: Content Analysis and Indexing—*Linguistic processing*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*

General Terms

Experimentation, Measurement, Performance

Keywords

Patent Enrichment, Information Extraction, Parallel, Large-Scale, GATE

1. INTRODUCTION

Patents are an important vehicle for protecting intellectual property and this importance is increasing in the current globalised and knowledge based economy. When researching new product ideas or filing new patents, inventors need to retrieve all relevant pre-existing know-how and/or exploit and enforce patents in their technological domain. However, this process is hindered by lack of richer metadata, which if present, would allow more powerful concept-based search to complement the traditional keyword-based approach.

Semantic annotation is the task of attaching metadata tags and/or ontology classes to text segments, as a prerequisite for knowledge access and retrieval tools and user interfaces. Automatic annotation is carried out by employing Information Extraction (IE) [4] techniques, which recognise automatically mentions of a given set of events, entities or relationships. From an algorithmic perspective, IE approaches fall in two broad categories: manually engineered ones (frequently based on pattern-matching like rules) (e.g., [13]) and machine learning ones (e.g. [2, 11]). Rule-based approaches are more suitable where a carefully engineered, high precision system is needed and there is no sufficient training data for a machine learning approach to be successful. From an operational perspective, IE tools can be deployed in both fully and semi-automatic applications (where users can inspect and, if needed, correct the automatically created metadata). In general, fully automatic methods are preferred when the volume of data is too large to make human post-annotation feasible, as is the case with patents.

In particular, patent processing and search require high recall methods, capable of operating robustly on large-volumes of data. Previous research on IE has been carried out mostly on smaller datasets from narrower domains, mostly news articles [12, 2, 7], with accuracy results exceeding 85%. The new challenge addressed here is in scaling up these methods to deal with the diversity and volume of patent data, without sacrificing computational performance and accuracy levels.

Applications of information extraction to patent annotations are quite scarce. [9] mostly focus on OCR and text classification, while discussing only briefly the importance and challenges of identifying references to figures and claims in patents. In this area they have only carried out a small feasibility study using the Xerox language processing tools, without providing any evaluation figures or sufficient implementational details. More recently, the PatExpert project [16] has developed some content extraction components, ba-

[*]Author names are ordered alphabetically.

PaIR'08, October 30, 2008, Napa Valley, California, USA.

sed on deeper linguistic analysis than the approach proposed here. The advantages of shallow IE methods such as ours are that they are more robust in face of language variability and also scale better in terms of computational efficiency. The latter objective is particularly important in our case, as the requirement is to process efficiently terabytes of patents.

This paper presents our shallow IE approach to automatic patent enrichment, tested in large-scale, parallel processing experiments on USPTO and EPO documents. Section 2 starts by defining the patent annotation task and examines its challenges. The information extraction tools are presented next (Section 3), including details on automatic annotation of sections, references and measurements. The key challenges encountered were dealing with ambiguities and errors in the data; and creation and maintenance of large, domain-independent dictionaries. Section 4 is dedicated to the third, most significant challenge, i.e., building an efficient, robust patent analysis pipeline, capable of dealing with terabytes of data. The accuracy of automatically created metadata is evaluated against a human-annotated gold standard, with results of over 90% on most annotation types (see Section 5). This excellent performance is achievable due to the relatively constrained, legal sub-language used in patents, although we did encounter significant differences in the way American and European patents are phrased and structured and the literature references and measurement units are extremely variable. In the end we summarise our findings and discuss future work.

2. THE PATENT ANNOTATION TASK

The experiments in this paper are based on two kinds of patents – American (USPTO) and European (EPO) ones. The reason behind choosing two data sources is because they differ in terms of already provided metadata as XML tags, formatting, quality, and legal language used.

The semantic annotation process adds new metadata in the form of XML tags, which are also mapped to a patent-specific ontology, encoded in OWL [8]. We chose to support two different annotation formats to aid interoperability and also to enable the use of ontology-based semantic query tools such as KIM [14] and OWLIM [10].

The automatically added metadata falls into two broad categories: wide and deep annotation types. Wide annotations are intended to cover meta-data types that apply to patents in general and do not depend on the specific subject area of the patent (as identified, e.g., by its IPC code). Examples of such meta-data include document sections and references to cited literature, examples, figures, claims, and other patents. Deep annotations are specific to one or more subject areas and are of interest to specialised patent searchers. The experiments reported here focus on automatic annotation of measurements, as they are very important for patent professionals, while also being very hard to find using keyword search, due to the diverse ways in which they are expressed in language.

