An Evaluation of Layout Features for Information Extraction from Calls for Papers

Karl-Michael Schneider
University of Passau
94030 Passau, Germany
schneide@phil.uni-passau.de

Abstract

We describe a feature-rich conditional random field model for the extraction of conference and workshop information (e.g. name, date, location, deadline) from calls for papers (CFPs). This has applications in the automatic construction of a conference knowledge base from a collection of CFPs. Relevant information in CFPs is often contained in regions that do not contain complete, grammatical sentences, but can be distinguished visually from other parts of the text by their formatting. We show that in this situation layout features, i.e. features that measure physical layout properties of a text, improve extraction accuracy considerably. On a corpus of CFPs we observe a 30% gain in F1 through the use of layout features.

1 Introduction

Information about relevant conferences and workshops is vital for people involved in research to present, discuss and publish their results and exchange ideas. Conference and workshop announcements are propagated to interested people via mailing lists as calls for papers (CFPs). A CFP includes such information as the conference or workshop name, date, location, website, various submission and notification dates, and possibly the name and website of an associated or co-located conference. A conference knowledge base (KB) presents this information for many conferences and workshops in a particular area (e.g. computer science, linguistics, artificial intelligence) in a structured way, with links between related or associated events.

Our goal is to build a conference KB automatically from a collection of CFPs. This involves two steps:

1. extracting relevant information from CFPs, which is the topic of this paper;
2. matching extracted instances from different CFPs that represent the same information but vary in their surface form (for instance, "ECML/PKDD 2005", "ECML/PKDD-2005", "ECML/PKDD'05" are all different spellings of the same conference name); this problem is known as coreference analysis [Kehler, 1997] or identity uncertainty [McCallum and Wellner, 2005].

Wellner *et al.* [2004] describe an integrated model that performs these two task simultaneously, with mutual benefit for both tasks. However, this paper focuses on the information extraction task only.

We apply conditional random fields (CRF) [Lafferty *et al.*, 2001] to information extraction from CFPs. This allows us to integrate various kinds of evidence from both content (i.e. tokens in a text) and layout (i.e. the physical structure of a text). CRFs have been applied successfully to a variety of sequence labeling tasks such as shallow parsing [Sha and Pereira, 2003], named entity recognition [Settles, 2004], information extraction [Peng and McCallum, 2004] and table recognition [Pinto *et al.*, 2003].

In particular we discuss the features we use to represent tokens in a CFP. CFPs differ from normal text by placing important information in regions that do not consist of complete, grammatical sentences, but instead are often characterized by rigid formatting, such as indented lines, centered lines, and lines separated by blank lines. Traditional information extraction techniques that rely on the grammatical structure of sentences (i.e. POS tags, syntactic structure), e.g. for extracting facts from news articles [Riloff and Jones, 1999], usually ignore the physical layout of the text and thus are not appropriate for information extraction from CFPs.

We describe a CRF model for information extraction that uses both token features (features measuring properties of tokens such as shape, occurrence in dictionaries, etc.) and layout features (features measuring the physical layout of the text surrounding a token). We observe a dramatic improvement in extraction performance through the use of these features.

The paper is organized as follows: In Sect. 2 we discuss related work. In Sect. 3 we review the general framework of CRFs. In Sect. 4 we describe the features used in our model. Section 5 describes our experiments, and Sect. 6 presents results. We finish the paper with some conclusions and an outlook to future work in Sect. 7.

2 Related Work

Layout features have been used previously in a variety of information extraction tasks. In [Peng and McCallum, 2004] a CRF is trained to extract various fields (such as author, title, etc.) from the header sections of research papers using a combination of linguistic and layout features. The features are very similar to ours. CFPs are similar to research papers in that most (though not all) of the important information is contained in highly formatted regions (the header section at the beginning) rather than in grammatical sentences. An important difference between this task and ours is that research paper headers consist only of header fields, with no intervening material. In contrast, the field instances in a CFP comprise only a small fraction of the tokens, making extraction a harder task. More-

over, many papers use standardized document layouts (e.g. through the use of LaTeX style files), whereas CFPs exhibit greater variation in form and layout.

