Extracting Person Names from Diverse and Noisy OCR Text

Thomas Packer, Joshua Lutes, Aaron Stewart, David Embley, Eric Ringger, Kevin Seppi
Department of Computer Science
Brigham Young University
Provo, Utah, USA
tpacker@byu.net

Abstract

Named entity recognition from scanned and OCRed historical documents can contribute to historical research. However, entity recognition from historical documents is more difficult than from natively digital data because of the presence of word errors and the absence of complete formatting information. We apply four extraction algorithms to various types of noisy OCR data found "in the wild" and focus on full name extraction. We evaluate the extraction quality with respect to hand-labeled test data and improve upon the extraction performance of the individual systems by means of ensemble extraction. We also evaluate the strategies with different applications in mind: the target applications (browsing versus retrieval) involve a trade-off between precision and recall. We illustrate the challenges and opportunities at hand for extracting names from OCRed data and identify directions for further improvement.

1 Introduction

Information extraction (IE) facilitates efficient knowledge acquisition for the benefit of many applications. Perhaps most importantly, IE from unstructured documents allows us to go beyond now-traditional keyword search and enables semantic search. Semantic search allows a user to search specifically for only those instances of an ambiguous name that belong to a semantic type such as *person* and to exclude instances of other entity types. By extracting information from noisy OCR data we aim to broaden the impact of IE technology to include printed documents that are otherwise inaccessible to digital tools. In particular, we are interested in books, newspapers, typed manuscripts, printed records, and other printed documents important for genealogy, family history and other historical research.

The specific task we target in the present study is the extraction of person names from a variety of types and formats of historical OCR documents. This task is an example of named entity recognition (NER) as described in (Nadeau and Sekine, 2007) and (Ratinov and Roth, 2009). Accurately and efficiently identifying names in noisy OCR documents containing many OCR errors presents a challenge beyond standard NER and requires adapting existing techniques or tools. Our applications of interest are search and machine-assisted browsing of document collections. Search requires names to be pre-identified and indexed. Machine-assisted browsing of document collections has greater tolerance for misidentified names.

There has been little published research on named entity extraction from noisy OCR data, but interest in this field is growing. Recent work by Grover et al. uses hand-written rules on two kinds of British parliamentary proceedings (2008). Earlier work by Miller et al. (2000) uses an HMM extractor on matched conditions: for their OCR task, they printed digital documents and scanned and OCRed the resulting copy to produce the OCR data for both training and test sets. To our knowledge, no published research targets the full extent of noisiness and diversity present in some real corpora or compares

competing NER techniques on the same OCR corpus.

In starting such a project, we had several questions: What variation of word error rate (WER) can be expected over multiple OCR engines and types of documents? What level of NER quality is achievable in a couple of months of development time, particularly when no annotated data is available for the corpus for training or evaluation purposes? How well can we do on a truly noisy and diverse corpus of OCR data? How do competing extraction approaches compare over different document types? Can improvements in extraction quality be gained by combining the strengths of different extractors?

We provide answers to these questions in the following sections. In §2 we describe the data we used as well as the names extracted. In §3 we present each of the basic extraction methods and examine their performance. In §4 we present a straightforward ensemble method for combining the basic extraction methods and show an improvement in performance over each of the component extractors. Finally, we conclude and discuss future work (§5).

2 Data and Task

The data used as input to our named entity recognizers is the OCR output for 12 titles spanning a diverse range of printed historical documents with relevance to genealogy and family history research. These documents are described in table 1. To the best of our knowledge, this collection has greater variety in formatting and genre than any other image-and-text corpus used in a published NER experiment. The data includes unstructured text (full sentences), structured (tabular) text including long lists of names and end-of-book indexes, and multi-column formatted text from the books and newspapers.

2.1 OCR

Three OCR engines were used in the production of the data used in this study. PrimeOCR, a commercial voting system utilizing six OCR engines, selects the best results from those engines (PrimeRecognition, 2009). Abby is a version of Abby FineReader used within an OCR engine produced by Kofax (Kofax, 2009). The newspapers were OCRed by an engine that was not identified by the corpus owner.

