Recognising Sets and Their Elements: Tree Kernels for Entity Instantiation Identification

Andrew McKinlay and Katja Markert
School of Computing
University of Leeds, UK
{scs4ajm,markert}@comp.leeds.ac.uk

Abstract

We apply tree kernels to *entity instantiations*. An entity instantiation is an entity relationship, in which a set of entities is mentioned, and then a member or subset of this set is introduced. We present the first reliably annotated intrasentential entity instantiation corpus, along with an extension to the intersentential annotations in McKinlay and Markert (2011). We then apply tree kernels to both inter- and intrasentential entity instantiations, showing comparable results to an extensive set of unstructured features. The combination of tree kernels and unstructured features leads to significant improvements over either method in isolation.

1 Introduction

In a previous paper, we define an entity instantiation as follows (McKinlay and Markert, 2011):

> An Entity Instantiation is a non-coreferent entity relationship, where a *set* of entities is mentioned, and then a *member* or *subset* is introduced.

Examples 1 and 2 show a set membership instantiation and a subset instantiation, respectively[1].

(1) a. **The two lawmakers** sparred in a highly personal fashion, violating usual Senate decorum.
    b. Their tone was good-natured, with *Mr. Packwood* saying he intended to offer [...]

(2) a. To the extent that the primary duty of personal staff involves local benefit-seeking, this indicates that political philosophy leads **congressional Republicans** to pay less attention to narrow constituent concerns.
    b. First, economists James Bennett and Thomas DiLorenzo find that *GOP senators* turn back roughly 10% more of their allocated personal staff budgets than Democrats do.

Entity Instantiations are highly context dependent and their interpretation requires careful consideration of prior mentions of both set and member/subset. In Example 3, one must refer back 2 sentences to establish that *'they'* is coreferent with *'the Montreal Protocol's legions of supporters'*, and the set from which *'Peter Teagan, a specialist in heat transfer'* is drawn. In Example 4, we need the knowledge that Mr. Mason is Jewish from the first sentence to establish the instantiation in the final sentence.

(3) But even though by some estimates it might cost the world as much as $100 billion between now and the year 2000 to convert to other coolants, foaming agents and solvents and to redesign equipment for these less efficient substitutes, the Montreal Protocol's legions of supporters say it is worth it. They insist that CFCs are damaging the earth's stratospheric ozone layer, which screens out some of the sun's ultraviolet rays. Hence, as **they** see it, if something isn't done earthlings will become ever more subject to sunburn and skin cancer.

[1] All examples in this paper are taken from the Penn Treebank Wall Street Journal corpus (Marcus et al., 1993) unless stated otherwise, and are occasionally abbreviated. Sets are highlighted in **bold** and members or subsets are shown in *italics*.

Peter Teagan, a specialist in heat transfer, is running a project at Arthur D. Little Inc., of Cambridge, Mass., to find alternative technologies that will allow industry to eliminate CFCs.

(4) [. . .] Or so it must seem to Jackie Mason, the veteran Jewish comedian appearing in a new ABC sitcom airing on Tuesday nights (9:30-10 p.m. EDT). Not only is Mr. Mason the star of "Chicken Soup," he's also the inheritor of a comedic tradition dating back to "Duck Soup," and he's currently a man in hot water. Here, in neutral language, is the gist of Mr. Mason's remarks, quoted first in the Village Voice while he was a paid spokesman for the Rudolph Giuliani mayoral campaign, and then in Newsweek after he and the campaign parted company. [. . .]

He said that **Jews** have contributed more to black causes over the years than vice versa.

Entity instantiations vary a great deal, in terms of internal structure, ordering and overlap with other phenomena. Example 1 shows a set member entity instantiation between an NP headed by a plural noun, and a named entity. In Example 5 the set member is coupled with an apposition — *'an analyst with Drexel Burnham Lambert'*. In Example 6, neither set nor set member are named entities, and the set is a complex plural noun phrase (NP) which is made up of several constituents. In Example 7, the member NP precedes the set NP, and recognition needs the interpretation of *'Capitol Hill'* as a metonymic reference to the U.S. Congress.

(5) a. But **other analysts** said that having Mr. Phillips succeed Mr. Roman would make for a smooth transition.
b. "Graham Phillips has been there a long time [. . .]", said *Andrew Wallach, an analyst with Drexel Burnham Lambert.*

(6) a. And Democrats, who are under increasing pressure from their leaders to reject the gains-tax cut, are finding **reasons to say no, at least for now.**
b. *A major reason* is that they believe the Packwood-Roth plan would lose buckets of revenue over the long run.

(7) a. However, the disclosure of the guidelines, first reported last night by NBC News, is already being interpreted on *Capitol Hill* as an unfair effort to pressure Congress.
b. It has reopened the bitter wrangling between **the White House and Congress** over who is responsible for the failure to oust Mr. Noriega and, more broadly, for difficulties in carrying out covert activities abroad.