The benefits from the automatic metadata enrichment process are three-fold. Firstly, IE is capable of dealing with variable language patterns and format irregularities much easier than text-based regular expressions. For example, references to other patents can be very diverse: U.S. Patent 4,524,128, Korean laid open utility model application No. 1999-007692. Secondly, once we markup the relevant parts of the patent, the IE tools can also carry out data normalisation. Again, taking an example from references to figures or similarly claims, expressions such as "Figures 1-3" or "Claims 5-10" imply references not just to the explicitly mentioned figure/claim numbers but also to all those in between. Lastly, by using text mining techniques we are capable to extract a significantly wider range of useful information and provide it as additional XML tags in the patent documents.

2.1 Section annotations

Patent documents are typically quite long, contain multiple required sections, and use highly formalised legal and technical terminology with the notable exception of literature references and measurements. Different aspects of the patent application are typically presented in a pre-defined set of sections and subsections (e.g. prior art, patent claims, technical problem addressed and effect). Both USPTO and EPO documents have at least three main parts, *the first page containing* bibliographical data and abstract, *the descriptions part*, and *the claims part*.

Automatic section recognition is based on identifying typical section titles and then partitioning the text automatically based on that. Pre-existing section markup is used, if available. For instance, BibliographicData, Abstract and Claims sections tend to be already annotated in patent documents so we use them directly. As we distinguish about 20 different types of sections, most of these still need to be detected automatically (the complete list appears in table 4).

2.2 Reference annotations

Reference annotations are used for parts of text that refer to either objects in the current document (e.g. figures, tables, etc.) or to other documents (e.g. scientific papers).

A reference annotation consists of two parts, a header indicating the type of the reference, and one or more identifiers which typically consist of a mixture of numbers and letters. For example, in *Figure 1 and 2* the header is *Figure* and the identifiers are *1* and *2*. In *U.S. Pat. No.3,765,999* the header is *U.S. Pat.* and the identifier is *No.3,765,999.*

Conjunctive phrases mentioning references to two or more objects of the same reference type are tagged initially as one Reference annotation, including the conjunction and all punctuation. For example, *Figures 1 and 2*; *Claims 1-3*; *Tables 1 to 10* are first annotated as one Reference each, of type Figure, Claim and Table respectively. The normalisation step then separates these into their constituency references, also including all implied references (e.g., to claim 2).

From an IE perspective, some types of references are much simpler to identify than others. For instance, there is significantly less variability in the way patents refer to figures, tables, claims, equations, and examples. References to other patents tend to be slightly more challenging, as they often include the inventor names, patent date, or even title, in addition to a simple header and identifier. The hardest of all are references to external sources, such as published papers (e.g., Hudson & Hay, Practical Immunology (Blackwell Scientific Publications, Oxford, UK, 1980), Chapter 8), which tend to be quite long and typically contain many abbreviations and idiosyncratic formatting. We have also observed significant differences between American and European patents in this respect and had to adapt the IE tools to deal with that accordingly.

2.3 Measurements annotations

Most measurements comprise a scalar value followed by a unit, e.g. 2x10 -7 Torr. Furthermore, two scalar values with or without unit can be contained in an interval. Sometimes there are also accompanying words, such as “less than” or “between” which are important for professional searchers and are therefore also marked by the IE tools, e.g., “less than about 0.0015 mm”, “2 x 10 5 to 2 x 10 7 cpm/ml”. Lastly, we also deal with relative measurements, such as percentages and ratios.

The main challenge in recognising measurements in patents comes from the large number of measurement units in existence (e.g., units used in physics patents are very different to those used in engineering ones). Another challenge is that some units have single letter abbreviations, which introduce ambiguities in many cases and therefore the wider context needs to be considered in order to determine, whether the sequence of numbers followed by a letter is indeed a measurement. One frequently encountered example are temperatures, e.g., “1C” where we need to distinguish correct temperature mentions from other cases, such as references to figures, examples, tables, etc. (as in “see Figure 1C”).

3. OUR APPROACH

3.1 IE tools

We have developed our information extraction system using GATE[1] [5]. GATE, the General Architecture for Text Engineering, is a framework providing support for a variety of language engineering tasks. It includes a vanilla information extraction system, ANNIE, and a large number of plugins for various tasks and applications, such as ontology support, information retrieval, support for different languages, WordNet, machine learning algorithms, and so on. The processing resources we use from ANNIE are as follows: tokeniser, gazetteer and finite state transduction grammars. The resources communicate via GATE’s annotation API, which is a directed graph of arcs bearing arbitrary feature/value data, and nodes rooting this data into document content (in this case text).

