ENTITY	r_{seed}	TASK	$\mathcal{D}$	$\lvert\mathcal{D}\rvert$	$\lvert\mathcal{M}(r_{seed}, \mathcal{D})\rvert$
DATE	`\d{2}/\d{2}/\d{2}`	DATE$_{\text{MidEast}}$	talk.politics.mideast	1k	7
		DATE$_{\text{WebKB}}$	WebKB	2k	86
		DATE$_{\text{Enron}}$	Enron	100k	25654
PHONE NUMBER	`\(\d{3}\)\d{3}-\d{4}`	PHONE$_{\text{ForSale}}$	misc.forsale	1k	88
COURSE NUMBER	`CS\d{3}`	COURSE$_{\text{WebKB}}$	WebKB	2k	1348
PHONE NUMBER	`\d{3}-\d{3}-\d{4}`	PHONE$_{\text{Enron}}$	Enron	100k	28994

Table 2: Extraction Tasks Overview. The seed regex for DATE is common on all corpora.

TASK	PRECISION				RECALL				F-SCORE				
	FREQ	GM$_{10}$	GM$_{1\%}$	OURS	$\mathcal{M}(r_{seed}, \mathcal{D})$	GM$_{10}$	GM$_{1\%}$	OURS	$\mathcal{M}(r_{seed}, \mathcal{D})$	FREQ	GM$_{10}$	GM$_{1\%}$	OURS
DATE$_{\text{MidEast}}$	0.009	0.009	0.045	**0.351**	0.072	0.072	0.072	**0.948**	0.135	0.017	0.016	0.055	**0.513**
DATE$_{\text{WebKB}}$	0.032	0.186	0.479	**1.000**	0.251	0.436	0.330	**0.857**	0.402	0.063	0.261	0.391	**0.923**
DATE$_{\text{Enron}}$	0.038	**0.982**	0.965	0.913	0.608	0.619	0.624	**0.887**	0.756	0.072	0.759	0.758	**0.900**
PHONE$_{\text{ForSale}}$	0.462	0.224	0.517	**0.984**	0.169	0.257	0.236	**0.483**	0.289	0.621	0.239	0.324	**0.648**
COURSE$_{\text{WebKB}}$	0.070	0.672	0.633	**0.994**	0.342	0.348	0.349	**0.855**	0.509	0.131	0.459	0.450	**0.919**
PHONE$_{\text{Enron}}$	0.118	**0.977**	0.970	0.766	0.684	0.687	0.687	**0.830**	0.812	0.211	**0.807**	0.805	0.796

Table 3: Comparative evaluation of accuracies of match set expansion algorithms

estimation of context features using the learned language models. The Levenshtein automaton from Schulz and Mihov (2002) makes $\mathcal{E}_t(m)$ estimation over all matches in $\mathbb{M}$ linear in $|\mathbb{M}|$. The linearity in $|\mathbb{M}|$ holds for the logistic regression training as well as the partial sort (Martınez 2005) in Line 8; note that k is a fraction of the size of the seed regex matches, which is much smaller than $\mathbb{M}$. With all the intra-iteration steps being linear, the sequence of iterations for the match set expansion is in $\mathcal{O}(num \times |\mathbb{M}|)$. Given the regex matches identified in the match expansion phase, the final regex recommender phase is much simpler and is in $\mathcal{O}(|\mathcal{G}^d_{r_{seed}}|)$ for fixed output sizes. We search over the generalization space $\mathcal{G}^d_{r_{seed}}$ much like in Murthy, P., and Deshpande (2012); heuristically limiting the size of $\mathcal{G}^d_{r_{seed}}$ by excluding unpromising generalization sub-trees would help scale the search to higher values of d.

Experiments

Experimental Setup

Extraction Tasks: We perform our empirical evaluation on a variety of extraction tasks over multiple real-world document corpora as shown in Table 2. The talk.politics.mideast and misc.forsale corpora are taken from the 20 Newsgroups dataset[6], whereas the Enron corpus is a random subset of 100k documents from the Enron Email Dataset[7]. The WebKB corpus[8] is another popular document dataset.

Baselines: Our empirical evaluation uses three baseline methods. A simple baseline, FREQ, labels all matches by regexes in $\mathcal{G}^d_{r_{seed}}$ as correct. GM$_{10}$ is the approach from Gupta and Manning (2014) using the recommended expansion rate of 10 instances per iteration. GM$_{1\%}$ is the adaptation of the same method, using the same expansion rate as our method, which is 1%. As stated earlier, the method from Gupta and Manning (2014) requires a set of positive instances to start the learning; we set it to $\mathcal{M}(r_{seed}, \mathcal{D})$.

Gold Standard: Our evaluation requires instances labeled as correct, for each extraction task. Since manual annotation of each token over thousands of documents is prohibitively expensive, we got labelings done by human annotators in two phases. In the first phase, our annotators labeled every regex in $\mathcal{G}^d_{r_{seed}}$ as either correct or incorrect with respect to the entity being considered. In the second phase, all matches of these correct regexes were shown to the annotators for getting labelings at the level of matches. Regexes labeled correct in the first phase and then found to have incorrect matches were specialized to regex(es) that span only correct matches. Thus, all correct regexes match only correct instances; these regexes are listed in the supplementary material. The matches by such correct regexes are used as the gold standard instances for evaluation purposes. The annotators also labeled the instances learned by the baselines, and the correct ones were added to the gold standard instances.

Parameters used: Our method uses three parameters: d, num, and p. We set these to 4, 150, and 1%, unless otherwise stated. We separately study the performance of our method across variations in these parameters.

Comparison of Match Set Expansion Stage

Table 3 illustrates the results from our comparative evaluation. As outlined earlier, we are interested in a good trade-off between precision and recall, and thus, a high F-score. FREQ peaks on recall with very low precision, whereas just considering $\mathcal{M}(r_{seed}, \mathcal{D})$ as the results results in the reverse behavior; since these were observed to achieve 1.0 on recall and precision respectively, those metrics are not shown in the table for brevity. Our method (**OURS**) beats the baselines on five of six extraction tasks, posting F-scores as much as 0.9 on three of them. On the sixth, it is seen to trail the leading method very closely. This illustrates the effectiveness of our modeling and establishes our method as the preferred method for the task.

Discussion: The baseline approach from Gupta and Manning (2014) identifies patterns of context words to characterize correct matches. This coarse-grained pattern-based framework does not allow discriminating between matches that bear context differences, but fall within the same pattern(s). Our context modeling is more fine-grained, with scoring depending on all words in the context window. We believe the enhanced effectiveness of our method is explained by the nature of our context modeling and blending it with content modeling in a logistic regression framework, as against just averaging the scores as Gupta and Manning (2014).

[6] http://qwone.com/~jason/20Newsgroups/

[7] https://www.cs.cmu.edu/~./enron/

[8] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-51/www/co-training/data/