In contrast to our previous work in McKinlay and Markert (2011), we also consider *intrasentential* entity instantiations. This introduces further variety, and the possibility of nested instantiations. In Example 8, the set member is nested within the conjunction that forms the set. In Example 9, the set member is also nested in the set, but this time as a subtree of the prepositional phrase that complements the set NP. Example 10 exhibits a different sort of nesting — the set is nested within the set member. There are also many intrasentential instantiations where the participant NPs do not overlap, such as Example 11.

(8) So if anything happened to me, I'd want to leave behind enough so that my 33-year-old husband would be able to pay off ***the mortgage*** **and some other debts**.

(9) [. . .] **several firms, including discount broker Charles Schwab & Co. and *Sears, Roebuck & Co. 's Dean Witter Reynolds Inc. unit***, have attacked program trading as a major market evil.

(10) When he is presented with a poster celebrating the organization's 20th anniversary, he recognizes a photograph of *one of* ***the founders*** and recalls time spent together in Camden.

(11) **Banking stocks** were the major gainers Monday amid hope that interest rates have peaked, as *Deutsche Bank and Dresdner Bank* added 4 marks each to 664 marks and 326 marks, respectively.

These complexities make entity instantiations difficult to identify. We address this complexity by using *tree kernels*, a method of learning from tree data.

In this paper we introduce the first corpus of intrasentential entity instantiations, and an expanded corpus of intersentential entity instantiations. We present the first algorithm for the classification of intrasentential instantiations, and the first application of tree kernels for both inter- and intrasentential instantiations.

2 Related Work

The only prior research which has tackled the problem of entity instantiations is our own in McKinlay and Markert (2011). We annotated a 25-text corpus of entity instantiations between adjacent sentences but not *within* sentences, and experimented with unstructured features, including lexical, contextual and world-knowledge features. We achieved good results on an artificially balanced set, but on the original, highly skewed data reported a highest F-Score of only 0.19 for set members and 0.14 for subsets.

Entity instantiations are also closely related to at least two important natural language processing problems: *relation extraction* and *bridging anaphora*.

Relation Extraction. Relation extraction (RE) is the detection and classification of binary semantic relationships between entities, such as Part-Of, Employed-By or Located-In. A considerable amount of research in this field is connected to the important MUC (MUC, 1998) and ACE programs (ACE, 2005), both of which provided RE corpora and shared evaluation metrics.

RE and detecting entity instantiations are similar problems; they both involve the discovery of binary semantic relations in context. There are two fundamental differences, however. Firstly the participants of entity instantiations are not restricted to mentions of entities representing concrete, real-world objects, but instead consider heterogeneous NPs. Secondly, whilst the evidence for an entity instantiation can be drawn from anywhere in the document or from existing world knowledge, RE schemes restrict the scope of their relations to within a sentence. Set membership and subset relations are not annotated as part of the RE corpora which formed part of the MUC and ACE programs.

SemEval-2 had a shared task, *Multi-Way Classification of Semantic Relations Between Pairs of Nominals* (Hendrickx et al., 2010), which does include a *Member-Collection* relation, and is somewhat different to the ACE/MUC RE paradigm. However, their task differs from ours in several ways. Firstly, they only consider relations which exist only between base NPs with common noun heads — named entities and pronouns are excluded. Additionally, and similarly to ACE/MUC, they do not mark relations which rely on discourse knowledge and restrict annotations to sentence internal relations. Also, rather than annotating full texts they focus on single sentences extracted from web searches.

Despite these important distinctions, the similarities mean that many of the methods used are relevant to entity instantiations, including the use of *kernel* methods. A range of work has applied tree kernels to the RE problem, applying kernels to shallow parses (Zelenko et al., 2003), dependency trees (Culotta and Sorensen, 2004; Bunescu and Mooney, 2005) and full constituency parses (Zhang et al., 2006; Zhou et al., 2007; Swampillai and Stevenson, 2011). Refinements include automatically deciding the portion of the tree required to learn the relation (Zhou et al., 2007) and combining unstructured features with tree kernels (Zhou et al., 2007; Swampillai and Stevenson, 2011).

Almost all RE research considers solely intrasentential relations. Swampillai and Stevenson (2011), however, apply tree kernels to the problem of *intersentential* RE. As a constituency parse tree pertains only to a single sentence, they join the two sentences containing the entities under a new ROOT node.

Other work has focused on unstructured features. Approaches include Bayesian networks (Roth and Yih, 2002), maximum entropy models (Kambhatla, 2004), Support Vector Machines (SVMs) (Zhou et al., 2005) and the inclusion of background knowledge (Chan and Roth, 2010; Sun et al., 2011).

Bridging Anaphora. Bridging anaphora are those anaphora which require inference from the reader to *bridge* the gap between anaphor and antecedent (Clark, 1975). The classical example is in the form of meronymy, as in Example 12[2] but bridging anaphora can also be connected to their antecedent by set membership, such as Example 13 (and Example 6). However, not all entity instantiations are bridged — Examples 1, 2, 5, 7, 8, 9, and 11 are amongst those that have non-anaphoric set members and subsets.