The **tokeniser** splits text into simple tokens, such as numbers, punctuation, symbols, and words of different types (e.g. with an initial capital, all upper case, etc.), adding a “Token” annotation to each. It does not need to be modified for different applications or text types.

Our application developed its own, patent-specific **gazetteers** (list of expressions) that aid the recognition of measurements and references (see below). The lists are compiled into finite state machines, which can match text tokens.

The **semantic tagger** (or JAPE transducer) consists of hand-crafted rules written in the JAPE pattern language [6], which describe patterns to be matched and annotations to be created. Patterns can be specified by describing a specific text string or annotation (e.g. those created by the tokeniser, gazetteer, document format analysis, etc.).

3.2 Building the Gazetteers

Rule-based IE systems comprise of a set of grammar rules based on some patterns and clue words. For example to locate a reference to a table, one could use the clue word *table* followed by a number. The idea of using gazetteers is to annotate such clue words in the text with all their inflections.

One approach is to use a set of hand-annotated examples to derive such lists, however this requires a thorough corpus analysis. This is the approach we used to build gazetteers for locating the references, based on the gold standard corpus (see Section 5.1). The reference gazetteers are rather small in size, 314 elements in total, and contain clue words such as *Figure*, *Table* and *Example* to name a few. They also contain entries such as *described in* or *Patent application no.* to help locate literature and patent references.

In case of measurements, a database[2] containing more than 30K entries was used to automatically populate a gazetteer list. The database also contains transformation rules for transforming one measurement value into another (e.g., inches to cm). Since a gazetteer is simply a list of entries, the information about transforming rules has been populated in an ontology. These rules will be used for answering semantic queries by transforming values in one measurement unit into the other on the fly.

3.3 Creating the Annotation Rules

A typical JAPE rule consists of two parts: left hand side (LHS) and right hand side (RHS). LHS consists of an annotation pattern that should be matched in the text and RHS declares the action that should be taken when the pattern specified in LHS is found in the document. An example of such a pattern is given below:

```
Rule: FindANumberFollowedByAUnit
(
  {Number}
  {MeasurementUnit}
):match
-->
:match.Measurement = {}
```

The pattern, specified above, will try to locate a sequence of annotations where the first annotation is of type *Number* and the next annotations is of type *MeasurementUnit* – the latter being created on the basis of the measurement unit gazetteer. If such a pattern is found, the entire sequence is annotated as the *Measurement* annotation. In total, the application has over 30 JAPE rules that identify measurement units in the text. These include identification of complex equations and intervals of measurements as well.

The process of identifying sections and locating various references in the text consists of executing similar rules over the text. Annotations produced by other processing resources such as the gazetteers are used in the rules.

Similar techniques are used along with the annotations produced by various gazetteer lists to identify annotations of type *Reference*. Each reference is then classified into a subtype such as Figure, Formula or Table. Unlike these references where the used keywords are part of the actual reference tag, contextual information is needed for patent and literature references. For example a reference to a literature can have keywords such as *described in* or *according to* in their left contexts. Below we give an example of such a pattern:

[1] Gate is freely available for download from http://gate.ac.uk/

[2] http://www.gnu.org/software/units

Table 1: Application pipeline.

Phase	Gate processing resource
1	Section Finder
2	English Tokeniser
3	Patent-specific gazetteer
4	Reference Finder
5	Measurements Finder

```
Rule: Patent
(
  {PatentContext}
  ({PatStart}{PatentNumber}):match
):match-with-context
-->
:match.Patent = {}
```

Given this pattern, it will match with the string such as *described in U.S. Patent 4,524,128*, where *described in*, *U.S. Patent* and *4,524,128* are annotated as *PatentContext, PatStart* and *PatentNumber* respectively. However, only the part that matches with *({PatStart}{PatentNumber})* is annotated as the *Patent* reference.

The application has over 30 JAPE rules that identify measurement units in the text. This include identification of complex equations and intervals of measurements as well. As explained in the previous subsection, the measurement gazetteer is used for identifying measurement units in the text. For example, the following pattern would annotate the text such as 40-50mph where *40* and *50* are the two numbers and *mph* is the measurement unit.

```
Rule: MeasurementInterval
(
  {Number}
  {Token.string == "-"}
  {Number}
  {Unit}
):span
-->
:span.Measurement = { type = "interval" }
```

3.4 The Application

The application pipeline consists of a number of processing resources that are executed sequentially, where some components rely on the output of earlier ones. An example of this is the Reference finder resource that depends on the output of the gazetteer. Table 1 lists the resources in their order of execution in our application.