Layout features have also been used for extracting tables from text [Hurst and Nasukawa, 2000; Pinto *et al.*, 2003]. In [Pinto *et al.*, 2003] layout features are used to locate tables in text, identify header and data cells and associate data cells with their corresponding header cells. They use a large variety of layout features that measure the occurrence of various amounts of whitespace indicative of table rows in text lines. Layout features such as "line begins with punctuation" and "line is the last line" are also used to learn to detect and extract signature lines and reply lines in E-mails [Carvalho and Cohen, 2004]. In both tasks an input text (web page with tables, E-mail) are considered sequences of lines rather than sequences of tokens, and features measure properties of lines. In contrast, we use features that measure properties of both lines and tokens.

In [Cox *et al.*, 2005] a conditional Markov model (CMM) tagger and a CRF are trained to extract up to 11 fields from workshop calls for papers using various token features, including orthography, POS tags and named entity tags, but no layout features. In addition, a relational model is used on top of the sequence model, that encodes domain-specific expectations, e.g. workshop acronyms resemble their names, and workshop dates occur after paper submission dates. The relational model improves performance by 5% f-score over the CMM alone but degrades performance of the CRF (probably because of the smaller window size used with the CRF). Moreover, extraction performance of the CRF is comparable to or better than that of the CMM with the relational model.

3 Conditional Random Fields

3.1 Model

Conditional random fields are discriminatively-trained undirected graphical models that are based on an exponential form and thus can combine overlapping, non-independent features easily [Lafferty *et al.*, 2001]. This allows us to integrate various kinds of evidence from both content and layout of a text. We use a linear-chain CRF, which maximizes the conditional probability of a label sequence, $\mathbf{y} = y_1 \dots y_T$, given an input sequence, $\mathbf{x} = x_1 \dots x_T$:

$$P_\Lambda(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp\left(\sum_{t=1}^{T}\sum_{k} \lambda_k f_k(y_{t-1}, y_t, \mathbf{x}, t)\right) \quad (1)$$

where $\Lambda = \{\lambda, \dots\}$ is a parameter vector, $Z_\mathbf{x}$ is a normalization constant that makes the probabilities of all label sequences sum to one, $f_k(y_{t-1}, y_t, \mathbf{x}, t)$ is a feature function, and λ_k is a learned weight associated with f_k. A feature function indicates the occurrence of an event consisting of a state transition $y_{t-1} \rightarrow y_t$ and a query to the input sequence $\mathbf{x}$ centered at the current time step t. For example, a feature function might have value 1 if the current state, y_t, is B-TI (indicating the beginning of a conference title) and the previous state, y_{t-1}, is O (meaning "not belonging to any entity") and the current word, x_t, is "Fifth", and value 0 otherwise.

A linear-chain CRF uses a global exponential model, in contrast to Maximum-Entropy Markov Models (MEMM) [McCallum *et al.*, 2000] that maximize the conditional probability of each state given the previous state and an observation, making them prone to the *label bias problem* [Lafferty *et al.*, 2001]. CRFs avoid this problem by using a model over state sequences rather than states.

3.2 Training

The weight λ_k for a feature function f_k indicates how likely the corresponding event is to occur. The weights are set to maximize the conditional log-likelihood of a set of labeled training sequences $D = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) : i = 1, \dots, M\}$:

$$\begin{aligned} &\sum_{i=1}^{M} \log P_\Lambda(\mathbf{y}^{(i)}, \mathbf{x}^{(i)}) - \sum_k \frac{\lambda_k^2}{2\sigma_k^2} \\ &= \sum_{i=1}^{M}\left(\sum_{t=1}^{T}\sum_k \lambda_k f_k(y_{t-1}^{(i)}, y_t^{(i)}, \mathbf{x}^{(i)}, t) - \log Z_{\mathbf{x}^{(i)}}\right) \quad (2) \\ &\quad - \sum_k \frac{\lambda_k^2}{2\sigma_k^2} \end{aligned}$$

