Open Knowledge Extraction through Compositional Language Processing

Benjamin Van Durme
Lenhart Schubert
University of Rochester (USA)
email: `vandurme@cs.rochester.edu`

Abstract

We present results for a system designed to perform *Open Knowledge Extraction*, based on a tradition of compositional language processing, as applied to a large collection of text derived from the Web. Evaluation through manual assessment shows that well-formed propositions of reasonable quality, representing general world knowledge, given in a logical form potentially usable for inference, may be extracted in high volume from arbitrary input sentences. We compare these results with those obtained in recent work on Open *Information* Extraction, indicating with some examples the quite different kinds of output obtained by the two approaches. Finally, we observe that portions of the extracted knowledge are comparable to results of recent work on *class attribute* extraction.

1 Introduction

Several early studies in large-scale text processing (Liakata and Pulman, 2002; Gildea and Palmer, 2002; Schubert, 2002) showed that having access to a sentence's syntax enabled credible, automated semantic analysis. These studies suggest that the use of increasingly sophisticated linguistic analysis tools could enable an explosion in available symbolic knowledge. Nonetheless, much of the subsequent work in extraction has remained averse to the use of the linguistic deep structure of text; this decision is typically justified by a desire to keep the extraction system as computationally lightweight as possible.

The acquisition of background knowledge is not an activity that needs to occur online; we argue that as long as the extractor will finish in a *reasonable* period of time, the speed of such a system is an issue of secondary importance. Accuracy and usefulness of knowledge should be of paramount concern, especially as the increase in available computational power makes such "heavy" processing less of an issue.

The system explored in this paper is designed for *Open Knowledge Extraction*: the conversion of arbitrary input sentences into general world knowledge represented in a logical form possibly usable for inference. Results show the feasibility of extraction via the use of sophisticated natural language processing as applied to web texts.

2 Previous Work

Given that the concern here is with *open* knowledge extraction, the myriad projects that target a few prespecified types of relations occurring in a large corpus are set aside.

Among early efforts, one might count work on deriving selectional preferences (e.g., Zernik (1992); Resnik (1993); Clark and Weir (1999)) or partial predicate-argument structure (e.g., Abney (1996)) as steps in the direction of open knowledge extraction, though typically few of the tuples obtained (often a type of subject plus a verb, or a verb plus a type of object) can be interpreted as complete items of world knowledge. Another somewhat relevant line of research was initiated by Zelle and Mooney (1996), concerned with learning to map NL database queries into formal DB queries (a kind of semantic interpretation). This was pursued further, for instance, by Zettlemoyer and Collins (2005) and Wong and Mooney (2007), aimed at learning log-linear models, or (in the latter case) synchronous CF grammars augmented with lambda operators, for mapping English queries to DB queries. However, this approach requires annotation of texts with logical forms, and extending this approach to general texts would seemingly require a massive corpus of hand-annotated text — and the logical forms would have to cover far more phenomena than are found in DB queries (e.g., attitudes, generalized quantifiers, etc.).

Another line of relevant work is that on semantic role labelling. One early example was MindNet (Richardson et al., 1998), which was based on collecting 24 semantic role relations from MRDs such as the *American Heritage Dictionary*. More recent representative efforts includes that of Gildea and Jurafsky (2002), Gildea and Palmer (2002), and Punyakanok et al. (2008). The relevance of this work comes from the fact that identifying the arguments of the verbs in a sentence is a first step towards forming predications, and these may in many cases correspond to items of world knowledge.

Liakata and Pulman (2002) built a system for recovering Davidsonian predicate-argument structures from the Penn Treebank through the application of a small set of syntactic templates targeting head nodes of verb arguments. The authors illustrate their results for the sentence “*Apple II owners, for example, had to use their television sets as screens and stored data on audiocassettes*” (along with the Treebank annotations); they obtain the following QLF, where verb stems serve as predicates, and arguments are represented by the head words of the source phrases:

```
have(e1,owner, (use(e3,owner,set), and as(e3,screen)),
           and (store(e2,owner,datum), and on(e2,audiocassette)))
```

For a test set of 100 Treebank sentences, the authors report recall figures for various aspects of such QLFs ranging from 87% to 96%. While a QLF like the one above cannot in itself be regarded as world knowledge, one can readily imagine postprocessing steps that could in many cases obtain credible propositions from such QLFs. How accurate the results would be with machine-parsed sentences is at this point unknown.

