Decreasing lexical data sparsity in statistical syntactic parsing - experiments with named entities

Deirdre Hogan, Jennifer Foster and Josef van Genabith
National Centre for Language Technology
School of Computing
Dublin City University
Dublin 9, Ireland
dhogan,jfoster,josef@computing.dcu.ie

Abstract

In this paper we present preliminary experiments that aim to reduce lexical data sparsity in statistical parsing by exploiting information about named entities. Words in the WSJ corpus are mapped to named entity clusters and a latent variable constituency parser is trained and tested on the transformed corpus. We explore two different methods for mapping words to entities, and look at the effect of mapping various subsets of named entity types. Thus far, results show no improvement in parsing accuracy over the best baseline score; we identify possible problems and outline suggestions for future directions.

1 Introduction

Techniques for handling lexical data sparsity in parsers have been important ever since the lexicalisation of parsers led to significant improvements in parser performance (Collins, 1999; Charniak, 2000). The original treebank set of non-terminal labels is too general to give good parsing results. To overcome this problem, in lexicalised constituency parsers, non-terminals are enriched with lexical information. Lexicalisation of the grammar vastly increases the number of parameters in the model, spreading the data over more specific events. Statistics based on low frequency events are not as reliable as statistics on phenomena which occur regularly in the data; frequency counts involving words are typically sparse.

Word statistics are also important in more recent unlexicalised approaches to constituency parsing such as latent variable parsing (Matsuzaki et al., 2005; Petrov et al., 2006). The basic idea of latent variable parsing is that rather than enrich the non-terminal labels by augmenting them with words, a set of enriched labels which can encapsulate the syntactic behaviour of words is automatically learned via an EM training mechanism.

Parsers need to be able to handle both low frequency words and words occurring in the test set which were unseen in the training set (unknown words). The problem of rare and unknown words is particularly significant for languages where the size of the treebank is small. Lexical sparseness is also critical when running a parser on data that is in a different domain to the domain upon which the parser was trained. As interest in parsing real world data increases, a parsers ability to adequately handle out-of-domain data is critical.

In this paper we examine whether clustering words based on their named entity category can be useful for reducing lexical sparsity in parsing. Intuitively word tokens in the corpus such as, say, ‘Dublin’ and ‘New York’ should play similar syntactic roles in sentences. Likewise, it is difficult to see how different people names could have different discriminatory influences on the syntax of sentences. This paper describes experiments at replacing word tokens with special named entity tokens (person names are mapped to PERSON tokens and so on). Words in the original WSJ treebank are mapped to entity types extracted from the BBN corpus (Weischedel and Brunstein, 2005) and a latent variable parser is trained and tested on the mapped corpus. Ultimately, the motivation behind grouping words together in this fashion is to make it easier for

Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011), pages 14–19,
Portland, Oregon, USA, 23 June 2011. ©2011 Association for Computational Linguistics

the parser to recognise regularities in the data.[1]

The structure of paper is as follows: A brief summary of related work is given in Section 2. This includes an outline of a common treatment of low frequency and rare words in constituency parsing, involving a mapping process that is similar to the named entity mappings. Section 3 presents the experiments carried out, starting with a short introduction of the named entity resource used in our experiments and a description of the types of basic entity mappings we examine. In §3.1 and §3.2 we describe the two different types of mapping technique. Results are presented in Section 4, followed by a brief discussion in Section 5 indicating possible problems and avenues worth pursuing. Finally, we conclude.

2 Related Work

Much previous work on parsing and multiword units (MWUs) adopts the words-with-spaces approach which treats MWUs as one token (by concatenating the words together) (Nivre and Nilsson, 2004; Cafferkey et al., 2007; Korkontzelos and Manandhar, 2010). Alternative approaches are that of Finkel and Manning (2009) on joint parsing and named entity recognition and the work of (Wehrli et al., 2010) which uses collocation information to rank competing hypotheses in a symbolic parser. Also related is work on MWUs and grammar engineering, such as (Zhang et al., 2006; Villavicencio et al., 2007) where automatically detected MWUs are added to the lexicon of a HPSG grammar to improve coverage.

Our work is most similar to the words-with-spaces approach. Our many-to-one experiments (see §3.1) in particular are similar to previous work on parsing words-with-spaces, except that we map words to entity types rather than concatenated words. Results are difficult to compare however, due to different parsing methodologies, different types of MWUs, as well as different evaluation methods.

Other relevant work is the integration of named entity types in a surface realisation task by Rajkumar et al. (2009) and the French parsing experiments of (Candito and Crabbé, 2009; Candito and Seddah, 2010) which involve mapping words to clusters based on morphology as well as clusters automatically induced via unsupervised learning on a large corpus.

