Automatic Classification of English Verbs Using Rich Syntactic Features

Lin Sun and Anna Korhonen
Computer Laboratory
University of Cambridge
15 JJ Thomson Avenue
Cambridge CB3 0FD, UK
ls418,alk23@cl.cam.ac.uk

Yuval Krymolowski
Department of Computer Science
University of Haifa
31905, Haifa
Israel
yuvalkry@gmail.com

Abstract

Previous research has shown that syntactic features are the most informative features in automatic verb classification. We experiment with a new, rich feature set, extracted from a large automatically acquired subcategorisation lexicon for English, which incorporates information about arguments as well as adjuncts. We evaluate this feature set using a set of supervised classifiers, most of which are new to the task. The best classifier (based on Maximum Entropy) yields the promising accuracy of 60.1% in classifying 204 verbs to 17 Levin (1993) classes. We discuss the impact of this result on the state-of-art, and propose avenues for future work.

1 Introduction

Recent research shows that it is possible, using current natural language processing (NLP) and machine learning technology, to automatically induce lexical classes from corpus data with promising accuracy (Merlo and Stevenson, 2001; Korhonen et al., 2003; Schulte im Walde, 2006; Joanis et al., 2007). This research is interesting, since lexical classifications, when tailored to the application and domain in question, can provide an effective means to deal with a number of important NLP tasks (e.g. parsing, word sense disambiguation, semantic role labeling), as well as enhance performance in many applications (e.g. information extraction, question-answering, machine translation) (Dorr, 1997; Prescher et al., 2000; Swier and Stevenson, 2004; Dang, 2004; Shi and Mihalcea, 2005).

Lexical classes are useful because they capture generalizations over a range of (cross-)linguistic properties. Being defined in terms of similar meaning components and (morpho-)syntactic behaviour of words (Jackendoff, 1990; Levin, 1993) they generally incorporate a wider range of properties than e.g. classes defined solely on semantic grounds (Miller, 1990). They can be used to build a lexical organization which effectively captures generalizations and predicts much of the syntax and semantics of a new word by associating it with an appropriate class. This can help compensate for lack of data for individual words in NLP.

Large-scale exploitation of lexical classes in real-world or domain-sensitive tasks has not been possible because existing manually built classifications are incomprehensive. They are expensive to extend and do not incorporate important statistical information about the likelihood of different classes for words. Automatic classification is a better alternative. It is cost-effective and gathers statistical information as a side-effect of the acquisition process.

Most work on automatic classification has focussed on verbs which are typically the main predicates in sentences. Syntactic features have proved the most informative in verb classification. Experiments have been reported using both (i) deep syntactic features (e.g. subcategorization frames (SCFs)) extracted using parsers and subcategorisation acquisition systems (Schulte im Walde, 2000; Korhonen et al., 2003; Schulte im Walde, 2006) and (ii) shallow ones (e.g. NPs/PPs preceding/following verbs) extracted using taggers and chunkers (Merlo and Stevenson, 2001; Joanis et al., 2007).

(i) correspond closely with features used for manual classification (Levin, 1993). They have proved successful in the classification of German (Schulte im Walde, 2006) and English verbs (Korhonen et al., 2003). Yet promising results have also been reported when using (ii) for English verb classification (Merlo and Stevenson, 2001; Joanis et al., 2007). This may indicate that (i) are optimal for the task when combined with additional syntactic information from (ii).

We investigate this matter by experimenting with a new, rich feature set which incorporates information about SCFs (arguments) as well as adjuncts. It was extracted from VALEX, a large automatically acquired SCF lexicon for English (Korhonen et al., 2006). We evaluate the feature set thoroughly using set of supervised classifiers, most of which are new in verb classification. The best performing classifier (Maximum Entropy) yields the accuracy of 60.1% on classifying 204 verbs into 17 Levin (1993) classes. This result is good, considering that we performed no sophisticated feature engineering or selection based on the properties of the target classification (Joanis et al., 2007). We propose various avenues for future work.