The pipeline is executed on one document at a time. Figure 1 shows an example of a processed document.

4. LARGE-SCALE EFFICIENCY TRIALS

4.1 Experiments

One of the most challenging tasks of any IE application is to adapt it to process a large amount of data without compromising on quality or performance. The main purpose of carrying out these experiments was to develop a highly optimised and equally accurate application that can exploit the hardware at its best. This is achieved by benchmarking the individual resources used in the application and by identifying those that need optimisation. In this section, we describe how we achieve this. We first describe the setup of our experiments, followed by some baseline results and some details on optimisation. Finally, we compare the baseline results with the optimised application results.

Table 2: Baseline Experiments.

Patent Type	No of Processes	KB/Sec	Time/Document USPTO or EPO
USPTO	1	8.06	10.54s
USPTO	4	29.95	2.84s
USPTO	8	53.15	1.60s
EPO	1	6.56	4.45s
EPO	4	27.41	1.08s
EPO	8	47.12	0.62s

The main purpose of the application is to consistently process large amount of patent data and produce metadata in the form of patent-specific annotations (i.e. sections, references and measurements) and other linguistic annotations (such as tokens, sentences etc.). In order to evaluate the consistency in the application's performance on a large dataset, experiments were carried out on a corpus consisting of 1.3 million USPTO (108GB) and 27 thousand EPO (780MB) documents in XML format with few attributes on each markup. The average sizes of USPTO and EPO documents were 85KB and 29.21KB respectively.

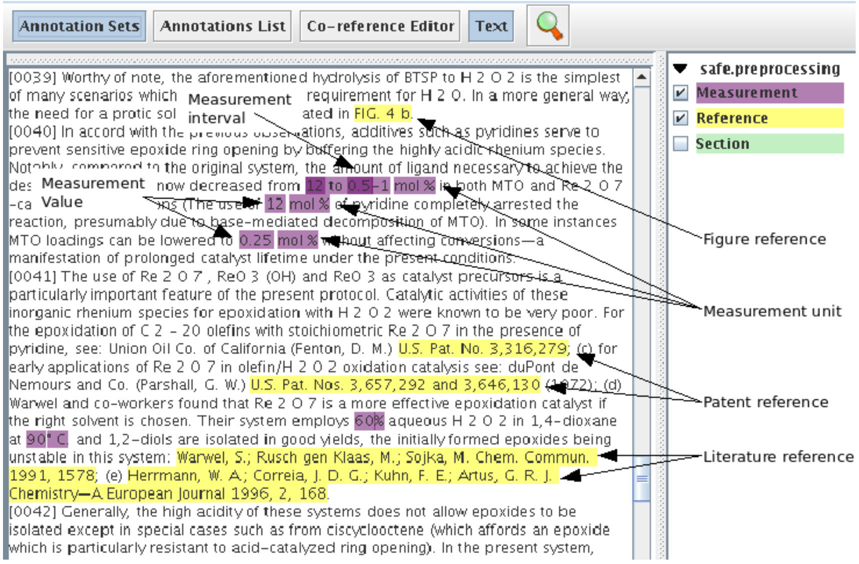
Our experiments were carried out on the IRF's Large Data Collier (LDC)[3]. This is an SGI Altix 4700 system comprising 20 processing nodes each with four 1.4GHz Itanium processor cores and 18GB RAM. The nodes are connected using SGI's high speed *NUMAlink* interconnect technology, allowing the whole cluster to appear as a single shared-memory system with a total of 80 processor cores and 360GB of main memory. Storage is provided via a fibre channel SAN.

In this environment, experiments were run with different numbers of processes running simultaneously. Table 2 gives details on the baseline experiments.

As shown in the table, the application was able to process 8.06KB per second when executed in a single process. In other words, it took a process 10.54 seconds to process a single USPTO document with an average size of 85KB. Whereas processing rates of 29.95 KB/Sec and 53.15 KB/Sec were observed for 4 and 8 parallel processes respectively. Even though the number of processes were increased to 4 and 8, the processing rate did not increase with the multiplication of the number of processes. Similar results can be seen for the EPO documents. The time taken to process one EPO document is bit less than the half of the time taken to process one USPTO document. The same is also true for the average sizes of USPTO and EPO documents. This indicates that the processing rate is linear as the size of the patent documents grows (see figure 2).