2.1 Parsing unknown words

Most state-of-the-art constituency parsers (e.g. (Petrov et al., 2006; Klein and Manning, 2003)) take a similar approach to rare and unknown words. At the beginning of the training process very low frequency words in the training set are mapped to special UNKNOWN tokens. In this way, some probability mass is reserved for occurrences of UNKNOWN tokens and the lexicon contains productions for such tokens ($X \rightarrow$ UNKNOWN), with associated probabilities. When faced with a word in the test set that the parser has not seen in its training set - the unknown word is mapped to the special UNKNOWN token.

In syntactic parsing, rather than map all low frequency words to one generic UNKNOWN type, it is useful to have several different clusters of unknown words, grouped according to morphological and other ‘surfacey’ clues in the original word. For example, certain suffixes in English are strong predictors for the part-of-speech tag of the word (e.g. ‘ly’) and so all low frequency words ending in ‘ly’ are mapped to ‘UNKNOWN-ly’. As well as suffix information, UNKNOWN words are commonly grouped based on information on capitalisation and hyphenation. Similar techniques for handling unknown words have been used for POS tagging (e.g. (Weischedel et al., 1993; Tseng et al., 2005)) and are used in the Charniak (Charniak, 2000), Berkeley (Petrov et al., 2006) and Stanford (Klein and Manning, 2003) parsers, as well as in the parser used for the experiments in this paper, an in-house implementation of the Berkeley parser.

3 Experiments

The BBN Entity Type Corpus (Weischedel and Brunstein, 2005) consists of sentences from the Penn WSJ corpus, manually annotated with named entities. The Entity Type corpus includes annota-

[1] It is true that latent variable parsers automatically induce categories for similar words, and thus might be expected to induce a category for say names of people if examples of such words occurred in similar syntactic patterns in the data. Nonetheless, the problem of data sparsity remains - it is difficult even for latent variable parsers to learn accurate patterns based on words which only occur say once in the training set.

type	count	examples
PERSON	11254	Kim Cattrall
PER_DESC	21451	president,chief executive officer,
FAC	383	office, Rockefeller Center
FAC_DESC	2193	chateau ,stadiums, golf course
ORGANIZATION	24239	Securities and Exchange Commission
ORG_DESC	15765	auto maker, college
GPE	10323	Los Angeles,South Africa
GPE_DESC	1479	center, nation, country
LOCATION	907	North America,Europe, Hudson River
NORP	3269	Far Eastern
PRODUCT	667	Maxima, 300ZX
PRODUCT_DESC	1156	cars
EVENT	296	Vietnam war,HUGO ,World War II
WORK_OF_ART	561	Revitalized Classics Take..
LAW	300	Catastrophic Care Act,Bill of Rights
LANGUAGE	62	Latin
CONTACT_INFO	30	555 W. 57th St.
PLANT	172	crops, tree
ANIMAL	355	hawks
SUBSTANCE	2205	gold,drugs, oil
DISEASE	254	schizophrenia,alcoholism
GAME	74	football senior tennis and golf tours

Table 1: Name expression entity types (sections 02-21)

unk map	NE map	#unks	*f*-score	POS
generic	**none (baseline 1)**	**2966 (4.08%)**	**88.69**	**95.57**
	ALL_NAMED	1908 (2.73%)	89.21	95.49
	$REDUCED$	2122 (3.02%)	89.43	96.08
	$Person$	2671 (3.68%)	88.98	95.55
	$Organisation$	2521 (3.55%)	89.38	95.92
	$Location$	2945 (4.05%)	89.00	95.62
sigs	**none (baseline 2)**	**2966 (4.08%)**	**89.72**	**96.51**
	ALL_NAMED	1908 (2.73%)	89.67	95.99
	$REDUCED$	2122 (3.02%)	89.53	96.65
	$Person$	2671 (3.68%)	89.32	96.47
	$Organisation$	2521 (3.55%)	89.53	96.64
	$Location$	2945 (4.05%)	89.20	96.52

Table 2: Many-to-One Parsing Results.

tion for three classes of named entity: name expressions, time expressions and numeric expressions (in this paper we focus on name expressions). These are further broken down into types. Table 1 displays name expression entity types, their frequency in the training set (sections 02-21), as well as some illustrative examples from the training set data.

We carried out experiments with different subsets of entity types. In one set of experiments, all name expression entities were mapped, with no restriction on the types (ALL_NAMED). We also carried out experiments on a reduced set of named entities - where only entities marked as *PERSON*, *ORGANIZATION*, or *GPE* and *LOCATION* were mapped ($REDUCED$). Finally, we ran experiments where only one type of named entity was mapped at a time. In all cases the words in the named entities were replaced by their entity type.