(12) I looked into *the room*. **The ceiling** was very high.

(13) I met *two people* yesterday. **The woman** told me a story.

[2]Examples 12 and 13 are from Clark (1975). The anaphor is in **bold**, the antecedent is in *italics*.

Theoretical linguistic literature has discussed set membership and subset bridging (Clark, 1975; Prince, 1981), and the phenomenon has been annotated in at least three corpora (Poesio, 2003; Nissim et al., 2004; Markert et al., 2012). Early computational approaches either used hand-crafted rules (Markert et al., 1996; Poesio et al., 1997; Vieira and Poesio, 2000) or focused solely on meronymy-based bridging (Markert et al., 2003; Poesio et al., 2004). More recent work has focused on learning the *information status* (IS) of an entity, rather than identifying its antecedent. The IS of an entity represents whether it is *new* to the reader, *old* because it is coreferent to a prior mention, or can be *mediated* from prior text, often by bridging. Most relevant to our work is the learning of fine-grained IS, which involves learning subtypes of the mediated category, including set membership. Rahman and Ng (2012) use the Switchboard corpus (Nissim et al., 2004), which includes a restricted version of set membership, and employ a feature set based on unigrams, markables and binary features based on hand-coded rules. Markert et al. (2012) learn fine-grained IS on a portion of OntoNotes corpus. They couple local features with a collective learning model, using links between instances based upon syntactic parent-child and precedence relations.

3 Annotation, Agreement and Corpus Study

We created a substantial corpus annotated for both inter- and intrasentential entity instantiations. Our initial corpus study in McKinlay and Markert (2011) covered 25 Penn Treebank (PTB) Wall Street Journal corpus (Marcus et al., 1993) texts, annotating solely between adjacent sentences. We first extended our intersentential annotation to cover an additional 50 PTB texts, and then added a second layer of intrasentential annotation to the same 75 texts.

3.1 Potential difficulties and Borderline Cases

We took inspiration from the Recognising Textual Entailment (RTE) task (Dagan et al., 2006). In RTE, the challenge is to automatically ascertain whether a text (T) *entails* a hypothesis (H). Rather than framing the problem as an issue of logical implicature, they regard RTE as an applied, empirical task:

> *We say that T entails H if, typically, a human reading T would infer that H is most likely true.* (Dagan et al. (2006))

We, as well, were interested in the phenomena from the perspective of a human reading the text, and so did not apply strict logical rules for identifying entity instantiation, and instead took an applied approach. While successful, this approach is not without drawbacks, and leads to a number of borderline cases.

The plural NPs[3] which act as sets in our corpus fall into 4 rough categories; extensionally defined, clearly intensionally defined, vaguely intensionally defined and generic. For those NPs which are either extensionally defined or are clearly intensional, set members are easy to identify. Examples 7 and 8 show extensionally defined sets, and the sets in Examples 2, 4 and 11 are clearly defined intensional examples.

The other two categories cause more difficulties. Not knowing the members in a vaguely intensionally defined set makes it difficult judging whether the relationship between NPs is a subset, coreference or set overlap. In Example 14, for instance, it is difficult to know for certain whether *'175'* and *'136'* are subsets of *'The 189 Democrats who supported the override yesterday'*, though it may be assumed to be the case.

(14) The 189 Democrats who supported the override yesterday compare with 175 who initially backed the rape-and-incest exemption two weeks ago and 136 last year.

In our annotation scheme, we make no distinction between those plural NPs which represent sets and those which represent generics, and allow instantiations to be drawn from both. This leads to annotation that is more akin to hyponymy than set membership or subset relationships, such as in Example 15.

[3] We restrict our set NPs to plural NPs, in order to reduce annotation effort. This does lead to the exclusion of some singular nouns which would be valid sets, such as *family, set* or *group*. In the future, we intend to include such nouns, either by means of a manually constructed list or using lexicosyntactic patterns.

Entity Instantiation	M&M (2011) corpus		Full corpus	
	# NP Pairs	%	# NP Pairs	%
Set Member	468	1.62	1477	1.89
Subset	180	0.62	641	0.82
No instantiation plural-singular NP pair	18 758	64.76	46 128	59.11
No instantiation plural-plural NP pair	9 560	33.00	29 793	38.18
Total	28 966	100.00	78 039	100.00

Table 1: Frequency of Intersentential Annotations, compared with 25 text corpus from McKinlay and Markert (2011).

(15) a. A customs official said the arrests followed a "Snake Day" at Utrecht University in the Netherlands, an event used by some collectors as an opportunity to obtain **rare snakes**.

b. British customs officers said they'd arrested eight men sneaking *111 rare snakes* into Britain [...]

Despite these problems, we still achieved substantial agreement. This is likely due to the genre of the texts involved; the financial-based newswire texts annotated tend to include many sets, subsets and members which are concrete, such as companies, countries and people. Applying this scheme to a genre of texts that contains more generics and less straightforwardly defined NPs, for example a philosophy text, could lead to a more problematic annotation. One possible way to improve agreement would be to introduce a layer of annotation that identified generic NPs, such as that employed by Reiter and Frank (2010), and prevent these generic NPs from participating in instantiations.

