An Unsupervised Model for Instance Level Subcategorization Acquisition

Simon Baker
Computer Laboratory
University of Cambridge
sb895@cam.ac.uk

Roi Reichart
Technion, IIT
Haifa, Israel
roiri@ie.technion.ac.il

Anna Korhonen
Computer Laboratory
University of Cambridge
alk23@cam.ac.uk

Abstract

Most existing systems for subcategorization frame (SCF) acquisition rely on supervised parsing and infer SCF distributions at type, rather than instance level. These systems suffer from poor portability across domains and their benefit for NLP tasks that involve sentence-level processing is limited. We propose a new unsupervised, Markov Random Field-based model for SCF acquisition which is designed to address these problems. The system relies on supervised POS tagging rather than parsing, and is capable of learning SCFs at instance level. We perform evaluation against gold standard data which shows that our system outperforms several supervised and type-level SCF baselines. We also conduct task-based evaluation in the context of verb similarity prediction, demonstrating that a vector space model based on our SCFs substantially outperforms a lexical model and a model based on a supervised parser [1].

1 Introduction

Subcategorization frame (SCF) acquisition involves identifying the arguments of a predicate and generalizing about its syntactic frames, where each frame specifies the syntactic type and number of arguments permitted by the predicate. For example, in sentences (1)-(3) the verb *distinguish* takes three different frames, the difference between which is not evident when considering the phrase structure categorization:

(1) **Direct Transitive**: [They]NP [distinguished]VP [the mast]NP [of [ships on the horizon]NP]PP .

(2) **Indirect Transitive:** [They]NP [distinguished]VP [between [me and you]ADVP]PP .

(3) **Ditransitive**: [They]NP [distinguished]VP [him]NP [from [the other boys]NP]PP.

As SCFs describe the syntactic realization of the verbal predicate-argument structure, they are highly valuable for a variety of NLP tasks. For example, verb subcategorization information has proven useful for tasks such as parsing (Carroll and Fang, 2004; Arun and Keller, 2005; Cholakov and van Noord, 2010), semantic role labeling (Bharati et al., 2005; Moschitti and Basili, 2005), verb clustering, (Schulte im Walde, 2006; Sun and Korhonen, 2011) and machine translation (hye Han et al., 2000; Hajič et al., 2002; Weller et al., 2013).

SCF induction is challenging. The argument-adjunct distinction is difficult even for humans, and is further complicated by the fact that both arguments and adjuncts can appear frequently in potential argument head positions (Korhonen et al., 2000). SCFs are also highly sensitive to domain variation so that both the frames themselves and their probabilities vary depending on the meaning and behavior of predicates in the domain in question (e.g. (Roland and Jurafsky, 1998; Lippincott et al., 2010; Rimell et al., 2013), Section 4).

Because of the strong impact of domain variation, SCF information is best acquired automatically. Existing data-driven SCF induction systems, however, do not port well between domains. Most existing systems rely on hand-written rules (Briscoe and Carroll, 1997; Korhonen, 2002; Preiss et al., 2007) or simple co-occurrence statistics (O'Donovan et al., 2005; Chesley and Salmon-Alt, 2006; Ienco et al., 2008; Messiant et al., 2008; Lenci et al., 2008; Altamirano and Alonso i Alemany, 2010; Kawahara and Kurohashi, 2010) applied to the grammatical dependency output of supervised statistical parsers. Even the handful of recent systems

[1] The verb similarity dataset used for the evaluation of our model is publicly available at ie.technion.ac.il/∼roiri/.

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 278–289,
October 25-29, 2014, Doha, Qatar.

that use modern machine learning techniques (Debowski, 2009; Lippincott et al., 2012; Van de Cruys et al., 2012; Reichart and Korhonen, 2013) use supervised parsers to pre-process the data[2].

Supervised parsers are notoriously sensitive to domain variation (Lease and Charniak, 2005). As annotation of data for each new domain is unrealistic, current SCF systems suffer from poor portability. This problem is compounded for the many systems that employ manually developed SCF rules because rules are inherently ignorant to domain-specific preferences. The few SCF studies that focused on specific domains (e.g. biomedicine) have reported poor performance due to these reasons (Rimell et al., 2013).

Another limitation of most current SCF systems is that they produce a *type*-level SCF lexicon (i.e. a lexicon which lists, for a given predicate, different SCF types with their relative frequencies). Such a lexicon provides a useful high-level profile of the syntactic behavior of the predicate in question, but is less useful for downstream NLP tasks (e.g. information extraction, parsing, machine translation) that involve sentence processing and can therefore benefit from SCF information at instance level. Sentences (1)-(3) demonstrate this limitation - a prior distribution over the possible syntactic frames of *distinguish* provides only a weak signal to a sentence level NLP application that needs to infer the verbal argument structure of its input sentences.

We propose a new unsupervised model for SCF induction which addresses these problems with existing systems. Our model does not use a parser or hand-written rules, only a part-of-speech (POS) tagger is utilizes in order to produce features for machine learning. While POS taggers are also sensitive to domain variation, they can be adapted to domains more easily than parsers because they require much smaller amounts of annotated data (Lease and Charniak, 2005; Ringger et al., 2007). However, as we demonstrate in our experiments, domain adaptation of POS tagging may not even be necessary to obtain good results on the SCF acquisition task.

Our model, based on the Markov Random Field (MRF) framework, performs instance-based SCF learning. It encodes syntactic similarities among verb instances across different verb types (derived from a lexical and POS-based feature representation of verb instances) as well as prior beliefs on the tendencies of specific instances of the same verb type to take the same SCF.

We evaluate our model against corpora annotated with verb instance SCFs (Quochi et al., 2012). In addition, following the Levin verb clustering tradition (Levin, 1993) which ties verb meanings with their syntactic properties, we evaluate the semantic predictive power of our clusters. In the former evaluation, our model outperforms a number of strong baselines, including supervised and type-level ones, achieving an accuracy of up to 69.2%. In the latter evaluation a vector space model that utilized our induced SCFs substantially outperforms the output of a type-level SCF system that uses the fully trained Stanford parser.