Examples of images and corresponding OCR output are given in figures 1 and 2. Figure 1 is an example of one of the poorer quality images and accompanying OCR output. Causes of poor quality are dark splotches in the noisy image and the fact that the OCR engine failed to recognize column boundaries during zoning. In figure 2, letter-spacing is incorrectly interpreted by the OCR engine, resulting in the introduction of superfluous spaces within words. This figure also illustrates the common problem of words that are split and hyphenated at line boundaries as well as other types of errors.

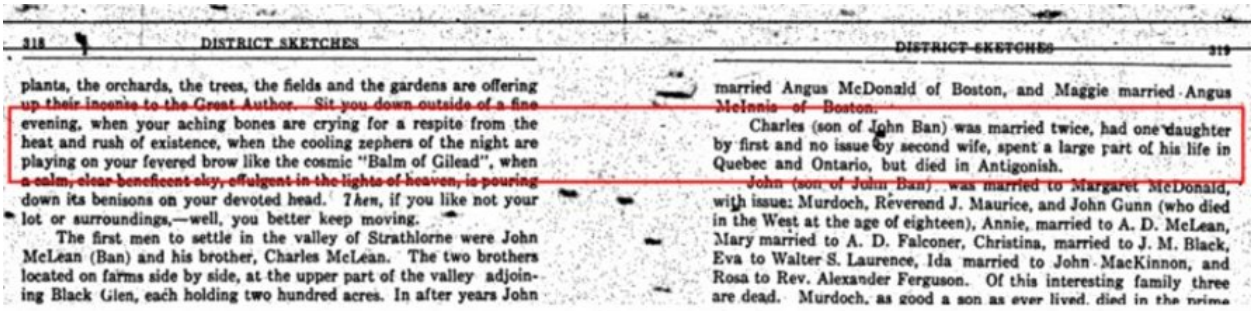
This particular collection of OCR documents were originally intended to be indexed for keyword search. Because the search application requires no more than a bag-of-words representation, much of the document structure and formatting, including punctuation and line boundaries in many cases, were discarded before the data was made available, which affects the quality of the documents with respect to NER. Furthermore, in parts of some of the documents, the original token ordering was not preserved consistently: in some cases this was caused by the OCR engine being unable to first separate columns into distinct sections of text, while in other cases this was more likely caused by the noisiness and poor thresholding (binarization) of the image. The quality of the original images on which OCR was performed varied greatly. Consequently, this corpus represents a very noisy and diverse setting for extracting information.

2.2 Annotation

Our task is the extraction of the full names of people, e.g., "Mrs Herschel Williams", from OCRed documents. Since the corpus was not originally intended as a public benchmark for NER, the pages used for development test and blind test data were hand-annotated for the current project. One to two pages from each document were annotated for each of the development test and blind test sets. The annotations consisted of marking person names, including titles. The number of names annotated in the blind test set is given in table 1 for each document in the corpus. Blind test pages were not inspected during the development of the extraction systems. All extractors, including ensembles, were applied to the

Title and Years	Genre	Engine	N	WER	Fc	Ff
Birmingham, Alabama; 1888-1890	City Directory	Abby	23	53	35	61
Portland, Oregon; 1878-1881	City Directory	Abby	69	21	44	55
Year Book of the First Church of Christ in **Hartford**; 1904	Church Year Book	Prime	13	28	38	37
The **New York** Church Year Book; 1859-60	Church Year Book	Prime	26	46	47	62
The **Blake** Family in England; 1891	Family History	Prime	0	NA	NA	NA
The **Libby** Family in America; 1602-1881	Family History	Prime	24	85	32	42
History and Genealogy of the Families of Old **Fairfield**	Local History	Abby	52	28	64	89
History of **Inverness** County, Nova Scotia	Local History	Abby	9	75	15	28
United States Ship **Ajax**; 1980	Navy Cruise Book	Abby	0	NA	NA	NA
United States Ship **Albany**; 1962-1964	Navy Cruise Book	Abby	9	114	0	34
Montclair Tribune; 1967-1968	Newspaper	Unk.	174	15	64	68
The **Story City** Herald; 1955	Newspaper	Unk.	91	92	44	55
Over all			490	56	38	53