It was also a part of these experiments to benchmark the individual processing resources and report their data processing rates and their share in the overall time taken by the entire application. As specified earlier, the motive was to identify components which needed further optimisation. Having obtained results on 8 parallel processes, the interim changes were instantly applied to remove linguistic compo-

[3] http://www.ir-facility.org/the_irf/semantic-supercomputing



[0039] Worthy of note, the aforementioned hydrolysis of BTSP to H 2 O 2 is the simplest of many scenarios which requirement for H 2 O. In a more general way, the need for a protic sol ated in FIG. 4 b.
[0040] In accord with the previous observations, additives such as pyridines serve to prevent sensitive epoxide ring opening by buffering the highly acidic rhenium species. Notably, compared to the original system, the amount of ligand necessary to achieve the des now decreased from 12 to 0.5-1 mol % in both MTO and Re 2 O 7 -ca ns (The use of 12 mol % of pyridine completely arrested the reaction, presumably due to base-mediated decomposition of MTO). In some instances MTO loadings can be lowered to 0.25 mol % without affecting conversions—a manifestation of prolonged catalyst lifetime under the present conditions.
[0041] The use of Re 2 O 7 , ReO 3 (OH) and ReO 3 as catalyst precursors is a particularly important feature of the present protocol. Catalytic activities of these inorganic rhenium species for epoxidation with H 2 O 2 were known to be very poor. For the epoxidation of C 2 - 20 olefins with stoichiometric Re 2 O 7 in the presence of pyridine, see: Union Oil Co. of California (Fenton, D. M.) U.S. Pat. No. 3,316,279; (c) for early applications of Re 2 O 7 in olefin/H 2 O 2 oxidation catalysis see: duPont de Nemours and Co. (Parshall, G. W.) U.S. Pat. Nos. 3,657,292 and 3,646,130 (1972); (d) Warwel and co-workers found that Re 2 O 7 is a more effective epoxidation catalyst if the right solvent is chosen. Their system employs 60% aqueous H 2 O 2 in 1,4-dioxane at 90° C. and 1,2-diols are isolated in good yields, the initially formed epoxides being unstable in this system: Warwel, S.; Rusch gen Klaas, M.; Sojka, M. Chem. Commun. 1991, 1578; (e) Herrmann, W. A.; Correia, J. D. G.; Kuhn, F. E.; Artus, G. R. J. Chemistry—A European Journal 1996, 2, 168.
[0042] Generally, the high acidity of these systems does not allow epoxides to be isolated except in special cases such as from ciscyclooctene (which affords an epoxide which is particularly resistant to acid-catalyzed ring opening). In the present system,

safe.preprocessing
☑ Measurement
☑ Reference
□ Section

Figure 1: Annotated patent document in GATE GUI.

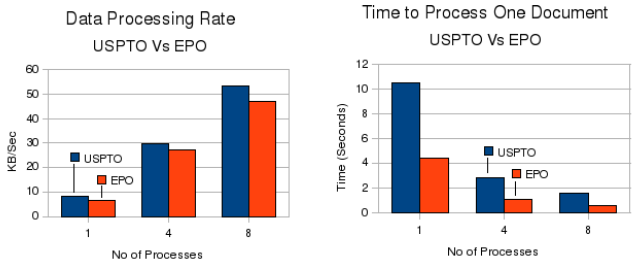


Figure 2: Baseline Results.

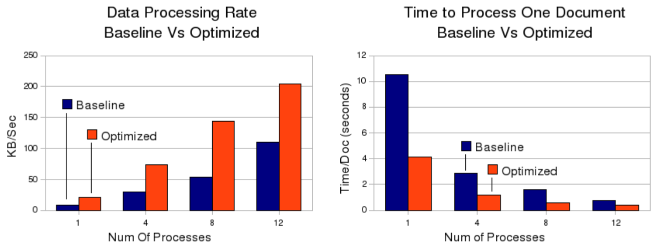


Figure 3: Baseline Vs Optimised.

nents that were identified as expensive and did not have significant impact on the overall results, such as, ANNIE's morphological analyser and named entity recogniser. Although this resulted in a slightly reduced number of linguistic annotations, it did not affect the automatic processing of patent-specific annotations. As a result, when the same experiment was performed with 12 parallel processes, the processing rate of 110.03 KB/Sec was achieved which is slightly less than double the processing rate for 8 parallel processes. Similarly the time taken to process one document decreased from 1.60 seconds to 0.77 seconds per document.

The patent processing application contains several linguistic components that depend on the output of the previous stages. One such example is the grammar that identifies measurement units, which requires word boundaries to be already annotated by the tokeniser. The benchmarking tool was also used to identify expensive resources, which are needed by subsequent grammars. Such resources were then either optimised or completely removed after refactoring the depending grammars. For example, instead of using a morphological analyser, the rules were instead modified to match all word forms under consideration.