In the same year, Schubert (2002) described a project aimed directly at the extraction of general world knowledge from Treebank text, and Schubert and Tong (2003) provided the results of hand-assessment of the resulting propositions. The Brown corpus yielded about 117,000 distinct simple propositions (somewhat more than 2 per sentence, of variable quality). Like Liakata and Pulman’s approach the method relied on the computation of unscoped logical forms from Treebank trees, but it abstracted propositional information along the way, typically discarding modifiers at deeper levels from LFs at higher levels, and also replacing NPs (including named entities) by their types as far as possible. Judges found about 2/3 of the output propositions (when automatically verbalized in English) acceptable as general claims about the world. The next section provides more detail on the extraction system, called KNEXT, employed in this work.

Clark et al. (2003), citing the 2002 work of Schubert, report undertaking a similar extraction effort for the 2003 Reuters corpus, based on parses produced by the Boeing parser, (see Holmback et al. (2000)), and obtained 1.1 million subject-verb-object fragments. Their goal was eventually to employ such tuples as common-sense expectations to guide the interpretation of text and the retrieval of possibly relevant knowledge in question-answering. This goal, unlike the goal of inferential use of extracted knowledge, does not necessarily require the extracted information to be in the form of logical propositions. Still, since many of their tuples were in a form that could be quite directly converted into propositional forms similar to those of Schubert, their work indicated the potential for scalability in parser-based approaches to information extraction or knowledge extraction.

A recent project aimed at large-scale, open extraction of tuples of text fragments representing verbal predicates and their arguments is TextRunner (Banko et al., 2007). This systems does part-of-speech tagging of a corpus, identifies noun phrases with a noun phrase chunker, and then uses tuples of nearby noun phrases within sentences to form apparent relations, using intervening material to represent the relation. Apparent modifiers such as prepositional phrases after a noun or adverbs are dropped. Every candidate relational tuple is classified as trustworthy (or not) by a Bayesian classifier, using such features as parts of speech, number of relevant words between the noun phrases, etc. The Bayesian classifier is obtained through training on a parsed corpus, where a set of heuristic rules determine the trustworthiness of apparent relations between noun phrases in that corpus. As a preview of an example we will discuss later, here are two relational tuples in the format extracted by TextRunner:[1]

> (the **people**) use (**force**),
> (the **people**) use (**force**) to impose (a **government**).

No attempt is made to convert text fragments such as "*the people*" or "*use _ to impose*" into logically formal terms or predicates. Thus much like semantic role-labelling systems, TextRunner is an *information* extraction system, under the terminology used here; however, it comes closer to knowledge extraction than the former, in that it often strips away much of the modifying information of complex terms (e.g., leaving just a head noun phrase).

2.1 KNEXT

KNEXT (Schubert, 2002) was originally designed for application to collections of manually annotated parse trees, such as the Brown corpus. In order to extract knowledge from larger text collections, the system has been extended for processing arbitrary text through the use of third-party parsers. In addition, numerous improvements have been made to the semantic interpretation rules, the filtering techniques, and other components of the system. The extraction procedure is as follows:

1. Parse each sentence using a Treebank-trained parser (Collins, 1997; Charniak, 1999).

2. Preprocess the parse tree, for better interpretability (e.g., distinguish different types of SBAR phrases and different types of PPs, identify temporal phrases, etc.).

3. Apply a set of 80 interpretive rules for computing unscoped logical forms (ULFs) of the sentence and all lower-level constituents in a bottom-up sweep; at the same time, *abstract* and collect phrasal logical forms that promise to yield stand-alone propositions (e.g., ULFs of clauses and of pre- or post-modified nominals are prime candidates). The ULFs are rendered in Episodic Logic (e.g., (Schubert and Hwang, 2000)), a highly expressive representation allowing for generalized quantifiers, predicate modifiers, predicate and sentence reification operators, and other devices found in NL. The abstraction process drops modifiers present in lower-level ULFs (e.g., adjectival premodifiers of nominal predicates) in constructing higher-level ULFs (e.g., for clauses). In addition, named entities are generalized as far as possible using several gazetteers (e.g., for male and female given names, US states, world cities, actors, etc.) and some morphological processing.

4. Construct complete sentential ULFs from the phrasal ULFs collected in the previous step; here some filtering is performed to exclude vacuous or ill-formed results.

[1] Boldface indicates items recognized as head nouns.

5. Render the propositions from the previous step in (approximate) English; again significant heuristic filtering is done here.