3.2 Intersentential Annotation

We follow our previous annotation method (McKinlay and Markert, 2011), automatically identifying plural and singular NPs, and separately displaying plural-plural NP pairs for subset annotation and plural-singular NP pairs for set member annotation. We also remove NPs that are appositions or predicates, and include the option to mark NPs as *"Not a mention"*, for excluding instances of non-referential *it*, idiomatic NPs and generic pronouns. The task of the annotator is then to indicate whether each NP pair is an instantiation. Each sentence pair is annotated both with sets in the first sentence and members/subsets in the second sentence, and sets in the second sentence and members/subsets in the first.

We annotated 50 PTB texts following this scheme, which combined with our original 25 texts gave us a corpus of 75 texts annotated for intersentential entity instantiations. Table 1 shows the frequency distribution of set members and subsets in both our original 25 texts and the full 75 text corpus.

3.3 Intrasentential Annotation

We added a layer of *intrasentential* entity instantiation annotation to the same 75 texts. We followed the same scheme of annotation as for the intersentential entity instantiations. However, we also included nested instantiations, such as those in Examples 8, 9 and 10.

3.3.1 Agreement Study and Gold Standard Corpus

Despite the differences between inter- and intrasentential annotation being minor, and the intersentential annotation scheme being previously shown to be reliable, we undertook a short agreement study. Five randomly selected texts were annotated by the two authors of this paper independently, and agreement was measured in the same three ways as in McKinlay and Markert (2011):

1. Does this pair of candidate NPs participate in a set membership/subset relationship or not?
2. Does this candidate set member/subset participate in a set membership/subset relationship with any potential set or not?
3. Is there an Entity Instantiation in this sentence?

Method	# Items Tested	Kappa	Agreement
1	3098 NP pairs	0.7493	97.81%
2	1414 NPs	0.7742	96.39%
3	237 sentences	0.7277	89.87%

Table 2: Intrasentential Agreement Statistics

Entity Instantiation	# NP pairs	%
Set Member	1 538	3.51
Subset	865	1.98
No instantiation plur-sing pair	24 363	55.63
No instantiation plur-plur pair	17 028	38.88
Total	43 794	100.00

Table 3: Frequency of Intrasentential Entity Instantiations in 75 texts

Relationship	Set Member		Other Sing-Plur pair	
Set NP Parent	1 065	(69.2%)	2 294	(9.4%)
Member NP Parent	55	(3.6%)	1 843	(7.6%)
Same Clause	84	(5.5%)	7 068	(29.0%)
Different Clause	334	(21.7%)	13 158	(54.0%)
Total	1 538	(100.0%)	24 363	(100.0%)

Table 4: Frequency of syntactic relationships between NPs in set member instantiations.

Relationship	Subset		Other Sing-Plur pair	
Set NP Parent	615	71.1%	1 489	8.7%
Subset NP Parent	85	9.8%	1 991	11.7%
Same Clause	90	10.4%	4 945	29.0%
Different Clause	75	8.7%	8 603	50.5%
Total	865	100.0%	17 028	100.0%

Table 5: Frequency of syntactic relationships between NPs in subset instantiations.

We achieve substantial agreement with all three metrics (see Table 2). Common disagreements consisted of matters of interpretation rather than any systematic problem with the scheme. One common disagreement, related to the issues mentioned in Section 3.1, was deciding whether two sets were in a subset relationship or overlapping, such as *'the key districts'* and *'the state's major cities'* in Example 16.

(16) With ballots from *most of the state's major cities* in by yesterday morning, the Republicans came away with 10% of the vote in several of **the key districts**.

The intrasentential annotation was then completed over the remaining 70 texts by the first author of this paper. The frequency distribution of these annotations is shown in Table 3. The final corpus of intersentential and intrasentential instantiations will be made publicly available, in a stand-off form, at `http://www.comp.leeds.ac.uk/markert/data.html`.

3.3.2 Intrasentential Syntactic Relationships

To gain an insight into the patterns tree kernels might learn, we computed the syntactic relationship between the two participant NPs in an entity instantiation, and compared this to the distribution of non-instantiations. We organised the relationships into four classes: the set NP was a parent of the member/subset NP (e.g Example 8), the member/subset NP was a parent of the set NP (e.g Example 10), the two NPs were not in a parent/child relationship but were in the same clause, and the two NPs were in different clauses (e.g. Example 11).

The results are shown in Tables 4 and 5. We found that in the majority of instantiations, the Set NP was a parent of the Member or Subset NP, and that the distribution of instantiations was significantly different from that of non-instantiations in both set members and subsets[4].

4 Experiments

We used a supervised machine learning approach to identify entity instantiations, treating set membership and subsets separately (see also McKinlay and Markert (2011)). We therefore divide our data set into two; plural-singular NP pairs that are labelled either *set member* or *no-instantiation* and plural-plural NP pairs that are labelled either *subset* or *no-instantiation*. We use the same feature set for both, employing two types of features; traditional unstructured features, and tree kernels.

