Projecting Semantic Roles via Tai Mappings

Hector-Hugo Franco-Penya
Trinity College Dublin
Dublin, Ireland.
francoph@scss.tcd.ie

Martin Emms
Trinity College Dublin
Dublin, Ireland.
mtemms@scss.tcd.ie

Abstract

This work takes the paradigm of projecting annotations within labelled data into unlabelled data, via a mapping, and applies it to Semantic Role Labelling. The projections are amongst dependency trees and the mappings are the Tai-mappings that underlie the well known tree edit-distance algorithm. The system was evaluated in seven different languages. A number of variants are explored relating to the amount of information attended to in aligning nodes, whether the scoring is distance-based or similarity-based , and the relative ease with which nodes can be ignored. We find that all of these have statistically significant impacts on the outcomes, mostly in language - independent ways, but sometimes language dependently.

1 Introduction

There are a number of pattern recognition scenarios that have the characteristics that one has some kind of *structured* test data (sequence, tree, graph, grid) within which some annotation is missing, and one wants to infer the missing annotation by exploiting fully annotated training data. A possible approach is to seek to define *alignments* between training and test cases, and to use these to *project* annotation from the training to test cases. This has been successfully used in computational biology, for example, to project annotation via sequence alignments (Marchler-Bauer et al., 2002) and graph alignments (Kolar et al., 2008). Semantic Role Labeling (SRL) can be seen as a further instance of this pattern recognition scenario and we will describe in this paper an SRL system which works by projecting annotations over an alignment. This paper extends results reported by Franco-Penya (2010). In the remainder of this section we give a brief overview of SRL. Section 2 then describes our system, followed in sections 3 and 4 by discussion of its evaluation.

For a number of languages, to an existing syntactic treebank, a layer of *semantic role* information has been added: the evolution of the Penn Treebank into PropBank is an example (Palmer et al., 2005). Role-label inventories and annotation principles vary widely, but our system has been applied to data annotated along the same lines as exemplified by PropBank. The example below illustrates a PropBank role-labelling.[1]

[*Revenue*]$_{A1}$ *edged* [*up*]$_{A5}$ [*3.4 %*]$_{A2}$ [*to $904 million*]$_{A4}$ [*from $874 million*]$_{A3}$ [*in last year's third quarter*]$_{TMP}$

A lexical item (such as *edge*), is given a frameset of enumerated core argument roles (A0 ... A5). In the example, A3 is the start point of the movement, and a minimal PropBank commitment is that A3 and the other enumerated role identifiers are used consistently across different tokens of *edge*. Across different lexical items, commitments concerning continuity in the use of the enumerated arguments are harder to state – see the conclusions in section 5. There are also named roles (such as TMP above), whose use across different items is intended to be consistent.

[1] To save space, this shows simply a labelling of subsequences, omitting syntactic information. Figure 2 shows annotation added to a syntactic structure.

Arising from the CoNLL-2009 SRL evaluation shared task (Hajič et al., 2009), for seven languages (**Ca**talan, **Ch**inese, **Cz**ech, **En**glish, **Ge**rman, **Ja**panese, **Sp**anish), there is data consisting of a role-annoation layer added to syntactic information. The syntactic information is expressed as dependency trees. In some cases this is derived from a primary constituent-structure representation (eg. English), and in other cases it is the 'native' representation (eg. Czech). For each language, tree nodes have four kinds of syntactic information: FORM: a word form; LEMMA: lemma of the word form; POS: part-of-speech tag (tag sets are language specific); DEPREL: the dependency relation to its head word (the relation-sets are language specific). Additionally in each tree, $\mathcal{T}$, a number of nodes are identified as predicate nodes. Each predicate node $p^{\mathcal{T}}$, is linked to a set of argument nodes, $args(p^{\mathcal{T}})$. For each $a_i^{\mathcal{T}} \in args(p^{\mathcal{T}})$, the link $(p^{\mathcal{T}}, a_i^{\mathcal{T}})$ is labelled with a role. The role-labeling follows the PropBank approach.

2 An Alignment-based SRL system

Sub-tree extraction A preliminary to the role-labelling process itself is to extract a sub-tree that is relevant to a given predicate and its arguments. Where p is a particular predicate node of a tree $\mathcal{T}$, let $sub\_tree(\mathcal{T}, p)$ stand for the relevant sub-tree of $\mathcal{T}$. There is considerable latitude in how to define this, and we define it in a simple way, via the least upper bound, lub, of p and $args(p)$: $sub\_tree(\mathcal{T}, p)$ includes all nodes on paths down from lub to p and $args(p)$. This is the same sub-tree notion as used by Moschitti et al. (2008). Figure 1 illustrates. Henceforth all trees will be assumed to have been extracted in this way.

