Automatically acquiring models of preposition use

Rachele De Felice and Stephen G. Pulman
Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
{rachele.defelice|stephen.pulman}@comlab.ox.ac.uk

Abstract

This paper proposes a machine-learning based approach to predict accurately, given a syntactic and semantic context, which preposition is most likely to occur in that context. Each occurrence of a preposition in an English corpus has its context represented by a vector containing 307 features. The vectors are processed by a voted perceptron algorithm to learn associations between contexts and prepositions. In preliminary tests, we can associate contexts and prepositions with a success rate of up to 84.5%.

1 Introduction

Prepositions have recently become the focus of much attention in the natural language processing community, as evidenced for example by the ACL workshops, a dedicated Sem-Eval task, and The Preposition Project (TPP, Litkowski and Hargraves 2005). This is because prepositions play a key role in determining the meaning of a phrase or sentence, and their correct interpretation is crucial for many NLP applications: AI entities which require spatial awareness, natural language generation (e.g. for automatic summarisation, QA, MT, to avoid generating sentences such as **I study at England*), automatic error detection, especially for non-native English speakers. We present here an approach to learning which preposition is most appropriate in a given context by representing the context as a vector populated by features referring to its syntactic and semantic characteristics. Preliminary tests on five prepositions - *in, of, on, to, with* - yield a success rate of between 71% and 84.5%. In Section 2, we illustrate our motivations for using a vector-based approach. Section 3 describes the vector creation, and Section 4 the learning procedure. Section 5 presents a discussion of some preliminary results, and Section 6 offers an assessment of our method.

2 Contextual features

Modelling preposition use is challenging because it is often difficult to explain why in two similar contexts a given preposition is correct in one but not the other. For example, we say A is *similar **to** B*, but *different **from** C*, or we *study **in** England*, but ***at** King's College*. Nor can we rely on co-occurrence with particular parts of speech (POS), as most prepositions have a reasonably wide distribution. Despite this apparently idiosyncratic behaviour, we believe that prepositional choice is governed by a combination of several syntactic and semantic features. Contexts of occurrence can be represented by vectors; a machine learning algorithm trained on them can predict with some confidence, given a new occurrence of a context vector, whether a certain preposition is appropriate in that context or not.

We consider the following macro-categories of features to be relevant: POS being modified; POS of the preposition's complement; given a RASP-style grammatical relation output (GR; see e.g. Briscoe et al. 2006), what GRs the preposition occurs in; named entity (NE) information - whether the modified or complement items are NEs; WordNet information - to which of the WordNet lexicographer

Proceedings of the 4th ACL-SIGSEM Workshop on Prepositions, pages 45–50,
Prague, Czech Republic, June 2007. ©2007 Association for Computational Linguistics

classes[1] the modified and complement nouns and verbs belong; immediate context - POS tags of ±2 word window around the preposition. For example, given a sentence such as *John drove **to** Cambridge*, we would note that this occurrence of the preposition ***to*** modifies a verb, its complement is a location NE noun, the verb it modifies is a 'verb of motion', the tags surrounding it are NNP, VBD, NNP[2], and it occurs in the relation 'iobj' with the verb, and 'dobj' with the complement noun.

Our 307-feature set aims to capture all the salient elements of a sentence which we believe could be involved in governing preposition choice, and which can be accurately recognised automatically. Our choice of features is provisional but based on a study of errors frequently made by learners of English: however, when we spot a misused preposition, it often takes some reflection to understand which elements of the sentence are making that preposition choice sound awkward, and thus we have erred on the side of generosity. In some cases it is easier: we observe that in the earlier example *England* is a location NE while *King's College* is an organisation NE: this distinction may be the trigger for the difference in preposition choice.

