Unsupervised Learning of Field Segmentation Models for Information Extraction

Trond Grenager
Computer Science Department
Stanford University
Stanford, CA 94305
grenager@cs.stanford.edu

Dan Klein
Computer Science Division
U.C. Berkeley
Berkeley, CA 94709
klein@cs.berkeley.edu

Christopher D. Manning
Computer Science Department
Stanford University
Stanford, CA 94305
manning@cs.stanford.edu

Abstract

The applicability of many current information extraction techniques is severely limited by the need for supervised training data. We demonstrate that for certain *field structured* extraction tasks, such as classified advertisements and bibliographic citations, small amounts of prior knowledge can be used to learn effective models in a primarily unsupervised fashion. Although hidden Markov models (HMMs) provide a suitable generative model for field structured text, general unsupervised HMM learning fails to learn useful structure in either of our domains. However, one can dramatically improve the quality of the learned structure by exploiting simple prior knowledge of the desired solutions. In both domains, we found that unsupervised methods can attain accuracies with 400 unlabeled examples comparable to those attained by supervised methods on 50 labeled examples, and that semi-supervised methods can make good use of small amounts of labeled data.

1 Introduction

Information extraction is potentially one of the most useful applications enabled by current natural language processing technology. However, unlike general tools like parsers or taggers, which generalize reasonably beyond their training domains, extraction systems must be entirely retrained for each application. As an example, consider the task of turning a set of diverse classified advertisements into a queryable database; each type of ad would require tailored training data for a supervised system. Approaches which required little or no training data would therefore provide substantial resource savings and extend the practicality of extraction systems.

The term *information extraction* was introduced in the MUC evaluations for the task of finding short pieces of relevant information within a broader text that is mainly irrelevant, and returning it in a structured form. For such "nugget extraction" tasks, the use of unsupervised learning methods is difficult and unlikely to be fully successful, in part because the nuggets of interest are determined only extrinsically by the needs of the user or task. However, the term *information extraction* was in time generalized to a related task that we distinguish as *field segmentation*. In this task, a document is regarded as a sequence of pertinent fields, and the goal is to segment the document into fields, and to label the fields. For example, bibliographic citations, such as the one in Figure 1(a), exhibit clear field structure, with fields such as *author*, *title*, and *date*. Classified advertisements, such as the one in Figure 1(b), also exhibit field structure, if less rigidly: an ad consists of descriptions of attributes of an item or offer, and a set of ads for similar items share the same attributes. In these cases, the fields present a salient, intrinsic form of linguistic structure, and it is reasonable to hope that field segmentation models could be learned in an unsupervised fashion.

In this paper, we investigate unsupervised learning of field segmentation models in two domains: bibliographic citations and classified advertisements for apartment rentals. General, unconstrained induction of HMMs using the EM algorithm fails to detect useful field structure in either domain. However, we demonstrate that small amounts of prior knowledge can be used to greatly improve the learned model. In both domains, we found that unsupervised methods can attain accuracies with 400 unlabeled examples comparable to those attained by supervised methods on 50 labeled examples, and that semi-supervised methods can make good use of small amounts of labeled data.

(a)	*AUTH*	*AUTH*	*AUTH*	*DATE*	*DATE*	*DATE*	*DATE*	*TTL*	*TTL*	*TTL*	*TTL*
	Pearl	,	J.	(	1988	)	.	Probabilistic	Reasoning	in	Intelligent
	TTL	*TTL*	*TTL*	*TTL*	*TTL*	*TTL*	*TTL*	*PUBL*	*PUBL*	*PUBL*	
	Systems	:	Networks	of	Plausible	Inference	.	Morgan	Kaufmann	.	