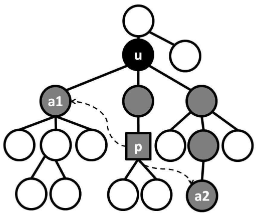

Figure 1: Sub-tree extraction: $a1, a2 = args(p)$, $u = lub(a1, a2, p)$.

Alignment and Projection Let an *alignment* of trees S and T be a *1-to-1*, *partial* mapping α : $S \mapsto T$, from the nodes of S *into* the nodes of T. If S is role-labelled, such an alignment projects a role for test tree argument a_i^T *if* it is aligned to a training tree argument, a_j^S, and the predicate nodes p^S and p^T are aligned: the role of a_j^S is projected to a_i^T. Such a role-projecting tree will be termed '*usable for T*'. Fig. 2 shows an example alignment between subtrees, with the aligned sub-trees shown in the context of the trees $\mathcal{S}$ and $\mathcal{T}$ from which they come. Argument nodes $\mathcal{T}_4$, $\mathcal{T}_6$ and $\mathcal{T}_7$ would receive projected labels $A1$, $A2$, and $A3$ from $\mathcal{S}_7$, $\mathcal{S}_{12}$ and $\mathcal{S}_{13}$. The first two are correct, whilst $\mathcal{T}_7$'s annotation should be A4.

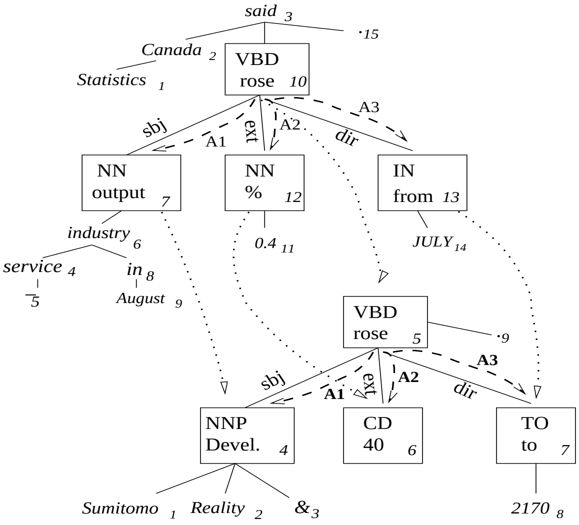

Figure 2: An example alignment.

Algorithm outline Let $[\Delta]^i_{d_i}$ be an equivalence class, contains all training samples at distance d_i to T. The training set can be thought of as a sequence of equivalence classes $[\Delta]^1_{d_1}, [\Delta]^2_{d_2}, \ldots$, for increasing distance.[2] The algorithm works with a PANEL of nearest neighbours, which is always a prefix of this sequence of equivalence classes. Where T is defined by predicate p and arguments $a_1 \ldots a_n$, the algorithm to predict a label for each a_i is

1. *make sorting of training trees and set* PANEL *of nearest neighbours to be first equivalence class* $[\Delta]^1_{d_1}$

2. (i) *make a set of predictions from the* usable *members of* PANEL (ii) *if there is a most fre-*

[2] Or alternatively a sequence of similarity equivalence classes, at decreasing similarities to T.

quent prediction, return it (iii) *if there is a tie, or no usable members, add next equivalence class to* PANEL *if possible and go to (i), else return 'unclassified'*

Tai mappings In this work, alignments are restricted to be so-called *Tai* mappings (Tai, 1979): amongst all possible 1-to-1, partial mappings from S into T, $\alpha : S \mapsto T$, these are mappings which respect *left-to-right order* and *ancestry*.[3] Then to select a *preferred* alignment, a score is assigned to it. The definitions relevant to this which are given below follow closely those of Emms and Franco-Penya (2012).

Because a mapping α is partial on S and *into* T, there is a set $\mathcal{D} \subseteq S$ ('deletions'), of those $i \in S$ which are not mapped *to* anything and a set $\mathcal{I} \subseteq T$ ('insertions'), of those $j \in T$ which are not mapped *from* anything, and the alignment scorings make reference to the sets α, $\mathcal{D}$ and $\mathcal{I}$.

We consider both a 'distance' scoring, $\Delta(\alpha : S \mapsto T)$, whose *minimum* value is used to select the alignment, and a 'similarity' scoring $\Theta(\alpha : S \mapsto T)$, whose *maximum* value is used. The 'distance' scoring, $\Delta(\alpha : S \mapsto T)$, is given by

$$\sum_{(i,j)\in\alpha} C^{\Delta}(i^{\gamma}, j^{\gamma}) + \sum_{i\in\mathcal{D}} C^{\Delta}(i^{\gamma}, \lambda) + \sum_{j\in\mathcal{I}} C^{\Delta}(\lambda, j^{\gamma})$$