Table 1: A summary of the documents used. In the first column, the nickname used throughout this paper is shown in bold. **Engine** = the OCR engine used for each title. The following numbers refer to the blind test set. **N** = number of person names annotated, **WER** = word error rate (percent) for OCR output, averaged over pages if more than one page, **Fc** = Coarse-grained F-measure for coarse-grained Majority Ensemble, **Ff** = Fine-grained F-measure for course-grained Majority Ensemble. Ensembles are defined in §4. Though there were no names annotated for the blind test set in Blake and Ajax, they are included above for completeness (they contributed to the training and development test sets).




```
evening when your aching bones are crying for a respite from the Charles son of hn Ban
was married twice had oneaught heat and rush of existence when the cooling zephers of the
night are by first and no issue y second wife spent a brow like large part of his playing
on your fevered the cosmic r Balm of Gilead life in quo Quebec and died a calm clear
```

Figure 1: Example of poor quality data found in Inverness. Names to be extracted are “Charles” and “John Bahn”.

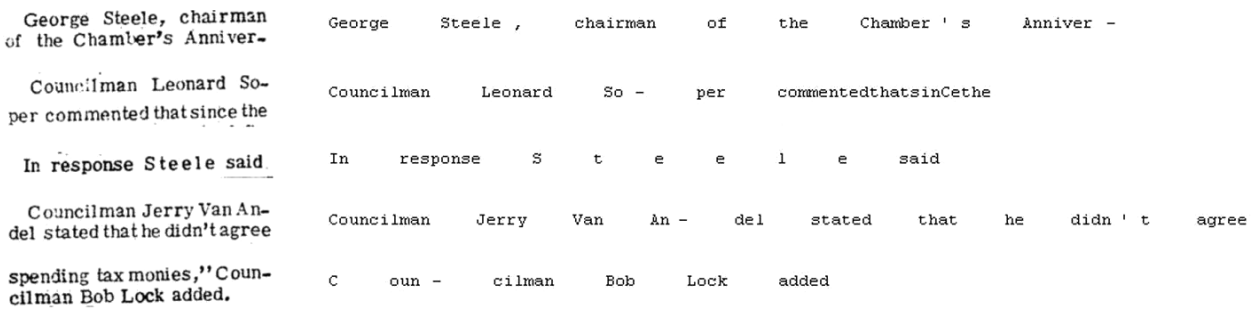


Figure 2: Pairs of image and corresponding OCR text from one page of Montclair.

same pages. When variations in individual systems were considered, the options which performed best on development test data were selected and executed on the blind test data, with scores for blind test data reported.

The OCR text in our corpus was sufficiently noisy to necessitate labeling guidelines that accommodate the errors. On the one hand, we considered labeling only named entities that appeared correctly in the OCR text; on the other hand, we considered labeling all named entities occurring in the original images. In the end, we settled on a middle ground to accommodate some character recognition errors: any token having a character error rate above 50% was excluded from annotation. In this, we attempted to balance the negative impact of removing too many tokens which could legitimately be identified by some named entity recognizers based solely on context and the negative impact to the real application of the extraction, which in our case was a name search engine index. Such an index would likely grow unnecessarily large if it were filled with garbled names for which users are unlikely to search or which are sufficiently dissimilar to real names.

2.3 Metrics

Precision, recall and F-measure scores were calculated for person names in both a coarse and fine manner. The coarse-grained metrics score extractor output in an all-or-none manner: they count an extracted full name as correct only if it matches a full name in the hand-labeled test set (including token positions/IDs). Using the above example, if an extractor misses the title "Mrs" and labels only "William Herschel" as a full name, then this is considered as one false positive (since "William Herschel" is not found among the manual annotations) *and* one false negative (since "Mrs William Herschel" is not found among the extractor's output). Thus this one mistake counts against both precision and recall.