The term $\sum_k \frac{\lambda_k^2}{2\sigma_k^2}$ is a Gaussian prior that is used for penalizing the log-likelihood in order to avoid over-fitting, and σ_k^2 is a variance [Peng and McCallum, 2004]. Maximizing (2) corresponds to matching the expected count of each feature according to the model to its adjusted empirical count:

$$\begin{aligned} &\sum_{i=1}^{M}\sum_{t=1}^{T} f_k(y_{t-1}^{(i)}, y_t^{(i)}, \mathbf{x}^{(i)}, t) - \frac{\lambda_k}{\sigma_k^2} \\ &= \sum_{i=1}^{M}\sum_{\mathbf{y}'} P_\Lambda(\mathbf{y}'|\mathbf{x}^{(i)}) \sum_{t=1}^{T} f_k(y'_{t-1}, y'_t, \mathbf{x}^{(i)}, t) \end{aligned}$$

The terms $\frac{\lambda_k}{\sigma_k^2}$ are used to discount the empirical feature counts. Any iterative procedure, such as traditional maximum entropy learning algorithms (GIS and IIS, [Della Pietra *et al.*, 1997]) can be used to maximize the log-likelihood in (2), but we use a procedure called *limited-memory quasi-Newton* (L-BFGS) [Sha and Pereira, 2003] because it converges much faster [Malouf, 2002; Sha and Pereira, 2003]. Since the log-likelihood function in a linear-chain CRF is convex (we assume a one-to-one correspondence between states and labels), learning is guaranteed to converge to a global maximum.

3.3 Inference

Information extraction with CRFs is seen as a sequence labeling task. We use the tokens from the text as the input sequence, and use three different types of symbols [Ramshaw and Marcus, 1995]: B-*type* (first token of an entity), I-*type* (subsequent tokens of an entity), and O (the token is not part of an entity). Thus an instance of a particular entity type is marked by the label sequence "B-*type* I-*type*..." An instance consisting of a single token is marked as "B-*type*".

We use the Viterbi algorithm to find the label sequence that maximizes the conditional probability of the label sequence given the input sequence:

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} P_\Lambda(\mathbf{y}|\mathbf{x})$$

subject to the consistency constraint that I-*type* must follow B-*type* or I-*type*.

4 Features

Each token in a CFP is represented as a set of binary features that measure lexical, contextual and spatial properties of the token. Table 1 presents a summary of the features that we use. The features can be divided into three groups:

Feature	Definition
Generic features	
w	word identity
ICAP	capitalized
ACAP	all uppercase
SCAP	single uppercase letter
MCAP	mixed case
ADIG	all digits
PUNC	punctuation symbol
URL	regular expression for URL
EMAIL	regular expression for E-mail address
HASUP	token contains uppercase letter
HASDIG	token contains digit
HASDASH	token contains "-"
HASPUNC	token contains punctuation symbol
ABBR	word ends with period
LOC	word occurs in gazetteer list
STATE	abbreviation of U.S. state name
COUNTRY	UK, U.S.A.
D_MO	january, february, march, april, may, june, july, august, september, october, november, december, jan, feb, mar, apr, may, jun, jul, aug, sep, sept, oct, nov, dec
D_DAY	monday, tuesday, wednesday, thursday, friday, saturday, sunday, mon, tue, wed, thu, fri, sat, sun
Domain features	
CNAME	conference name
CNUMY	conference number or year
DAY	day of week or day of month
DAYS	range of days
YEAR	four-digit year
SYEAR	two-digit year
ROM	roman number
NTH	number attribute
D_INST	center, centre, college, department, institute, school, univ., university
D_ORG	association, consortium, council, group, society
D_EV	colloquium, conf., conference, congress, convention, forum, meeting, round, roundtable, seminar, summit, symposium, table, track, workshop
D_ATTR	annual, autumn, biannual, biennial, european, fall, int., interdisciplinary, international, joint, national, special, spring, summer, winter
D_TH	st, nd, rd, th
Layout features	
BOL	first token in the line
EOL	last token in the line
BOT	first line in the text
EOT	last line in the text
BLANK	line contains no visible characters
PUNCTLN	line contains only punctuation characters
INDENT	line is indented
FIRST10	first 10 lines in the text
FIRST20	first 20 lines in the text

Table 1: List of features used (features with prefix "D_" are dictionary features and are case-insensitive)

