Accelerating Human Authorship of Information Extraction Rules

Dayne Freitag, John Cadigan, John Niekrasz, Robert Sasseen
SRI International
9988 Hibert Street, Suite 203
San Diego, CA 92131 USA
firstname.lastname@sri.com

Abstract

We consider whether machine models can facilitate the human development of rule sets for information extraction. Arguing that rule-based methods possess a speed advantage in the early development of new extraction capabilities, we ask whether this advantage can be increased further through the machine facilitation of common recurring manual operations in the creation of an extraction rule set from scratch. Using a historical rule set, we reconstruct and describe the putative manual operations required to create it. In experiments targeting one key operation—the enumeration of words occurring in particular contexts—we simulate the process of corpus review and word list creation, showing that several simple interventions greatly improve recall as a function of simulated labor.

1 Introduction

To maximize accuracy and robustness under the state of the art in information extraction (IE), one trains machine learning (ML) models, typically underpinned by neural language models, on large numbers of sentence-level annotations (Ma and Hovy, 2016; Zhang et al., 2018; Wadden et al., 2019). If annotations are sufficiently numerous, this approach yields robust extraction capabilities that are difficult to implement through other means. And it has methodological advantages, inasmuch as the annotations serve as a precise extensional definition of a given extraction problem, one that can be trivially exploited in the development of improved extractors through new learning algorithms and architectures.

However, this approach imposes certain costs that are not immediately apparent and that manifest as diminished agility, including:

- *Labor overhead*. Success depends critically on consistent annotation at scale, often requiring a team of trained annotators, the development of clear annotation guidelines, and the employment of a review process.
- *Domain fragility*. The resulting extractors are often domain- or genre-specific, suffering substantial degradation when applied to texts from different, even adjacent, domains. Recent research on domain transfer and few-shot learning offers mitigations (e.g., Huang et al. 2020), but techniques from this research often can only be applied to problems proximal to those for which annotations exist, and often result in models with lower accuracy. Typically, additional annotation is required (Bai et al., 2022).
- *Use case myopia*. These challenges push the IE research community toward problems of putative general utility, such as named entity recognition. To the extent that these "canonical" problems target relatively complex information (e.g., event recognition), they suffer substantial practical limitations. For example, the set of event types encountered in news reporting is practically unbounded, while the types distinguished in canonical resources number in the dozens (LDC, 2005).

Most concerningly, the community's shared focus on a small number of canonical problems, while it fosters replicability and fundamental progress, inhibits progress on methods that would enable the practical deployment of IE on a truly broad range of problems. Many real-world problems involve data or use cases too distinctive to be solved with community models, and many candidate customers of IE lack the resources for adequate data annotation.

Rule-based approaches to IE offer an alternative for the deployment of competent novel extractors (Appelt and Onyshkevych, 1998; Valenzuela-Escárcega et al., 2016). While they suffer from certain limitations—limited retargetability, reduced

Proceedings of the 29th International Conference on Computational Linguistic, pages 45–55
October 12–17, 2022.

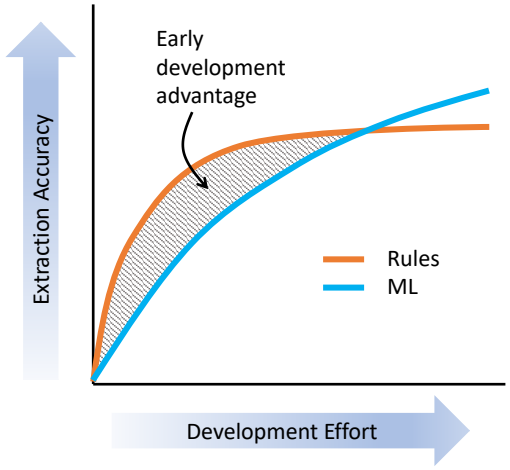

Figure 1: A notional deployment curve comparing the accuracy of rule-based and ML-based extractors as a function of labor investment.

recall, etc.—they possess one significant advantage over ML-based approaches, as illustrated in Figure 1. Specifically, in the early stages of an effort pursuing novel extractors, they support very rapid deployment. Hours of effort often suffice to implement usable extractors, where an equivalent ML-based extractor would require days or weeks. We argue that this "early deployment advantage" makes rule-based IE an important tool in real-world settings. Importantly, rule-based methods and ML are not mutually exclusive. We have previously presented evidence that rule-based extractors can be used to *annotate* training data for ML, and that the resulting models generalize the rules in useful ways (Freitag et al., 2022).