2 Previous Work

Several SCF acquisition systems are available for English (O'Donovan et al., 2005; Preiss et al., 2007; Lippincott et al., 2012; Van de Cruys et al., 2012; Reichart and Korhonen, 2013) and other languages, including French (Messiant, 2008), Italian (Lenci et al., 2008), Turkish (Uzun et al., 2008), Japanese (Kawahara and Kurohashi, 2010) and Chinese (Han et al., 2008). The prominent input to these systems are grammatical relations (GRs) which express binary dependencies between words (e.g. direct and indirect objects, various types of complements and conjunctions). These are generated by some parsers (e.g. (Briscoe et al., 2006)) and can be extracted from the output of others (De-Marneffe et al., 2006).

Two representative systems for English are the Cambridge system (Preiss et al., 2007) and the BioLexicon system which was used to acquire a substantial lexicon for biomedicine (Venturi et al., 2009). These systems extract GRs at the verb instance level from the output of a parser: the RASP general-language unlexicalized parser[3] (Briscoe et al., 2006) and the lexicalized Enju parser tuned to the biomedical domain (Miyao and Tsujii, 2005), respectively. They generate potential SCFs by mapping GRs to a predefined SCF inventory using a set of manually developed rules (the Cambridge system) or by simply considering the sets of GRs including verbs in question as potential SCFs (BioLexicon). Finally, a type level lexicon

[2](Lippincott et al., 2012) does not use a parser, but the syntactic frames induced by the system do not capture *sets of* arguments for verbs, so are not SCFs in a traditional sense.

[3]A so-called unlexicalized parser is a parser trained without explicit SCF annotations.

is built through noisy frame filtering (based on frequencies or on external resources and annotations), which aims to remove errors from parsing and argument-adjunct distinction. Clearly, these systems require extensive manual work: a-priori definition of an SCF inventory and rules, manually annotated sentences for training a supervised parser, SCF annotations for parser lexicalization, and manually developed resources for optimal filtering.

A number of recent works have applied modern machine learning techniques to SCF induction, including point-wise co-occurrence of arguments (Debowski, 2009), a Bayesian network model (Lippincott et al., 2012), multi-way tensor factorization (Van de Cruys et al., 2012) and Determinantal Point Processes (DPPs) -based clustering (Reichart and Korhonen, 2013). However, all of these systems induce type-level SCF lexicons and, except from the system of (Lippincott et al., 2012) that is not capable of learning traditional SCFs, they all rely on supervised parsers.

Our new system differs from previous ones in a number of respects. First, in contrast to most previous systems, our system provides SCF analysis for each verb instance in its sentential context, yielding more precise SCF information for systems benefiting from instance-based analysis. Secondly, it addresses SCF induction as an unsupervised clustering problem, avoiding the use of supervised parsing or any of the sources of manual supervision used in previous works. Our system relies on POS tags - however, we show that it is not necessary to train a tagger with in-domain data to obtain good performance on this task, and therefore our approach provides a more domain-independent solution to SCF acquisition.

We employ POS-tagging instead of unsupervised parsing for two main reasons. First, while a major progress has been made on unsupervised parsing (e.g. (Cohen and Smith, 2009; Berg-Kirkpatrick et al., 2010)), the performance is still considerably behind that of supervised parsing. For example, the state-of-the-art discriminative model of (Berg-Kirkpatrick et al., 2010) achieves only 63% directed arc accuracy for WSJ sentences of up to 10 words, compared to more than 95% obtained with supervised parsers. Second, current unsupervised parsers produce unlabeled structures which are substantially less useful for SCF acquisition than labeled structures produced by supervised parsers (e.g. grammatical relations).

Finally, a number of recent works addressed related tasks such as argument role clustering for SRL (Lang and Lapata, 2011a; Lang and Lapata, 2011b; Titvo and Klementiev, 2012) in an unsupervised manner. While these works differ from ours in the task (clustering arguments rather than verbs) and the level of supervision (applying a supervised parser), like us they analyze the verb argument structure at the instance level.

3 Model

We address SCF induction as an unsupervised verb instance clustering problem. Given a set of plain sentences, our algorithm aims to cluster the *verb instances* in its input into syntactic clusters that strongly correlate with SCFs. In this section we introduce a Markov Random Field (MRF) model for this task: Section 3.1 describes our model's structure, components and objective; Section 3.2 describes the model potentials and the knowledge they encode; and Section 3.3 describes how clusters are induced from the model.

3.1 Model Structure

We implement our model in the MRF framework (Koller and Friedman, 2009). This enables us to encode the two main sources of information that govern SCF selection in verb instances: (1) At the sentential context, the verbal syntactic frame is encoded through syntactic features. Verb instances with similar feature representations should therefore take the same syntactic frame; and (2) At the global context, per verb type SCF distributions tend to be Zipfian (Korhonen et al., 2000). Instances of the same verb type should therefore be biased to take the same syntactic frame.

Given a collection of plain input sentences, we denote the number of verb instances in the collection with n, and the number of data-dependent equivalence classes (ECs) with K (see below for their definition), and define an undirected graphical model (MRF), $G = (V, E, L)$. We define the vertex set as $V = X \cup C$, with $X = \{x_1, \ldots, x_n\}$ consisting of one vertex for every verb instance in the input collection, and $C = \{c_1 \ldots c_K\}$ consisting of one vertex for each data-dependent EC. The set of labels used by the model, L, corresponds to the syntactic frames taken by the verbs in the input data. The edge set E is defined through the model's potentials that are described below.