- Generic token features describe word identity (i.e. the token itself is a feature), orthographic properties (e.g. capitalized, all uppercase, mixed case, all digits, punctuation), membership in certain token classes (month names, week days, URLs, E-mails), and occurrence in a gazetteer list.[1] We map capitalized words and uppercase words to lowercase after generating capitalization features. We look up sequences of up to five consecutive tokens in the gazetteer list and assign a feature to each token of a matching sequence. In addition we have a list of names of U.S. states and a short list of country abbreviations since these were not in the gazetteer list.
- Domain-specific features indicate parts of dates and numerical expressions that are typical for CFPs (e.g. 2005, 3-7, 19th), occurrence in a dictionary of words that are common in CFPs (e.g. conference, annual, international) and regular expression pattern matches that define common orthographic patterns in CFPs (e.g. ACL'03, ECML/PKDD).
- Layout features measure the position of a token in a line (i.e. whether the token is at the beginning or end of a line), the position of the line containing a token in the text (first/last line, first 10 and first 20 lines) and formatting properties of the line containing a token (indented lines, empty lines, lines consisting only of punctuation characters). The rationale for the first 10 and first 20 lines features is that in most CFPs the relevant information about a conference appears at the beginning of the text, usually within the first 10 or 20 lines (but never at the end of a CFP). Note that the feature BLANK can never occur (because all features occur with tokens, and no token occurs in a blank line). However, features BLANK-i and BLANK+i represent valuable information about the physical layout of the text.

For each token we collect the features for that token as well as for the two preceeding and following tokens, and for the line containing the token as well as for the two preceeding and following lines. For example, consider the following feature set:

> W=9th BOL HASDIG DAY NTH W-1=papers
> ICAP-1 W-2=for W+1=european ACAP+1
> D_ATTR+1 W+2=workshop ACAP+2 D_EV+2
> FIRST10 FIRST20 INDENT FIRST10-1
> FIRST20-1 BLANK-1 BOT-2 FIRST10-2
> FIRST20-2 INDENT-2 FIRST10+1 FIRST20+1
> BLANK+1 FIRST10+2 FIRST20+2
> INDENT+2

This feature set indicates the following properties of a token and its surrounding text:

- the token is the word "9th",
- it occurs in the token sequence "for Papers 9th EUROPEAN WORKSHOP" (for example, it could occur in the sequence "Call for Papers 9th EUROPEAN WORKSHOP ON NATURAL LANGUAGE GENERATION"),
- it appears at the beginning of a line,
- the line containing the token is the third line,
- the previous and next line are empty,

[1] obtained from http://www.world-gazetteer.com/

- the line containing the token and the lines two lines up (first) and down (fifth) are indented,
- all of them are among the first 10 and first 20 lines in the text.

5 Experiments

5.1 Dataset

Our dataset consists of 263 CFPs received by the author from various mailing lists between August 2002 and January 2004, and from February 2005 to May 2005.[2] We remove duplicate and near duplicate messages (based on their Nilsimsa digest[3]) and use only the plain text part of each message and remove mailing list signatures and email headers that occur in the text.

We apply only minimal tokenization. We separate punctuation, double quotes and parentheses from preceeding and following words but do not separate a period from the preceeding word if the word is a single capital letter or appears on a hand-crafted list of known abbreviations ("Dr", "Prof", "Int", etc.). Also, we do not separate dashes and single quotes from preceeding and following material because these symbols are often part of conference names, e.g. "ACL'05", "ICML-2005".

Each CFP is manually annotated for seven fields:

- Name (e.g. "ACL 2005")
- Title (e.g. "42nd Annual Meeting of the Association for Computational Linguistics")
- Date (i.e. when the conference takes place)
- Location (i.e. where the conference takes place)
- URL (of the conference web site)
- Deadline (for paper submission)
- Conjoined (i.e. the name and title of the main conference if the event is part of a larger conference, e.g. a workshop held in conjunction with a conference)

The total number of tokens is 203,151, with 7,217 tokens (3.6%) belonging to field instances.

For the experiments, we split the data into a training and testing set. We use the first 128 CFPs (from August 2002 to January 2004) for training and the remaining 135 CFPs (from February 2005 to May 2005) for testing.