In this paper, we consider a tighter integration between rule-based IE and ML, one in which ML *facilitates* the authorship of rules by offering options and suggestions to the human technician. In a new extraction problem area lacking annotations, the rule author is confronted with a difficult search problem—a difficulty that increases with the expressiveness of the rule language. We hypothesize that ML can be used to simplify the search in ways that dramatically reduce effort. This paper is an attempt to illuminate the dimensions along which such assistance is possible. We approach this through analysis of a historical rule set for extracting quantitative claims from the scientific literature on solar materials. By inspecting how various language features were used in pursuit of a performant extraction model, we attempt to infer some of the operations employed by the author in the initial construction and subsequent refinement of an improving rule set. And we provide preliminary quantitative evidence that some simple interventions could have substantially accelerated a key operation: the creation of problem-specific word lists.

To summarize, we make the following contributions in this paper:

- We introduce the concept of *facilitated rule authorship* for information extraction, a research objective with the potential to dramatically decrease the cost of deploying performant IE on new problems.
- We use a historical extraction rule set to illuminate the operations that human authors employ in their search through the space of possible rule sets. Our intent is to focus attention on human deficits that might be mitigated through focused application of ML.
- We conduct experiments to address one such deficit, the creation of problem-specific word lists, and provide quantitative estimates of the labor savings that can be realized through various approaches to facilitation.

2 Related Work

The use of declarative, efficiently executable rules for information extraction was a common feature of early work in the area, which led to the creation of several rule frameworks (Appelt and Onyshkevych, 1998; Reiss et al., 2008; Thakker et al., 2009). Motivated by the difficulty of purely manual rule creation, early applications of machine learning to the problem sought to facilitate aspects of the authoring process, particularly the creation of what we call *word sets* and what the literature often calls *dictionaries* or *semantic lexicons* (Riloff, 1993; Soderland et al., 1995). This line of research led to some general methods for exploiting syntagmatic search (contextual patterns) for the assembly of paradigmatic resources (lexicons) (Jones et al., 1999), but by treating the lexicon as an end in its own right, it begged the question of ultimate utility for the downstream task of information extraction.

Early successes in lexicon induction gave rise to research pursuing end-to-end extraction through supervised rule or pattern induction (Freitag, 1998; Soderland, 1999; Freitag and Kushmerick, 2000; Califf and Mooney, 2003). This work offered

the advantage of improved replicability and retargetability, replacing the highly technical activity of rule creation with the more transparent activity of data annotation. However, once annotated data was available in sufficient volumes, rule-based representations were eventually outperformed by less constrained representations better able to integrate diverse signals in the data (Freitag and McCallum, 1999; Lafferty et al., 2001; Collobert et al., 2011).

The center of gravity in subsequent research has focused on models able to exploit large volumes of annotated data and the acquisition of data in sufficient volumes to realize their advantages. Because annotation overhead hampers application of these methods to new problems, the field continues to investigate approaches to reducing annotation overheads, including few-shot learning (Han et al., 2018; Fritzler et al., 2019; Huang et al., 2020) and transfer learning (Wang et al., 2018; Huang et al., 2018; Yang and Katiyar, 2020). In some cases, these methods make it possible to achieve impressive competence in a new task with very few training examples. But note that such approaches, inasmuch as they often transfer extraction knowledge from known target types or from highly resourced domains to adjacent ones, do not eliminate the need for annotation. And it is often questionable whether the resulting models are sufficiently performant for downstream use without supplementation.

If rule-based approaches to extraction have ceased to be a major research focus, they remain an important tool in the toolkit of practitioners (Chiticariu et al., 2013) and available as features of several general-purpose NLP toolkits (Thakker et al., 2009; Kluegl et al., 2016; Honnibal et al., 2020). Although new rule frameworks occasionally feature in the more recent literature (Chang and Manning, 2014; Valenzuela-Escárcega et al., 2016; Khaitan et al., 2008; Krishnamurthy et al., 2008), these works are almost exclusively descriptive, failing to provide empirical benchmarks that would facilitate continued research in the area. In particular, the process of authoring rules has received no prior empirical scrutiny, making it difficult to corroborate perceived advantages of rule-based methods.

3 A Historical Rule Set

As part of a project attempting to document progress in solar materials research, we developed an extractor for quantatitive "claims," statements that communicated some important scientific measurement. Our experimental data consisted of approximately 160K abstracts from the Web of Science[1] on solar energy research from 1968 to 2014. As is typical in projects like this, IE was not the focus of the effort, but only a means to assemble structured data for downstream analysis, which in this case sought to summarize diachronic progress on several key research dimensions.