The fine-grained metrics are more forgiving and would be more appropriate for a document browsing application as opposed to searching for a complete name. They will give partial credit if any part of a name is recognized because they look for matches between the individual tokens in the hand annotated data and the extracted data. Continuing the example above, the extractor that recognizes only "William Herschel" as a full name will have two true positives, one false negatives and no false positives.

These two metrics partially acknowledge the same issues addressed by the MUC evaluation metrics described in (Nadeau and Sekine, 2007) in which evaluation is decomposed into two complementary dimensions: TEXT and TYPE.

3 Basic Extraction Methods and Results

We built four person name recognizers while exploring possible adaptations of existing named entity recognition methods to the genre and especially the noisiness of our corpus. The work required a couple of months, with each extractor being built by a different researcher. The extractors are designated as dictionary-based, regular expression based, MEMM (maximum-entropy Markov model) and CRF (conditional random field). In these four extractors, we are comparing solutions from two competing disciplines for NER: the hand-written, rule-based approach and the supervised machine learning approach.

We applied them individually and collectively (within the ensemble extractors) on the blind test data and report a summary of their results in figures 3 and 5 (coarse metrics), and 4 and 6 (fine metrics). Only the results for the coarse-grained ensembles are reported in these four figures.

3.1 Dictionary-Based Extractor

The dictionary extractor is a simple extractor, intended as a baseline and requiring about 20 to 30 hours to develop. It identifies any token as part of a name if it is found in a case-sensitive name dictionary. It then combines each contiguous sequence of name tokens into a full name if it meets a few constraints. The following constraints were developed manually while inspecting the names in a few of the pages in the "training" (unlabeled, non-test) data: a name must either contain one or more tokens that are not initials or must contain exactly two initials (for partial credit when only two initials can be identified). A name must also consist of five or fewer tokens.

Name dictionaries include the following, collected from online sources: a given name dictionary

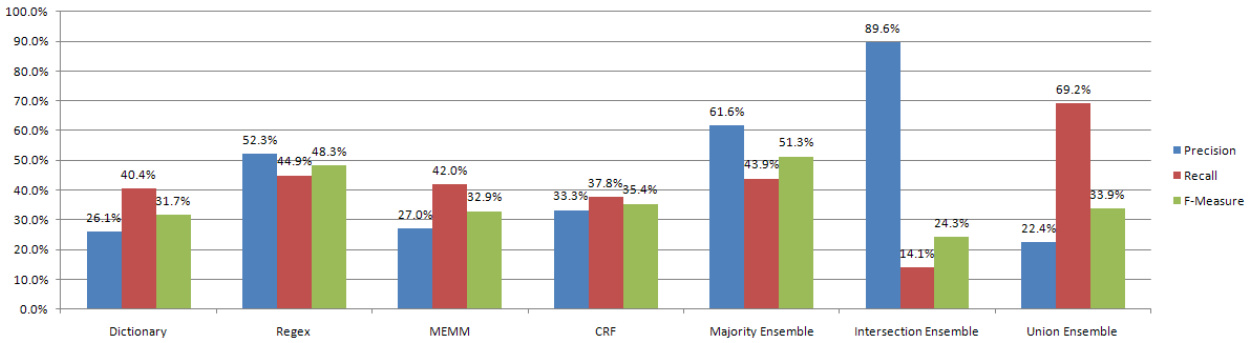


Figure 3: Coarse-grained precision, recall and F-measure for person names in the blind test set.