Having thus optimised the application, the experiments on all 1.3 million documents were repeated to collect new benchmark results. Table 3 shows a comparison between the baseline results and the results obtained for the optimised application. Figure 3 explains these results through graphs.

Table 3: Baseline Vs Optimised Application.

No of	KB/seconds		seconds/Doc	
processes	Baseline	Optimised	Baseline	Optimised
1	8.06	20.7	10.54	4.11
4	29.95	73.86	2.84	1.15
8	53.15	144.44	1.6	0.59
12	110.03	203.76	0.77	0.42

As shown in table 3, the processing rates for 1, 4 and 8 parallel processes are almost 2.5 times higher, whereas with 12 processes the optimised application performes 1.8 times better. This is most likely due to the fact that the entire LDC system is a cluster of nodes where the memory and input/output operations are being shared among the processor nodes. With 12 processes (in comparison to the fewer 8 processes), this means further segregation of the shared LDC resources.

After automatic annotation the 1.3 million documents require around 139 GB when stored as stand-off XML files plus content text files. The average file size is 85 KB before and 113 KB (standoff and content) after processing.

Finally, to summarise the results, the baseline application took 264 hours (11 days) to process 1.3 million USPTO documents at the processing rate of 110.03 KB/Sec, whereas the optimised application took 142 hours (5.92 days) to process the same number of documents at a processing rate of 203.76 KB/Sec. Given that the hardware has 80 processor nodes in total, the overall speed can be improved even further.

In order to be able to estimate the number of annotations that the application produces per document, 20 documents (both from the USPTO and EPO documents) were obtained at random. These contained 147 section annotations, 604 measurements, 1,351 references and 150,140 linguistic annotations. Based on these results, it would be reasonable to estimate that each document in our corpus contains an average of 105 end-user (patent specific) annotations and 7507 linguistic annotations.

5. ACCURACY EVALUATION

5.1 Gold standard

Patent documents typically contain one or more IPC codes indicating the nature of the invention, so we used these codes to select patents relating to particular technology categories. We selected the patents from two very different fields, mechanical engineering and biomedical technology, to better examine the diversity in the data.

We manually annotated some USPTO documents and EPO documents in several iterations, in order to test the annotation user interface, refine the annotation definitions, and evaluate the automatic processing. The evaluation corpus used in this paper consists of 23 USPTO and 28 EPO documents. The reason behind annotating more EPO than USPTO documents is that the latter tend to be more uniform in terms of formatting, language used, sectioning, etc.

We consider that a corpus of 51 documents, 376,713 words or 2,490,666 characters is big enough to test our system. For the section annotations, in addition we also extracted all heading annotations from the 1.3 millions documents and checked that the most frequent ones are found by our system.

In order to ensure more consistent gold standard data, manual annotation was carried out in two passes. First two annotators would markup each patent documents independently and then an expert checked the two sets and corrected any disagreements and missed annotations. In total, more than 10 different human annotators were involved. Throughout the process we also controlled the inter-annotator agreement, which ensures good quality of the human-annotated data.

5.2 Results on the Gold Standard

Accuracy evaluation is an essential part of the development of information extraction applications and is carried out by comparing the annotations produced by the automatic system against those in the gold standard.

The reported results make use of traditional evaluation metrics for information extraction [3]: precision, recall, and F-measure. Precision measures the number of correctly identified items as a percentage of the number of items identified. It measures how many of the items that the system identified were actually correct, regardless of whether it also failed to retrieve correct items. The higher the precision, the better the system is at ensuring that what is identified is correct. Recall measures the number of correctly identified items as a percentage of the total number of correct items measuring how many of the items that should have been identified actually were identified. The higher the recall rate, the better the system is at not missing correct items. The F-measure [15] is often used in conjunction with Precision and Recall, as a weighted average of the two – usually an application requires a balance between Precision and Recall.

Table 4: Corpus statistics. USPTO contains 23 documents and EPO contains 28 documents.