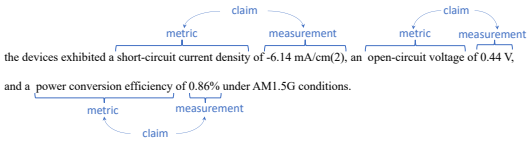

Figure 2: Several examples of the *claim* relation in a sentence from the solar energy literature.

As shown in Figure 2, a *claim* is a binary relation between two domain-specific entities or concepts: a quantitative expression or *measurement*, and the corresponding *metric* or quantity being measured. For greatest flexibility in downstream analysis, our definition of *claim* was inclusive, encompassing any expression reflecting the result of a scientific measurement. As the example in the figure illustrates, the two phrasal extraction targets pose different challenges. Measurements, consisting typically of a number and a unit of measurement, exhibit strong orthographic regularities, parts of which could be exploited with regular expressions. Metrics, on the other hand, are noun phrases.

To address this extraction challenge, we employed VALET, a recently described IE rule syntax and framework implemented in python (Freitag et al., 2022). The earlier version of VALET used in this work lacked several of the features of the current framework. In particular, the rule author had no access to syntactic information. Thus, the problem of extraction amounted to scanning tokens in the input stream sequentially, relying on orthographic and lexical clues to decide when the left and right boundaries of the two phrasal targets were observed. We briefly describe VALET's provisions for such scanning to simplify later exposition. Readers interested in more detail or a review of VALET's more recent features are referred to the paper or the more extensive documentation in the public release.

A statement or rule in VALET is a sequence of

[1] https://clarivate.com/webofsciencegroup/solutions/web-of-science/

Type	Example
regex	`/^[a-z]/i`
set	`{ a an the }i`
reference	`&myclass`

Table 1: The types of atomic token class expression available in this study.

three things: a name, a piece of syntax indicatin the type of rule, and an expression defining th rule's behavior. The evaluation of such a statement yields an *extractor*, which can be applied directly to text (e.g., via scripts from the command line) or incorporated into subsequent statements through reference to the rule's name.

The rules in this study relied on two types of expressions, *token class expressions* and *phrase expressions*. The example token class expression

determiner: { a an the }i

defines a case-insensitive extractor matching the individual words listed between the braces. Table 1 lists the full set of atomic token class expressions historically available to the rule author. A full token class expression is a Boolean combinations of these classes using the operators `and`, `or`, and `not`. Thus, the token class

notdet: not &determiner

matches any token that is not a determiner.

Phrase expressions employ a regular expression syntax to match multi-token sequences, enabling the rule author to mix previously defined token classes with literal tokens. In addition, phrase expressions can co-refer, enabling context-free composition. Consider the rules:

```
cap : /^[A-Z]/
honor : { Dr Mr Mrs Ms }
caps -> &cap+
person -> &honor .? @caps
```

The `person` phrasal extractor in this example recognizes person mentions prefixed by an honorific, incorporating a separate phrase extractor for sequences of capitalized tokens (`caps`) by reference. (Unlike in standard character-level regular expressions, the optional '.' has no special significance and matches period tokens in the input literally.)

The rule set used to extract claims consists of 34 rules (15 token class expressions and 19 phrasal extractors). We next review the structure of this rule set and consider what it implies about the process of human rule development.

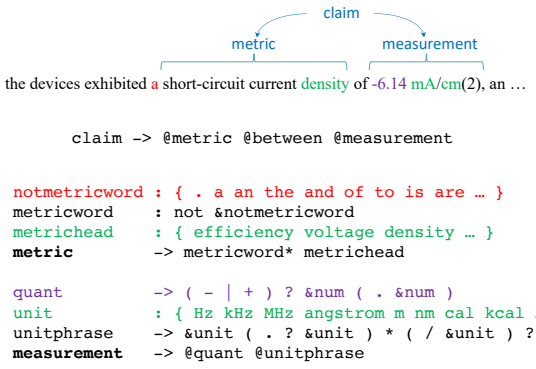

Figure 3: An excerpt of the rule set for extraction quantitative claims.

4 Rule Search

A human technician confronting a new extraction problem faces a daunting challenge. Even if they possess the means to observe the effects of any changes to rules, a protracted exploratory process is required to arrive at an effective solution to any extraction problem other than the most trivial. To understand where ML and automation might facilitate that process, we first seek to form an intuition regarding the solution structure for extracting claims, as representative of a broader class of similar problems, then enumerate potential points of intervention in the putative search that produced this solution.