As an example of KNEXT output, the sentence:

> *Cock fights, however, are still legal in six of the United States, perhaps because we still eat chicken regularly, but no-longer dogs.*

yields a pair of propositions expressed logically as:

> [(K (NN cock.n (PLUR fight.n))) legal.a],
> [(DET (PLUR person.n)) eat.v (K chicken.n)]

and these are automatically rendered in approximate English as:

> COCK FIGHTS CAN BE LEGAL.
> PERSONS MAY EAT CHICKEN.

As can be seen, KNEXT output does not conform to the ⟨relation, arg1, arg2, ...⟩, *tuple* style of knowledge representation favored in information extraction (stemming from that community's roots in populating DB tables under a fixed schema). This is further exemplified by the unscoped logical form:[2]

> [(DET (PLUR person.n)) want.v (Ka (rid.a (of.p (DET dictator.n))))]

which is verbalized as PERSONS MAY WANT TO BE RID OF A DICTATOR and is supported by the text fragment:

> *... and that if the Spanish people wanted to be rid of Franco, they must achieve this by ...*

Later examples will be translated into a more conventional logical form.

One larger collection we have processed since the 2002-3 work on Treebank corpora is the British National Corpus (BNC), consisting of 100 million words of mixed-genre text passages. The quality of resulting propositions has been assessed by the hand-judging methodology of Schubert and Tong (2003), yielding positive judgements almost as frequently as for the Brown Treebank corpus. The next section, concerned with the web corpus collected and used by Banko et al. (2007), contains a fuller description of the judging method. The BNC-based KB, containing 6,205,877 extracted propositions, is publicly searchable via a recently developed online knowledge browser.[3]

[2] Where Ka is an action/attribute reification operator.

[3] `http://www.cs.rochester.edu/u/vandurme/epik`

3 Experiments

The experiments reported here were aimed at a comparative assessment of linguistically based knowledge extraction (by KNEXT), and pattern-based information extraction (by TextRunner, and by another system, aimed at class attribute discovery). The goal being to show that logically formal results (i.e. *knowledge*) based on syntactic parsing may be obtained at a subjective level of accuracy similar to methods aimed exclusively at acquiring correspondences between string pairs based on shallow techniques.

Dataset Experiments were based on sampling 1% of the sentences from each document contained within a corpus of 11,684,774 web pages harvested from 1,354,123 unique top level domains. The top five contributing domains made up 30% of the documents in the collection.[4] There were 310,463,012 sentences in all, the sample containing 3,000,736. Of these, 1,373 were longer than a preset limit of 100 tokens, and were discarded.[5] Sentences containing individual tokens of length greater than 500 characters were similarly removed.[6]

As this corpus derives from the work of Banko et al. (2007), each sentence in the collection is paired with zero or more *tuples* as extracted by the TextRunner system.

Note that while websites such as `Wikipedia.org` contain large quantities of (semi-)structured information stored in lists and tables, the focus here is entirely on natural language sentences. In addition, as the extraction methods discussed in this paper do not make use of intersentential features, the lack of sentence to sentence coherence resulting from random sampling had no effect on the results.

Extraction Sentences were processed using the syntactic parser of Charniak (1999). From the resultant trees, KNEXT extracted 7,406,371 *propositions*, giving a *raw* average of 2.47 per sentence. Of these, 4,151,779 were unique, so that the average extraction frequency per sentence is 1.78 unique propositions. Post-processing left 3,975,197 items, giving a per sentence expectation of 1.32 unique, filtered propositions. Selected examples regarding knowledge about people appear in Table 1.

For the same sample, TextRunner extracted 6,053,983 *tuples*, leading to a raw average of 2.02 tuples per sentence. As described by its designers, TextRunner is an *information* extraction system; one would be mistaken in using these results to say that KNEXT "wins" in raw extraction volume, as these numbers are not in fact directly comparable (see section on *Comparison*).

Table 1: Verbalized propositions concerning the class PERSON

A PERSON MAY...			
SING TO A GIRLFRIEND	RECEIVE AN ORDER FROM A GENERAL	KNOW STUFF	PRESENT A PAPER
EXPERIENCE A FEELING	CARRY IMAGES OF A WOMAN	BUY FOOD	PICK_UP A PHONE
WALK WITH A FRIEND	CHAT WITH A MALE-INDIVIDUAL	BURN A SAWMILL	FEIGN A DISABILITY
DOWNLOAD AN ALBUM	MUSH A TEAM OF (SEASONED SLED DOGS)		RESPOND TO A QUESTION
SING TO A GIRLFRIEND	OBTAIN SOME_NUMBER_OF (PERCULA CLOWNFISH)		LIKE (POP CULTURE)