(b)	*SIZE*	*SIZE*	*SIZE*	*SIZE*	*SIZE*	*FEAT*	*FEAT*	*FEAT*	*FEAT*	*FEAT*	*FEAT*
	Spacious	1	Bedroom	apt	.	newly	remodeled	,	gated	,	new
	FEAT	*FEAT*	*FEAT*	*FEAT*	*FEAT*	*NBRHD*	*NBRHD*	*NBRHD*	*NBRHD*	*NBRHD*	*NBRHD*
	appliance	,	new	carpet	,	near	public	transportion	,	close	to
	NBRHD	*NBRHD*	*NBRHD*	*RENT*	*RENT*	*RENT*	*CONTACT*				
	580	freeway	,	$	500.00	Deposit	(510)655-0106				

(c)	*RB*	,	*PRP*	*VBD*	*RB*	*NNP*	*NNP*	.
	No	,	it	was	n't	Black	Monday	.

Figure 1: Examples of three domains for HMM learning: the bibliographic citation fields in (a) and classified advertisements for apartment rentals shown in (b) exhibit field structure. Contrast these to part-of-speech tagging in (c) which does not.

2 Hidden Markov Models

Hidden Markov models (HMMs) are commonly used to represent a wide range of linguistic phenomena in text, including morphology, parts-of-speech (POS), named entity mentions, and even topic changes in discourse. An HMM consists of a set of states S, a set of observations (in our case words or tokens) W, a transition model specifying $\mathrm{P}(s_t|s_{t-1})$, the probability of transitioning from state s_{t-1} to state s_t, and an emission model specifying $\mathrm{P}(w|s)$ the probability of emitting word w while in state s. For a good tutorial on general HMM techniques, see Rabiner (1989).

For all of the unsupervised learning experiments we fit an HMM with the same number of hidden states as gold labels to an unannotated training set using EM.[1] To compute hidden state expectations efficiently, we use the Forward-Backward algorithm in the standard way. Emission models are initialized to almost-uniform probability distributions, where a small amount of noise is added to break initial symmetry. Transition model initialization varies by experiment. We run the EM algorithm to convergence. Finally, we use the Viterbi algorithm with the learned parameters to label the test data.

All baselines and experiments use the same tokenization, normalization, and smoothing techniques, which were not extensively investigated. Tokenization was performed in the style of the Penn Treebank, and tokens were normalized in various ways: numbers, dates, phone numbers, URLs, and email addresses were collapsed to dedicated tokens, and all remaining tokens were converted to lowercase. Unless otherwise noted, the emission models use simple add-λ smoothing, where λ was 0.001 for supervised techniques, and 0.2 for unsupervised techniques.

3 Datasets and Evaluation

The bibliographic citations data is described in McCallum et al. (1999), and is distributed at *http://www.cs.umass.edu/~mccallum/*. It consists of 500 hand-annotated citations, each taken from the reference section of a different computer science research paper. The citations are annotated with 13 fields, including *author*, *title*, *date*, *journal*, and so on. The average citation has 35 tokens in 5.5 fields. We split this data, using its natural order, into a 300-document training set, a 100-document development set, and a 100-document test set.

The classified advertisements data set is novel, and consists of 8,767 classified advertisements for apartment rentals in the San Francisco Bay Area downloaded in June 2004 from the Craigslist website. It is distributed at *http://www.stanford.edu/~grenager/*. 302 of the ads have been labeled with 12 fields, including *size*, *rent*, *neighborhood*, *features*, and so on. The average ad has 119 tokens in 8.7 fields. The annotated data is divided into a 102-document training set, a 100-document development set, and a 100-document test set. The remaining 8465 documents form an unannotated training set.

In both cases, all system development and parameter tuning was performed on the development set,

[1] EM is a greedy hill-climbing algorithm designed for this purpose, but it is not the only option; one could also use coordinate ascent methods or sampling methods.