4.1 Rule Set Structure

Figure 3 presents an excerpt from the most productive portion of the claims rule set in simplified form. In this segment, a claim is a `metric` expression separated by intervening language (captured by the `between` rule, not shown in the figure) from a following `measurement`. The key rules implementing `metric` and `measurement` are shown at the bottom of the figure, with coloring to draw attention to how several key components align to the example text.

The phrase highlighted in purple is an example of an extraction constituent exploiting orthographic regularities. The appropriate structure of the numeric portion of a `measurement` follows very predictable patterns and is amenable to succinct characterization. As a consequence,

the corresponding rule provides high-recall access to good candidate regions where mentions of `measurement` and `claim` might be found. Such substructures provide a natural starting point at the beginning of rule set construction, providing the technician a means to review a large number of candidate expressions to form initial intuitions about the nature of a given extraction problem.

The rules in green correspond to a very common feature in rule-based extraction models: essentially special-purpose lexicons that list precisely the tokens that may occur in a particular context. Such a feature is critical for the identification of `metric` mentions, which lack the orthographic clues afforded by `measurement` mentions. In this technical domain, the concepts subject to measurement are practically finite, and variations in metric are often indicated through qualifiers prepended to a key head word (e.g., by prepending the phrase "short-circuit current" to "density"). Note that even if the set of possible head words is finite, it need not be small. Thus, if the rule author opts to approach an entity recognition challenge through enumeration, they still face a signficant challenge in many cases.

Finally, the rule in red employs a similar strategy toward a different objective. Specifically, it lists a set of stop words that a `metric` phrase may *not* contain and indirectly defines the start of the phrase as the first word following this boundary class. This objective can be addressed more conveniently through reference to parts of speech—something supported in more recent versions of VALET—but both the problem of delimiting mentions and the strategy of explicit exclusion are relevant in any rule writing endeavor.

4.2 Search Operations

Although we only possess the final product, we are now in a position to infer a plausible sequence of steps by which this rule set was created. Figure 4 depicts such a sequence, with colors to distinguish the various textual regions and classes of operations that were involved. While the actual sequence is unknown, the required activities or operations can be inferred with certainty from the structure of the ultimate rule set. In this section, we describe each of these operations and speculate about opportunities for automation or facilitation.

4.2.1 Anchoring (■)

The starting point for our extraction of claims is the numeric portion of the `measurement`, which, as noted, is suggestive of the presence of a claim and largely accessible through surface features. We write a simple rule that matches any numeric token —a rule that overgenerates by design—and use it to inspect `measurement` candidates. Although we show only a single match of this putative rule, it would presumably match multiple spans in the example. However, what matters is that, to a first approximation, this simple rule matches all textual regions in which we might expect a `claim` to appear. Note that this step depends heavily on the technician's intuition and is difficult to automate in settings lacking annotated data.

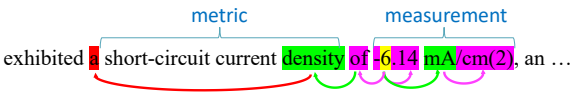

Figure 4: A likely sequence of operations in authoring a rule set to extract claims, including anchoring (■), elaboration (■), positive word set (■), and negative word set (■).

4.2.2 Elaboration (■)

This syntagmatic operation can be *internal* or *contextual* and involves extension beyond the match boundaries returned by a current rule. In the example, if `6` is the anchoring match, an obvious first step is to elaborate the rule so that it encompasses the entire phrase `-6.14`. Here, we have an early rule that reliably matches tokens or sub-phrases of an extraction target, and we use it to elaborate the *internal* structure of the target.

The interstitial text between measurements and metrics (`of` in this example) provides an example of *contextual* elaboration. We use anchoring numeric expressions to investigate and characterize the "bridge" language that separates our two target entities. If, as is often the case, this language is highly stereotpyical and expressed in a vocabulary of manageable size, we can create a rule for it in a way analagous to our elaboration of the anchoring numeric expression.

The operation of elaboration, which we have defined rather coarsely, almost certainly involves more specific actions that are currently difficult to articulate, but a thought experiment might point the way to forms of machine facilitation possible in the short term. Consider the state of affairs after the initial seed rule and the fact that we match `6` but want a rule matching `-6.14`. Instead of editing the rule directly, the technician might indicate a handful of

elaborations, say by dragging their cursor over the complete number phrase in each case, hoping that the system can suggest an accurate elaboration.