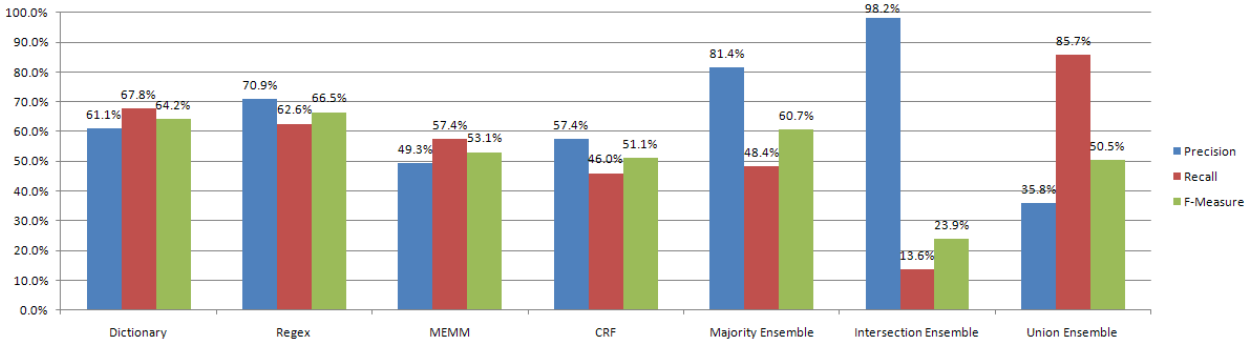


Figure 4: Fine-grained precision, recall and F-measure for person names in the blind test set.

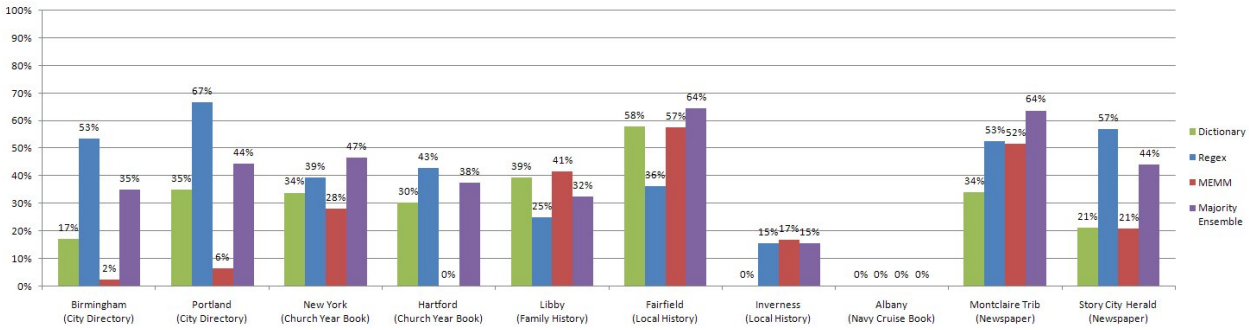


Figure 5: Coarse-grained F-measure for person names for each title in the blind test set.

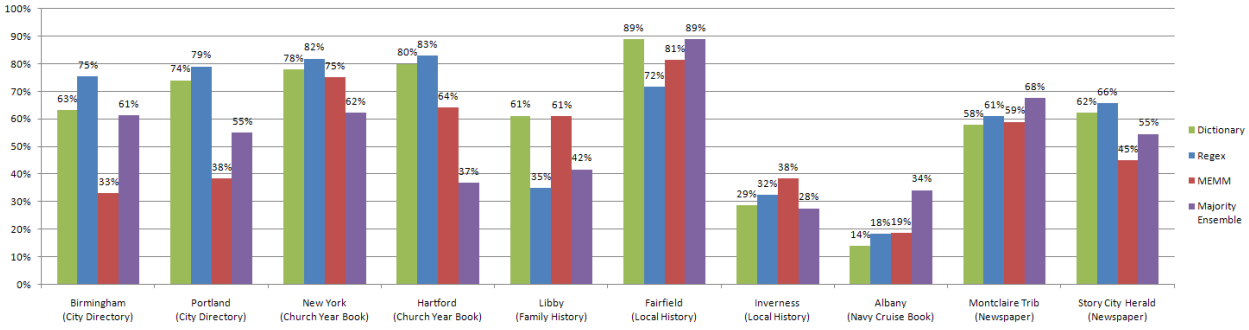


Figure 6: Fine-grained F-measure for person names for each title in the blind test set.

(18,000 instances), a surname dictionary (150,000 instances), a list of common initial letters (capital letters A through W) and a list of titles (10 handwritten instances including "Mr" and "Jr").

The surname dictionary was pruned by sorting the original list by an approximation of $P(label = Surname|word)$ computed automatically from statistics collected from a corpus of web pages and then removing the low-scoring words from the dictionary below a cut-off that was determined by maximizing extraction accuracy over the development test (validation) set.