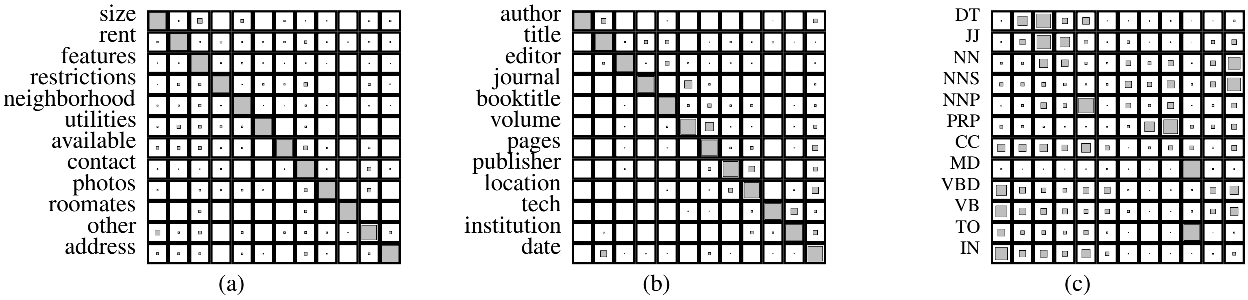


Figure 2: Matrix representations of the target transition structure in two field structured domains: (a) classified advertisements (b) bibliographic citations. Columns and rows are indexed by the same sequence of fields. Also shown is (c) a submatrix of the transition structure for a part-of-speech tagging task. In all cases the column labels are the same as the row labels.

and the test set was only used once, for running final experiments. Supervised learning experiments train on documents selected randomly from the annotated training set and test on the complete test set. Unsupervised learning experiments also test on the complete test set, but create a training set by first adding documents from the test set (without annotation), then adding documents from the annotated training set (without annotation), and finally adding documents from the unannotated training set. Thus if an unsupervised training set is larger than the test set, it fully contains the test set.

To evaluate our models, we first learn a set of model parameters, and then use the parameterized model to label the sequence of tokens in the test data with the model's hidden states. We then compare the similarity of the guessed sequence to the human-annotated sequence of gold labels, and compute accuracy on a per-token basis.[2] In evaluation of supervised methods, the model states and gold labels are the same. For models learned in a fully unsupervised fashion, we map each model state in a greedy fashion to the gold label to which it most often corresponds in the gold data. There is a worry with this kind of greedy mapping: it increasingly inflates the results as the number of hidden states grows. To keep the accuracies meaningful, all of our models have exactly the same number of hidden states as gold labels, and so the comparison is valid.

[2]This evaluation method is used by McCallum et al. (1999) but otherwise is not very standard. Compared to other evaluation methods for information extraction systems, it leads to a lower penalty for boundary errors, and allows long fields also contribute more to accuracy than short ones.

4 Unsupervised Learning

Consider the general problem of learning an HMM from an unlabeled data set. Even abstracting away from concrete search methods and objective functions, the diversity and simultaneity of linguistic structure is already worrying; in Figure 1 compare the field structure in (a) and (b) to the parts-of-speech in (c). If strong sequential correlations exist at multiple scales, any fixed search procedure will detect and model at most one of these levels of structure, not necessarily the level desired at the moment. Worse, as experience with part-of-speech and grammar learning has shown, induction systems are quite capable of producing some uninterpretable mix of various levels and kinds of structure.

Therefore, if one is to preferentially learn one kind of inherent structure over another, there must be some way of constraining the process. We could hope that field structure is the strongest effect in classified ads, while parts-of-speech is the strongest effect in newswire articles (or whatever we would try to learn parts-of-speech from). However, it is hard to imagine how one could bleach the local grammatical correlations and long-distance topical correlations from our classified ads; they are still English text with part-of-speech patterns. One approach is to vary the objective function so that the search prefers models which detect the structures which we have in mind. This is the primary way supervised methods work, with the loss function relativized to training label patterns. However, for unsupervised learning, the primary candidate for an objective function is the data likelihood, and we don't have another suggestion here. Another approach is to inject some prior knowledge into the search procedure by carefully choosing the starting point; indeed smart initialization has been critical to success in many previous unsupervised learning experiments. The central idea of this paper is that we can instead restrict the entire search domain by constraining the model class to reflect the desired structure in the data, thereby directing the search toward models of interest. We do this in several ways, which are described in the following sections.