3 Vector construction

The features are acquired from a version of the British National Corpus (BNC) processed by the C&C tools pipeline (Clark and Curran, to appear). The output of the C&C tools pipeline, which includes stemmed words, POS tags, NER, GRs and Combinatory Categorial Grammar (CCG) derivations of each sentence, is processed by a Python script which, for each occurrence of a preposition in a sentence, creates a vector for that occurrence and populates it with *0s* and *1s* according to the absence or presence of each feature in its context. Each vector therefore represents a corpus-seen occurrence of a preposition and its context. For each preposition we then construct a dataset to be processed by a machine learning algorithm, containing all the vectors which do describe that preposition's contexts, and an equal number of those which do not: our hypothesis is that these will be sufficiently different from the 'positive' contexts that a machine learning algorithm will be able to associate the positive vectors more strongly to that preposition.

4 Testing the approach

To test our approach, we first experimented with a small subset of the BNC, about 230,000 words (9993 sentences, of which 8997 contained at least one preposition). After processing we were left with over 33,000 vectors associated with a wide range of prepositions. Of course there is a certain amount of noise: since the vectors describe what the parser has tagged as prepositions, if something has been mistagged as one, then there will be a vector for it. Thus we find in our data vectors for things such as *if* and *whether*, which are not generally considered prepositions, and occasionally even punctuation items are misanalysed as prepositions; however, these represent only a small fraction of the total and so do not constitute a problem.

Even with a relatively large number of vectors, data sparseness is still an issue and for many prepositions we did not find a large number of occurrences in our dataset. Because of this, and because this is only a preliminary, small-scale exploration of the feasibility of this approach, we decided to initially focus on only 5 common prepositions[3]: ***in** (4278 occurrences), **of** (7485), **on** (1483), **to** (4841[4]), **with** (1520)*. To learn associations between context vectors and prepositions, we use the Voted Perceptron algorithm (Freund and Schapire 1999). At this stage we are only interested in establishing whether a preposition is correctly associated with a given context or not, so a binary classifier such as the Voted Perceptron is well-suited for our task. At a later stage we aim to expand this approach so that a notification of error or inappropriateness is paired with suggestions for other, more likely prepositions. A possible implementation of this is the output of a

[1] These are 41 broad semantic categories (e.g. 'noun denoting a shape', 'verb denoting a cognitive process') to which all nouns and verbs in WordNet are assigned.

[2] Penn Treebank tagset.

[3] These prepositions often occur in compound prepositions such as *in front of*; their inclusion in the data could yield misleading results. However out of 33,339 vectors, there were only 463 instances of compound prepositions, so we do not find their presence skews the results.

[4] Here *to* includes occurrences as an infinitival marker. This is because the tagset does not distinguish between the two occurrences; also, with a view to learner errors, its misuse as both a preposition and an infinitival marker is very common.

ranked list of the probability of each preposition occurring in the context under examination, especially as of course there are many cases in which more than one preposition is possible (cf. *the folder* ***on*** *the briefcase* vs. *the folder* ***in*** *the briefcase*).

We use the Weka machine learning package to run the Voted Perceptron. Various parameters can be modified to obtain optimal performance: the number of epochs the perceptron should go through, the maximum number of perceptrons allowed, and the exponent of the polynomial kernel function (which allows a linear function such as the perceptron to deal with non-linearly separable data), as well as, of course, different combinations of vector features. We are experimenting with several permutations of these factors to ascertain which combination gives the best performance. Preliminary results obtained so far show an average accuracy of 75.6%.

5 Results and Discussion

We present here results from two of the experiments, which consider two possible dimensions of variation: the polynomial function exponent, *d*, and the presence of differing subsets of features: WordNet or NE information and the ±2 POS tag window. Tests were run 10 times in 10-fold cross-validation.

5.1 The effect of the *d* value

The value of *d* is widely acknowledged in the literature to play a key role in improving the performance of the learning algorithm; the original experiment described in Freund and Schapire (1999) e.g. reports results using values of *d* from 1 to 6, with *d=2* as the optimal value. Therefore our first investigation compared performance with values for *d* set to *d=1* and *d=2*, with the other parameters set to 10 epochs and 10,000 as the maximum number of perceptrons allowed (Table 1).