Annotation type	USPTO	EPO
Section.Abstract	23	28
S.BackgroundArt	19	22
S.BestMode	2	5
S.BibliographicData	23	28
S.Bibliography	0	8
S.Claims	23	0
S.CrossReferenceToR.A.	6	1
S.DetailedDescription	11	18
S.DisclosureOfInvention	3	6
S.DrawingDescription	16	20
S.Effects	1	2
S.Examples	17	25
S.PreferredEmbodiment	10	7
S.PriorArt	4	6
S.Sponsorship	2	0
S.SummaryOfTheInvent.	20	18
S.TechnicalField	14	17
S.UsageOfInvention	1	6
Annotations/Doc	8.5	8
Reference.Claim	352	2
R.Example	99	264
R.Figure	375	570
R.Formula	79	66
R.Literature	114	488
R.Patent	92	182
R.Table	59	105
Annotations/Doc	51	60
Annotations/1000 Char	1.4	1
M.scalarValue	1998	3409
Measurement.unit	1613	2994
M.interval	432	375
Annotations/Doc	176	242
Annotations/1000 Char	4.9	4
Characters per document	827,294 35,969	1,663,372 59,406

Overall, the evaluation figures obtained are above 85%, which makes them suitable for immediate deployment and use in end-user applications (see Table 5). The main exception are references to other patents and external publications/literature, where the results are not yet final, as the rules are still under development. Concerning the recognition of measurement intervals in the 28 EPO documents, the lower results can be explained by their more versatile forms that are not easy to generalise.

In the case of measurements, for comparison's sake, the highest results obtained in the Matrixware TempRanger experiment[4] were 75.51% precision and 88.48% recall, while only identifying temperature expressions. Therefore, our measurement grammars not only achieve higher performance, but it also captures a much wider range of measurements.

Table 5: Evaluation figures per annotation type on the USPTO 23 documents and EPO 28 documents gold standard for micro-averaged precision, recall and F1-score.

Annotation type	USPTO			EPO		
	P.	R.	F1	P.	R.	F1
S.BackgroundArt	74	74	74	56	68	61
S.DrawingDescr.	75	75	75	84	80	82
Section.Examples	65	65	65	61	56	58
S.SummaryOf.	89	80	84	83	83	83
S.TechnicalField	80	57	67	94	94	94
Reference.Claim	100	100	100	100	100	100
R.Example	97	100	99	100	99	99
R.Figure	99	99	99	99	98	98
R.Formula	99	99	99	100	100	100
R.Literature	69	75	72	70	74	72
R.Patent	76	77	77	72	84	78
R.Table	100	98	99	100	100	100
M.scalarValue	96	93	94	94	92	93
Measurement.unit	95	92	93	94	93	93
M.interval	93	92	93	82	81	82

5.3 Comparison against Pre-Existing Markup

There is some pre-existing markup for some kinds of references and sections in the USPTO documents (there is no such markup in the EPO documents that we have at present). This enabled us to also compare our automatically created annotations against those already in the data.

At present, there is pre-existing markup only for two kinds of references, those to figures and claims, therefore these are the only ones we can compare on. Table 6 presents the precision, recall and F1 results, when the pre-existing markup is used as a gold standard. It can be seen that most pre-existing references to figures and claims are identical to those obtained by our grammars, in particular for the claim reference. However, there are quite a few different annotations between the two for figure references.

Through manual inspection, we found that figure references are more complicated than that for claim references. While most pre-existing annotations are correct for the simpler cases, e.g., “FIG. 1” and “FIG. 5A”, there are many mistakes for more complex references such as “FIGS. 5A to 5 D” (only “FIGS. 5A” is tagged while 5D is missed), “FIGS. 4B, a device” (tagged wrongly as “FIGS. 4B, a”), and “FIG. 4 a” (only “FIG. 4” is tagged). In contrast, our grammars produced the correct annotations in such cases. There are also slight differences in the way conjunctions are tagged, where pre-existing and automatically produced annotations differ slightly, consequently lowering the accuracy results.

To summarise, for reference identification it is better to rely entirely on those produced by our system, because on one hand, there are mistakes and inconsistencies in the pre-existing markup, and on the other, the claim and figure references produced by our system are very reliable (see the respective precision and recall in Table 5.

Table 6: Comparing the pre-processing results (as the key set) with those in the original markups (as the response set) for the two types of reference on the 23 USPTO documents: the micro-average over the two sets of documents for the F-measures (Precision, Recall, F1) for each reference type.

Annotation type	Precision	Recall	F1
Reference.Claim	95	97	96
Reference.Figure	91	87	88

We also carried out a similar analysis of the section tags. There the pre-existing markup covers section types such as cross-reference-to-related-applications, summary-of-invention and detailed-description. We found that the first one is quite reliable but the other two are less so. For example, one detailed-description annotation covers also the section “BEST MODE FOR CARRYING OUT THE INVENTION”, while another one covers not only the correct section “DETAILED DESCRIPTION OF THE INVENTION” but also the examples section. Yet another example is a summary-of-invention annotation covering the sections for technical field, background and summary. While this might not be problematic for some applications, our goal is to provide as detailed and fine grained section identification as possible, in order to enable users to search only within the desired parts. Consequently, our automatic approach currently uses only the cross-reference-to-related-applications tags, plus the annotations for the sections containing bibliographic data, abstract, and claims.