We introduce our target classification in section 2 and syntactic features in section 3. The classification techniques are presented in section 4. Details of the experimental evaluation are supplied in section 5. Section 6 provides discussion and concludes with directions for future work.

2 Test Verbs and Classes

We adopt as a target classification Levin's (1993) well-known taxonomy where verbs taking similar diathesis alternations are assumed to share meaning components and are organized into a semantically coherent class. For instance, the class of "*Break* Verbs" (class 45.1) is partially characterized by its participation in the following alternations:

1. **Causative/inchoative alternation**:
   Tony broke the window ↔ *The window broke*
2. **Middle alternation**:
   Tony broke the window ↔ *The window broke easily*
3. **Instrument subject alternation**:
   Tony broke the window with the hammer ↔ *The hammer broke the window*

	LEVIN CLASS	EXAMPLE VERBS
9.1	PUT	bury, place, install, mount, put
10.1	REMOVE	remove, abolish, eject, extract, deduct
11.1	SEND	ship, post, send, mail, transmit
13.5.1	GET	win, gain, earn, buy, get
18.1	HIT	beat, slap, bang, knock, pound
22.2	AMALGAMATE	contrast, match, overlap, unite, unify
29.2	CHARACTERIZE	envisage, portray, regard, treat, enlist
30.3	PEER	listen, stare, look, glance, gaze
31.1	AMUSE	delight, scare, shock, confuse, upset
36.1	CORRESPOND	cooperate, collide, concur, mate, flirt
37.3	MANNER OF SPEAKING	shout, yell, moan, mutter, murmur
37.7	SAY	say, reply, mention, state, report
40.2	NONVERBAL EXPRESSION	smile, laugh, grin, sigh, gas
43.1	LIGHT EMISSION	shine, flash, flare, glow, blaze
45.4	CHANGE OF STATE	soften, weaken, melt, narrow, deepen
47.3	MODES OF BEING WITH MOTION	quake, falter, sway, swirl, teeter
51.3.2	RUN	swim, fly, walk, slide, run

Table 1: Test classes and example verbs

Alternations are expressed as pairs of SCFs. Additional properties related to syntax, morphology and extended meanings of member verbs are specified with some classes. The taxonomy provides a classification of 4,186 verb senses into 48 broad and 192 fine-grained classes according to their participation in 79 alternations involving NP and PP complements.

We selected 17 fine-grained classes and 12 member verbs per class (table 2) for experimentation. The small test set enabled us to evaluate our results thoroughly. The classes were selected to (i) include both syntactically and semantically similar and different classes (to vary the difficulty of the classification task), and to (ii) have enough member verbs whose predominant sense belongs to the class in question (we verified this according to the method described in (Korhonen et al., 2006)). As VALEX was designed to maximise coverage most test verbs had 1000-9000 occurrences in the lexicon.

3 Syntactic Features

We employed as features distributions of SCFs specific to given verbs. We extracted them from the recent VALEX (Korhonen et al., 2006) lexicon which provides SCF frequency information for 6,397 English verbs. VALEX was acquired automatically from five large corpora and the Web (using up to 10,000 occurrences per verb) using the subcategorization acquisition system of Briscoe and Carroll (1997). The system incorporates RASP, a domain-independent robust statistical parser (Briscoe and

Carroll, 2002), and a SCF classifier which identifies 163 verbal SCFs. The basic SCFs abstract over lexically-governed particles and prepositions and predicate selectional preferences.