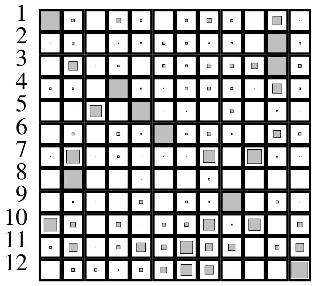
4.1 Baselines

To situate our results, we provide three different baselines (see Table 1). First is the most-frequent-field accuracy, achieved by labeling all tokens with the same single label which is then mapped to the most frequent field. This gives an accuracy of 46.4% on the advertisements data and 27.9% on the citations data. The second baseline method is to pre-segment the unlabeled data using a crude heuristic based on punctuation, and then to cluster the resulting segments using a simple Naïve Bayes mixture model with the Expectation-Maximization (EM) algorithm. This approach achieves an accuracy of 62.4% on the advertisements data, and 46.5% on the citations data.

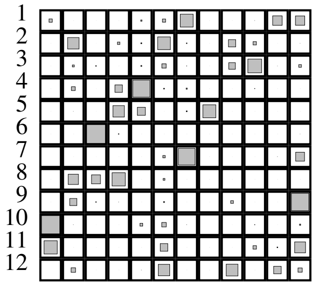
As a final baseline, we trained a supervised first-order HMM from the annotated training data using maximum likelihood estimation. With 100 training examples, supervised models achieve an accuracy of 74.4% on the advertisements data, and 72.5% on the citations data. With 300 examples, supervised methods achieve accuracies of 80.4 on the citations data. The learning curves of the supervised training experiments for different amounts of training data are shown in Figure 4. Note that other authors have achieved much higher accuracy on the the citation dataset using HMMs trained with supervision: McCallum et al. (1999) report accuracies as high as 92.9% by using more complex models and millions of words of BibTeX training data.

4.2 Unconstrained HMM Learning

From the supervised baseline above we know that there is some first-order HMM over $|S|$ states which captures the field structure we're interested in, and we would like to find such a model without supervision. As a first attempt, we try fitting an unconstrained HMM, where the transition function is initialized randomly, to the unannotated training data. Not surprisingly, the unconstrained approach leads to predictions which poorly align with the desired field segmentation: with 400 unannotated training documents, the accuracy is just 48.8% for the advertisements and 49.7% for the citations: better than the single state baseline but far from the supervised baseline. To illustrate what is (and isn't) being learned, compare typical transition models learned by this method, shown in Figure 3, to the maximum-likelihood transition models for the target annotations, shown in Figure 2. Clearly, they aren't anything like the target models: the learned classified advertisements matrix has some but not all of the desired diagonal structure, and the learned citations matrix has almost no mass on the diagonal, and appears to be modeling smaller scale structure.



(a) Classified Advertisements



(b) Citations

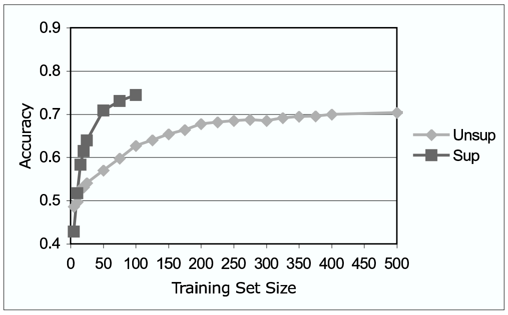
Figure 3: Matrix representations of typical transition models learned by initializing the transition model uniformly.