5.2 Performance Measure

Following [Peng and McCallum, 2004] we measure performance using two different sets of metrics: word-based and instance-based. For word-based evaluation, we define *TP* as the number of distinct words in all hand-tagged instances of a field that occur in at least one extracted instance of that field; *FN* as the number of distinct words in hand-tagged instances that do not occur in an extracted instance; and *FP* as the number of distinct words in all extracted instances of a field that do not occur in at least one hand-tagged instance of the field. These counts are summed over all CFPs in the test set. Word precision, recall and F1 are defined as $prec = \frac{TP}{TP+FP}$, $recall = \frac{TP}{TP+FN}$, $F1 = \frac{2 \times prec \times recall}{prec+recall}$.

[2]Our results are not directly comparable to Cox *et al.* [2005]; unfortunately we became aware of their work only after our experiments and were unable to obtain their corpus and use it in time for the publication of this paper.

[3]http://ixazon.dynip.com/ cmeclax/nilsimsa.html

Instance-based evaluation considers an extracted instance correct only if it is identical to a hand-tagged instance of the same field. Thus in instance-based evaluation an extracted instance with even a single added or missing word is counted as an error. Instance precision and instance recall are the percentage of extracted instances of a field that are identical to a hand-tagged instance, and the percentage of hand-tagged instances that are extracted by the CRF, respectively. Instance F1 is defined accordingly as in word-based evaluation.

Note that instance-based recall/precision/F1 is not necessarily lower than word-based recall/precision/F1. As an example, consider two instances u_1u_2 and $v_1v_2v_3$. If u_1u_2, v_1v_2 and w_1w_2 are extracted, instance-based recall, precision and F1 are 50%, 33.3% and 40%, respectively, while word-based recall, precision and F1 are 80%, 66.7% and 72.7%, respectively. However, if u_1u_2 and $w_1w_2w_3$ are extracted, instance-based recall and precision are both 50% while word-based recall and precision are only 40%.

We report the word-based and instance-based measures for each field. Overall performance is measured by calculating precision and recall from counts summed over all fields and calculating F1 from overall precision and recall (called "micro average" in the information retrieval literature). This favors fields that occur more frequently than others. In addition, we calculate the average of the per-field F1 values (called "macro average" in the information retrieval literature). This gives equal weight to all fields.

5.3 Training CRFs

We use a Java implementation of CRFs [McCallum, 2002]. Training with the full feature set took about four hours on an Athlon AMD 800 MHz CPU with Linux operating system and converged after 156 iterations.

6 Results

6.1 Performance Evaluation

Table 2 shows per-field and overall performance. Word-based F1 is around 80% for most fields, except Conjoined and Name which are significantly lower. As expected, instance-based F1 is lower than word-based F1 for most fields, except Name which is 1.3% higher and URL which is equal to word-based F1 because URLs are single tokens. For Conjoined and Title instance-based F1 is much lower than word-based F1 (around 15–18%), presumably because on average instances of Conjoined and Title consist of more tokens than other fields, making them more prone to instance-based errors.

Notice also that performance is significantly lower than in [Peng and McCallum, 2004] for the research paper extraction task. However, field extraction from CFPs is a more difficult task because most tokens in a CFP do not belong to a field instance, whereas research paper headers consist only of header fields. In the CFP task there are three types of extraction errors: (i) assigning a word to the wrong field, (ii) assigning a word that belongs to a field to no field, (iii) assigning a non-field word to some field. In the research paper task only the first error type can occur.

6.2 Effects of Different Kinds of Features

To analyze the contribution of different kinds of features we trained four different models, using (i) only generic features, (ii) generic and domain features, (iii) generic and layout features, (iv) all features (the latter model is identical to

Field	Instances	W-Recall	W-Precision	W-F1	I-Recall	I-Precision	I-F1
Conjoined	93	41.6%	66.1%	51.0%	28.0%	48.1%	35.4%
Date	168	72.7%	90.8%	80.8%	64.9%	79.6%	71.5%
Deadline	161	68.9%	92.0%	78.8%	59.6%	80.7%	68.6%
Location	120	72.1%	90.8%	80.4%	64.2%	82.8%	72.3%
Name	78	46.7%	78.1%	58.5%	48.7%	77.6%	59.8%
Title	136	80.9%	79.8%	80.3%	61.8%	63.6%	62.7%
URL	131	71.8%	87.9%	79.0%	71.8%	87.9%	79.0%
Micro average	887	70.2%	84.1%	76.5%	59.1%	75.8%	66.4%
Macro average				72.7%			64.2%