[4]We used a χ^2 test for consistency in a 4×2 table with 3 degrees of freedom, giving $\chi^2 = 4605$ for set members and $\chi^2 = 3123$ for subsets, both corresponding to $p < 0.00000001$.

4.1 Unstructured features

Our unstructured features are identical to those presented in McKinlay and Markert (2011). They comprise five categories; *surface*, *salience*, *syntactic*, *contextual* and *knowledge*, and contain features that relate to a single NP, and those that represent cross-NP relationships. We list them briefly below, further details of the features can be found in McKinlay and Markert (2011).

Surface features. The unigrams, part-of-speech tags, lemmas and head words of each NP. Also included is Levenshtein's distance between the corresponding strings, the distance in characters and words between NP pairs, and a boolean feature which represents the order of the NPs.

Salience features. The grammatical role of each NP, whether it is the first mention of that entity in the sentence or document, the number of prior mentions and the overall number of mentions of the entity in the document.

Syntactic features. Syntactic parallelism and pre- and post-modification of each NP. The modification type includes values that represent apposition, conjunction, pre modification and bare nouns.

Contextual features. The Levin class (Levin, 1993) of each NP's head verb, as well as the verb itself, whether each NP is in a quotation, and an approximation of the discourse relations present in the two sentences by identifying likely discourse connectives and mapping them to their most frequent explicit relation in the Penn Discourse Treebank (PDTB) (Prasad et al., 2008).

Knowledge-based features. WordNet-based features which express synonymy/hyponymy between potentials members/subsets and sets. A feature which searches Freebase (Bollacker et al., 2008), for potential set member/subset NPs and compares the *topics* (loosely hyponyms) of matching entries to the potential set NP. A Point-wise Mutual Information feature derived from Google hit counts, based on the notion that the pattern "X and other Y", where X is a potential set member or subset and Y is a potential set, indicates hyponymy (Hearst, 1992; Markert and Nissim, 2005). A feature which establishes whether the animacy of the two NPs matches.

4.2 Tree Kernels

The unstructured features discussed in Section 4.1 are presented to the machine learner as a vector. Tree features are instead presented as structured data, and the learner works directly with this structured form.

We used two trees — Shortest Path Enclosed Tree (SPET) and Shortest Path Tree (SPT), which have been previously used for RE (Zhang et al., 2006; Swampillai and Stevenson, 2011). We also included two variations in the lexicalisation of these trees; full delexicalisation, in which all terminal nodes are removed, and partial delexicalisation, in which all terminals which represent nouns are removed.

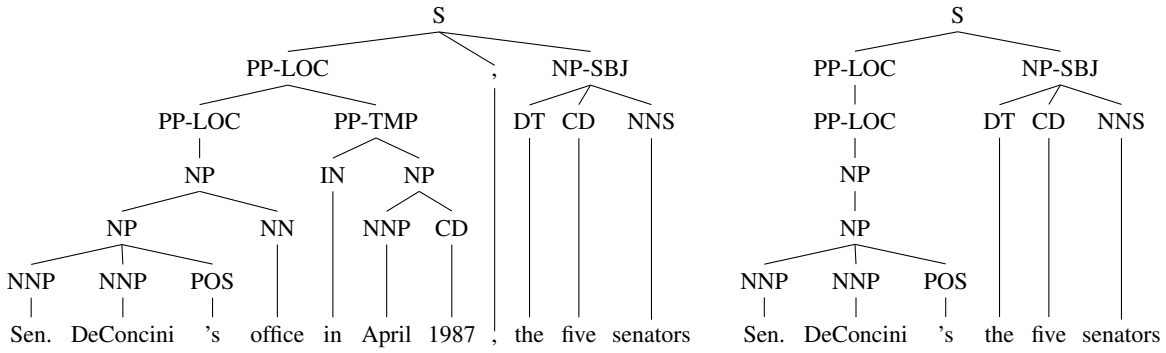
The SPET is the shortest path between the two NPs, inclusive of all nodes in between. SPT is identical, but *exclusive* of all nodes in between. Example 17 shows a sentence with two NPs underlined. Figure 1(a) and 1(b) show the SPET and SPT that connects them, respectively. We replace the node label of the subtree that represents the set member/subset NP with the node MEMBER, and node label of the subtree that represents the set NP with SET.

(17) In a highly unusual meeting in Sen. DeConcini's office in April 1987, the five senators asked federal regulators to ease up on Lincoln.

For intersentential entity instantiations we followed Swampillai and Stevenson (2011), joining the trees of the two sentences under a single node called ROOT and then extracting the trees as above.

4.3 Experimental Set Up

We considered the problems of intersentential and intrasentential instantiations separately, reasoning that intrasentential instantiations are a sufficiently different phenomena, and occur in patterns not found in intersentential instantiations. Our intuition was that syntax played a stronger role in identifying intrasentential instantiations, and that the tree kernels would have a greater impact on the intrasentential data.