3.1 Many-to-one Mapping

In the many-to-one mapping all words in a named entity were replaced with one named entity type token. This approach is distinct from the words-with-spaces approach previously pursued in parsing where, for example, ‘New York’ would be replaced with ‘New_York’. Instead, in our experiments ‘New York’ is replaced with ‘GPE’ (geo-political entity). In both approaches, the parser is forced to respect the multiword unit boundary (and analyses which contain constituents that cross the MWU boundary will not be considered by the parser). Intuitively, this should help parser accuracy and speed. The advantage of mapping the word tokens to their entity type rather than to a words-with-spaces token is that in addition we will be reducing data sparsity.

One issue with the many-to-one mapping is that in evaluation exact comparison with a baseline result is difficult because the tokenisation of test and gold sets is different. When named entities span more than one word, we are reducing the number of words in the sentences. As parsers tend to do better on short sentences than on long sentences, this could make parsing somewhat easier. However, we found that the average number of words in a sentence before and after this mapping does not change by much. The average number of words in the development set is 23.9. When we map words to named entity tokens (ALL_NAMED), the average drops by just one word to 22.9.[2]

3.2 One-to-one Mapping

In the one-to-one experiments we replaced each word in named entity with a named entity type token (e.g. Ada Lovelace → pperson pperson).[3] The motivation was to measure the effect of reducing word sparsity using named entities without altering the original tokenisation of the data.[4]

[2] A related issue is that the resulting parse tree will lack an analysis for the named entity.

[3] The entity type was given an extra letter where needed (e.g. ‘pperson’) to avoid the conflation of a mapped entity token with an original word (e.g. ‘person’) in the corpus.

[4] Note, where there is punctuation as part of a named entity we do not map the punctuation.

unk map	NE map	#unks	*f*-score	POS
generic	**none (baseline 1)**	**2966 (4.08%)**	**88.69**	**95.57**
	ALL_NAMED	1923 (2.64%)	89.28	94.99
	$REDUCED$	2122 (2.90%)	88.76	95.76
	$Person$	2654(3.65%)	88.95	95.57
	$Organisation$	2521 (3.45%)	88.80	95.59
	$Location$	2945 (4.04%)	88.88	95.66
sigs	**none (baseline 2)**	**2966 (4.08%)**	**89.72**	**96.51**
	ALL_NAMED	1923 (2.64%)	89.36	95.64
	$REDUCED$	2122 (2.90%)	89.01	96.32
	$Person$	2654(3.65%)	89.30	96.52
	$Organisation$	2521 (3.45%)	89.29	96.30
	$Location$	2945 (4.04%)	89.55	96.54

Table 3: One-to-One Parsing Results

In an initial experiment, where the mapping was simply the word to the named entity type, many sentences received no parse. This happened often when a named entity consisted of three or more words and resulted in a sentence such as 'But while the Oorganization Oorganization Oorganization Oorganization did n't fall apart Friday'. We found that refining the named entity by adding the number of the word in the entity to the mapping resolved the coverage problem. The example sentence is now: 'But while the Oorganization1 Oorganization2 Oorganization3 Oorganization4 did n't fall apart Friday'. See §5 for a possible explanation for the parser's difficulty with one-to-one mappings to coarse grained entity types.

4 Results

Table 2 and Table 3 give the results for the many-to-one and one-to-one experiments respectively. Results are given against a baseline where unknowns are given a 'generic' treatment (baseline 1) - i.e. they are not clustered according to morphological and surface information - and for the second baseline (baseline 2), where morphological or surface feature markers (sigs) are affixed to the unknowns.[5]

The results indicate that though lexical sparsity is decreasing, insofar as the number of unknown words ($\#unks$ column) in the development set decreases with all named entity mappings, the named entity clusters are not informative enough and parser accuracy falls short of the previous best result. For all experiments, a pattern that emerges is that mapping words to named entities improves results when low frequency words are mapped to a generic UNKNOWN token. However, when low frequency words are mapped to more fine-grained UNKNOWN tokens, mapping words to named entities decreases accuracy marginally.

If a particular named entity occurs often in the text then data sparsity is possibly not a problem for this word. Rather than map all occurrences of a named entity to its entity type, we experimented with mapping only low frequency entities. These named entity mapping experiments now mirror more closely the unknown words mappings - low frequency entities are mapped to special entity types, then the parser maps all remaining low frequency words to UNKNOWN types. Table 4 shows the effect of mapping only entities that occur less than 10 times in the training set, to the *person* type and the *reduced* set of entity types. Results somewhat improve for all but one of the one-to-one experiments, but nonetheless remain below the best baseline result. There is still no advantage in mapping low frequency person name words to, say, the *person* cluster, rather than to an UNKNOWN-plus-signature cluster.