Here $(.)^{\gamma}$ gives the label of a node, and a function C^{Δ} is used to give label-dependent costs to the members of α, $\mathcal{D}$ and $\mathcal{I}$. This is the major parameter of the distance scoring and the various setting for it which were considered are detailed below; at a general level, to accord minimally with intuition, it should always be the case that $C^{\Delta}(x,y) \geq 0$, and $C^{\Delta}(x,y) \geq C^{\Delta}(x,x)$ for non-identical x and y. A 'similarity' scoring, $\Theta(\alpha : S \mapsto T)$, is given by

$$\sum_{(i,j)\in\alpha} C^{\Theta}(i^{\gamma}, j^{\gamma}) - \sum_{i\in\mathcal{D}} C^{\Theta}(i^{\gamma}, \lambda) - \sum_{j\in\mathcal{I}} C^{\Theta}(\lambda, j^{\gamma})$$

where C^{Θ} is a function defining costs for the members of α, $\mathcal{D}$ and $\mathcal{I}$. To accord minimally with intuition, $C^{\Theta}(x,y) \leq C^{\Theta}(x,x)$. Additionally, in this work, we also assume that $C^{\Theta}(x,\lambda) = C^{\Theta}(\lambda,x) = 0$.

Besides ranking alternative alignments between fixed S and T, the minimum distance alignment score defines a 'distance', $\Delta(S,T)$, for the pair (S,T), and maximum similarity alignment scores define a 'similarity', $\Theta(S,T)$, and these are used to rank alternative neighbours for a tree to be labelled. The algorithm to calculate $\Delta(S,T)$ and $\Theta(S,T)$ follows very closely that of Zhang and Shasha (1989): although originally proposed in the context of 'distance' and minimisation, it is straightforwardly adaptable to the context of 'similarity' and maximisation.

Cost settings On this data-set the label is in general a 4-tuple (p,d,l,f) of part-of-speech, dependency-relation, lemma, and word form. Four settings for the swap costs, $C^{\Delta}(x,y)$, are considered: B('binary'), T('ternary'), H('hamming') and FT('frame ternary'), based on the matches/mis-matches on these features. For any given feature a, let a^{δ} represent match/mis-match on a, with 1 for mis-match and 0 for match. The different swap settings are then defined as below

	$C^{\Delta}(x,y)$	values
B	$p^{\delta} \times d^{\delta}$	$0, 1$
T	$\frac{1}{2}[p^{\delta} + d^{\delta}]$	$0, \frac{1}{2}, 1$
H	$\frac{1}{4}[p^{\delta} + d^{\delta} + l^{\delta} + f^{\delta}]$	$0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1$
FT	$\frac{1}{2}[p^{\delta} + d^{\delta}] + fr^{\delta}$	$0, \frac{1}{2}, 1, \frac{3}{2}, 2$

For FT, fr refers to a synthesised attribute: for predicate nodes, fr is the frame identifier, and otherwise $fr = \_$. The effect is that $fr^{\delta} = 1$ if one node is a predicate and the other is not, or if both are predicates but from different frames. Besides these swap settings, the deletion cost $C^{\Delta}(x,\lambda) = C^{\Delta}(\lambda,x)$ was varied between 1 and 0.5. From each swap setting $C^{\Delta}(x,y)$, two 'similarity' settings were derived by $C^{\Theta}(x,y) = \delta - C^{\Delta}(x,y)$, for $\delta = 1$ or 2.

Some aspects of these choices are based on results concerning distance and similarity which we previously established (Emms and Franco-Penya, 2012), where we showed that the conversion

$$\begin{aligned} C^{\Theta}(x,y) &= 2\kappa - C^{\Delta}(x,y) \\ C^{\Theta}(x,\lambda) &= C^{\Delta}(x,\lambda) - \kappa \\ C^{\Theta}(\lambda,y) &= C^{\Delta}(\lambda,y) - \kappa \end{aligned}$$

[3] More precisely, where $anc(x,y)$ and $left(x,y)$ denote the 'ancestor of' and 'to the left of' relations, $\forall(i,j) \in \alpha, \forall(i',j') \in \alpha$, the mapping must satisfy (i) $left(i,i')$ iff $left(j,j')$ and (ii) $anc(i,i')$ iff $anc(j,j')$

converts C^{Δ} to C^{Θ} such that the same ordering is induced over the alternative alignments for any given pair (S, T). Call such settings **A-duals**. For a given choice from B/T/H/FT, the four above-mentioned settings for C^{Δ} and C^{Θ} thus stand in the following A-duality relationships:

distance	A-dual similarity
(a) $C^{\Delta}(x, \lambda) = \mathbf{1}$	(c) $C^{\Theta}(x, y) = \mathbf{2} - C^{\Delta}(x, y)$
(b) $C^{\Delta}(x, \lambda) = \mathbf{0.5}$	(d) $C^{\Theta}(x, y) = \mathbf{1} - C^{\Delta}(x, y)$

For such dual settings, the labelling potential of a given training sample is necessarily *identical* under the two settings. However, this A-duality property is distinct, potentially, from **N-duality**, which is the property that the two settings induce the same ranking of candidate neighbours $\{S_1 \dots S_N\}$ to a given T. Because the role-labelling system is driven both by neighbour and alignment ordering, it is an empirical question whether or not A-dual settings will deliver the same outcomes.