We used the noisy *unfiltered* version of VALEX which includes 33 SCFs per verb on average[1]. Some are genuine SCFs but some express adjuncts (e.g. *I sang in the party* could be SCF PP). A lexical entry for each verb and SCF combination provides e.g. the frequency of the entry (in active and passive) in corpora, the POS tags of verb tokens, the argument heads in argument positions, and the prepositions in PP slots. We experimented with three feature sets:

1. **Feature set 1**: SCFs and their frequencies
2. **Feature set 2**: Feature set 1 with two high frequency PP frames parameterized for prepositions: the simple PP (e.g. *they apologized* **to** *him*) and NP-PP (e.g. *he removed the shoes* **from** *the bag*) frames.
3. **Feature set 3**: Feature set 2 with three additional high frequency PP frames parameterized for prepositions: the NP-FOR-NP (e.g. *he bought a book* **for** *him*), NP-TO-NP (e.g. *he gave a kiss* **to** *her*), and OC-AP, EQUI, AS (e.g. *he condemned him* **as** *stupid*) frames.

In feature sets 2 and 3, 2-5 PP SCFs were refined according to the prepositions provided in the VALEX SCF entries (e.g. PP_at, PP_on, PP_in) because Levin specifies prepositions with some SCFs / classes. The scope was restricted to the 3-5 highest ranked PP SCFs to reduce the effects of sparse data.

4 Classification

4.1 Preparing the Data

A feature vector was constructed for each verb. VALEX includes 107, 287 and 305 SCF types for feature sets 1, 2, and 3, respectively. Each feature corresponds to a SCF type, and its value is the relative frequency of the SCF with the verb in question. Some of the feature values are zero, because most verbs take only a subset of the possible SCFs.

4.2 Machine Learning Methods

We implemented three methods for classification: the K nearest neighbours (KNN), support vector machines (SVM), and maximum entropy (ME). To our knowledge, only SVM has been previously used for verb classification. The free parameters were optimised for each feature set by (i) defining the value range (as explained below), and by (ii) searching for the optimal value on the training data using 10 fold cross validation (section 5.2).

4.2.1 K Nearest Neighbours

KNN is a memory-based classification method based on the distances between verbs in the feature space. For each verb in the test data, we measure its distance to each verb in the training data. The verb class label is the most frequent label in the top K closest training verbs. We use the entropy-based Jensen-Shannon (JS) divergence as the distance measure:

$$JS(P,Q) = \frac{1}{2}\left[D(P\|\frac{P+Q}{2}) + D(Q\|\frac{P+Q}{2})\right]$$

The range of the parameter K is 2-20.

4.2.2 Support Vector Machines

SVM (Vapnik, 1995) tries to find a maximal margin hyperplane to separate between two groups of verb feature vectors. In practice, a linear hyperplane does not always exist. SVM uses a kernel function to map the original feature vectors to higher dimension space. The 'maximal margin' optimizes our choice of dimensionality to avoid over-fitting. We use Chang and Lin (2001) 's LIBSVM library to implement the SVM. Following Hsu et al. (2003), we use the radial basis function as the kernel function:

$$K(x_i, x_j) = \exp\left(-\gamma||x_i - x_j||^2\right), \gamma > 0$$

γ and the cost of the error term C (the penalty for margin errors) are optimized. The search ranges of Hsu et al. (2003) are used:

$$C = 2^{-5}, 2^{-3}, \ldots, 2^{15}, 2^{17}\,; \gamma = 2^{-17}, 2^{-15}, \ldots, 2^{1}, 2^{3}$$

4.2.3 Maximum Entropy

ME constructs a probabilistic model that maximizes entropy on test data subject to a set of feature constraints. If verb x is in class 10.1 and takes the SCF 49 (NP-PP) with the relative frequency of 0.6 in feature function f, we have

$$f(x,y) = 0.6 \text{ if } y = 10.1 \text{ and } x = 49$$

The expected value of a feature f with respect to the empirical distribution (training data) is

$$\tilde{\mathcal{E}}(f) \equiv \textstyle\sum_{x,y} \tilde{p}(x,y) f(x,y)$$

The expected value of the feature f (on test data) with respect to the model $p(y|x)$ is

[1] The SCF accuracy of this lexicon is 23.7 F-measure, see (Korhonen et al., 2006) for details.