5 Discussion

Our results thus far suggest that clusters based on morphology or surface clues are more informative than the named entity clusters.

For the one-to-one mappings one obvious problem that emerged is that all words in entities (including function words for example) get mapped to a generic named entity token. A multi-word named entity has its own internal syntactic structure, reflected for example in its sequence of part-of-speech tags. By replacing each word in the entity with the generic entity token we end up loosing information about words, conflating words that take different part-of-speech categories, and in fact make parsing more difficult. The named entity clusters in this case are too coarse-grained and words with different syntactic properties are merged into the one cluster, something we would like to avoid.

In future work, as well as avoiding mapping more complex named entities, we will refine the named entity clusters by attaching to the entity type signatures similar to those attached to the UNKNOWN

[5] For all experiments, a split-merge cycle of 5 was used. Following convention, sections 02-21 were used for training. Sections 22 and 24 (sentences less than or equal to 100 words) were used for the development set. As experiments are ongoing we do not report results on a test set.

unk map	NE map	one2one *f*-score	many2one *f*-score
generic	$Person$	88.95	88.98
	$Person < 10$	88.97	89.05
	$Reduced$	88.76	89.43
	$Reduced < 10$	89.51	88.85
sigs	$Person$	89.30	89.32
	$Person < 10$	89.49	89.33
	$Reduced$	89.01	89.53
	$Reduced < 10$	89.42	89.15

Table 4: Measuring the effect of mapping only low frequency named entities.

types. It would also be interesting to examine the effect of mapping other types of named entities, such as dates and numeric expressions. Finally, we intend trying similar experiments on out-of-domain data, such as social media text where unknown words are more problematic.

6 Conclusion

We have presented preliminary experiments which test the novel technique of mapping word tokens to named entity clusters, with the aim of improving parser accuracy by reducing data sparsity. While our results so far are disappointing, we have identified possible problems and outlined future experiments, including suggestions for refining the named entity clusters so that they become more syntactically homogenous.

References

Conor Cafferkey, Deirdre Hogan, and Josef van Genabith. 2007. Multi-word units in treebank-based probabilistic parsing and generation. In *Proceedings of the 10th International Conference on Recent Advances in Natural Language Processing (RANLP-07)*, Borovets, Bulgaria.

Marie Candito and Benoit Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the International Workshop on Parsing Technologies (IWPT-09)*.

Marie Candito and Djamé Seddah. 2010. Lemmatization and statistical lexicalized parsing of morphologically-rich languages. In *Proceedings of the NAACL/HLT Workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL)*.

Eugene Charniak. 2000. A maximum entropy-inspired parser. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics (NAACL)*.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Jenny Rose Finkel and Christopher D. Manning. 2009. Joint parsing and named entity recognition. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-2009)*.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association of Computational Linguistics (ACL)*.

Ioannis Korkontzelos and Suresh Manandhar. 2010. Can recognising multiword expressions improve shallow parsing? In *Proceedings of the Conference of the North American Chapter of the ACL (NAACL-10)*, Los Angeles, California.

Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic cfg with latent annotations. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 75–82, Ann Arbor, June.

Joakim Nivre and Jens Nilsson. 2004. Multiword units in syntactic parsing. In *Workshop on Methodologies and Evaluation of Multiword Units in Real-World Applications*.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL*, Sydney, Australia, July.

Rajakrishnan Rajkumar, Michael White, and Dominic Espinosa. 2009. Exploiting named entity classes in ccg surface realisation. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-09)*.

Huihsin Tseng, Daniel Jurafsky, and Christopher Manning. 2005. Morpholgical features help pos tagging of unknown words across language varieties. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*.

Aline Villavicencio, Valia Kordoni, Yi Zhang, Marco Idiart, and Carlos Ramisch. 2007. Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Eric Wehrli, Violeta Seretan, and Luke Nerima. 2010. Sentence analysis and collocation identification. In *Proceedings of the Workshop on Multiword Expression: From Theory to Applications (MWE)*.

Ralph Weischedel and Ada Brunstein. 2005. BBN pronoun coreference and entity type corpus. In *Tehcnical Report*.

Ralph Weischedel, Richard Schwartz, Jeff Palmucci, Marie Meteer, and Lance Ramshaw. 1993. Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19(2).

Yi Zhang, Valia Kordoni, Aline Villavicencio, and Marco Idiart. 2006. Automated multiword expression prediction for grammar engineering. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*.