We can see that the results, as a first attempt at this approach, are encouraging, achieving a success rate of above 80% in two cases. Performance on *on* is somewhat disappointing, prompting the question whether this is because less data was available for it (although *with*, with roughly the same sized dataset, performs better), or if there is something intrinsic to the syntactic and semantic properties of this preposition that makes its use harder to pinpoint. The average performance of 75.6 - 77% is a promising starting point, and offers a solid base on which to proceed with a finer tuning of the various parameters, including the feature set, which could lead to better results. The precision and recall support our confidence in this approach, as there are no great differences between the two in any dataset: this means that the good results we are achieving are not coming at the expense of one or the other measure.

If we compare results for the two values of *d*, we note that, contrary to expectations, there is no dramatic improvement. In most cases it is between less than 1% and just over that; only *on* shows a marked improvement of 4%. However, a positive trend is evident, and we will continue experimenting with variations on this parameter's value to determine its optimal setting.

5.2 The effect of various feature categories

As well as variations on the learning algorithm itself, we also investigate how different types of features affect performance. This is interesting not only from a processing perspective - if some features are not adding any useful information then they may be disregarded, thus speeding up processing time - but also from a linguistic one. If we wish to use insights from our work to assist in the description of preposition use, an awareness of the extent to which different elements of language contribute to preposition choice is clearly of great importance.

Here we present some results using datasets in which we have excluded various combinations of the NE, WordNet and POS tag features. The WordNet and POS macrocategories of features are the largest sets - when both are removed, the vector is left with only 31 features - so it is interesting to note how this affects performance. Furthermore, the WordNet information is in a sense the core 'lexical semantics' component, so its absence allows for a direct comparison between a model 'with semantics' and one without. However, the WordNet data is also quite noisy. Many lexical items are assigned to several categories, because we are not doing any sense resolution on our data. The POS tag features represent 'context' in its most basic sense, detached from strict syntactic and semantic considerations; it is useful to examine the contribution this type of less sophisticated information can make.

	d=1				**d=2**			
Preposition	%correct	Precision	Recall	F-score	%correct	Precision	Recall	F-score
in	76.30%	0.75	0.78	0.77	76.61%	0.77	0.77	0.77
of	83.64%	0.88	0.78	0.83	84.47%	0.87	0.81	0.84
on	65.66%	0.66	0.65	0.65	69.09%	0.69	0.69	0.69
to	81.42%	0.78	0.87	0.82	82.43%	0.81	0.85	0.83
with	71.25%	0.73	0.69	0.70	72.88%	0.73	0.72	0.73
av.	75.65%	0.76	0.75	0.75	77.10%	0.77	0.77	0.77

Table 1: The effect of the *d* value

	All features	**No W.Net**	**No POS**	**No NER**	**No WN + POS**	**GRs only**
% correct	83.64%	83.47%	81.46%	83.33%	81.00%	81.46%
Precision	0.88	0.89	0.76	0.88	0.74	0.93
Recall	0.78	0.76	0.91	0.77	0.94	0.68
F-score	0.83	0.82	0.83	0.82	0.83	0.78

Table 2: OF: the effect of various feature categories (d=1)

Full results cannot be presented due to space restrictions: we present those for ‘of’, which are representative. In almost case, the dataset with all features included is the one with the highest percentage of correct classifications, so all features do indeed play a role in achieving the final result. However, among the various sets variation is of just 1 or 2%, nor do f-scores vary much. There are some interesting alternations in the precision and recall scores and a closer investigation of these might provide some insight into the part played by each set of features: clearly there are some complex interactions between them rather than a simple monotonic combination.