$$\mathcal{E}(f) \equiv \sum_{x,y} \tilde{p}(x)p(y|x)f(x,y)$$

$\tilde{p}(x)$ is the empirical distribution of x in the training data. We constrain $\mathcal{E}(f)$ to be the same as $\tilde{\mathcal{E}}(f)$

$$\mathcal{E}(f) = \tilde{\mathcal{E}}(f)$$

The model must maximize the entropy $H(Y|X)$

$$H(Y|X) \equiv -\sum_{x,y} \tilde{p}(x)p(y|x)\log p(y|x)$$

The constraint-optimization problem is solved by the Lagrange multiplier (Pietra et al., 1997). We used Zhang (2004)'s maximum entropy toolkit for implementation. The number of iterations i (5-50) of the parameter estimation algorithm is optimised.

5 Experiments

5.1 Methodology

We split the data into training and test sets using two methods. The first is 'leave one out' *cross-validation* where one verb in each class is held out as test data, and the remaining N-1 (i.e. 11) verbs are used as training data. The overall accuracy is the average accuracy of N rounds. The second method is *resampling*. For each class, 3 verbs are selected randomly as test data, and 9 are used as training data. The process is repeated 30 times, and the average result is recorded.

5.2 Measures

The methods are evaluated using first accuracy – the percentage of correct classifications out of all the classifications:

$$Accuracy = \frac{truePositives}{truePositives+falseNegatives}$$

When evaluating the performance at class level, precision and recall are calculated as follows:

$$Precision = \frac{truePositives}{truePositives+falsePositives}$$
$$Recall = \frac{truePositives}{truePositives+falseNegatives}$$

F-score is the balance over recall and precision. We report the average F-score over the 17 classes. Given there are 17 classes in the data, the accuracy of randomly assigning a verb into one of the 17 classes is $1/17 \approx 5.8\%$.

5.3 Results from Quantitative Evaluation

Table 2 shows the average performance of each classifier and feature set according to 'leave one out' cross-validation[2]. Each classifier performs considerably better than the random baseline. The simple KNN method produces the lowest accuracy (44.1-54.9) and SVM and ME the best (47.1-57.9 and 47.5-59.3, respectively).

The performance of all methods improves sharply when moving from the feature set 1 to the refined feature set 2: both accuracy and F-measure improve by over 10%. When moving from feature set 2 to the sparser feature set 3 (which includes a higher number of low frequency PP features) KNN worsens clearly (c. 5% in accuracy and F-measure) while the improvement in other methods is very small. This suggests that KNN deals worse than other methods with sparse data.

The resampling results in table 3 reveal that some classifiers perform worse than others when less training data is available[3]. KNN produces considerably lower results, particularly with the sparse feature set 3: 28.2 F-measure vs. 48.2 with cross-validation. Also SVM performs worse with feature set 3: 54.6 F-measure vs. 58.2 with cross-validation. ME thus appears the most robust method with smaller training data, producing results comparable with those in cross-validation.

Figure 1 shows the F-measure for 17 individual classes when the methods are used with feature set 3. Levin classes 40.2, 29.2, and 37.3 (see table 2) (the ones taking fewer prepositions with higher frequency) have the best average performance (65% or more), and classes 47.3, 45.4 and 18.1 the worst (40% or less). ME outperforms SVM with 9 of the 17 classes.

5.4 Qualitative Evaluation

We did some qualitative analysis to trace the origin of error types produced by ME with feature set 3. Examination of the worst performing class 47.3 (MODES OF BEING INVOLVING MOTION verbs) illustrates well the various error types. 10 of the 12 verbs in this class are classified incorrectly:

- **3** in class 43.1 (LIGHT EMISSION verbs): Verbs in 47.3 and 43.1 describe intrinsic properties of their subjects (e.g. *a jewel sparkles*, *a flag flutters*). Their similar alternations and PP SCFs make it difficult to separate them on syntactic grounds.
- **2** in class 51.3.2 (RUN verbs): 47.3 and 51.3.2 share the meaning component of motion. Their members take similar alternations and SCFs, which causes the confusion.