(a) Shortest Path Enclosed Tree (SPET)

(b) Shortest Path Tree (SPT)

Figure 1: Examples of trees used for tree kernel learning.

We applied 10-fold cross-validation for testing and training in all our experiments, keeping pairs of NPs from the same text in the same fold to avoid over-training based on specific topical unigrams that may occur in a single text. We used SVM-LIGHT-TK (Moschitti, 2006b), an extension to SVM^{light} (Joachims, 1999) as our learner. We found that a linear kernel gave best results for flat features, and so used it in all experiments. For all tree kernel experiments we used the Subset tree kernel. We used addition, rather than multiplication, to combine tree and flat feature kernels.

The data contains many more negative examples than positive ones (only 3.7% of the 121,833 candidate pairs are positive). Previously we experimented with *balanced* data sets (McKinlay and Markert, 2011) — data sets in which the numbers of Entity Instantiations and non-Instantiations are equal — to demonstrate the utility of their features. We, however, focus on the original skewed data sets.

For comparison we include two baselines: *majority*, which predicts the majority class in each fold, and *unigram* which has two features, representing the unigrams of the two NPs.

4.4 Intrasentential Results and Discussion

The results of our intrasentential experiments are shown in Table 6. Precision, Recall and F-Score are calculated for the positive instances and SPTP represents SPT, Partially Lexicalised where as SPETF represents SPET, Fully Lexicalised and so on. We also include results of a feature ablation study, in which we removed each group of features in turn. We performed a similar experiment with our tree kernels, based on removing each of the 4 tree kernels in turn. We then combined the full set of tree kernels with the full feature set, and the best performing feature set with the best performing tree kernel combination according to the results of our ablation.

All our algorithms beat the baselines significantly[5]. In the unstructured feature ablation, the best performing algorithm involves the omission of contextual features for both set members and subsets.

The tree kernels have a slightly worse accuracy than the unstructured features but provide a higher precision. There are no significant differences between the performance of each tree kernel combination; there seems to be no difference between partial and full lexicalisation or between including or omitting intervening context in terms of accuracy. This suggests that a few structural features that all 4 representations have in common are important.

The combination of the best unstructured features and best tree kernels leads to significant improvements over either method in isolation for both set members and subsets. Also, the combination of all trees and all features is significantly better than the best unstructured and tree methods for subsets.

[5]McNemar's χ^2 test (1 d.f.) was used for all significance tests on results. Minimum χ^2 values were 280 for set members and 101 for subsets, both corresponding to $p < 0.00000001$.

	Set Members				Subsets			
Feature set	**Accuracy**	**P**	**R**	**F**	**Accuracy**	**P**	**R**	**F**
Majority	94.1%	—	—	—	95.2%	—	—	—
Unigrams	94.1%	—	—	—	95.2%	—	—	—
Unstructured Features								
All features	96.9%	0.847	0.578	0.687	$96.8\%^{\eta}$	0.842	0.425	0.565
All features - Surface	$96.2\%^{\delta}$	0.805	0.475	0.597	$96.1\%^{\gamma}$	0.774	0.282	0.414
All features - Salience	$95.7\%^{\gamma}$	0.836	0.337	0.481	$96.2\%^{\gamma}$	0.867	0.265	0.406
All features - Syntax	96.6%	0.835	0.538	0.654	$96.1\%^{\gamma}$	0.791	0.271	0.404
All features - Contextual	$97.0\%^{\alpha}$	0.849	0.597	0.701	$97.0\%^{\alpha}$	0.834	0.471	0.602
All features - World Knowledge	$96.7\%^{\delta}$	0.834	0.552	0.665	$96.6\%^{\gamma}$	0.852	0.788	0.833
Tree kernels								
SPTP+SPTF+SPETP+SPETF	96.7%	0.894	0.495	0.637	96.7%	0.937	0.342	0.501
SPTF+SPETP+SPETF	96.6%	0.897	0.486	0.630	96.7%	0.940	0.345	0.504
SPTP+SPETF+SPETP	$96.7\%^{\beta}$	0.914	0.491	0.638	96.7%	0.937	0.343	0.504
SPTP+SPTF+SPETF	96.6%	0.892	0.492	0.634	$96.7\%^{\beta}$	0.940	0.345	0.504
SPTP+SPTF+SPETP	96.7%	0.908	0.494	0.640	96.7%	0.934	0.343	0.502
Combination kernels								
All Trees + All features	97.0%	0.884	0.579	0.699	$97.2\%^{\epsilon}$	0.934	0.461	0.618
SPTF + SPTP + SPETF + All - Contextual	$97.1\%^{\epsilon}$	0.889	0.591	0.710	$97.3\%^{\epsilon}$	0.936	0.476	0.631
SPTP + SPETF + SPETP + All - Contextual	97.1%	0.886	0.586	0.705	$97.3\%^{\epsilon}$	0.935	0.479	0.633

Table 6: Intrasentential results.