4.3 Diagonal Transition Models

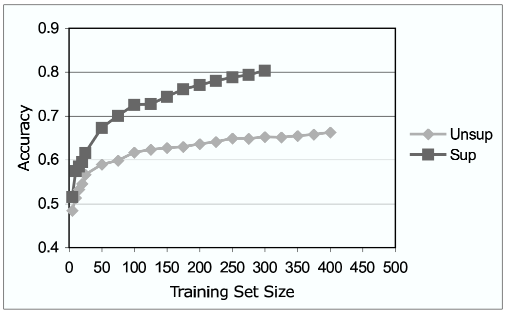
To adjust our procedure to learn larger-scale patterns, we can constrain the parametric form of the transition model to be

$$\mathrm{P}(s_t|s_{t-1}) = \begin{cases} \sigma + \frac{(1-\sigma)}{|S|} & \text{if } s_t = s_{t-1} \\ \frac{(1-\sigma)}{|S|} & \text{otherwise} \end{cases}$$

where $|S|$ is the number of states, and σ is a global free parameter specifying the self-loop probability:



(a) Classified advertisements



(b) Bibliographic citations

Figure 4: Learning curves for supervised learning and unsupervised learning with a diagonal transition matrix on (a) classified advertisements, and (b) bibliographic citations. Results are averaged over 50 runs.

the probability of a state transitioning to itself. (Note that the expected mean field length for transition functions of this form is $\frac{1}{1-\sigma}$.) This constraint provides a notable performance improvement: with 400 unannotated training documents the accuracy jumps from 48.8% to 70.0% for advertisements and from 49.7% to 66.3% for citations. The complete learning curves for models of this form are shown in Figure 4. We have tested training on more unannotated data; the slope of the learning curve is leveling out, but by training on 8000 unannotated ads, accuracy improves significantly to 72.4%. On the citations task, an accuracy of approximately 66% can be achieved either using supervised training on 50 annotated citations, or unsupervised training using 400 unannotated citations. [3]

Although σ can easily be reestimated with EM (even on a per-field basis), doing so does not yield

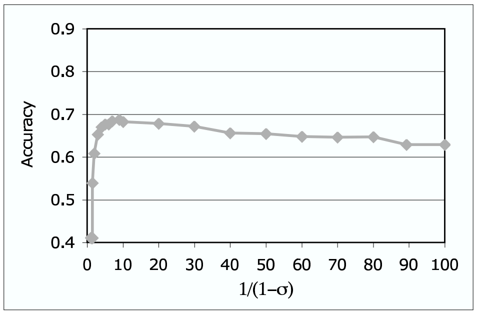


Figure 5: Unsupervised accuracy as a function of the expected mean field length $\frac{1}{1-\sigma}$ for the classified advertisements dataset. Each model was trained with 500 documents and tested on the development set. Results are averaged over 50 runs.

better models.[4] On the other hand, model accuracy is not very sensitive to the exact choice of σ, as shown in Figure 5 for the classified advertisements task (the result for the citations task has a similar shape). For the remaining experiments on the advertisements data, we use $\sigma = 0.9$, and for those on the citations data, we use $\sigma = 0.5$.

4.4 Hierarchical Mixture Emission Models

Consider the highest-probability state emissions learned by the diagonal model, shown in Figure 6(a). In addition to its characteristic content words, each state also emits punctuation and English function words devoid of content. In fact, state 3 seems to have specialized entirely in generating such tokens. This can become a problem when labeling decisions are made on the basis of the function words rather than the content words. It seems possible, then, that removing function words from the field-specific emission models could yield an improvement in labeling accuracy.

One way to incorporate this knowledge into the model is to delete stopwords, which, while perhaps not elegant, has proven quite effective in the past. A better founded way of making certain words unavailable to the model is to emit those words from all states with equal probability. This can be accomplished with the following simple hierarchical mixture emission model

$$\mathrm{P}_h(w|s) = \alpha \mathrm{P}_c(w) + (1-\alpha)\mathrm{P}(w|s)$$

where P_c is the common word distribution, and α is

[3]We also tested training on 5000 additional unannotated citations collected from papers found on the Internet. Unfortunately the addition of this data didn't help accuracy. This probably results from the fact that the datasets were collected from different sources, at different times.