[4] `en.wikipedia.org`, `www.answers.com`, `www.amazon.com`, `www.imdb.com`, `www.britannica.com`

[5] Typically enumerations, e.g., *There have been 29 MET deployments in the city of Florida since the inception of the program : three in Ft. Pierce , Collier County , Opa Locka , ...*

[6] For example, *Kellnull phenotypes can occur through splice site and splice-site / frameshift mutations301,302 450039003[...]3000 premature stop codons and missense mutations.*

1. A REASONABLE GENERAL CLAIM
   e.g., A grand-jury may say a proposition
2. TRUE BUT TOO SPECIFIC TO BE USEFUL
   e.g., Bunker walls may be decorated with seashells
3. TRUE BUT TOO GENERAL TO BE USEFUL
   e.g., A person can be nearest an entity
4. SEEMS FALSE
   e.g., A square can be round
5. SOMETHING IS OBVIOUSLY MISSING
   e.g., A person may ask
6. HARD TO JUDGE
   e.g., Supervision can be with a company

Figure 1: Instructions for categorical judging

Evaluation Extraction quality was determined through manual assessment of verbalized propositions drawn randomly from the results. Initial evaluation was done using the method proposed in Schubert and Tong (2003), in which judges were asked to label propositions according to their category of acceptability; abbreviated instructions may be seen in Figure 1.[7] Under this framework, category one corresponds to a strict assessment of acceptability, while an assignment to any of the categories between one and three may be interpreted as a weaker level of acceptance. As seen in Table 2, average acceptability was judged to be roughly 50 to 60%, with associated Kappa scores signalling fair (0.28) to moderate (0.48) agreement.

Table 2: Percent propositions labeled under the given category(s), paired with Fleiss' Kappa scores. Results are reported both for the authors (judges one and two), along with two volunteers

Category	**% Selected**	**Kappa**	**% Selected**	**Kappa**
1	49%	0.4017	50%	0.2822
1, 2, or 3	54%	0.4766	60%	0.3360
	judges		judges w/ volunteers	

Judgement categories at this level of specificity are useful both for system analysis at the development stage, as well as for training judges to recognize the disparate ways in which a proposition may not be acceptable. However, due to the rates of agreement observed, evaluation moved to the use of a five point sliding scale (Figure 2). This scale allows for only a single axis of comparison, thus collapsing the various ways in which a proposition may or may not be flawed into a single, general notion of acceptability.

[7] Judges consisted of the authors and two volunteers, each with a background in linguistics and knowledge representation.

THE STATEMENT ABOVE IS A REASONABLY CLEAR, ENTIRELY PLAUSIBLE GENERAL CLAIM AND SEEMS NEITHER TOO SPECIFIC NOR TOO GENERAL OR VAGUE TO BE USEFUL:

1. I agree.
2. I lean towards agreement.
3. I'm not sure.
4. I lean towards disagreement.
5. I disagree.

Figure 2: Instructions for scaled judging

The authors judged 480 propositions sampled randomly from amongst bins corresponding to frequency of support (i.e., the number of times a given proposition was extracted). 60 propositions were sampled from each of 8 such ranges.[8] As seen in Figure 3, propositions that were extracted at least twice were judged to be more acceptable than those extracted only once. While this is to be expected, it is striking that as frequency of support increased further, the level of judged acceptability remained roughly the same.

4 Comparison

To highlight differences between an extraction system targeting knowledge (represented as logical statements) as compared to information (represented as segmented text fragments), the output of KNEXT is compared to that of TextRunner for two select inputs.

4.1 Basic

Consider the following sentence:

> *A defining quote from the book, "An armed society is a polite society", is very popular with those in the United States who support the personal right to bear arms.*

From this sentence TextRunner extracts the tuples:[9]

> (A defining **quote**) is a (polite **society** "),
> (the personal **right**) to bear (**arms**).

We might manually translate this into a crude sort of logical form:

> IS-A(A-DEFINING-QUOTE, POLITE-SOCIETY-"),
> TO-BEAR(THE-PERSONAL-RIGHT, ARMS).

[8] $(0,2^0,2^1,2^3,2^4,2^6,2^8,2^{10},2^{12})$, i.e., (0,1], (1,2], (2,8], ...

[9] Tuple *arguments* are enclosed in parenthesis, with the items recognized as *head* given in bold. All non-enclosed, conjoining text makes up the tuple *predicate*.