α SVM flat-feature algorithm with highest accuracy
β Tree Kernel Algorithm with highest accuracy
γ Significantly *worse* than α, $p < 0.001$.
δ Significantly *worse* than α, $p < 0.05$.
ϵ Significantly *better* than α ($p < 0.05$) and β ($p < 0.001$)

4.5 Intersentential Results and Discussion

Intersentential instantiation identification is a more difficult problem than its intrasentential counterpart. The best F-scores achieved by us on the original data, rather than the artificially created balanced set, were 0.1938 and 0.1414 for set members and subsets respectively, and involved oversampling the positive instances (McKinlay and Markert, 2011). Our classifier had very poor recall without oversampling — 0.0289 for set members, 0.0266 for subsets — leading to F-Scores of 0.0527 and 0.0465.

In our experiments on the expanded corpus, we found that SVM-LIGHT-TK with the same options as our intrasentential experiments led to a classifier which always predicted the majority class, giving us a Precision, Recall and F-Score of 0. We had more success by using the cost-factor parameter to penalise errors on positive examples more heavily in the training process[6]. The value of the cost-parameter was set to $f(\text{Negative Examples})/f(\text{Positive Examples})$.

The results of our intersentential experiments are shown in Table 7. We improve over our previous non-oversampled results for set members and subsets, and the oversampled results for set members. However, as the corpus is triple the size, direct comparison is difficult. We find that whilst our tree kernels are more accurate than their unstructured counterparts, recall is much poorer, meaning that the unstructured features in isolation have the best F-Scores. The only algorithms that perform significantly differently to the unigram baseline are the unstructured set member classifier, which has worse accuracy but an increased F-Score, and the two subset classifiers which use tree kernels, which have higher accuracy but lower F-Scores. Our intuition that tree kernels would have less impact on intersentential instantiations, as they are not as syntax-dependent, appears accurate.

[6]Applying this additional setting to our intrasentential data produced classifiers with similar accuracy as before, but with reduced precision and increased recall. For example, on the All Trees + All Features combination, the classifier using the cost factor parameter scored an Accuracy/P/R/F of 96.5/71.0/69.5/70.2 for set members and 97.1/78.8/54.1/64.2 for subsets.

	Set Members				Subsets			
Feature set	**Accuracy**	**P**	**R**	**F**	**Accuracy**	**P**	**R**	**F**
Majority	96.9%	—	—	—	97.9%	—	—	—
Unigrams	95.1%	0.166	0.143	0.153	97.1%	0.058	0.023	0.033
All Unstructured	94.8%†	0.216	0.257	0.235	97.0%	0.146	0.086	0.108
All Trees	95.2%	0.217	0.214	0.215	97.8%†	0.042	0.002	0.003
All Trees + All Unstructured	95.2%	0.217	0.214	0.215	97.7%†	0.250	0.041	0.070

Table 7: Intersentential results.

† Significantly different from unigram baseline, $p < 0.001$.

5 Conclusion and Future Work

In this paper we make two novel contributions; the introduction of intrasentential entity instantiations, and the application of tree kernels to the detection of both intra- and intersentential entity instantiations. Our corpus of intrasentential entity instantiations is annotated with good agreement, and our statistics show that the majority of intrasentential instantiations have strong syntactic links between participating NPs. We then use tree kernels to learn directly from constituency parse tree data. Our tree kernels perform comparably to much larger and more varied set of unstructured features, that needed access to outside world knowledge sources. In addition, the combination of those unstructured features and tree kernels leads to significant improvements over either method in isolation on intrasentential data. Our best algorithms are highly precise.

In the future, we wish to explore the annotation of entity instantiations beyond adjacent sentences, and apply our scheme to genres other than newswire. We wish to explore different tree representations, such as those based on dependency structures, and different tree kernels, such as the more general Partial Tree Kernel (Moschitti, 2006a). We intend to improve our classification results by employing a global model for the joint learning of inter- and intrasentential entity instantiations.

Entity instantiations also have the potential to be useful for a number of applications, including discourse relation classification, sentiment analysis and summarisation. We wish to investigate the impact of entity instantiations on these applications.

Acknowledgements

Andrew McKinlay is funded by an EPSRC Doctoral Training Grant.

References

ACE (2000-2005). Automatic Content Extraction. `http://www.ldc.upenn.edu/Projects/ACE/`.

Bollacker, K., C. Evans, P. Paritosh, T. Sturge, and J. Taylor (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of ACM SIGMOD 2008*, pp. 1247–1250.

Bunescu, R. and R. Mooney (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of HLT/EMNLP 2005*, pp. 724–731.

Chan, Y. and D. Roth (2010). Exploiting background knowledge for relation extraction. In *Proceedings of COLING 2010*, pp. 152–160.

Clark, H. (1975). Bridging. In *Proceedings of the 1975 Workshop on Theoretical Issues in Natural Language Processing*, pp. 169–174.

Culotta, A. and J. Sorensen (2004). Dependency tree kernels for relation extraction. In *Proceedings of ACL 2004*, pp. 423.