Table 2: Extraction results with the full feature set

Features	generic	generic+domain	generic+layout	generic+domain+layout
micro Word-F1	58.8%	61.4% (+4.4%)	74.9% (+27.4%)	76.5% (+30.1%)
macro Word-F1	54.0%	57.2% (+5.9%)	70.4% (+30.4%)	72.7% (+34.6%)
micro Instance-F1	50.3%	52.6% (+4.6%)	65.0% (+29.2%)	66.4% (+32.0%)
macro Instance-F1	46.4%	49.2% (+6.0%)	62.3% (+34.3%)	64.2% (+38.4%)

Table 3: Contribution of different kinds of features (numbers in parentheses show relative improvement over generic features alone)

that in the previous section). We compare the overall performance of the four models in Table 3. Both domain and layout features improve the performance over using only generic features, both individually and in combination. Using the full feature set increases instance-based macro averaged F1 by 38% (relative) over using only generic features. Layout features have the biggest impact, resulting in a 34% relative increase in F1 over the generic features and 30% over the combination of generic and domain features. Domain features alone contribute only a 6% improvement over the generic features.

Table 4 shows the per-field improvement in instance-based F1 due to layout features. The biggest improvement (64% relative) is obtained for Name, and for Title and Location the relative improvement is 40%. These fields are highly correlated with formatting in CFPs. For the Deadline field the improvement is relatively small (only 7%). We explain this with the observation that deadlines are typically surrounded by unambiguous lexical material. This is confirmed in Table 5 which shows the features with highest weights in the trained CRF for transitions that start an instance from the default state O. According to Table 5 a good indicator for the start of a deadline is when the token two tokens to the left is one of the words "deadline" (as in "Paper submission deadline: August 5, 2005"), "submissions", "submission", "due", "abstract" or the current token is a day name. (Some of the features in Table 5 such as W-2=2004 and W=esslli clearly occur due to our limited training data.)

State	Features with highest weights
B-NA	W-1=(BOL ACAP W-2=systems MCAP W-2=papers
B-TI	PUNC-1 ICAP-1 W-1=: W-1=the D_ATTR+1 BOL
B-DA	W-2=date W-2=dates D_EV-2 W-2=conference PUNC-1 BOL-2
B-LO	B-LOC YEAR-2 ADIG-2 PUNC-1 W-1=, W-2=2004
B-DL	W-2=deadline W-2=submissions W-2=submission W-2=due DAY W-2=abstracts
B-UR	URL BLANK-2 W=www.irit.fr/cgi-bin/voir-congres W-2=site W-2=website W-2=workshop
B-CJ	W-1=with HASUP W=esslli D_EV+2 W-2=with W-2=information

Table 5: Features with highest weights in the trained CRF, where the previous state is O

7 Conclusions and Future Work

This paper applies conditional random fields to a practical problem: extracting important knowledge from calls for papers for academic conferences and related events. We demonstrate the effectiveness of layout features in the absence of grammatical structure, which is typical for those regions in CFPs that contain the key information about an event, obtaining an improvement in instance-based average F1 by 30%.

Extraction performance in our experiments is reasonable but not optimal, probably due to the relatively small training corpus. Increasing the amount of training data would be expected to help improve the performance. However, annotating training data manually is labor-intensive. In future work we intend to employ bootstrapping [Lin *et al.*, 2003] to reduce the amount of manual work in obtaining training data.

References

[Carvalho and Cohen, 2004] Vitor R. Carvalho and William W. Cohen. Learning to extract signature and reply lines from email. In *Prod. First Conference on Email and Anti-Spam (CEAS)*, Mountain View, CA, 2004.

[Cox *et al.*, 2005] Christopher Cox, Jamie Nicolson, Jenny Rose Finkel, Christopher Manning, and Pat Langley. Template sampling for leveraging domain knowledge in information extraction. In *PASCAL Challenges Workshop*, Southampton, U.K., April 2005.