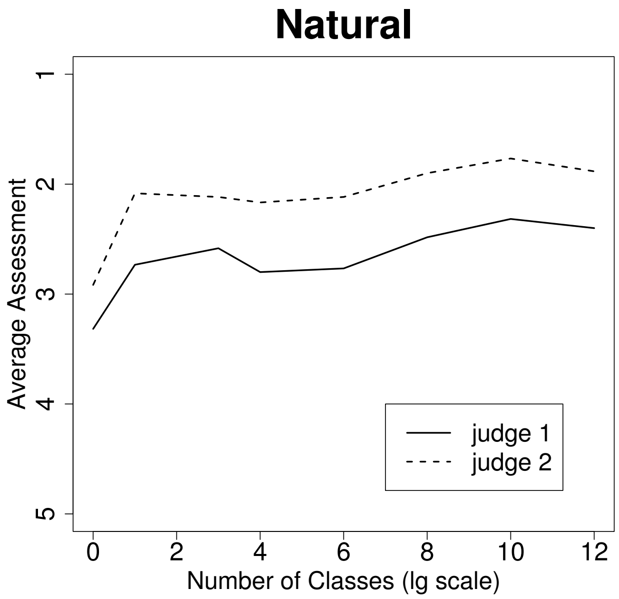

Figure 3: As a function of frequency of support, average assessment for propositions derived from *natural* sentences

Better would be to consider only those terms classified as head, and make the assumption that each tuple argument implicitly introduces its own quantified variable:

∃x,y. QUOTE(x) & SOCIETY(y) & IS-A(x,y),
∃x,y. RIGHT(x) & ARMS(y) & TO-BEAR(x,y).

Compare this to the output of KNEXT:[10]

∃x. SOCIETY(x) & POLITE(x),
∃x,y,z. THING-REFERRED-TO(x) & COUNTRY(y) & EXEMPLAR-OF(z,y) & IN(x,z),
∃x. RIGHT(x) & PERSONAL(x),
∃x,y. QUOTE(x) & BOOK(y) & FROM(x,y),
∃x. SOCIETY(x) & ARMED(x),

which is automatically verbalized as:

A SOCIETY CAN BE POLITE,
A THING-REFERRED-TO CAN BE IN AN EXEMPLAR-OF A COUNTRY,
A RIGHT CAN BE PERSONAL,
A QUOTE CAN BE FROM A BOOK,
A SOCIETY CAN BE ARMED.

[10] For expository reasons, scoped, simplified versions of KNEXT's ULFs are shown. More accurately propositions are viewed as weak *generic conditionals*, with a non-zero lower bound on conditional frequency, e.g., [∃x. QUOTE(x)] $\Rightarrow_{0.1}$ [∃y. BOOK(y) & FROM(x,y)], where x is dynamically bound in the consequent.

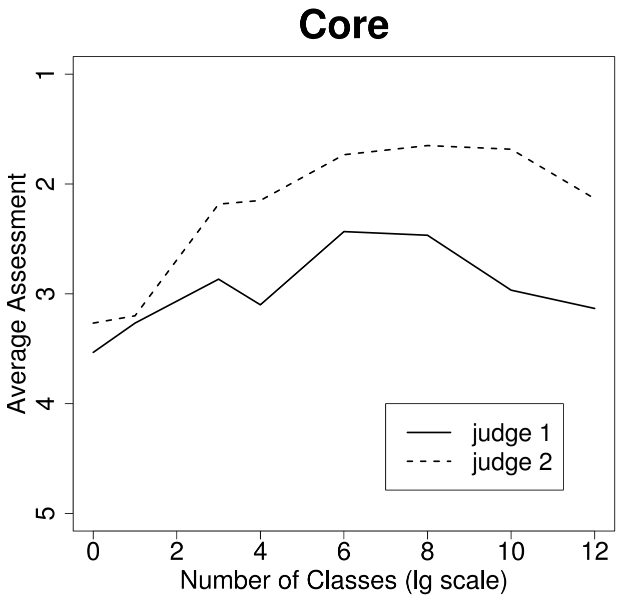

Figure 4: As a function of frequency of support, average assessment for propositions derived from *core* sentences

4.2 Extended Tuples

While KNEXT uniquely recognizes, e.g., adjectival modification and various types of possessive constructions, TextRunner more aggressively captures constructions with extended cardinality. For example, from the following:

> *James Harrington in The Commonwealth of Oceana uses the term anarchy to describe a situation where the people use force to impose a government on an economic base composed of either solitary land ownership, or land in the ownership of a few.*

TextRunner extracts 19 tuples, some with three or even four arguments, thus aiming beyond the binary relations that most current systems are limited to. That so many tuples were extracted for a single sentence is explained by the fact that for most tuples containing $N > 2$ arguments, TextRunner will also output the same tuple with $N - 1$ arguments, such as:

(the **people**) use (**force**),
(the **people**) use (**force**) to impose (a **government**),
(the **people**) use (**force**) to impose (a **government**) on (an economic **base**).