We will refer to settings (a) and (b) as 'dist(del=1)' and 'dist(del=0.5)', and settings (c) and (d) as 'sim(2-swap)' and 'sim(1-swap)'.

3 Experiment Procedure

The seven languages of the CoNLL-2009 SRL task were used (Hajič et al., 2009), with the same division into training sets and evaluation sets.

Due to our computational limitations the English training data set was limited to the first 20,000 sentences and the Czech training data set was limited to the first 10,000 sentences.

A simple labeling accuracy score is reported. When a two-way contrast is considered, the significance of a difference in outcomes under the two settings was determined by the McNemar test (McNemar, 1947), at levels of significance $p < 0.05$ and $p < 0.001$, as do Fürstenau and Lapata (2011) in their role-labelling work.

4 Results

4.1 Contrasting Swap Settings

Figure 3 shows a graph with the accuracy for the seven languages on the evaluation data set, using the dist(del=1) setting.[4] The languages are sorted by the accuracy of the H setting. The baseline (first bar) corresponds to the accuracy that would be attained by always choosing the label that is most frequent for that language.

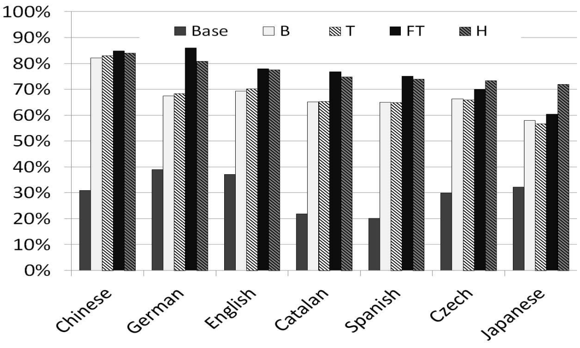

Figure 3: Accuracy of Tree edit distance across the seven languages.

Looking first at the H setting, the performance across the languages ranges from 73%(Czech) to 84%(Chinese), which in all cases substantially out-performs the majority-class base line (which is in the range 20(Spanish)–40%(German)). The other cost-settings also clearly out-perform this base line. It also seems that the variation in accuracies across the languages is not relatable to the variation in this majority-class base line.

The performance under the B and T settings is perhaps surprising, when one recalls that these two settings pay no attention at all to lexical differences between nodes, and refer only to the dependency relations and the part of speech. Nonetheless, even in this setting of tolerance to node exchange, it seems that in concert with the structure-respecting requirements of Tai mappings, surprisingly high accuracies can be obtained.

Chinese has the highest overall accuracy, with even the very simplest settings reaching relatively high accuracy. It is interesting to note that amongst the highest performing other systems which have used the same data, performance on the Chinese data-set has tended to be *below* that of other languages, with Che et al. (2009) and Dai et al. (2009) reporting over 81 F1 in all languages except Chinese where the F1 reported was 76.38 and 76.23. It suggests that tree distance and labelling by alignment methods may be especially suitable for the Chinese data set.

Japanese reports the worst results, especially

[4] The full table of values, including the 3 out-of-domain cases, appears as the first column in Table 4.

for the B/T/FT settings. This is probably due to the fact that, on inspection, the Japanese data gives 96.1% of syntactic dependencies the same dependency relation, practically canceling the contribution of the DEPREL feature.

The outcomes for the Spanish and Catalan data sets are very similar to each other. This is not unexpected as for the most part one is a translation of the other and they were annotated in the same way with the same set of labels.

Table 1 summarises the outcomes of pair-wise comparisons of the swap settings. Under '1st > 2nd', !l,*m,**n appears if there were l data-sets on which the 1st setting out-performed the 2nd setting, for m of these the outcomes were significantly different ($p = 0.05$), and for n of these the difference holds at a stricter significance level ($p = 0.001$).

	Settings	1st > 2nd	2nd > 1st	avge 2nd-1st
total	B-T	!4, *2, **1	!6, *3, **3	0.341%
	B-FT	!0, *0, **0	!10, *10, **10	7.45%
	B-H	!0, *0, **0	!10, *10, **9	7.779%
	T-FT	!0, *0, **0	!10, *10, **10	7.109%
	T-H	!0, *0, **0	!10, *9, **9	7.438%
	FT-H	!7, *6, **4	!3, *3, **3	0.329%

Table 1: Comparing swap-settings, for dist(del=1), on the 7 evaluation data-sets, and 3 out-of-domain data-sets.