We encode information in the model through three main sets of potentials: one set of *singleton potentials* - defined over individual model vertexes, and two sets of *pairwise potentials* - defined between pairs of vertexes. The first set consists of a singleton potential for each vertex in the model. Reflecting the Zipfian distribution of SCFs across the instances of the same verb type, these potentials encourage the model to assign such verb instances to the same frame (cluster). The information encoded in these potentials is induced via a pre-processing clustering step. The second set consists of a pairwise potential for each pair of vertexes $x_i, x_j \in X$ - that is, for each verb instance pair in the input, across verb types. These potentials encode the belief, computed as feature-based similarity (see below), that their verb instance arguments implement the same SCF.

Finally, potentials from the last set bias the model to assign the same SCF to high cardinality sets of cross-type verb instances based on their syntactic context. While these are pairwise potentials defined between verb instance vertexes (X) and EC vertexes (C), they are designed so that they bias the assignment of all verb instance vertexes that are connected to the same EC vertex towards the same frame assignment ($l \in L$). The two types of pairwise potentials complement each other by modeling syntactic similarities among verb instance pairs, as well as among higher cardinality verb instance sets.

The resulted maximum aposteriori problem (MAP) takes the following form:

$$MAP(V) = \arg\max_{x,c \in V} \sum_{i=1}^{n} \theta_i(x_i) + \sum_{i=1}^{n}\sum_{j=1}^{n} \theta_{i,j}(x_i, x_j) +$$

$$\sum_{i=1}^{n}\sum_{j=1}^{K} \phi_{i,j}(x_i, c_j) \cdot I(x_i \in EC_j) + \sum_{i=1}^{K}\sum_{j=1}^{K} \xi_{i,j}(c_i, c_j)$$

where the predicate $I(x_i \in EC_j)$ returns 1 if the i-th verb instance belongs the j-th equivalence class and 0 otherwise. The ξ pairwise potentials defined between EC vertexes are very simple potentials designed to promise different assignments for each pair of EC vertexes. They do so by assigning a $-\infty$ score to assignments where their argument vertexes take the same frame and a 0 otherwise. In the rest of this section we do not get back to this simple set of potentials.

A graphical illustration of the model is given in Figure 1. Note that we could have selected a richer model structure, for example, by defining a similarity potential over all verb instance vertexes that share an equivalence class. However, as the figure demonstrates, even the structure of the pruned version of our model (see Section 3.3) usually contains cycles, which makes inference NP-hard (Shimony, 1994). Our design choices aim to balance between the expressivity of the model and the complexity of inference. In Section 3.3 we describe the LP relaxation algorithm we use for inference.

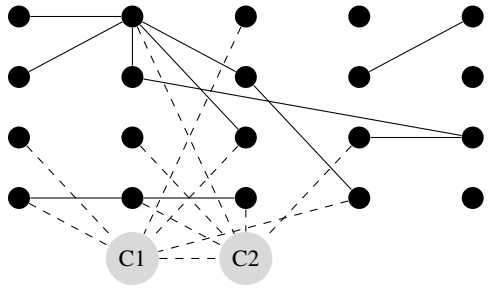

Figure 1: A graphical illustration of our model (after pruning, see Sec. 3.3) for twenty verb instances ($|X| = 20$), each represented with a black vertex, and two equivalence classes (ECs), each represented with a gray vertex ($|C| = 2$). Solid lines represent edges (and $\theta_{i,j}$ pairwise potentials) between verb instance vertexes. Dashed lines represent edges between verb instance vertexes and EC vertexes ($\phi_{i,j}$ pairwise potentials) or between EC vertexes ($\xi_{i,j}$ pairwise potentials) .

3.2 Potentials and Encoded Knowledge

Pairwise Syntactic Similarity Potentials. The pairwise syntactic similarity potentials are defined for each pair of verb instance vertexes, $x_i, x_j \in X$. They are designed to encourage the model to assign verb instances with similar fine-grained feature representations to the same frame ($l \in L$) and verb instances with dissimilar representations to different frames. For this aim, for every verb pair i, j with feature representation vectors v_i, v_j and verb instance vertexes $x_i, x_j \in X$, we define the following potential function:

$$\theta_{i,j}(x_i = l_1, x_j = l_2) = \left\{ \begin{array}{ll} \lambda(v_i, v_j) & \text{if } l_1 = l_2 \\ 0 & \text{otherwise} \end{array} \right\}$$

Where $l_1, l_2 \in L$ are label pairs and λ is a verb instance similarity function. Below we describe the feature representation and the λ function.

The verb instance feature representation is defined through the following process. For each

word instance in the input sentences we first build a basic feature representation (see below). Then, for each verb instance we construct a final feature representation defined to be the concatenation of that verb's basic feature representation with the basic representations of the words in a size 2 window around the represented verb. The final feature representation for the i-th verb instance in our dataset is therefore defined to be $v_i = [w_{-2}, w_{-1}, vb_i, w_{+1}, w_{+2}]$, where w_{-k} and w_{+k} are the basic feature representations of the words in distance $-k$ or $+k$ from the i-th verb instance in its sentence, and vb_i is the basic feature representation of that verb instance.

Our basic feature representation is inspired from the feature representation of the MST parser (McDonald et al., 2005) except that in the parser the features represent a directed edge in the complete directed graph defined over the words in a sentence that is to be parsed, while our features are generated for word n-grams. Particularly, our feature set is a concatenation of two sets derived from the MST set described in Table 1 of (McDonald et al., 2005) in the following way: (1) In both sets the parent word in the parser's set is replaced with the represented word; (2) In one set every child word in the parser's set is replaced by the word to the left of the represented word and in the other set it is replaced by the word to its right. This choice of features allows us to take advantage of a provably useful syntactic feature representation without the application of any parse tree annotation or parser.

We compute the similarity between the syntactic environments of two verb instances, i, j, using the following equation:

$$\lambda(v_i, v_j) = W \cdot \cos(v_i, v_j) - S$$

Where W is a hyperparameter designed to bias verb instances of the same verb type towards the same frame. Practically, W was tuned to be 3 for instances of the same type, and 1 otherwise [4].