Dagan, I., O. Glickman, and B. Magnini (2006). The PASCAL Recognising Textual Entailment Challenge. In J. Quiñonero Candela, I. Dagan, B. Magnini, and F. d'Alché Buc (Eds.), *Machine Learning Challenges*, Volume 3944, Chapter 9, pp. 177–190. Springer Berlin Heidelberg.

Hearst, M. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of COLING 1992*, pp. 539–545.

Hendrickx, I., S. N. Kim, Z. Kozareva, P. Nakov, D. Ó Séaghdha, S. Padó, M. Pennacchiotti, L. Romano, and S. Szpakowicz (2010). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pp. 33–38.

Joachims, T. (1999). Making large-scale support vector machine learning practical. In *Advances in kernel methods*, pp. 169–184. MIT Press.

Kambhatla, N. (2004). Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of ACL 2004*, pp. 22.

Levin, B. (1993). *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

Marcus, M., M. Marcinkiewicz, and B. Santorini (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics 19*(2), 313–330.

Markert, K., Y. Hou, and M. Strube (2012). Collective classification for fine-grained information status. In *Proceedings of ACL 2012*, pp. 8–14.

Markert, K., N. Modjeska, and M. Nissim (2003). Using the web for nominal anaphora resolution. In *Proceedings of EACL 2003 Workshop on the Computational Treatment of Anaphora*, pp. 39–46.

Markert, K. and M. Nissim (2005). Comparing knowledge sources for nominal anaphora resolution. *Computational Linguistics 31*(3), 367–402.

Markert, K., M. Strube, and U. Hahn (1996). Inferential realization constraints on functional anaphora in the centering model. In *Proceedings of CogSci 1996*, pp. 609–614.

McKinlay, A. and K. Markert (2011, September). Modelling entity instantiations. In *Proceedings of RANLP 2011*, pp. 268–274.

Moschitti, A. (2006a). Efficient convolution kernels for dependency and constituent syntactic trees. In *Proceedings of ECML 2006*, pp. 318–329.

Moschitti, A. (2006b). Making tree kernels practical for natural language learning. In *Proceedings of EACL 2006*, Volume 6, pp. 113–120.

MUC (1987-1998). Message Understanding Conferences. The NIST MUC website: `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/`.

Nissim, M., S. Dingare, J. Carletta, and M. Steedman (2004). An annotation scheme for information status in dialogue. In *Proceedings of LREC 2004*.

Poesio, M. (2003). Associative descriptions and salience: A preliminary investigation. In *Proceedings of the EACL 2003 Workshop on The Computational Treatment of Anaphora*.

Poesio, M., R. Mehta, A. Maroudas, and J. Hitzeman (2004). Learning to resolve bridging references. In *Proceedings of ACL 2004*, pp. 143.

Poesio, M., R. Vieira, and S. Teufel (1997). Resolving bridging references in unrestricted text. In *Proceedings of Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pp. 1–6.

Prasad, R., N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, and B. Webber (2008). The Penn Discourse TreeBank 2.0. In *Proceedings of LREC 2008*, pp. 2961–2968.

Prince, E. (1981). Toward a Taxonomy of Given-New Information. *Radical Pragmatics 3*, 223–255.

Rahman, A. and V. Ng (2012). Learning the fine-grained information status of discourse entities. In *Proceedings of EACL 2012*.

Reiter, N. and A. Frank (2010). Identifying generic noun phrases. In *Proceedings of ACL 2010*, pp. 40–49.

Roth, D. and W. Yih (2002). Probabilistic reasoning for entity & relation recognition. In *Proceedings of COLING 2002*, pp. 1–7. ACL.

Sun, A., R. Grishman, and S. Sekine (2011). Semi-supervised relation extraction with large-scale word clustering. In *Proceedings of ACL-HLT 2011*, pp. 521–529.

Swampillai, K. and M. Stevenson (2011, September). Extracting relations within and across sentences. In *Proceedings of RANLP 2011*, pp. 25–32. RANLP 2011 Organising Committee.

Vieira, R. and M. Poesio (2000). An empirically based system for processing definite descriptions. *Computational Linguistics 26*(4), 539–593.

Zelenko, D., C. Aone, and A. Richardella (2003). Kernel methods for relation extraction. *The Journal of Machine Learning Research 3*, 1083–1106.

Zhang, M., J. Zhang, J. Su, and G. Zhou (2006). A composite kernel to extract relations between entities with both flat and structured features. In *Proceedings of COLING/ACL 2006*, pp. 825–832.

Zhou, G., J. Su, J. Zhang, and M. Zhang (2005). Exploring various knowledge in relation extraction. In *Proceedings of ACL 2005*, pp. 427–434.

Zhou, G., M. Zhang, D. Ji, and Q. Zhu (2007). Tree Kernel-Based Relation Extraction with Context-Sensitive Structured Parse Tree Information. In *Proceedings of EMNLP-CoNLL 2007*, pp. 728–736.