In addition, tuples may overlap, without one being a proper subset of another:

(a **situation**) where (the **people**) use (**force**),
(**force**) to impose (a **government**),
(a **government**) on (an economic **base**) composed of
(either solitary land **ownership**).

This overlap raises the question of how to accurately quantify system performance. When measuring average extraction quality, should samples be drawn randomly across tuples, or from originating sentences? If from tuples, then sample sets will be biased (for good or ill) towards fragments derived from complex syntactic constructions. If sentence based, the system fails to be rewarded for extracting as much from an input as possible, as it may conservatively target only those constructions most likely to be correct. With regards to volume, it is not clear whether adjuncts should each give rise to additional facts added to a final total; optimal would be the recognition of such optionality. Failing this, perhaps a tally may be based on unique predicate head terms?

As a point of merit according to its designers, TextRunner does not utilize a parser (though as mentioned it does part of speech tagging and noun phrase chunking). This is said to be justified in view of the known difficulties in reliably parsing open domain text as well as the additional computational costs. However, a serious consequence of ignoring syntactic structure is that incorrect bracketing across clausal boundaries becomes all too likely, as seen for instance in the following tuple:

(**James Harrington**) uses (the term **anarchy**) to describe (a **situation**) where (the **people**),

or in the earlier example where *from the book, "An armed society* appears to have been erroneously treated as a post-nominal modifier, intervening between the first argument and the *is-a* predicate.

KNEXT extracted the following six propositions, the first of which was automatically filtered in post-processing for being overly vague:[11]

⋆ A MALE-INDIVIDUAL CAN BE IN A NAMED-ENTITY OF A NAMED-ENTITY,
A MALE-INDIVIDUAL MAY USE A (TERM ANARCHY),
PERSONS MAY USE FORCE,
A BASE MAY BE COMPOSED IN SOME WAY,
A BASE CAN BE ECONOMIC,
A (LAND OWNERSHIP) CAN BE SOLITARY.

5 Extracting from Core Sentences

We have noted the common argument against the use of syntactic analysis when performing large-scale extraction viz. that it is too time consuming to be worthwhile. We are skeptical of such a view, but decided to investigate whether an argument-bracketing system such as TextRunner might be used as an extraction *preprocessor* to limit what needed to be parsed.

For each TextRunner tuple extracted from the sampled corpus, *core* sentences were constructed from the predicate and noun phrase arguments,[12] which were then used as input to KNEXT for extraction.

[11] The authors judge the third, fifth and sixth propositions to be both well-formed and useful.

[12] Minor automated heuristics were used to recover, e.g., missing articles dropped during tuple construction.

From 6,053,981 tuples came an equivalent number of core sentences. Note that since TextRunner tuples may overlap, use of these reconstructed sentences may lead to skewed propositional frequencies relative to “normal” text. This bias was very much in evidence in the fact that of the 10,507,573 propositions extracted from the core sentences, only 3,787,701 remained after automatic postprocessing and elimination of duplicates. This gives a per-sentence average of 0.63, as compared to 1.32 for the original text.

While the raw number of propositions extracted for each version of the underlying data look similar, 3,975,197 (natural) vs. 3,787,701 (core), the actual overlap was less than would be expected. Just 2,163,377 propositions were extracted jointly from both natural and core sentences, representing a percent overlap of 54% and 57% respectively.

Table 3: Mean judgements (lower is better) on propositions sampled from those supported either exclusively by natural or core sentences, or those supported by both

	Natural	**Core**	**Overlap**
judge 1	3.35	3.85	2.96
judge 2	2.95	3.59	2.55

Quality was evaluated by each judge assessing 240 randomly sampled propositions for each of: those extracted exclusively from natural sentences, those extracted exclusively from core sentences, those extracted from both (Table 3). Results show that propositions exclusively derived from core sentences were most likely to be judged poorly. Propositions obtained both by KNEXT alone and by KNEXT- processing of TextRunner-derived core sentences (the overlap set) were particularly likely to be judged favorably.