[Della Pietra *et al.*, 1997] Stephen Della Pietra, Vincent J. Della Pietra, and John Lafferty. Inducing features of

Field	Conjoined	Date	Deadline	Location	Name	Title	URL
without layout	29.2%	62.6%	64.0%	50.0%	36.4%	43.5%	58.8%
with layout	35.4%	71.5%	68.6%	72.3%	59.8%	62.7%	79.0%

Table 4: Instance-based F1 improvements for individual fields through the use of layout features

random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(4):380–393, 1997.

[Hurst and Nasukawa, 2000] Matthew Hurst and Tetsuya Nasukawa. Layout and language: Integrating spatial and linguistic knowledge for layout understanding tasks. In *Proc. 18th Int. Conference on Computational Linguistics (COLING'00)*, pages 334–340, Saarbrücken, Germany, 2000.

[Kehler, 1997] Andrew Kehler. Probabilistic coreference in information extraction. In Claire Cardie and Ralph Weischedel, editors, *Proc. 2nd Conference on Empirical Methods in Natural Language Processing (EMNLP'97)*, pages 163–173, Somerset, NJ, 1997. Association for Computational Linguistics.

[Lafferty *et al.*, 2001] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conference on Machine Learning (ICML-2001)*, pages 282–289, San Francisco, CA, 2001. Morgan Kaufmann.

[Lin *et al.*, 2003] Winston Lin, Roman Yangarber, and Ralph Grishman. Bootstrapped learning of semantic classes from positive and negative examples. In *Proc. ICML-2003 Workshop on The Continuum from Labeled to Unlabeled Data*, pages 103–110, Washington, DC, 2003.

[Malouf, 2002] Robert Malouf. A comparison of algorithms for maximum entropy parameter estimation. In *Proc. Sixth Conference on Natural Language Learning (CoNLL-2002)*, pages 49–55, Taipei, Taiwan, 2002.

[McCallum and Wellner, 2005] Andrew McCallum and Ben Wellner. Conditional models of identity uncertainty with application to noun coreference. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 905–912, Cambridge, MA, 2005. MIT Press.

[McCallum *et al.*, 2000] Andrew McCallum, Dayne Freitag, and Fernando Pereira. Maximum entropy markov models for information extraction and segmentation. In *Proc. 17th International Conference on Machine Learning (ICML-2000)*, pages 591–598, San Francisco, CA, 2000. Morgan Kaufmann.

[McCallum, 2002] Andrew Kachites McCallum. MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu/, 2002.

[Peng and McCallum, 2004] Fuchun Peng and Andrew McCallum. Accurate information extraction from research papers using conditional random fields. In *Proc. HLT-NAACL 2004*, pages 329–336, Boston, Massachusetts, 2004.

[Pinto *et al.*, 2003] David Pinto, Andrew McCallum, Xing Wei, and W. Bruce Croft. Table extraction using conditional random fields. In *Proc. 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2003)*, pages 235–242, Toronto, Canada, 2003.

[Ramshaw and Marcus, 1995] Lance A. Ramshaw and Mitchell P. Marcus. Text chunking using transformation-based learning. In *Proc. ACL Third Workshop on Very Large Corpora*, pages 82–94, 1995.

[Riloff and Jones, 1999] Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proc. 16th National Conference on Artificial Intelligence and 11th Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI 1999)*, pages 474–479, Orlando, Florida, 1999. AAAI Press.

[Settles, 2004] Burr Settles. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proc. Int. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA)*, pages 104–107, Geneva, Switzerland, 2004.

[Sha and Pereira, 2003] Fei Sha and Fernando C. N. Pereira. Shallow parsing with conditional random fields. In *Proc. HLT-NAACL 2003*, pages 134–141, Edmonton, Canada, 2003.

[Wellner *et al.*, 2004] Ben Wellner, Andrew McCallum, Fuchun Peng, and Michael Hay. An integrated, conditional model of information extraction and coreference with application to citation matching. In *Proc. 20th Annual Conference on Uncertainty in Artificial Intelligence (UAI-04)*, pages 593–601, Arlington, Virginia, 2004. AUAI Press.