While the cosine function is the standard measure of similarity between two vectors, its values are in the $[0, 1]$ range. In the MRF modeling framework, however, we must encode a negative pairwise potential value between two vertexes in order to encourage the model to assign different labels (frames) to them. We therefore added the positive hyperparameter S which was tuned, without access to gold standard manual annotations, so that there is an even number of negative and positive pairwise syntactic similarity potentials after the model is pruned (see Section 3.3) [5].

Type Level Singleton Potentials. The goal of these potentials is to bias verb instances of the same type to be assigned to the same syntactic frame while still keeping the instance based nature of our algorithm. For this aim, we applied Algorithm 1 for pre-clustering of the verb instances and encoded the induced clusters into the local potentials of the corresponding $x \in X$ vertexes. For every $x \in X$ the singleton potential is therefore defined to be:

$$\theta_i(x_i = l) = \left\{ \begin{array}{ll} F \cdot \max \lambda & \text{if } l \text{ is induced by Algorithm 1} \\ 0 & \text{otherwise} \end{array} \right\}$$

where $\max \lambda$ is the maximum λ score across all verb instance pairs in the model and $F = 0.2$ is a hyperparamter.

Algorithm 1 has two hyperparameters: T and M, the first is a similarity cut-off value used to determine the initial set of clusters, while the second is used to determine whether two clusters are similar enough to be merged. We tuned these hyperparameters, without manually annotated data, so that the number of clusters induced by this algorithm will be equal to the number of gold standard SCFs. T was tuned so that the first part of the algorithm generates an excessive number of clusters, and M was then tuned so that these clusters are merged to the desired number of clusters.

The λ function, used to measure the similarity between two verbs, is designed to bias the instances of the same verb type to have a higher similarity score. Algorithm 1 therefore tends to assign such instances to the same cluster. In our experiments that was always the case for this algorithm.

High Cardinality Verb Sets Potentials. This set of potentials aims to bias larger sets of verb instances to share the same SCF. It is inspired by (Rush et al., 2012) who demonstrated, that syntactic structures that appear at the same syntactic context, in terms of the surrounding POS tags, tend to manifest similar syntactic behavior. While they demonstrated the usefulness of their method for dependency parsing and POS tagging, we implement it for higher level SCFs.

We identified syntactic contexts that imply similar SCFs for verb instances appearing inside them.

[4] All hyperparameters that require gold-standard annotation for tuning, were tuned using held-out data (Section 4).

[5] The values in practice are $S = 0.43$ for labour legislation and $S = 0.38$ for environment.

Algorithm 1 Verb instance pre-clustering algorithm. $\hat{\lambda}$ is the average λ score between the members of its cluster arguments. T and M are hyperparametes tuned without access to gold standard data.

Require: $K = \emptyset$
  for all $x \in X$ **do**
    for all $k \in K$ **do**
      for all $u \in k$ **do**
        if $\lambda(v_x, v_u) > T$ **then**
          $k = k \cup \{x\}$
          Go to next x
        end if
      end for
    end for
    $k1 = \{x\}$
    $K = K \cup k1$
  end for
  for all $k_1, k_2 \in K$: $k_1 \neq k_2$ **do**
    if $\hat{\lambda}(k_1, k_2) > M$ **then**
      Merge (k_1, k_2)
    end if
  end for

Contexts are characterized by the coarse POS tag to the left and to the right of the verb instance. While the number of context sets is bounded only by the number of frames our model is designed to induce, in practice we found that defining two equivalence sets led to the best performance gain, and the sets we used are presented in Table 1.

In order to encode this information into our MRF, each set of syntactic contexts is associated with an equivalence class (EC) vertex $c \in C$ and the verb instance vertexes of all verbs that appear in a context from that set are connected with an edge to c. The pairwise potential between a vertex $x \in X$ and its equivalence class is defined to be:

$$\phi_{i,j}(x_i = l_1, c_j = l_2) = \left\{ \begin{array}{ll} U & \text{if } l_1 = l_2 \\ 0 & \text{otherwise} \end{array} \right\}$$

$U = 10$ is a hyperparameter that strongly biases x vertexes to get the same SCF as their EC vertex.

3.3 Verb Cluster Induction

In this section we describe how we induce verb instance clusters from our model. This process is based on the following three steps: (1) Graph pruning; (2) Induction of an Ensemble of approximate MAP inference solutions in the resulted graphical model; and, (3) Induction of a final clustering solution based on the ensemble created at step 2. Below we explain the necessity of each of these steps and provide the algorithmic details.

EC-1		EC-2	
Left	Right	Left	Right
,	D	V	T
N	D	R	T
V	.	N	D
R	D	R	N

Table 1: POS contexts indicative for the syntactic frame of the verb instance they surround. D: *determiner*, N: *noun*, V: *verb*, T: the preposition *'to'* (which has its own POS tag in the WSJ POS tag set which we use), R: *adverb*. EC-1 and EC-2 stand for the first and second equivalence class respectively. In addition, the following contexts where associated with both ECs: $(T, D), (T, N), (N, N)$ and (V, I) where I stands for a preposition.

Graph Pruning. The edge set of our model consists of an edge for every pair of verb instance vertexes and of the edges that connect verb instance vertexes and equivalence class vertexes. This results in a large tree-width graph which substantially complicates MRF inference. To alleviate this we prune all edges with a positive score lower than p_+ and all edges with a negative score higher than p_-, where p_+ and p_- are manually tuned hyperparametes [6].

MAP Inference. For most reasonable values of p_+ and p_- our graph still contains cycles even after it is pruned, which makes inference NP-hard (Shimony, 1994). Yet, thanks to our choice of an edge-factorized model, there are various approximate inference algorithms suitable for our case.