On the one hand, many sentential fragments ignored by TextRunner yield KNEXT propositions; on the other, TextRunner’s output may be assembled to produce sentences yielding propositions that KNEXT otherwise would have missed. Ad-hoc analysis suggests these new propositions derived with the help of TextRunner are a mix of noise stemming from bad tuples (usually a result of the aforementioned incorrect clausal bracketing), along with genuinely useful propositions coming from sentences with constructions such as appositives or conjunctive enumerations where TextRunner outguessed the syntactic parser as to the correct argument layout. Future work may consider whether (syntactic) language models can be used to help prune core sentences before being given to KNEXT.

Figure 4 differs from Figure 3 at low frequency of support. This is the result of the partially redundant tuples extracted by TextRunner for complex sentences; the core verb-argument structures are those most likely to be correctly interpreted by KNEXT, while also being those most likely to be repeated across tuples for the same sentence.

6 Class Properties

While TextRunner is perhaps the extraction system most closely related to KNEXT in terms of generality, there is also significant overlap with work on *class attribute*

Table 4: By frequency, the top ten attributes a class MAY HAVE. Emphasis added to entries overlapping with those reported by Paşca and Van Durme. Results for starred classes were derived without the use of prespecified lists of instances

COUNTRY	government, war, team, history, rest, coast, census, economy, *population*, independence
DRUG*	*side effects*, influence, *uses*, doses, manufacturer, efficacy, release, graduates, plasma levels, safety
CITY*	makeup, heart, center, *population*, history, side, places, name, edge, area
PAINTER*	*works*, art, brush, skill, lives, sons, friend, order quantity, muse, eye
COMPANY	windows, products, word, page, review, film, team, award, studio, director

extraction. Paşca and Van Durme (2007) recently described this task, going on to detail an approach for collecting such attributes from search engine query logs. As an example, the search query "*president of Spain*" suggests that a *Country* may have a *president*.

If one were to consider attributes to correspond, at least in part, to things a class MAY HAVE, CAN BE, or MAY BE, then a subset of KNEXT's results may be discussed in terms of this specialized task. For example, for the five classes used in those authors' experiments, Table 4 contains the top ten most frequently extracted things each class MAY HAVE, as determined by KNEXT, without any targeted filtering or adaptation to the task.

Table 5: Mean assessed acceptability for properties occurring for a single class (1), and more than a single class (2+). Final column contains Pearson correlation scores

	1	2+	1	2+	corr.
MAY HAVE	2.80	2.35	2.50	2.28	0.68
MAY BE	3.20	2.85	2.35	2.13	0.59
CAN BE	3.78	3.58	3.28	2.75	0.76
	judge 1		judge 2		

For each of these three types of attributive categories the authors judged 80 randomly drawn propositions, constrained such that half (40 for each) were supported by a single sentence, while the other half were required only to have been extracted at least twice, but potentially many hundreds or even thousands of times. As seen in Table 5, the judges were strongly correlated in their assessments, where for MAY HAVE and MAY BE they were lukewarm (3.0) or better on the majority of those seen.

In a separate evaluation judges considered whether the number of classes sharing a given attribute was indicative of its acceptability. For each unique attributive propo-

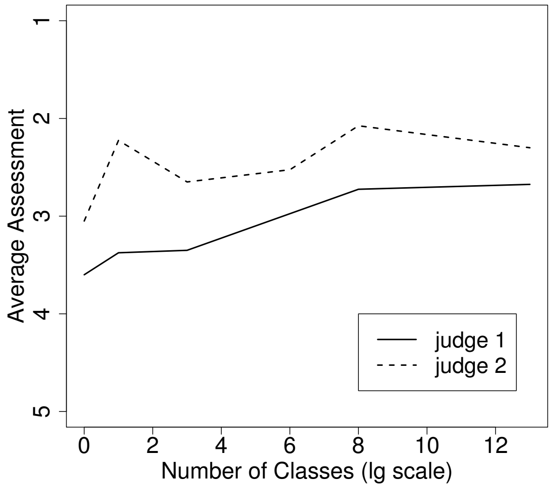

Figure 5: Mean quality of class attributes as a function of the number of classes sharing a given property

sition the class in "subject" position was removed, leaving fragments such as that bracketed: A ROBOT [CAN BE SUBHUMAN]. These attribute fragments were tallied and binned by frequency,[13] with 40 then sampled from each. For a given attribute selected, a single attributive proposition matching that fragment was randomly drawn. For example, having selected the attribute CAN BE FROM A US-CITY, the proposition SOME_NUMBER_OF SHERIFFS CAN BE FROM A US-CITY was drawn from the 390 classes sharing this property. As seen in Figure 5, acceptability rose as a property became more common.