The T and B settings turn out to give rather similar outcomes. Table 1 shows that T's margin over B averages out to 0.341%. There are 3 strictly significant cases where T out-performs B, and 1 cases in the other direction.

In its turn the H setting always out-performs the T setting, 9 times out of 10 at the strictest significance level, with the average margin being 7.438%. Thus penalising lexical difference seems always to improve performance, and nearly always substantially, though for the Chinese and out-of-domain English data sets, the margin for H over T falls to less than 1%.

The outcomes with FT are more language dependent. FT out-performs H more often (7!,6*) than H out-performs FT (3!,3*) and English is the one data-set on which the two are not significantly different. Japanese shows the highest margin in favour of H (11.5%) whilst German shows the highest margin in favour of FT (5.2%). The poor relative performance of FT for Japanese is again probably a function of the uninformative nature of its dependency annotation.

For all languages FT out-performs T, at the strictest significance level, with the margin averaging out to 7.1%. For German the margin is especially large at 17.8%.

4.2 Contrasting Representations

Table 2 compares the results obtained with trees to results obtained with a *linearised* version, using just a node sequence corresponding to the sequences of words that spans the predicate and argument nodes. Encoding these as linear trees, the tree-matching in this case reduces to the standard Levenshtein matching.

Settings	Tree > Lev.	Lev > Tree	avge Lev-Tree
B	*8, **6	*0, **0	-2.83%
T	*10, **10	*0, **0	-4.574%
FT	*10, **10	*0, **0	-7.758%
H	*10, **10	*0, **0	-8.113%

Table 2: Comparing Tree and Levenshtein outcomes.

As is evident, for all languages and all swap-settings, the alignment on the linear representation gives substantially poorer results than the alignment on the trees. For T/FT/H in each single experiment the tree version produce better score than the linear version, and in B it was never detected a statistical advantage of the linear version over the tree version. This indicates that the Tai-mapping constraints on the tree-representation definitely leverage information that is beneficial to this labeling task.

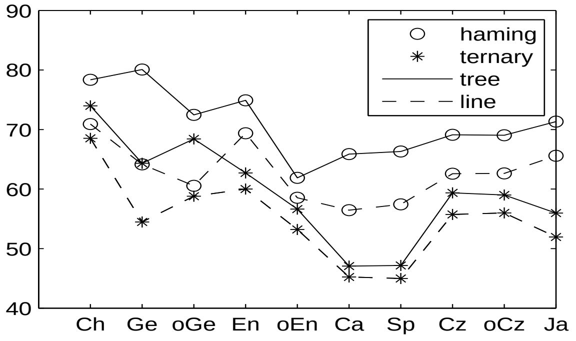

Figure 4: Tree vs Levenshtein distance (T & H).

Setting		max(dist)	max(sim)	difference
Chinese	B **	82.82%	**83.46%!**	0.6459%
	T =	83.98**%!**	83.86%	-0.1155%
	FT *	85.88**%!**	85.53%	-0.3464%
	H **	85.06%	**85.88%!**	0.8227%
German	B *	67.41**%!**	65.74%	-1.676%
	T *	68.25**%!**	65.74%	-2.514%
	FT **	**87.71%!**	85.01%	-2.7%
	H =	81.38**%!**	80.82%	-0.5587%
o-German	B *	69.51**%!**	67.59%	-1.926%
	T *	69.1**%!**	67.09%	-2.01%
	FT *	75.46**%!**	73.03%	-2.429%
	H =	72.7%	73.53**%!**	0.8375%
English	B **	69.34%	**70.21%!**	0.8674%
	T =	70.79**%!**	70.48%	-0.3092%
	FT =	78.75**%!**	78.49%	-0.2619%
	H *	78.72%	79.43**%!**	0.7171%
o-English	B =	63.08%	63.67**%!**	0.5944%
	T =	66.26**%!**	65.14%	-1.119%
	FT =	69.83**%!**	69.51%	-0.3147%
	H =	66.68%	67.52**%!**	0.8392%
Catalan	B **	**65.7%!**	64.11%	-1.587%
	T **	**65.7%!**	63.88%	-1.818%
	FT **	**79.35%!**	77.75%	-1.596%
	H =	75.8**%!**	75.58%	-0.2217%
Spanish	B **	**65.5%!**	64.23%	-1.268%
	T **	**65.7%!**	63.98%	-1.717%
	FT **	**77.25%!**	76.2%	-1.049%
	H =	75.16%	75.64**%!**	0.4736%
Czech	B =	66.28**%!**	66.21%	-0.07138%
	T **	**66.69%!**	66%	-0.6935%
	FT *	70.05**%!**	69.53%	-0.5226%
	H **	74.05%	**74.77%!**	0.7138%
o-Czech	B *	64.42%	64.92**%!**	0.497%
	T =	65.08**%!**	64.73%	-0.3529%
	FT **	**69.24%!**	68.17%	-1.066%
	H *	72.56%	73.36**%!**	0.7995%
Japanese	B =	57.93**%!**	57.31%	-0.6207%
	T **	**57.97%!**	55.86%	-2.106%
	FT **	**61.52%!**	58.94%	-2.577%
	H **	**75.34%!**	71.32%	-4.025%
total	B	*4, **2	*3, **2	-0.4545%
	T	*6, **4	*0, **0	-1.276%
	FT	*8, **5	*0, **0	-1.286%
	H	*1, **1	*4, **2	0.03983%
	all	*19, **12	*7 ,**4	-0.7441%