We applied the message passing algorithm for linear-programming (LP) relaxation of the MAP assignment (MPLP, (Sontag et al., 2008)). LP relaxation algorithms for the MAP problem define an upper bound on the original objective which takes the form of a linear program. Consequently, a minimum of this upper bound can be found using standard LP solvers or, more efficiently, using specialized message passing algorithms (Yanover et al., 2006). The MPLP algorithm described in (Sontag et al., 2008) is appealing in that it iteratively computes tighter upper bounds on the MAP objective (for details see their paper).

Cluster Ensemble Generation and a Final Solution. As our MAP objective is non-convex,

[6] The values used in practice are $p_+ = 0.28, p_- = -0.17$ for the labour legislation dataset, and $p_+ = 0.25$, $p_- = -0.20$ for the environment set.

the convergent point of an optimization algorithm applied to it is highly sensitive to its initialization. To avoid convergence to arbitrary local maxima which may be of poor quality, we turn to a perturbation protocol where we repeatedly introduce random noise to the MRF's potential functions and then compute the approximate MAP solution of the resulted model using the MPLP algorithm. Noising was done by adding an ϵ term to the $lambda$ values described in section 3.2 [7]. This protocol results in a set of cluster (label) assignments for the involved verb instances, which we treat as an ensemble of experts from which a final, high quality, solution is to be induced.

The basic idea in ensemble learning is that if several experts independently cluster together two verb instances, our belief that these verbs belong in the same cluster should increase. (Reichart et al., 2012) implemented this idea through the k-way normalized cut clustering algorithm (Yu and Shi, 2003). Its input is an undirected graph $\hat{G} = (\hat{V}, \hat{E}, \hat{W})$ where $\hat{V}$ is the set of vertexes, $\hat{E}$ is the set of edges and $\hat{W}$ is a non-negative and symmetric edge weight matrix. To apply this model to our task, we construct the input graph $\hat{G}$ from the labelings (frame assignments) contained in the ensemble. The graph vertexes $\hat{V}$ correspond to the verb instances and the (i, j)-th entry of the matrix $\hat{W}$ is the number of ensemble members that assign the same label to the i-th and j-th verb instances.

For $A, B \subseteq \hat{V}$ define:

$$links(A, B) = \sum_{i \in A, j \in B} \hat{W}(i, j)$$

Using this definition, the normalized link ratio of A and B is defined to be:

$$NormLinkRatio(A, B) = \frac{links(A, B)}{links(A, \hat{V})}$$

The k-way normalized cut problem is to minimize the links that leave a cluster relative to the total weight of the cluster. Denote the set of clusterings of $\hat{V}$ that consist of k clusters by $\hat{C} = \{\hat{c_1}, \ldots \hat{c_t}\}$ and the j-th cluster of the i-th clustering by $\hat{c_{ij}}$. Then

$$c^* = \underset{\hat{c_i} \in \hat{C}}{\operatorname{argmin}} \sum_{j=1}^{k} NormLinkRatio(\hat{c_{ij}}, \hat{V} - \hat{c_{ij}})$$

The algorithm of (Yu and Shi, 2003) solves this problem very efficiently as it avoids the heavy eigenvalues and eigenvectors computations required by traditional approaches.

4 Experiments and Results

Our model is unique compared to existing systems in two respects. First, it does not utilize supervision in the form of either a supervised syntactic parser and/or manually crafted SCF rules. Consequently, it induces unnamed frames (clusters) that are not directly comparable to the named frames induced by previous systems. Second, it induces syntactic frames at the verb instance, rather than type, level. Evaluation, and especially comparison to previous work, is therefore challenging.

We therefore evaluate our system in two ways. First, we compare its output, as well as the output of a number of clustering baselines, to the gold standard annotation of corpora from two different domains (the only publicly available ones with instance level SCF annotation, to the best of our knowledge). Second, in order to compare the output of our system to a rule-based SCF system that utilizes a supervised syntactic parser, we turn to a task-based evaluation. We aim to predict the degree of similarity between verb pairs and, following (Pado and Lapata, 2007) , we do so using a *syntactic-based* vector space model (VSM). We construct three VSMs - (a) one that derives features from our clusters; (b) one whose features come from the output of a state-of-the-art verb type level, rule based, SCF system (Reichart and Korhonen, 2013) that uses a modern parser (Klein and Manning, 2003); and (c) a standard lexical VSM. Below we show that our system compares favorably in both evaluations.

Data. We experimented with two datasets taken from different domains: labor legislation and environment (Quochi et al., 2012). These datasets were created through web crawling followed by domain filtering. Each sentence in both datasets may contain multiple verbs but only one target verb has been manually annotated with a SCF. The labour legislation domain dataset contains 4415 annotated verb instances (and hence also

[7] ϵ was accepted by first sampling a number in the $[0, 1]$ range using the Java psuodorandom generator and then scaling it to 1% of $\cos(v_i, v_j)$. This value was tuned, without access to gold standard manual annotations, so that there is an even number of negative and positive pairwise syntactic similarity potentials after the model is pruned (Section 3.3).

sentences) of 117 types, and the environmental domain dataset contains 4503 annotated verb instances of 116 types. In both datasets no verb type accounts for more than 4% of the instances and only up to 35 verb types account for 1% of the instances or more. The lexical difference between the corpora is substantial: they share only 42 annotated verb types in total, of which only 2 verb types (responsible for 4.1% and 5.2% of the instances in the environment and labor legislation domains respectively) belong to the 20 most frequent types (responsible for 37.9% and 46.85% of the verb instances in the respective domains) of each corpus.

The 29 members of the SCF inventory are detailed in (Quochi et al., 2012). Table 2, presenting the distribution of the 5 highest frequency frames in each corpus, demonstrates that, in addition to the significant lexical difference, the corpora differ to some extent in their syntactic properties. This is reflected by the substantially different frequencies of the "dobj:iobj-prep:su" and "dobj:su" frames.