The resulting problem resembles grammar induction (Lang et al., 1998), but has some features distinct from the typical framing of that endeavor. For one thing, the examples are embedded, and the surrounding text, which must not be matched by the final rule, provides useful constraint on any proposed "grammar." Second, although the number of ground-truth examples (those touched by the human) may be small, the number of candidate examples can be huge. If the rule author indicates that `6` should be followed by `.14`, we can in principle notice that the pattern *num.num* is a common theme in the textual regions selected by our anchoring rule. Finally, the induction process has access to a rich and extensible set of elements, as well as a human collaborator to assist in choosing them (or proposing new ones). For example, we have already defined a class of numeric tokens to implement our seed rule, which is available, where appropriate, for characterizing the `14` part of our example expression. Similarly, when we turn our attention to units of measurement, we may define a class that includes both `mA` and `cm`, affording the induction algorithm an easy path to elaborate a rule matching individual unit tokens (e.g., `mA`) to the extended syntax exemplified in the figure (`mA/cm(2)`).

4.2.3 Enumeration (■ and ■)

This paradigmatic operation can be used to address two opposing needs. When we pursue *positive enumeration* (■), we are attempting to specify exactly the set of tokens that may appear in some context in an extraction target, such as the head word of `metric` phrases or possible unit abbreviations in `measurement` phrases (the rules shown in green in Figure 3). In constrast, *negative enumeration* (■) is akin to the definition of stop word lists and can be used to delimit extraction targets, as in the rule shown in red in Figure 3.

In contrast to elaboration, enumeration is a well-defined activity, one that should be readily amenable to machine facilitation. We possess at least two levers that might be used to implement such facilitation. First, if the rule author's approach is to build out from a core component, as in Figure 4, the resulting word sets will be populated with the tokens occurring in proximity to our currently implemented rule set. For example, once we can recognize the numeric portion of a `measurement` accurately, we can tabulate the tokens that tend to follow such expressions (perhaps ranking them by pointwise mutual information with the numeric expression) to derive a noisy word list that can be quickly reviewed and codified in a new token class.

A generalization of this approach, and an approach ultimately offering more flexibility, is to exploit corpus co-occurrence statistics to infer lexical affinities (e.g., through distributional distances or embeddings). Using an authoring framework equipped with such information, a technician might point at a token in context and be presented with a list of semantically comparable tokens, again with the option of selecting some subset to define a new token class.

5 Experiments

5.1 Framing

Our experiments investigate the feasibility and value of automated facilitation of word set enumeration. We simulate a rule author constructing the two word sets shown in green in Figure 3, one for `metric` head words (`metrichead`) and one for units of measurement (`unit`). We investigate two settings. In one, we suppose that before this process begins, the user has created a high-recall anchoring rule that captures some aspect of the context in which the new class of words is expected to appear. In the other setting, there is no such anchor, and the user must rely on other means to find good candidate inclusions.

Unassisted, the user must scan the corpus sequentially, considering candidate words the nominating procedure provides. This is our unit of cost: the review of an individual word for inclusion or exclusion. As each new word is added to the set, our recall of claims improves. Our experiments investigate precisely this trade-off: How can we maximize recall while minimizing human effort? We measure two forms of recall: *word recall*, or the fraction of words found in the respective ground-truth word set; and *claim recall*, the fraction of ground-truth claims found when the current word set is used in place of the ground-truth one. Note that our experiments consider only the situation in which the user has access to some nomination procedure. If a nomination procedure is entirely lacking, labor requirements are presumably higher than any of our experimental alternatives.

For cases where there is no anchoring rule, we must rely on knowledge of the current word set and corpus analysis to suggest additions. For this variant we imagine an iterative setting in which, at each step, the system analyzes the current version of a partial word set and draws on its models of the corpus to present a ranked list of candidate words additions to the user. The user then repeatedly scans down this list until a single good addition is found, then requests a new list. For both of our targets, the initial word set is a singleton containing the first `metric` head word (or `unit`, respectively) encountered in a sequential scan of the corpus.[2]

We experiment with two approaches to unanchored candidate ranking. In each case, we compute a ranking over all terms in the corpus vocabulary that are not in the partial word set and not previously reviewed. The first approach (call it *FT-centroid*) uses FastText embeddings. We rank all possible additions according to their cosine distance from the centroid of the words in the current word set. We also experimented with a variant, *FT-max*, that uses *maximum* cosine similarity. This variant produced results comparable to *FT-centroid*.

The second approach (call it *IT-set*) employs an information-theoretic analysis, where each word in the corpus vocabulary is represented as a distribution over observation contexts. We consider words occurring up to two tokens removed from the reference observation, encoding each unique token and offset as a distinct context (e.g., "`the`" one token to the left is a distinct context from "`the`" two tokens to the right of the reference word). The matrix formed from the set of such distributions is then submitted to a coclustering operation that groups rows and columns while minimizing Shannon information loss. This results in a dense distributional embedding for each word as a distribution over context clusters. We then compute the Hellinger distance between all word pairs and rank all candidate word set additions in descending order by mean distance from words in the current word set.