Table 3: max(dist) vs max(sim): max(dist) is best of dist(del=1) and dist(del=0.5), max(sim) is best of sim(swap=2) and sim(swap=1).

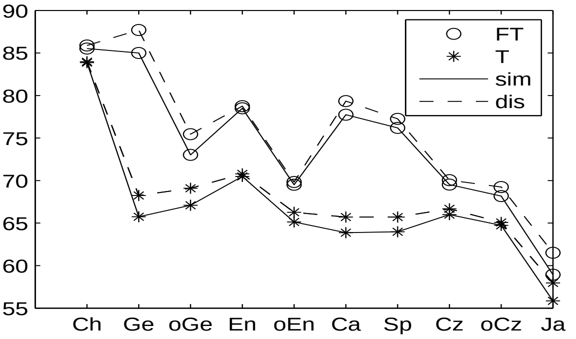

Figure 5: Distance outcomes compared to Similarity outcomes (T and FT).

4.3 Contrasting Distance and Similarity

Table 3 compares the best results of distance and similarity. In the case of distance, the best value is between dist(del=1) and dist(del=0.5), and in the case of similarity, the best value is between sim(2-swap) and sim(1-swap). Figure 5 plots the this contrast for the T and FT swap settings.

For the T overall the tree distance performs better than tree similarity. For T, this occurs 4 times at the strictest significance level, 6 times at the less strict level, there is no language where similarity outperforms distance, and the average margin is 1.28%. For B, this trend is less clear, with just 2 cases where distance out-performed similarity at the strictest significance level. The margins are small and average out at 0.45% in favour of the distance setting.

For FT, this occurs 5 times at the strictest significance level, 8 times at the less strict level, there is no language where similarity outperforms distance, and the average margin is 1.29%. German, Catalan, Spanish and Japanese show the largest margin, whilst Chinese, English and Czech the smallest.

For the H setting, overall the margin between distance and similarity is small, 0.04%. Similarity out-performs distance 2 times at the strictest significance level and 4 times at the less strict level. Japanese is unusual, being the only case where the comparison is statistically significantly in favour of *distance*, by a margin of 4.03%.

Recall from section 2 that the distance and similarity settings can be paired off as alignment-duals, namely ’dist(del=1)’ with ’sim(2-swap)’ and ’dist(del=0.5)’ with ’sim(1-swap)’. For such dual settings the labelling potential of a given training sample is necessarily *identical* under the two settings. We noted that this does not theoretically guarantee identical system outcomes: A-duality does not imply N-duality, that is identical *neighbour*-ordering. The experiments actually show that it is indeed not the case for these data-sets that the A-dual settings are also N-dual settings. Space precludes giving details of this, but it is implied by Table 3 and Fig. 5: for accuracies a_1 and a_2 attained by the distance alternatives, if N-duality held, the dual similarity settings would also attain accuracies a_1 and a_2, and we would observe no differences between maximum distance outcomes and maximum similarity outcomes.