3.2 Regular Expression Rule-Based Extractor

The regular expression rule-based (Regex) extractor was based on the Ontology-based Extraction System (OntoES) of Embley et al.(Embley et al., 1999). OntoES was designed to extract a variety of information from terse, data-rich, structured and semi-structured text found in certain types of web pages such as car sale ads. In the current work, we adapt OntoES to work with noisy, unstructured text and therefore do not make use of many of its features associated with conceptual modeling and web page structure. Like the dictionary-based extractor, the Regex extractor also uses dictionaries to recognize tokens that should be considered components of a person name. Matching of entries in these dictionaries is stage-wise case-sensitive. By this we mean that the extractor first finds matching tokens in a case-sensitive manner. Then for each page in which a dictionary entry is found, the extractor looks for case-insensitive matches of that word. The Regex extractor then labels any token pattern as a full name wherever one of the following regular expression patterns is found. Note that the patterns are described in Perl5 regular expression syntax.

optional title, given name, optional initial, surname:

```
\b({Title}\s+){0,1}({First})\s+
([A-Z]\s+){0,1}{Last}\b
```

title, surname:

```
\b{Title}\s+{Last}\b
```

title, capitalized words:

```
\b{Title}([A-Z][A-Za-z]*){1,3}\b
```

surname, title, given names or initials:

```
\b({Last})(\s+{Title})?
(\s+({First}|[A-Z])){1,2}\b
```

initials, surname:

```
\b([A-Z]\s+){1,2}{Last}\b
```

The name dictionaries used in the Regex extractor include the following: a given name dictionary (5,000 names taken from the 1990 study of the US Census), a surname dictionary (89,000 names, taken from the same US Census study) and a list of titles (773 titles manually taken from Wikipedia[1]). Counts exclude stop-words which had been removed (570 words). Other words used to eliminate false-positives were also taken from short lists of mutually exclusive categories: US States (149), street signs (11) and school suffixes (6).

Among the four base extractors, figures 3 and 4 show that the Regex extractor generally produces the highest quality extractions overall. Much of the improvement exhibited by the Regex extractor over the simpler dictionary extractor comes from the regular expression pattern matching which constrains possible matches to only the above patterns. The Regex extractor does less well on family and local histories (e.g. Libby and Fairfield) where the given regular expressions do not consistently apply: there are many names that consist of only a single given name. This could be corrected with contextual clues.

3.3 Maximum Entropy Markov Model

The MEMM extractor is a maximum entropy Markov model similar to that used in (Chieu and Ng, 2003) and trained on CoNLL NER training data (Sang and Meulder, 2003) in the newswire genre. Because of the training data, this MEMM was trained to recognize persons, places, dates and organizations in unstructured text, but we evaluated it only on the person names in the OCR corpus.

The feature templates used in the MEMM follow. For dictionary features, there was one feature template per dictionary, with dictionaries including all the dictionaries used by the previous two extractors.

- current word
- previous tag
- previous previous tag
- bigram of previous two tags
- next word
- current word's suffix and prefix, lengths 1 through 10 characters

[1] http://en.wikipedia.org/wiki/Title

- all upper case word (case-folded word)
- current word starts with an upper case character
- current word starts with an upper case character and is not the first word of a sentence
- next word starts with an upper case character
- previous word starts with an upper case character
- contains a number
- contains a hyphen
- the word is in dictionary

The validation / development test set was used to select the most promising variation of the MEMM. Variations considered but rejected included the use of a character noise model in conjunction with an allowance for small edit distances (from zero to three) when matching dictionary entries, similar in spirit to, though less well developed than (Wang et al., 2009). Variations also included additional feature templates based on centered 5-grams.

By way of comparison, this same MEMM was trained and tested on CoNLL data, where it achieved 83.1% F-measure using the same feature templates applied to the OCR data, as enumerated. This is not a state-of-the-art CoNLL NER system but it allows for more flexible experimentation.