6. CONCLUSION

This paper presented a large-scale, parallel IE system to automatically annotate USPTO and EPO documents with relevant new metadata, in order to enable richer searches by inventors and companies. This system has been tested on a 1.3 million documents corpus (more than 100 GB) with 12 parallel processes and achieves in its optimised version a data rate of 200 KB/seconds or less than 6 days of processing. Processing is highly parallelisable and the overall time can be reduced further by using more servers.

Our system supports not only batch-mode automatic annotation but can also be used with an interface to check/correct the annotations and to search using the newly generated tags. Notably, ANNIC [1] a tool present in GATE, allows us to input semantic queries such as *Find all length measurements.* We are in the process of extending the possibilities to queries like *Find a measurement greater than 10 cm* or *Find a measurement in the description section.* We already have an ontology of units that is linked to the measurement annotations, which will allow us to normalise all measurements in order to make semantic queries a lot more efficient and also independent of the measurement unit used (e.g., a query for inches can retrieve patents with units in cm).

Our system can also be deployed with a web interface that allows a large number of annotators to correct the automatic

[4] These unpublished results were provided to us in email communication with Matrixware staff.

annotations. This is done in order to create a gold standard that will be used for machine learning and also to allow further evaluation of the rule-based processing components. As machine learning IE applications can be deployed only when sufficient data has been annotated, this has not been possible without first developing the rule-based system describe in this paper. Another strand of our future work will be to create an efficient machine learning system to improve further the precision and recall on all annotation types and introduce new ones.

7. ACKNOWLEDGMENTS

This work has been supported by a Matrixware/IRF research grant.

8. REFERENCES

[1] N. Aswani, V. Tablan, K. Bontcheva, and H. Cunningham. Indexing and Querying Linguistic Metadata and Document Content. In *Proceedings of Fifth International Conference on Recent Advances in Natural Language Processing (RANLP2005)*, Borovets, Bulgaria, 2005.

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

[3] N. Chinchor. Muc-4 evaluation metrics. In *Proceedings of the Fourth Message Understanding Conference*, pages 22–29, 1992.

[4] H. Cunningham. Information Extraction, Automatic. *Encyclopedia of Language and Linguistics, 2nd Edition*, pages 665–677, 2005.

[5] 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 (ACL'02)*, 2002.

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

[7] D. Day, P. Robinson, M. Vilain, and A. Yeh. MITRE: Description of the *Alembic* System Used for MUC-7. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998.

[8] M. Dean, G. Schreiber, S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. OWL web ontology language reference. W3C recommendation, W3C, Feb 2004. http://www.w3.org/TR/owl-ref/.

[9] D. Hull, S. Ait-Mokhatar, M. Chuat, A. Eisele, E. Gaussier, G. Grefenstette, P. Isabelle, C. Samuelsson, and F. Segond. Language technologies and patent search and classification. *World Patent Information*, 23:265–268, 2001.

[10] A. Kiryakov. OWLIM: balancing between scalable repository and light-weight reasoner. In *Proc. of WWW2006*, Edinburgh, Scotland, 2006.

[11] Y. Li, K. Bontcheva, and H. Cunningham. SVM Based Learning System For Information Extraction. In M. N. J. Winkler and N. Lawerence, editors, *Deterministic and Statistical Methods in Machine Learning*, LNAI 3635, pages 319–339. Springer Verlag, 2005.

[12] D. Maynard, K. Bontcheva, and H. Cunningham. Towards a semantic extraction of Named Entities. In *Recent Advances in Natural Language Processing*, Bulgaria, 2003.

[13] 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*, pages 257–274, Tzigov Chark, Bulgaria, 2001.

[14] B. Popov, A. Kiryakov, D. Ognyanoff, D. Manov, and A. Kirilov. KIM – A semantic platform for information extraction and retrieval. *Natural Language Engineering*, 10:375–392, 2004.

[15] C. van Rijsbergen. *Information Retrieval.* Butterworths, London, 1979.

[16] L. Wanner, R. Baeza-Yates, S. Brugmann, J. Codina, B. Diallo, E. Escorsa, M. Giereth, Y. Kompatsiaris, S. Papadopoulos, E. Pianta, G. Piella, I. Puhlmann, G. Rao, M. Rotard, P. Schoester, L. Serafini, and V. Zervaki. Towards Content-oriented Patent Document Processing. *World Patent Information*, 30(1):21–33, 2008.