4.4 Contrasting two Distance settings

Cost setting		del=1	del=.5	difference
Chinese	B **	82.14%	**82.82%!**	0.6748%
	T **	83%	**83.98%!**	0.9743%
	FT **	84.78%	**85.88%!**	1.093%
	H **	83.93%	**85.06%!**	1.129%
German	B *	67.41**%!**	65.83%	-1.583%
	T =	68.25**%!**	67.41%	-0.838%
	FT *	86.03%	87.71**%!**	1.676%
	H =	80.82%	81.38**%!**	0.5587%
o-German	B *	69.51**%!**	67.76%	-1.759%
	T =	69.1**%!**	68.84%	-0.2513%
	FT =	74.62%	75.46**%!**	0.8375%
	H =	72.53%	72.7**%!**	0.1675%
English	B =	69.34**%!**	69.18%	-0.1675%
	T **	70.12%	**70.79%!**	0.6656%
	FT **	77.98%	**78.75%!**	0.7687%
	H **	77.57%	**78.72%!**	1.147%
o-English	B =	63.08**%!**	63.01%	-0.06993%
	T =	65.77%	66.26**%!**	0.4895%
	FT *	69.83**%!**	68.39%	-1.434%
	H *	66.68**%!**	64.9%	-1.783%
Catalan	B **	65.12%	**65.7%!**	0.5764%
	T *	65.23%	65.7**%!**	0.47%
	FT **	76.77%	**79.35%!**	2.572%
	H **	74.76%	**75.8%!**	1.038%
Spanish	B **	64.97%	**65.5%!**	0.5243%
	T **	64.89%	**65.7%!**	0.8118%
	FT **	75.04%	**77.25%!**	2.216%
	H **	73.95%	**75.16%!**	1.209%
Czech	B **	**66.28%!**	65.47%	-0.8158%
	T **	65.84%	**66.69%!**	0.8464%
	FT =	69.99%	70.05**%!**	0.06119%
	H **	73.28%	**74.05%!**	0.7699%
o-Czech	B **	**64.42%!**	63.21%	-1.21%
	T =	64.73%	65.08**%!**	0.3457%
	FT **	**69.24%!**	68.19%	-1.044%
	H =	72.56**%!**	72.23%	-0.3313%
Japanese	B =	57.93**%!**	57.4%	-0.5266%
	T *	56.69%	57.97**%!**	1.279%
	FT *	60.43%	61.52**%!**	1.091%
	H **	71.92%	**75.34%!**	3.423%
total	B	*4, **2	*3, **3	-0.4356%
	T	*0, **0	*6, **4	0.4793%
	FT	*2, **1	*6, **4	0.7837%
	H	*1, **0	*6, **6	0.7327%

Table 4: Tree distance with del=1 vs del=0.5.

Table 4 contrasts the two tree-distance settings[5], one where the deletion cost is 1 and another where the deletion cost is 0.5.

The main observation is that there is a tendency for the del=0.5 setting to out-perform del=1. The differences are small, usually less than one per cent. Figure 6 plots the outcomes for T and FT, and Figure 7 shows H outcomes.

For B, 3 times del=0.5 was strictly statistically better than del=1, but the converse was also the case 2 times, with margin averaging out at 0.44% in favour of del=1. For T, del=0.5 often out-performed del=1, and significantly so (*6,**4), whilst del=1 never significantly out-performed del=0.5. Averaged over all the data-sets, there is a small margin for del=0.5: 0.48%.

For FT, again del=0.5 often out-performed del=1, and significantly so (*6,**4), with the margins being a little larger than the case for T. For two of the out-of-domain data-sets (o-Cz and o-En), the relationship reverses, with del=1 statistically significantly out-performing the del=0.5 setting. Averaging there is a margin in favour of del=0.5 of 0.78%, with largest margin shown by Spanish (2.21%) and Catalan (2.57%). See Figure 6.

For H, the effect of switching from del=0.5 to del=1 follows a very similar pattern to that found for FT and where del=0.5 exceeds del=1, the effect is a little more pronounced than it was for FT, with 6 cases strictly statistically significant. Again for o-Cz, the relationship is reversed, and for the other two out-of-domain data-sets either del=1 is significantly better or statistically indistinguishable. Averaged across all the data-sets the margin is 0.73% in favour of del=0.5 for H.

Concerning the out-of-domain data-sets, one could speculate that the reason why the del=0.5 setting does not improve over the del=1 setting is that this makes swaps relatively more costly, and that the out-of-domain data-sets require a greater tolerance to swaps.

5 Conclusions and future work

In an approach to projecting annotations over Tai-mappings, we have explored the effects of variations of possible parameters. We have considered a number of settings concerning the costing of swaps, referring to greater and lesser amounts of

[5] The higher of the two values is used in table 3 for max(dist).

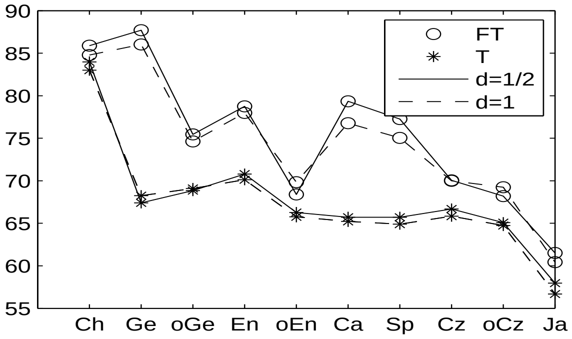

Figure 6: Comparing dist(del=1)to dist(del=0.5) (T and FT).

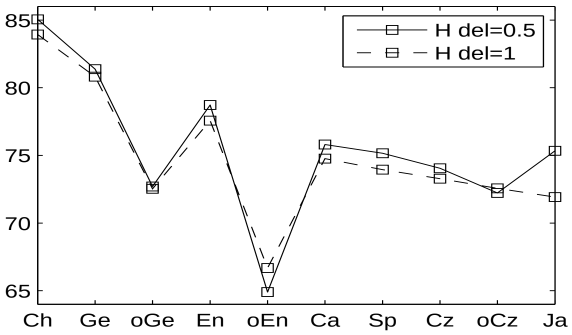

Figure 7: Comparing dist(del=1)to dist(del=0.5) (H).

the available information. With the H setting, using syntactic and lexical information, the performance across the languages ranges from 74% to 85%, and the performance under the B and T settings, which attend to no lexical information was also substantial. The linearised representation was shown to be clearly out-performed by the tree representation. Distance- and similarity-based alignment scoring were shown to give different outcomes, with distance overall out-performing similarity on the B, T and FT settings, but with this no longer the case for the H setting.