[2]Recall is not shown as it is identical here with accuracy.

[3]Recall that the amount of training data is smaller with resampling evaluation, see section 5.2.

	Feature set 1			Feature set 2			Feature set 3		
	ACC	P	F	ACC	P	F	ACC	P	F
RAND	5.8			5.8			5.8		
KNN	44.1	48.4	44.0	54.9	56.9	53.9	49.5	47.0	48.2
ME	47.5	49.4	47.6	59.3	61.4	59.9	59.3	61.9	60.0
SVM	47.1	50.4	47.8	57.8	59.4	57.9	57.8	60.1	58.2

Table 2: 'Leave one out' cross-validation results for KNN, ME, and SVM

	Feature set 1			Feature set 2			Feature set 3		
	ACC	P	F	ACC	P	F	ACC	P	F
RAND	5.8			5.8			5.8		
KNN	37.3	39.9	36.5	42.7	47.2	42.6	27.1	34.2	28.2
ME	47.1	47.3	47.0	58.1	59.1	58.1	60.1	60.5	59.8
SVM	47.3	50.2	47.7	56.8	59.5	57.1	54.4	56.5	54.6

Table 3: Re-sampling results for KNN, ME, and SVM

- **2** in class 37.7 (SAY verbs) and 1 in class 37.3 (MANNER OF SPEAKING verbs): 47.3 differs in semantics and syntax from 37.7 and 37.3. The confusion is due to idiosyncratic properties of individual verbs (e.g. *quake, wiggle*).
- **1** in class 36.1 (CORRESPOND verbs): 47.3 and 36.1 are semantically very different, but their members take similar intransitive and PP SCFs with high frequency.
- **1** in class 45.4 (OTHER CHANGE OF STATE verbs): Classes 47.3 and 45.3 are semantically different. Their similar PP SCFs explains the misclassification.

Most errors concern classes which are in fact semantically related. Unfortunately there is no gold standard which would comprehensively capture the semantic relatedness of Levin classes. Other errors concern semantically unrelated but syntactically similar classes – cases which we may be able to address in the future with careful feature engineering. Some errors relate to syntactic idiosyncracy. These show the true limits of lexical classification - the fact that the correspondence between the syntax and semantics of verbs is not always perfect.

6 Discussion and Conclusion

Our best results (e.g. 60.1 accuracy and 59.8 F-measure of ME) are good, considering that no sophisticated feature engineering / selection based on the properties of the target classification was performed in these experiments. The closest comparison point is the recent experiment reported by Joanis et al. (2007) which involved classifying 835 English verbs to 14 Levin classes using SVM. Features were specifically selected via analysis of alternations that are used to characterize Levin classes. Both shallow syntactic features (syntactic slots obtained using a chunker) and deep ones (SCFs extracted using Briscoe and Carroll's system) were used. The accuracy was 58% with the former and only 38% with the latter. This experiment is not directly comparable with ours as we classified a smaller number of verbs (204) to a higher number of Levin classes (17) (i.e. we had less training data) and did not select the optimal set of features using Levin's alternations. We nevertheless obtained better accuracy with our best performing method, and better accuracy (47%) with the same method (SVM) when the comparable feature set 1 was acquired using the very same subcategorization acquisition system.

It is likely that using larger and noisier SCF data explains the better result, suggesting that rich syntactic features incorporating information about both arguments and adjuncts are ideal for verb classification. Further experiments are required to determine the optimal set of features. In the future, we plan to experiment with different (noisy and filtered) versions of VALEX and add to the comparison a shallower set of features (e.g. NP and PP slots in VALEX regardless of the specific SCFs). We will also improve the features e.g. by enriching them with additional syntactic information available in VALEX lexical entries.

Acknowledgement

This work was partially supported by the Royal Society, UK.