As a pre-processing step we first POS tagged the datasets with the Stanford tagger (Toutanova et al., 2003) trained on the standard POS training sections of the WSJ PennTreebank corpus.

4.1 Evaluation Against SCF Gold Standard

Experimental Protocol The computational complexity of our algorithm does not allow us to run it on thousands of verb instances in a feasible time. We therefore repeatedly sampled 5% of the sentences from each dataset, ran our algorithm as well as the baselines (see below) and report the average performance of each method. The number of repetitions was 40 and samples were drawn from a uniform distribution while still promising that the distribution of gold standard SCFs in each sample is identical to their distribution in the entire dataset. Before running this protocol, 5% of each corpus was kept as held-out data on which hyperparameter tuning was performed.

Evaluation Measures and Baselines. We compare our system's output to instance-level gold standard annotation. We use standard measures for clustering evaluation, one measure from each of the two leading measure types: the V measure (Rosenberg and Hirschberg, 2007), which is an information theoretic measure, and greedy many-to-one accuracy, which is a mapping-based measure. For the latter, each induced cluster is first mapped to the gold SCF frame that annotates the highest number of verb instances this induced cluster also annotates and then a standard instance-level accuracy score is computed (see, e.g., (Reichart and Rappoport, 2009)). Both measures scale from 100 (perfect match with gold standard) to 0 (no match).

As mentioned above, comparing the performance of our system with respect to a gold standard to the performance of previous type-level systems that used hand-crafted rules and/or supervised syntactic parsers would be challenging. We therefore compare our model to the following baselines: (a) The *most frequent class (MFC)* baseline which assigns all verb instances with the SCF that is the most frequent one in the *gold standard annotation* of the data; (b) The *Random* baseline which simply assigns every verb instance with a randomly selected SCF; (c) Algorithm 1 of section 3.2 which generates unsupervised verb instance clustering such that verb instances of the same type are assigned to the same cluster; and (d) Finally, we also compare our model against versions where everything is kept fixed, except a subset of potentials which is omitted. This enables us to study the intricacies of our model and the relative importance of its components. For all models, the number of induced clusters is equal to the number of SCFs in the gold standard.

Results Table 3 presents the results, demonstrating that our full model substantially outperforms all baselines. For the first two simple heuristic baselines (*MFC* and *Random*) the margin is higher than 20% for both the greedy M-1 mapping measure and the V measure. Note tat the V score of the MFC baseline is 0 by definition, as it assigns all items to the same cluster. The poor performance of these simple baselines is an indication of the difficulty of our task.

Recall that the type level clustering induced by Algorithm 1 is the main source of type level information our model utilizes (through its singleton potentials). The comparison to the output of this algorithm (the *Type Pre-clustering* baseline) therefore shows the quality of the instance level refinement our model provides. As seen in table 3, our model outperforms this baseline by 6.9% for the M-1 measure and 5.2% for the V measure.

In order to compare our model to its components we exclude either the EC potentials (ϕ and ξ) only (Model - EC), or the EC and the singleton potentials (θ_i, Model - EC - Type pre-clustering). The results show that our model gains much more

Environment		Labour Legislation	
SCF	Frequency	SCF	Frequency
dobj:su	46%	dobj:su	39%
su	9%	dobj:iobj-prep:su	15%
iobj-prep:su	8%	su	10%
dobj:iobj-prep:su	6%	su:xcompto-vbare	8%
su:xcompto-vbare	6%	iobj-prep:su	7%

Table 2: Top 5 most frequent SCFs for the Environment and Labour Legislation datasets used in our experiments.

	Environment		Labour Legislation	
	M-1	V	M-1	V
Full Model	**66.4**	**57.3**	**69.2**	**55.6**
Baselines				
MFC	46.2	0	39.4	0
Random	34.6	28.1	36.5	27.8
Type Pre-clustering	60.1	52.1	62.3	51.4
Model Components				
Model - EC	64.9	56.2	67.4	54.6
Model - EC - Type pre-clustering	48.3	48.9	45.7	44.7

Table 3: Results for our full model, the baselines (*Type Pre-clustering*: the pre-clustering algorithm (Algorithm 1 of section 3.2), *MFC*: the most frequent class (SCF) in the gold standard annotation and *Random*: random SCF assignment) and the model components. The full model outperforms all other models across measures and datasets.

from the type level information encoded through the singleton potentials than from the EC potentials. Yet, EC potentials do lead to an improvement of up to 1.5% in M-1 and up to 1.1% in V and are therefore responsible for up to 26.1% and 21.2% of the improvement over the type pre-clustering baseline in terms of M-1 and V, respectively.

4.2 Task Based Evaluation

We next evaluate our model in the context of vector space modeling for verb similarity prediction (Turney and Pantel, 2010). Since most previous word similarity works used noun datasets, we constructed a new verb pair dataset, following the protocol used in the collection of the wordSimilarity-353 dataset (Finkelstein et al., 2002).

Our dataset consists of 143 verb pairs, constructed from 122 unique verb lemma types. The participating verbs appear ≥ 10 times in the concatenation of the labour legislation and the environment datasets. Only pairs of verbs that were considered at least remotely similar by human judges (independent of those that provided the similarity scores) were included. A similarity score between 1 and 10 was assigned to each pair by 10 native English speaking annotators and were then averaged in order to get a unique pair score.

Our first baseline is a standard VSM based on lexical collocations. In this model features correspond to the number of collocations inside a size 2 window of the represented verb with each of the 5000 most frequent nouns in the Google n-gram corpus (Goldberg and Orwant, 2013). Since our corpora are limited in size, we use the collocation counts from the Google corpus.