For anchored review, we introduce the *count* method which simply ranks matches to the anchor rule according to their marginal corpus frequency, suppressing any that the user has already reviewed. To simulate the putative process the historical author followed, we introduce *sequential*, a variant

[2]This is "temperature" for `metrichead` and "degrees" for `unit`.

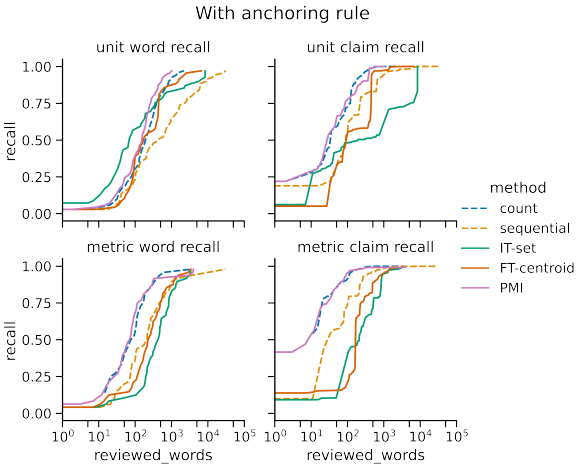

Figure 5: Word and claim recall as a function of words reviewed, using an anchoring rule for nomination.

that considers candidates in corpus order and performs no tracking of already reviewed words. In this case, a word may be reviewed more than once.

Building on the *count* approach, we also explore a point-wise mutual information variant (*PMI*). Instead of ranking by count, we rank by pointwise mutual information between word occurrence and matches of the anchoring rule. This closely follows the counting approach but has extra information about how frequently a word matches the anchoring rule. Finally, we experiment with an anchored variant of the rankers (*IT-set* and *FT-centroid*) that limits their nominations to words proximal to the anchor rule.

5.2 Results

Figure 5 presents results from our experiments employing anchoring rules. In these experiments, the rule used for `unit` nominated any word immediately following a numeric expression. The rule for `metrichead` uses the same numeric expression rule, extended with the rule used to model the intervening text typically found between such expressions and a preceding metric head word (e.g., the word "of" in Figure 3). Obviously, the anchor we use for `metrichead` is more selective than that for `unit`. In the plots, we use a dashed style for *count* and *sequential*, which, because of their simplicity, are useful baselines in both sets of experiments.

As the results make clear, this simplicity does not imply inferior performance. In an outcome that represented something of a surprise for us, corpus-analytic rankers offer benefit to the process of word

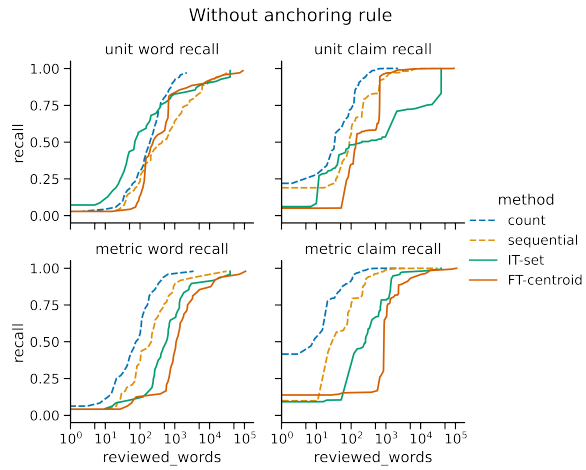

Figure 6: Word and claim recall as a function of words reviewed in the absence of an anchoring rule. The methods *count* and *sequential* do use such a rule and are included in the plot for the sake of comparison.

set enumeration only under limited conditions. In particular, early in the process, *IT-set* apparently nominates more pertinent `unit` words, but the effect disappears as the word set is built out and (crucially) does not apply to claim recall, arguably the more important metric. If the objective is not to find a good set of words alone, but instead to find a set that maximizes extraction recall, it is difficult to improve on a review prioritized by corpus frequency. *PMI* adds incremental benefit in some cases and does not appear to hurt on balance. The key appears to be the selection of a good anchoring rule.

Figure 6 displays the results of our experiments lacking an anchor rule (except for the dashed lines, which are included to make comparison with anchor-based methods easier). Here, *IT-set* continues to display its relative strength on the word recall metric, but the results for claim recall are much more ambiguous. More work is required to resolve this ambiguity, which is relevant to very agile deployment. In cases where reasonable recall is desired as early as possible, we care about, say, the 0.5 or 0.75 recall levels in the plots. Our experiments lead to no clear recommendation for this use case. Presumably, what is required is a variant of these methods that incorporates corpus frequency more prominently into the score used in ranking.