[4]While it may be surprising that disallowing reestimation of the transition function is helpful here, the same has been observed in acoustic modeling (Rabiner and Juang, 1993).

State	10 Most Common Words
1	. $ no ! month deposit , pets rent available
2	, . room and with in large living kitchen -
3	. a the is and for this to , in
4	[NUM1] [NUM0] , bedroom bath / - . car garage
5	, . and a in - quiet with unit building
6	- . [TIME] [PHONE] [DAY] call [NUM8] at

(a)

State	10 Most Common Words
1	[NUM2] bedroom [NUM1] bath bedrooms large sq car ft garage
2	$ no month deposit pets lease rent available year security
3	kitchen room new , with living large floors hardwood fireplace
4	[PHONE] call please at or for [TIME] to [DAY] contact
5	san street at ave st # [NUM:DDD] francisco ca [NUM:DDDD]
6	of the yard with unit private back a building floor
Comm.	*CR* . , and - the in a / is with : of for to

(b)

Figure 6: Selected state emissions from a typical model learned from unsupervised data using the constrained transition function: (a) with a flat emission model, and (b) with a hierarchical emission model.

a new global free parameter. In such a model, before a state emits a token it flips a coin, and with probability α it allows the common word distribution to generate the token, and with probability $(1-\alpha)$ it generates the token from its state-specific emission model (see Vaithyanathan and Dom (2000) and Toutanova et al. (2001) for more on such models). We tuned α on the development set and found that a range of values work equally well. We used a value of 0.5 in the following experiments.

We ran two experiments on the advertisements data, both using the fixed transition model described in Section 4.3 and the hierarchical emission model. First, we initialized the emission model of P_c to a general-purpose list of stopwords, and did not reestimate it. This improved the average accuracy from 70.0% to 70.9%. Second, we learned the emission model of P_c using EM reestimation. Although this method did not yield a significant improvement in accuracy, it learns sensible common words: Figure 6(b) shows a typical emission model learned with this technique. Unfortunately, this technique does not yield improvements on the citations data.

4.5 Boundary Models

Another source of error concerns field boundaries. In many cases, fields are more or less correct, but the boundaries are off by a few tokens, even when punctuation or syntax make it clear to a human reader where the exact boundary should be. One way to address this is to model the fact that in this data fields often end with one of a small set of boundary tokens, such as punctuation and new lines, which are shared across states.

To accomplish this, we enriched the Markov process so that each field s is now modeled by two states, a non-final $s^- \in S^-$ and a final $s^+ \in S^+$. The transition model for final states is the same as before, but the transition model for non-final states has two new global free parameters: λ, the probability of staying within the field, and μ, the probability of transitioning to the final state given that we are staying in the field. The transition function for non-final states is then

$$\mathrm{P}(s'|s^-) = \begin{cases} (1-\mu)(\lambda + \frac{(1-\lambda)}{|S^-|}) & \text{if } s' = s^- \\ \mu(\lambda + \frac{(1-\lambda)}{|S^-|}) & \text{if } s' = s^+ \\ \frac{(1-\lambda)}{|S^-|} & \text{if } s' \in S^- \backslash s^- \\ 0 & \text{otherwise.} \end{cases}$$

Note that it can bypass the final state, and transition directly to other non-final states with probability $(1-\lambda)$, which models the fact that not all field occurrences end with a boundary token. The transition function for non-final states is then

$$\mathrm{P}(s'|s^+) = \begin{cases} \sigma + \frac{(1-\sigma)}{|S^-|} & \text{if } s' = s^- \\ \frac{(1-\sigma)}{|S^-|} & \text{if } s' \in S^- \backslash s^- \\ 0 & \text{otherwise.} \end{cases}$$

Note that this has the form of the standard diagonal function. The reason for the self-loop from the final state back to the non-final state is to allow for field internal punctuation. We tuned the free parameters on the development set, and found that $\sigma = 0.5$ and $\lambda = 0.995$ work well for the advertisements domain, and $\sigma = 0.3$ and $\lambda = 0.9$ work well for the citations domain. In all cases it works well to set $\mu = 1 - \lambda$. Emissions from non-final states are as