We used our model to generate a vector representation of each verb in the following way. We run the model 5000 times, each time over a set of verbs consisting of one instance of each of the 122 verb types participating in the verb similarity set. The output of each such run is transformed to a binary vector for each participating verb, where all coordinates are assigned the value of 0, except from the one that corresponds to the cluster to which the verb was assigned which has the value of 1. The final vector representation is a concatenation of the 5000 binary vectors. Note that for this task we did not use the graph cut algorithm to generate a final clustering from the multiple MRF

runs. Instead we concatenated the output of all these runs into one feature representation that facilitates similarity prediction. For our model we estimated the verb pair similarity using the Tanimato similarity score for binary vectors:

$$T(X,Y) = \frac{\sum_i X_i \wedge Y_i}{\sum_i x_i \vee Y_i}$$

For the baseline model, where the features are collocation counts, we used the standard cosine similarity.

Our second baseline is identical to our model, except that: (a) the data is parsed with the Stanford parser (version 3.3.0, (Klein and Manning, 2003)) which was trained with sections 2-21 of the WSJ corpus; (b) the phrase structure output of the parser is transformed to the CoNLL dependency format using the official CoNLL 2007 conversion script (Johansson and Nugues, 2007); and then (c) the SCF of each verb instance is inferred using the rule-based system used by (Reichart and Korhonen, 2013). The vector space representation for each verb is then created using the process we described for our model and the same holds for vector comparison. This baseline allows direct comparison of frames induced by our SCF model with those derived from a supervised parser's output.

We computed the Pearson correlation between the scores of each of the models and the human scores. The results demonstrate the superiority of our model in predicting verb similarity: the correlation of our model with the human scores is 0.642 while the correlation of the lexical collocation baseline is 0.522 and that of the supervised parser baseline is only 0.266. The results indicate that in addition to their good alignment with SCFs, our clusters are also highly useful for verb meaning representation. This is in line with the verb clustering theory of the Levin tradition (Levin, 1993) which ties verb meaning with their syntactic properties. We consider this an intriguing direction of future work.

5 Conclusions

We presented an MRF-based unsupervised model for SCF acquisition which produces verb instance level SCFs as output. As opposed to previous systems for the task, our model uses only a POS tagger, avoiding the need for a statistical parser or manually crafted rules. The model is particularly valuable for NLP tasks benefiting from SCFs that are applied across text domains, and for the many tasks that involve sentence-level processing.

Our results show that the accuracy of the model is promising, both when compared against gold standard annotations and when evaluated in the context of a task. In the future we intend to improve our model by encoding additional information in it. We will also adapt it to a multilingual setup, aiming to model a wide range of languages.

Acknowledgments

The first author is supported by the Commonwealth Scholarship Commission (CSC) and the Cambridge Trust.

References

Ivana Romina Altamirano and Laura Alonso i Alemany. 2010. IRASubcat, a highly customizable, language independent tool for the acquisition of verbal subcategorization information from corpus. In *Proceedings of the NAACL 2010 Workshop on Computational Approaches to Languages of the Americas*.

Abhishek Arun and Frank Keller. 2005. Lexicalization in crosslinguistic probabilistic parsing: The case of french. In *Proceedings of ACL-05*.

Taylor Berg-Kirkpatrick, Alexander Bouchard-Cote, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Proceedings of NAACL-HLT-10*.

Akshar Bharati, Sriram Venkatapathy, and Prashanth Reddy. 2005. Inferring semantic roles using subcategorization frames and maximum entropy model. In *Proceedings of CoNLL-05*.

Ted Briscoe and John Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of ANLP-97*.

Ted Briscoe, John Carroll, and Rebecca Watson. 2006. The second release of the rasp system. In *Proceedings of ACL-COLING-06*.

John Carroll and Alex Fang. 2004. The automatic acquisition of verb subcategorisations and their impact on the performance of an HPSG parser. In *Proceedings of IJCNLP-04*.

Paula Chesley and Susanne Salmon-Alt. 2006. Automatic extraction of subcategorization frames for french. In *Proceedings of LREC-06*.

Kostadin Cholakov and Gertjan van Noord. 2010. Using unknown word techniques to learn known words. In *Proceedings of EMNLP-10*.

Shay Cohen and Noah Smith. 2009. Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *Proceedings of NAACL-HLT-09*.

Marie-Catherine De-Marneffe, Bill Maccartney, and Christopher Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC-06*.

Lukasz Debowski. 2009. Valence extraction using EM selection and co-occurrence matrices. *Proceedins of LREC-09*.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eitan Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20:116–131.

Yoav Goldberg and Jon Orwant. 2013. A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Proceedings of (*SEM)-13*. Association for Computational Linguistics.

Jan Hajič, Martin mejrek, Bonnie Dorr, Yuan Ding, Jason Eisner, Daniel Gildea, Terry Koo, Kristen Parton, Gerald Penn, Dragomir Radev, and Owen Rambow. 2002. Natural language generation in the context of machine translation. Technical report, Center for Language and Speech Processing, Johns Hopkins University, Baltimore. Summer Workshop Final Report.

Chung hye Han, Benoit Lavoie, Martha Palmer, Owen Rambow, Richard Kittredge, Tanya Korelsky, and Myunghee Kim. 2000. Handling structural divergences and recovering dropped arguments in a korean/english machine translation system. In *Proceedings of the AMTA-00*.

Dino Ienco, Serena Villata, and Cristina Bosco. 2008. Automatic extraction of subcategorization frames for italian. In *Proceedings of LREC-08*.

Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for english. In *Proceedings of NODALIDA-07*.

Daisuke Kawahara and Sadao Kurohashi. 2010. Acquiring reliable predicate-argument structures from raw corpora for case frame compilation. In *Proceedings of LREC-10*.

Dan Klein and Christopher Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of ACL-03*.

Daphne Koller and Nir Friedman. 2009. *Probabilistic graphical models: principles and techniques*. The MIT Press.

Anna Korhonen, Genevieve Gorrell, and Diana McCarthy. 2000. Statistical filtering and subcategorization frame acquisition. In *Proceedings of EMNLP-00*.

Anna Korhonen. 2002. Semantically motivated subcategorization acquisition. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*.