6 Discussion

This work is an initial step in a line of inquiry that could lead to better tooling in support of more agile extraction. The key insight is that once we have a performant rule set, one that we are willing to treat as authoritative, we can simulate the process that led to its creation and experiment with new modes of facilitation in pursuit of greater labor savings and model robustness. Critical to such research, and a focus of future work, is a credible cost model that quantifies levels of authoring effort. Not only would such a model provide a more precise characterization of the "early deployment advantage" of rules over ML, but it could help widen this advantage as an objective function for simulations of the authoring process.

Of course, this approach has certain shortcomings. For one, any model, including our historical rule set, that is not developed and vetted against a thoroughly annotated data sample is typically an approximation, usually one that is recall-limited. In our previous work, we sought to overcome this limitation by using the rule set to generate a large annotated sample to train a high-recall sequence labeler (Freitag et al., 2022). Here, we treat the rules as definitional, but it seems clear that some of the "false positive" elements nominated by our corpus-analytic rankers belong in the definition. For example, only one of the top ten terms nominated for `metrichead` by *IT-set* after two iterations of review was in the historical word set, but many of the excluded nominations appear plausible (e.g., *reflectance*, *oxidation*, or *transmittance*). Many of these words presumably occur rarely (if at all) as part of claim expressions, and our performance metric's emphasis on maximizing recall punishes rankers that promote terms in the tail of the distribution, but a complete account of claim language in this domain might want to include them.

A salient feature of all of these results is our ability to reach full recall quickly using a high-quality anchoring rule and a relatively simple ranking policy. But this outcome may in partly reflect a circularity in the experimental methodology. Our anchoring rules are elements of the historical model, and they therefore necessarily enable us to review all sentences that the rule set considers relevant. A key unanswered question is: what do these anchors miss? Our previous work, which used this rule set to train an ML extractor, yielded apparently valid claim expressions that the rule set does not sanction (Freitag et al., 2022). Perhaps methods such as *IT-set* and *FT-centroid*, which seem wasteful of human effort, can be used to identify alternative or

develop more general anchors.

More generally, the structure of a historical rule set is a reflection of the rule language and tooling available to the author, and conclusions drawn from a study of such a rule set may overlook promising points of integration between rule-based methods and machine learning or corpus analytics. For example, the current VALET framework supports on-demand application of *IT-set* via an interactive dialog presenting a large list of words deemed to be close to a chosen word in the text. The user can select any of the words in the list and ask the development UI to generate a new word set expression. Similarly, VALET offers a "radius" statement that matches words within some distance of a seed set in lexical embedding space. And we have begun investigating a trainable word set feature that engages the user in an active learning loop to derive a customized word matcher, one that can in principle exploit contextual embeddings.

While such features are potentially powerful, they sacrifice transparency and fine-grained control–two attractive aspects of rule-based methods. In this respect, they are in the tradition of alternative approaches to rapid IE deployment, such as Snorkel (Ratner et al., 2017), which seeks to learn performant extractors from collections of noisy "labeling functions." Such approaches, for problems on which they work, can lead to impressive labor savings, but they are difficult to control and optimize. But note that while Snorkel-like approaches and traditional rule-based methods approach the IE objective from different angles—Snorkel through redundant, high-recall labelers, rule-based methods through high-precision set covering—they are fundamentally compatible and offer interesting opportunities for hybridization. Trivially, a framework like VALET can be used to conveniently implement labeling functions. By the same token, Snorkel points the way to a mode of rule set application distinct from the typical disjunctive mode.

7 Conclusion

Rule-based methods remain an important component of any toolset addressing the broader problem of information extraction, especially in cases where existing extraction models or sources of annotated data are misaligned to new use cases. A trained technician, outfitted with a suitable rule authoring framework, can create a performant extractor for a new problem in a fraction of the time required to produce a ML model of comparable accuracy. Moreover, we have shown that some simple facilitations, based on an analysis of the rule authoring process, can serve to increase this "early deployment advantage." And by treating rule development as the focus of empirical investigation, we have pointed the way toward future systems in which rules and ML are combined creatively to lower the barrier to entry in the creation of custom extraction solutions.

References

Douglas E. Appelt and Boyan Onyshkevych. 1998. The common pattern specification language. Technical report, SRI International.