	Ads	Citations
Baseline	46.4	27.9
Segment and cluster	62.4	46.5
Supervised	74.4	72.5
Unsup. (learned trans)	48.8	49.7
Unsup. (diagonal trans)	70.0	66.3
+ Hierarchical (learned)	70.1	39.1
+ Hierarchical (given)	70.9	62.1
+ Boundary (learned)	70.4	64.3
+ Boundary (given)	71.9	68.2
+ Hier. + Bnd. (learned)	71.0	—
+ Hier. + Bnd. (given)	72.7	—

Table 1: Summary of results. For each experiment, we report percentage accuracy on the test set. Supervised experiments use 100 training documents, and unsupervised experiments use 400 training documents. Because unsupervised techniques are stochastic, those results are averaged over 50 runs, and differences greater than 1.0% are significant at p=0.05% or better according to the t-test. The last 6 rows are not cumulative.

before (hierarchical or not depending on the experiment), while all final states share a boundary emission model. Note that the boundary emissions are not smoothed like the field emissions.

We tested both supplying the boundary token distributions and learning them with reestimation during EM. In experiments on the advertisements data we found that learning the boundary emission model gives an insignificant raise from 70.0% to 70.4%, while specifying the list of allowed boundary tokens gives a significant increase to 71.9%. When combined with the given hierarchical emission model from the previous section, accuracy rises to 72.7%, our best unsupervised result on the advertisements data with 400 training examples. In experiments on the citations data we found that learning boundary emission model hurts accuracy, but that given the set of boundary tokens it boosts accuracy significantly: increasing it from 66.3% to 68.2%.

5 Semi-supervised Learning

So far, we have largely focused on incorporating prior knowledge in rather general and implicit ways. As a final experiment we tested the effect of adding a small amount of supervision: augmenting the large amount of unannotated data we use for unsupervised learning with a small amount of annotated data. There are many possible techniques for semi-supervised learning; we tested a particularly simple one. We treat the annotated labels as observed variables, and when computing sufficient statistics in the M-step of EM we add the observed counts from the

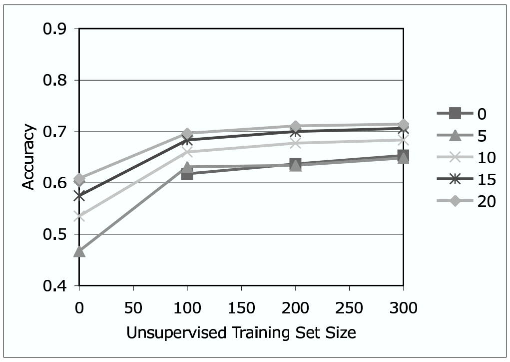


Figure 7: Learning curves for semi-supervised learning on the citations task. A separate curve is drawn for each number of annotated documents. All results are averaged over 50 runs.

annotated documents to the expected counts computed in the E-step. We estimate the transition function using maximum likelihood from the annotated documents only, and do not reestimate it. Semi-supervised results for the citations domain are shown in Figure 7. Adding 5 annotated citations yields no improvement in performance, but adding 20 annotated citations to 300 unannotated citations boosts performance greatly from 65.2% to 71.3%. We also tested the utility of this approach in the classified advertisement domain, and found that it did not improve accuracy. We believe that this is because the transition information provided by the supervised data is very useful for the citations data, which has regular transition structure, but is not as useful for the advertisements data, which does not.

6 Previous Work

A good amount of prior research can be cast as supervised learning of field segmentation models, using various model families and applied to various domains. McCallum et al. (1999) were the first to compare a number of supervised methods for learning HMMs for parsing bibliographic citations. The authors explicitly claim that the domain would be suitable for unsupervised learning, but they do not present experimental results. McCallum et al. (2000) applied supervised learning of Maximum Entropy Markov Models (MEMMs) to the domain of parsing Frequently Asked Question (FAQ) lists into their component field structure. More recently, Peng and McCallum (2004) applied supervised learning of Conditional Random Field (CRF) sequence models to the problem of parsing the headers of research papers.