Joel Lang and Mirella Lapata. 2011a. Unsupervised semantic role induction via split-merge clustering. In *Proceedings of ACL-11*.

Joel Lang and Mirella Lapata. 2011b. Unsupervised semantic role induction with graph partitioning. In *Proceedings of EMNLP-11*.

Matthew Lease and Eugene Charniak. 2005. Parsing biomedical literature. In *Proceedings of IJCNLP-05*.

Alessandro Lenci, Barbara Mcgillivray, Simonetta Montemagni, and Vito Pirrelli. 2008. Unsupervised acquisition of verb subcategorization frames from shallow-parsed corpora. In *Proceedings of LREC-08*.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. Chicago, IL.

Tom Lippincott, Anna Korhonen, and Diarmuid Oseaghdha. 2010. Exploring subdomain variation in biomedical language. *BMC Bioinformatics*.

Tom Lippincott, Aanna Korhonen, and Diarmuid Oseaghdha. 2012. Learning syntactic verb frames using graphical models. In *Proceedings of ACL-12*.

Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In *Proceedings of ACL-05*.

Cedric Messiant, Anna Korhonen, and Thierry Poibeau. 2008. LexSchem: A large subcategorization lexicon for French verbs. In *Proceedings of LREC-08*.

Cedric Messiant. 2008. A subcategorization acquistion system for french verbs. In *Proceedings of ACL08-SRW*.

Yusuke Miyao and Junichi Tsujii. 2005. Probabilistic disambiguaton models for wide-coverage hpsg parsing. In *Proceedings of ACL-05*.

Alessandro Moschitti and Roberto Basili. 2005. Verb subcategorization kernels for automatic semantic labeling. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*.

Ruth O'Donovan, Michael Burke, Aoife Cahill, Josef van Genabith, and Andy Way. 2005. Large-scale induction and evaluation of lexical resources from the penn-ii and penn-iii treebanks. *Computational Linguistics*, 31:328–365.

Sebastian Pado and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33:161–199.

Judita Preiss, Ted Briscoe, and Anna Korhonen. 2007. A system for large-scale acquisition of verbal, nominal and adjectival subcategorization frames from corpora. In *Proceedings of ACL-07*.

Valeria Quochi, Francesca Frontini, Roberto Bartolini, Olivier Hamon, Marc Poch, Muntsa Padr, Nuria Bel, Gregor Thurmair, Antonio Toral, and Amir Kamram. 2012. Third evaluation report. evaluation of panacea v3 and produced resources. Technical report.

Roi Reichart and Anna Korhonen. 2013. Improved lexical acquisition through dpp-based verb clustering. In *Proceedings of ACL-13*.

Roi Reichart and Ari Rappoport. 2009. The nvi clustering evaluation measure. In *Proceedings of CoNLL-09*.

Roi Reichart, Gal Elidan, and Ari Rappoport. 2012. A diverse dirichlet process ensemble for unsupervised induction of syntactic categories. In *Proceedings of COLING-12*.

Laura Rimell, Thomas Lippincott, Karin Verspoor, Helen Johnson, and Anna Korhonen. 2013. Acquisition and evaluation of verb subcategorization resources for biomedicine. *Journal of Biomedical Informatics*, 46:228–237.

Eric Ringger, Peter McClanahan, Robbie Haertel, George Busby, Marc Carmen, James Carroll, Kevin Seppi, and Deryle Lonsdale. 2007. Active learning for part-of-speech tagging: Accelerating corpus annotation. In *Proceedings of the ACL-07 Linguistic Annotation Workshop*.

Douglas Roland and Daniel Jurafsky. 1998. subcategorization frequencies are affected by corpus choice. In *Proceedings of ACL-98*.

Andrew Rosenberg and Julia Hirschberg. 2007. V measure: a conditional entropybased external cluster evaluation measure. In *Proceedings of EMNLP-07*.

Alexander Rush, Roi Reichart, Michael Collins, and Amir Globerson. 2012. Improved parsing and pos tagging using inter-sentence consistency constraints. In *Proceedings of EMNLP-12*.

Sabine Schulte im Walde. 2006. Experiments on the automatic induction of german semantic verb classes. *Computational Linguistics*, 32(2):159–194.

Solomon Shimony. 1994. Finding the maps for belief networks is np-hard. *Artificial Intelligence*, 68:399–310.

David Sontag, Talya Meltzer, Amir Globerson, Tommi Jaakkola, and Yair Weiss. 2008. Tightening lp relaxations for map using message passing. In *Proceedings of UAI-08*.

Lin Sun and Anna Korhonen. 2011. Hierarchical verb clustering using graph factorization. In *Proceedings of EMNLP-11*.

Ivan Titvo and Alexandre Klementiev. 2012. A bayesian approach to unsupervised semantic role induction. In *Proceedings of EMNLP-12*.

Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL-03*.

Peter Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.

Tim Van de Cruys, Laura Rimell, Thierry Poibeau, and Anna Korhonen. 2012. Multi-way tensor factorization for unsupervised lexical acquisition. In *Proceedings of COLING-12*.

Giulia Venturi, Simonetta Montemagni, Simone Marchi, Yutaka Sasaki, Paul Thompson, John McNaught, and Sophia Ananiadou. 2009. Bootstrapping a verb lexicon for biomedical information extraction. *Computational Linguistics and Intelligent Text Processing*, 5449:137–148.

Marion Weller, Alexander Fraser, and Sabine Schulte im Walde. 2013. Using subcategorization knowledge to improve case prediction for translation to german. In *Proceedings of ACL-13*.

Chen Yanover, Talya Meltzer, and Yair Weiss. 2006. Linear programming relazations and belief propogataion an empitical study. *JMLR Special Issue on Machine Learning and Large Scale Optimization*.

Stella Yu and Jianbo Shi. 2003. Multiclass spectral clustering. In *Proceedings of ICCV-13*.