Fan Bai, Alan Ritter, and Wei Xu. 2022. Pre-train or Annotate? Domain Adaptation with a Constrained Budget. ArXiv:2109.04711 [cs].

Mary Elaine Califf and Raymond J. Mooney. 2003. Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research*, 4(Jun):177–210.

Angel X. Chang and Christopher D. Manning. 2014. TokensRegex: Defining cascaded regular expressions over tokens. Technical report, Stanford University Computer Science Department.

Laura Chiticariu, Yunyao Li, and Frederick R. Reiss. 2013. Rule-Based Information Extraction is Dead! Long Live Rule-Based Information Extraction Systems! In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 827–832, Seattle, Washington, USA. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537.

Dayne Freitag. 1998. Toward general-purpose learning for information extraction. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 404–408.

Dayne Freitag, John Cadigan, Robert Saseen, and Paul Kalmar. 2022. VALET: rule-based information extraction for rapid deployment. In *13th Conference on Language Resources and Evaluation (LREC 2022)*.

Dayne Freitag and Nicholas Kushmerick. 2000. Boosted wrapper induction. In *Proceedings of AAAI-2000*.

Dayne Freitag and Andrew McCallum. 1999. Information extraction with hmms and shrinkage. In *Proceedings of the AAAI-99 workshop on machine learning for information extraction*, pages 31–36. Orlando, Florida.

Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 993–1000.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python.

Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. 2020. Few-shot named entity recognition: A comprehensive study. *arXiv preprint arXiv:2012.14978*.

Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-shot transfer learning for event extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170.

Rosie Jones, Andrew McCallum, Kamal Nigam, and Ellen Riloff. 1999. Bootstrapping for text learning tasks. In *IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications*.

Sanjeet Khaitan, Ganesh Ramakrishnan, Sachindra Joshi, and Anup Chalamalla. 2008. Rad: A scalable framework for annotator development. In *2008 IEEE 24th International Conference on Data Engineering*, pages 1624–1627. IEEE.

Peter Kluegl, Martin Toepfer, Philip-Daniel Beck, Georg Fette, and Frank Puppe. 2016. UIMA Ruta: Rapid development of rule-based information extraction applications. *Natural Language Engineering*, 22(1):1–40. Publisher: Cambridge University Press.

Rajasekar Krishnamurthy, Yunyao Li, Sriram Raghavan, Frederick Reiss, Shivakumar Vaithyanathan, and Huaiyu Zhu. 2008. SystemT: A System for Declarative Information Extraction. *SIGMOD Record*, 37(4):7.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*.

Kevin J. Lang, Barak A. Pearlmutter, and Rodney A. Price. 1998. Results of the Abbadingo one DFA learning competition and a new evidence-driven state merging algorithm. In *Grammatical Inference*, pages 1–12. Springer.

LDC. 2005. *ACE (Automatic Content Extraction) English Annotation Guidelines for Events*. Linguistic Data Consortium.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074.

Alexander J. Ratner, Stephen H. Bach, Henry R. Ehrenberg, and Chris Ré. 2017. Snorkel: Fast training set generation for information extraction. In *Proceedings of the 2017 ACM international conference on management of data*, pages 1683–1686.

Frederick Reiss, Sriram Raghavan, Huaiyu Zhu, Frederick Reiss, Sriram Raghavan, Rajasekar Krishnamurthy, Huaiyu Zhu, and Shivakumar Vaithyanathan. 2008. An algebraic approach to rule-based information extraction. In *In ICDE*, pages 933–942. IEEE.

Ellen Riloff. 1993. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI)*.

Stephen Soderland. 1999. Learning information extraction rules for semi-structured and free text. *Machine learning*, 34(1):233–272. Publisher: Springer.

Stephen Soderland, David Fisher, Jonathan Aseltine, and Wendy Lehnert. 1995. CRYSTAL: inducing a conceptual dictionary. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*.

Dhaval Thakker, Taha Osman, and Phil Lakin. 2009. GATE JAPE grammar tutorial.

Marco A Valenzuela-Escárcega, Gus Hahn-Powell, and Mihai Surdeanu. 2016. Odin's runes: A rule language for information extraction. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 322–329.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789.

Zhenghui Wang, Yanru Qu, Liheng Chen, Jian Shen, Weinan Zhang, Shaodian Zhang, Yimei Gao, Gen Gu, Ken Chen, and Yong Yu. 2018. Label-aware double transfer learning for cross-specialty medical

named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1–15.

Yi Yang and Arzoo Katiyar. 2020. Frustratingly simple few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6365–6375.

Yuhao Zhang, Peng Qi, and Christopher D Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215.