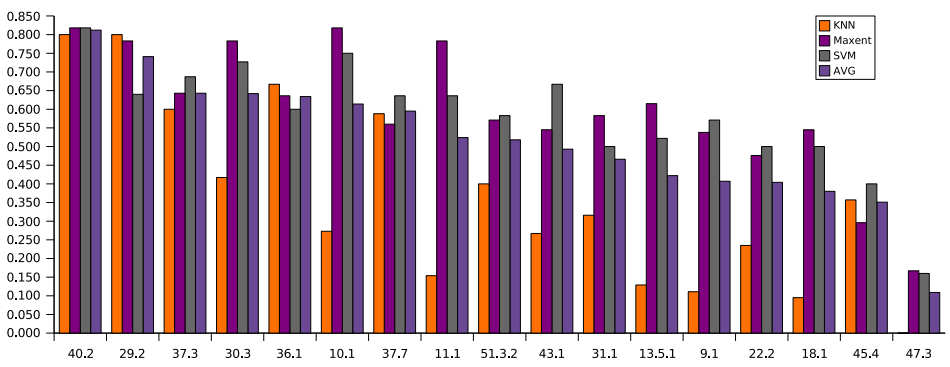

Figure 1: Class level F-score for feature set 3 (cross-validation)

References

E. J. Briscoe and J. Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of the* 5^{th} *ACL Conference on Applied Natural Language Processing*, pages 356–363, Washington DC.

E. J. Briscoe and J. Carroll. 2002. Robust accurate statistical annotation of general text. In *Proceedings of the* 3^{rd} *LREC*, pages 1499–1504, Las Palmas, Gran Canaria.

C. Chang and J. Lin, 2001. *LIBSVM: a library for support vector machines*.

H. T. Dang. 2004. *Investigations into the Role of Lexical Semantics in Word Sense Disambiguation*. Ph.D. thesis, CIS, University of Pennsylvania.

B. J. Dorr. 1997. Large-scale dictionary construction for foreign language tutoring and interlingual machine translation. *Machine Translation*, 12(4):271–322.

W. Hsu, C. Chang, and J. Lin. 2003. A practical guide to support vector classification.

R. Jackendoff. 1990. *Semantic Structures*. MIT Press, Cambridge, Massachusetts.

E. Joanis, S. Stevenson, and D. James. 2007. A general feature space for automatic verb classification. *Natural Language Engineering*, Forthcoming.

A. Korhonen, Y. Krymolowski, and Z. Marx. 2003. Clustering polysemic subcategorization frame distributions semantically. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 64–71.

A. Korhonen, Y. Krymolowski, and T. Briscoe. 2006. A large subcategorization lexicon for natural language processing applications. In *Proceedings of LREC*.

B. Levin. 1993. *English Verb Classes and Alternations*. Chicago University Press, Chicago.

P. Merlo and S. Stevenson. 2001. Automatic verb classification based on statistical distributions of argument structure. *Computational Linguistics*, 27(3):373–408.

G. A. Miller. 1990. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312.

S. D. Pietra, J. D. Pietra, and J. D. Lafferty. 1997. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393.

D. Prescher, S. Riezler, and M. Rooth. 2000. Using a probabilistic class-based lexicon for lexical ambiguity resolution. In *18th International Conference on Computational Linguistics*, pages 649–655, Saarbrücken, Germany.

S. Schulte im Walde. 2000. Clustering verbs semantically according to their alternation behaviour. In *Proceedings of COLING*, pages 747–753, Saarbrücken, Germany.

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

L. Shi and R. Mihalcea. 2005. Putting pieces together: Combining FrameNet, VerbNet and WordNet for robust semantic parsing. In *Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, Mexico.

R. Swier and S. Stevenson. 2004. Unsupervised semantic role labelling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 95–102, Barcelona, Spain, August.

V. N. Vapnik. 1995. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA.

L. Zhang, 2004. *Maximum Entropy Modeling Toolkit for Python and C++*, December.