7 Conclusions

Work such as TextRunner (Banko et al., 2007) is pushing extraction researchers to consider larger and larger datasets. This represents significant progress towards the greater community's goal of having access to large, expansive stores of general world knowledge.

The results presented here support the position that advances made over decades of research in parsing and semantic interpretation do have a role to play in large-scale knowledge acquisition from text. The price paid for linguistic processing is not excessive, and an advantage is the logical formality of the results, and their versatility, as indicated by the application to class attribute extraction.

[13] Ranges: $(0, 2^0, 2^1, 2^3, 2^6, \infty)$

Acknowledgements We are especially grateful to Michele Banko and her colleagues for generously sharing results, and to Daniel Gildea for helpful feedback. This work was supported by NSF grants IIS-0328849 and IIS-0535105.

References

Abney, S. (1996). Partial Parsing via Finite-State Cascades. *Natural Language Engineering 2*(4), 337–344.

Banko, M., M. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni (2007). Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 2670–2676.

Charniak, E. (1999). A Maximum-Entropy-Inspired Parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2000)*, pp. 132–139.

Clark, P., P. Harrison, and J. Thompson (2003). A Knowledge-Driven Approach to Text Meaning Processing. In *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pp. 1–6.

Clark, S. and D. Weir (1999). An iterative approach to estimating frequencies over a semantic hierarchy. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*, pp. 258–265.

Collins, M. (1997). Three Generative, Lexicalised Models for Statistical Parsing. In *Proceedings of the 35th Annual Conference of the Association for Computational Linguistics (ACL-97)*, pp. 16–23.

Gildea, D. and D. Jurafsky (2002). Automatic labeling of semantic roles. *Computational Linguistics 28*(3), 245–288.

Gildea, D. and M. Palmer (2002). The necessity of syntactic parsing for predicate argument recognition. In *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics (ACL-02)*, Philadelphia, PA, pp. 239–246.

Holmback, H., L. Duncan, and P. Harrison (2000). A word sense checking application for Simplified English. In *Proceedings of the 3rd International Workshop on Controlled Language Applications (CLAW00)*, pp. 120–133.

Liakata, M. and S. Pulman (2002). From Trees to Predicate Argument Structures. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING-02)*, pp. 563–569.

Paşca, M. and B. Van Durme (2007). What You Seek is What You Get: Extraction of Class Attributes from Query Logs. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 2832–2837.

Punyakanok, V., D. Roth, and W. tau Yih (2008). The Importance of Syntactic Parsing and Inference in Semantic Role Labeling. *Computational Linguistics 34*(2), 257–287.

Resnik, P. (1993). Semantic classes and syntactic ambiguity. In *Proceedings of ARPA Workshop on Human Language Technology*, pp. 278–283.

Richardson, S. D., W. B. Dolan, and L. Vanderwende (1998). MindNet: Acquiring and Structuring Semantic Information from Text. In *Proceedings of the 17th International Conference on Computational linguistics (COLING-98)*, pp. 1098–1102.

Schubert, L. K. (2002). Can we derive general world knowledge from texts? In *Proceedings of the 2nd International Conference on Human Language Technology Research (HLT 2002)*, pp. 94–97.

Schubert, L. K. and C. H. Hwang (2000). Episodic Logic meets Little Red Riding Hood: A comprehensive, natural representation for language understanding. In L. Iwanska and S. Shapiro (Eds.), *Natural Language Processing and Knowledge Representation: Language for Knowledge and Knowledge for Language*, pp. 111–174.

Schubert, L. K. and M. H. Tong (2003). Extracting and evaluating general world knowledge from the Brown corpus. In *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pp. 7–13.

Wong, Y. W. and R. J. Mooney (2007). Learning Synchronous Grammars for Semantic Parsing with Lambda Calculus. In *Proceedings of the 45th Annual Conference of the Association for Computational Linguistics (ACL-07)*, pp. 960–967.

Zelle, J. M. and R. J. Mooney (1996). Learning to Parse Database Queries using Inductive Logic Programming. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-96)*, pp. 1050–1055.

Zernik, U. (1992). Closed yesterday and closed minds: Asking the right questions of the corpus to distinguish thematic from sentential relations. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING-02)*, pp. 1305–1311.

Zettlemoyer, L. and M. Collins (2005). Learning to Map Sentences to Logical Form: Structured Classification with Probabilistic Categorial Grammars. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pp. 658–666.