Figures 5 and 6 show the greatest quality difference with respect to the other extractors in the two city directories (Birmingham and Portland). These directories essentially consist of lists of the names of people living in the respective cities, followed by terse information about them such as addresses and business names, one or two lines per person. Furthermore, the beginning of each entry is the name of the person, starting with the surname, which is less common in the data on which the MEMM was trained. The contrast between the newswire genre and most of the test data explains its relatively poor performance overall. Previous studies on domain mismatch in supervised learning but especially in NER (Vilain et al., 2007) document similar dramatic shortfalls in performance.

3.4 Conditional Random Field

The CRF extractor uses the conditional random field implementation in the Mallet toolkit (McCallum, 2002). It was trained and executed in the same way as the MEMM extractor described above, including the use of identical feature templates. Training and testing on the CoNLL data, as we did with the MEMM extractor, yielded a 87.0% F-measure.

The CRF extractor is the only one of the four base extractors not included in the ensemble. Adding the CRF resulted in slightly lower scores on the development test set. We also ran the ensemble with the CRF but without the MEMM, resulting in a 2% lower score on the development test set, ruling it out. Separate experiments on CoNLL test data with artificial noise introduced showed similarly worse behavior by the CRF, relative to the MEMM.

4 Ensemble Extraction Methods and Results

We combined the decisions of the first three base extractors described above using a simple voting-based ensemble. The ensemble interprets a full name in each base extractor's output as one vote in favor of that entity as a person name. The general ensemble extractor is parameterized by a threshold, t, indicating how many of the base extractors must agree on a person name before it can be included in the ensemble's output. By varying this parameter, we produced the three following ensemble extractors:

- Union ($t = 1$): any full name identified by any of the base extractors is output.
- Majority ($t = 2$): if a majority of the base extractors (two or more) recognizes the same text as a name, then that name is recognized.
- Intersection ($t = 3$): the three base extractors must be unanimous in choosing a full name to be extracted before that name will be output.

Figure 3 shows that the Majority Ensemble outperforms each base extractor in terms of F-measure.

A second set of ensembles was developed. They are identical to the three except that they allowed each base extractor to vote on individual tokens. This fine-grained ensemble did not produce accuracies as high as the coarse-grained approach when using the coarse-grained metrics, but when we use the fine-grained metrics it did better, achieving 68% F-measure over the entire corpus while the coarse-grained ensemble achieved only 60.7% F-measure. The highest-scoring base extractor (Regex) achieved 66.5% using the fine-grained metric. So, again, an ensemble did better than each base extractor regardless of the metric (coarse or fine), as long as the matching version of the ensemble was applied.

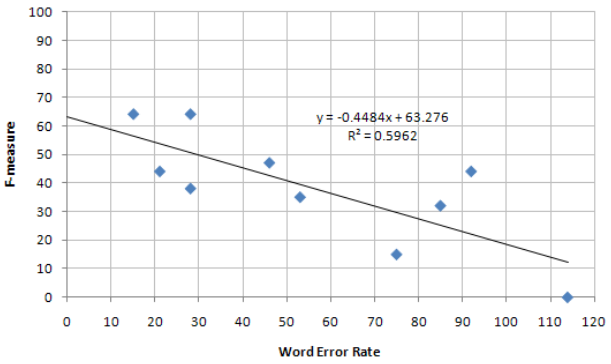


Figure 7: Coarse F-measure of the coarse majority voting ensemble for person names as a function of word error rate for pages in the blind test set.

5 Conclusions and Future Work

In conclusion, we answer the questions posed in the introduction. WER varies widely in this dataset: the average is much higher than the 20% reported in other papers (Miller et al., 2000). In a plot of WER versus NER performance shown in figure 7, the linear fit is substantially poorer than for the data reported in the work of Miller et al.

Ranges of 0–64% or 28–89% F-measure for NER can be expected on noisy OCR data, depending on the document and the metric. Figure 7 shows some but not perfect correlation between NER quality and WER. Among those errors that directly cause greater WER, different kinds of errors affect NER quality to different degrees.