There has also been some previous work on unsupervised learning of field segmentation models in particular domains. Pasula et al. (2002) performs limited unsupervised segmentation of bibliographic citations as a small part of a larger probabilistic model of identity uncertainty. However, their system does not explicitly learn a field segmentation model for the citations, and encodes a large amount of hand-supplied information about name forms, abbreviation schemes, and so on. More recently, Barzilay and Lee (2004) defined *content models*, which can be viewed as field segmentation models occurring at the level of discourse. They perform unsupervised learning of these models from sets of news articles which describe similar events. The *fields* in that case are the topics discussed in those articles. They consider a very different set of applications from the present work, and show that the learned topic models improve performance on two discourse-related tasks: information ordering and extractive document summarization. Most importantly, their learning method differs significantly from ours; they use a complex and special purpose algorithm, which is difficult to adapt, while we see our contribution to be a demonstration of the interplay between model family and learned structure. Because the structure of the HMMs they learn is similar to ours it seems that their system could benefit from the techniques of this paper. Finally, Blei and Moreno (2001) use an HMM augmented by an aspect model to automatically segment documents, similar in goal to the system of Hearst (1997), but using techniques more similar to the present work.

7 Conclusions

In this work, we have examined the task of learning field segmentation models using unsupervised learning. In two different domains, classified advertisements and bibliographic citations, we showed that by constraining the model class we were able to restrict the search space of EM to models of interest. We used unsupervised learning methods with 400 documents to yield field segmentation models of a similar quality to those learned using supervised learning with 50 documents. We demonstrated that further refinements of the model structure, including hierarchical mixture emission models and boundary models, produce additional increases in accuracy. Finally, we also showed that semi-supervised methods with a modest amount of labeled data can sometimes be effectively used to get similar good results, depending on the nature of the problem.

While there are enough resources for the citation task that much better numbers than ours can be and have been obtained (with more knowledge and resource intensive methods), in domains like classified ads for lost pets or used bicycles unsupervised learning may be the only practical option. In these cases, we find it heartening that the present systems do as well as they do, even without field-specific prior knowledge.

8 Acknowledgements

We would like to thank the reviewers for their consideration and insightful comments.

References

R. Barzilay and L. Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *Proceedings of HLT-NAACL 2004*, pages 113–120.

D. Blei and P. Moreno. 2001. Topic segmentation with an aspect hidden Markov model. In *Proceedings of the 24th SIGIR*, pages 343–348.

M. A. Hearst. 1997. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

A. McCallum, K. Nigam, J. Rennie, and K. Seymore. 1999. A machine learning approach to building domain-specific search engines. In *IJCAI-1999*.

A. McCallum, D. Freitag, and F. Pereira. 2000. Maximum entropy Markov models for information extraction and segmentation. In *Proceedings of the 17th ICML*, pages 591–598. Morgan Kaufmann, San Francisco, CA.

H. Pasula, B. Marthi, B. Milch, S. Russell, and I. Shpitser. 2002. Identity uncertainty and citation matching. In *Proceedings of NIPS 2002.*

F. Peng and A. McCallum. 2004. Accurate information extraction from research papers using Conditional Random Fields. In *Proceedings of HLT-NAACL 2004*.

L. R. Rabiner and B.-H. Juang. 1993. *Fundamentals of Speech Recognition*. Prentice Hall.

L. R. Rabiner. 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

K. Toutanova, F. Chen, K. Popat, and T. Hofmann. 2001. Text classification in a hierarchical mixture model for small training sets. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 105–113. ACM Press.

S. Vaithyanathan and B. Dom. 2000. Model-based hierarchical clustering. In *UAI-2000*.