There were also some language-specific findings, amongst them that Japanese results with B/T/FT settings were noticeably poor, almost certainly due to that data-set's lack of variation in dependency labels, and that whilst mostly lowering the cost of deletion improved performance, this was not the case for the out-of-domain data-sets.

The non-negligible performance under the B and T settings, which attend to no lexical information is perhaps worth further scrutiny. If across different lexical items, PropBank aimed for no continuity of use of enumerated core arguments, intuition would suggest that in these B and T settings, performance should be very low. Concerning the A0 and A1 roles, PropBank commits to some consistency concerning possession of prototypically agent and patient properties, and for other enumerated roles, to consistency within certain verb groups. One direction for future research would be to investigate what aspects of performance are due to consistent use across lexical items, by automatically introducing inconsistency by permutations amongst the identifiers for a particular item.

There has been little comparable work using an alignment approach to SRL an exception being Fürstenau and Lapata (2011), though there are significant differences: they work with FrameNet (Fillmore et al., 2004), and use an alignment not constrained by ancestry or linear order. Also, rather than taking each unlabelled item, T, in turn, and using its nearest labelled neighbours, $\{S_1, \ldots, S_n\}$, their aim is to take each labelled exemplar, S, from a frame *lexicon*, in turn, and use it to project annotation to its nearest neighbours $\{T_1, \ldots T_n\}$ in an unlabelled corpus. For all of these reasons, we cannot at the moment make any meaningful quantitative comparisons with their work. Nonetheless, it seems reasonable to expect the contrasts we have described concerning cost settings and distance versus similarity to apply to the kind of data-set expansion scenario they discuss and investigating whether this is so is a potential avenue for further research. Conversely it would be interesting to see how our findings are effected if we were to replace our notion of alignments which must be Tai mappings, with the notion of alignment from their work.

Acknowledgements

All calculations were performed on the Lonsdale cluster maintained by the Trinity Centre for High Performance Computing. This cluster was funded through grants from Science Foundation Ireland.

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Trinity College Dublin.

References

Wanxiang Che, Zhenghua Li, Yongqiang Li, Yuhang Guo, Bing Qin, and Ting Liu. 2009. Multilingual Dependency-based Syntactic and Semantic Parsing. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 49–54, Boulder, Colorado, June. Association for Computational Linguistics.

Qifeng Dai, Enhong Chen, and Liu Shi. 2009. An Iterative Approach for Joint Dependency Parsing and Semantic Role Labeling. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 19–24, Boulder, Colorado, June. Association for Computational Linguistics.

Martin Emms and Hector-Hugo Franco-Penya. 2012. On order equivalences between distance and similarity measures on sequences and trees. In *Proceedings of ICPRAM 2012 International Conference on Pattern Recognition Application and Methods*.

C.J. Fillmore, J.Ruppenhofer, and C.F. Baker, 2004. *Frontiers in Linguistics*, chapter Framenet and representing the link between semantic and syntactic relations.

Hector-Hugo Franco-Penya. 2010. Edit Tree Distance alignments for Semantic Role Labelling. In *Proceedings of the ACL 2010 Student Research Workshop*, pages 79–84, Uppsala, Sweden. Association for Computational Linguistics.

Hagen Fürstenau and Mirella Lapata. 2011. Semi-supervised semantic role labeling via structural alignment. *Computational Linguistics*, 38(June 2011):135—-171.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Adam Meyers, Joakim Nivre, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL-2009)*.

Michal Kolar, Michael Lassig, and Johannes Berg. 2008. From protein interactions to functional annotation: graph alignment in Herpes. *BMC Systems Biology*, 2(1).

Aron Marchler-Bauer, Anna R Panchenko, Benjamin A Shoemaker, Paul A Thiessen, Lewis Y Geer, and Stephen H Bryant. 2002. Cdd: a database of conserved domain alignments with links to domain three-dimensional structure. *Nucleic Acids Res*, 30(1):281–3.

Q McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Alessandro Moschitti, Daniele Pighin, and Roberto Basili. 2008. Tree kernels for semantic role labeling. *Computational Linguistics*, 34(2):193–224.

Martha Palmer, Paul Kingsbury, and Daniel Gildea. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Kuo-Chung Tai. 1979. The tree-to-tree correction problem. *Journal of the ACM (JACM)*, 26(3):433.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal of Computing*, 18:1245–1262.