The Libby text's WER was lower because of poor character-level recognition (word order was actually good) while Inverness had more errors in word order where text from two columns has been incorrectly interleaved by the OCR engine (its character-level recognition was good). From error analysis on such examples, it seems likely that word order errors play a bigger role in extraction errors than do character recognition errors.

We also conclude that combining basic methods can produce higher quality NER. Each of the three ensembles maximizes a different metric. The Majority Ensemble achieves the highest F-measure over the entire corpus, compared to any of the base extractors and to the other ensembles. The Intersection Ensemble achieves the highest precision and the Union Ensemble achieves the highest recall. Each of these results is useful for a different application. If the intended application is a person name search engine, users do not want to manually sift through many false-positives; with a sufficiently large corpus containing millions of book and newspaper titles, a precision of 89.6% would be more desirable than a precision of 61.6%, even when only 14.1% of the names available in the corpus can be recognized (low recall). Alternatively, if higher recall is necessary for an application in which no instances should be missed, then the high-recall Union Ensemble could be used as a filter of the candidates to be shown. Browsing and exploration of a data set for every case may be such an application. High-recall name browsing could facilitate manual labeling or checking.

This work is a starting point against which to compare techniques which we hope will be more effective in automatically adapting to new document formats and genres in the noisy OCR setting. One way to adapt the supervised machine learning approaches is in applying a more realistic noise model of OCR errors to the CoNLL data. Another is to use semi-supervised machine learning techniques to take advantage of the large volume of unlabeled and previously unused data available in each of the titles in this corpus. We plan to contrast this with the more laborious method of producing labeled training data from within the present corpus. Additional feature engineering and additional labeled pages for evaluation are also in order. The rule-based Regex extractor could also be adapted automatically to differing document or page formats by filtering a larger set of regular expressions in the first of two passes over each document. Finally, we plan to combine NER with work on OCR error correction (Lund and Ringger, 2009) to see if the combination can improve accuracies jointly in both OCR and information extraction.

6 Acknowledgements

We would like to acknowledge Ancestry.com and Lee Jensen of Ancestry.com for providing the OCR data from their free-text collection and for financial support. We would also like to thank Lee Jensen for discussions regarding applications of this work and the related constraints.

References

Hai Leong Chieu and Hwee Tou Ng. 2003. Named entity recognition with a maximum entropy approach. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 160–163.

D. W. Embley, D. M. Campbell, Y. S. Jiang, S. W. Liddle, D. W. Lonsdale, Y. -K. Ng, and R. D. Smith. 1999. Conceptual-model-based data extraction from multiple-record web pages. *Data & Knowledge Engineering*, 31(3):227–251, November.

Claire Grover, Sharon Givon, Richard Tobin, and Julian Ball. 2008. Named entity recognition for digitised historical texts. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*.

Kofax. 2009. Kofax homepage. http://www.kofax.com/.

W. B Lund and E. K Ringger. 2009. Improving optical character recognition through efficient multiple system alignment. In *Proceedings of the 2009 joint international conference on Digital libraries*, pages 231–240.

Andrew Kachites McCallum. 2002. MALLET: a machine learning for language toolkit. http://mallet.cs.umass.edu/.

David Miller, Sean Boisen, Richard Schwartz, Rebecca Stone, and Ralph Weischedel. 2000. Named entity extraction from noisy input: speech and OCR. In *Proceedings of ANLP-NAACL 2000*, pages 316–324.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26.

PrimeRecognition. 2009. PrimeOCR web page. http://www.primerecognition.com/augprime/prime_ocr.htm.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155, Boulder, Colorado. Association for Computational Linguistics.

E. F.T.K Sang and F. De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, volume 922, page 1341.

Marc Vilain, Jennifer Su, and Suzi Lubar. 2007. Entity extraction is a boring solved problem: or is it? In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers on XX*, pages 181–184, Rochester, New York. Association for Computational Linguistics.

Wei Wang, Chuan Xiao, Xuemin Lin, and Chengqi Zhang. 2009. Efficient approximate entity extraction with edit distance constraints. In *Proceedings of the 35th SIGMOD international conference on Management of data*, pages 759–770, Providence, Rhode Island, USA. ACM.