Such small variations allow us to conclude that these sets of features are not hampering peformance (because their absence does not in general lead to *better* results), but also that they may not be a major discriminating factor in preposition choice: grammatical relations seem to be the strongest feature - only 18 components of the vector! This does not imply that semantics, or the immediate context of a word, play no role: it may just be that the way this data is captured is not the most informative for our purposes. However, we must also consider if something else in the feature set is impeding better performance, or if this is the best we can achieve with these parameters, and need to identify more informative features. We are currently working on expanding the feature set, considering e.g. subcategorisation information for verbs, as well as experimenting with the removal of other types of features, and using the WordNet data differently. On the other hand, we also observe that each macrocategory of features does contribute something to the final result. This could suggest that there is no one magic bullet-like feature which definitely and faultlessly identifies a preposition but rather, as indeed we know by the difficulties encountered in finding straightforward identification criteria for prepositions, this depends on a complex interrelation of features each of which contributes something to the whole.

6 Evaluation and related work

6.1 Error detection evaluation

One of our motivations in this work was to investigate the practical utility of our context models in an error detection task. The eventual aim is to be able, given a preposition context, to predict the most likely preposition to occur in it: if that differs from the one actually present, we have an error. Using real learner English as testing material at our current stage of development is too complex, however. This kind of text presents several challenges for NLP and for our task more specifically, such as spelling mistakes - misspelled words would not be recognised by WordNet or any other lexical item-based component. Furthermore, often a learner’s error cannot simply be described in terms of one word needing to be replaced by another, but has a more complex structure. Although it is our intention to be able to process these kinds of texts eventually, as an interim evaluation we felt that it was best to focus just on texts where the only feature susceptible to error was a preposition. We therefore devised a simple artificial error detection task using a corpus in which errors are artificially inserted in otherwise correct text, for which we present interim results (the dataset is currently quite small) and we compare it against a ‘brute force’ baseline, namely using the recently released Google n-gram data to predict the most likely preposition.

We set up a task aimed at detecting errors in the use of *of* and *to*, for which we had obtained the best results in the basic classification tests reported earlier, and we created for this purpose a small corpus using BBC news articles, as we assume the presence of errors there, spelling or otherwise, is extremely unlikely. Errors were created by replacing correct occurrences of one of the prepositions with another, incorrect, one, or inserting *of* or *to* in place of other prepositions. All sentences contained at least one preposition. Together with a set of sentences where the prepositions were all correct, we obtained a set of 423 sentences for testing, consisting of 492 preposition instances. The aim was to replicate both kinds of errors one can make in using prepositions[5].

We present here some results from this small scale task; the data was classified by a model of the algorithm trained on the BNC data with all features included, 10 epochs, and *d=2*. If we run the task on the vectors representing all occurrences of each of the prepositions, and ask the classifier to distinguish between correct and incorrect usages, we find the percentage of correct classifications as follows:

Prep	Accuracy	Precision	Recall
of	75.8	0.72	0.68
to	81.35	0.76	0.74
Average:	78.58	0.74	0.71

These results show both high precision and high recall, as do those for the dataset consisting of correct occurrences of the preposition and use of another preposition instead of the right one: (*of* - 75%, *to* - 67% - these are accuracy figures only, as precision and recall make no sense here.) This small task shows that it is possible to use our model to reliably check a text for preposition errors.

However, these results need some kind of baseline for comparison. The most obvious baseline would be a random choice between positive and negative (i.e. the context matches or does not match the preposition) which we would expect to be successful 50% of the time. Compared to that the observed accuracies of 75% or more on all of these various classification tasks is clearly significant, representing a 50% or more reduction in the error rate.

However, we are also working on a more challenging baseline consisting of a simple 3-gram lookup in the Google n-gram corpus (ca. 980 million 3-grams). For example, given the phrase *fly _ Paris*, we could decide to use ***to*** rather than ***at*** because we find 10,000 occurrences of *fly* ***to*** *Paris* and hardly any of *fly* ***at*** *Paris*. In a quick experiment, we extracted 106 three-word sequences, consisting of one word each side of the preposition, from a random sample of the BBC dataset, ensuring each type of error was equally represented. For each sequence, we queried the Google corpus for possible prepositions in that sequence, selecting the most frequent one as the answer. Despite the very general nature of some of the 3-grams (e.g. *one of the*), this method performs very well: the n-gram method scores 87.5% for *of* (vs. our 75.8%) and 72.5% for *to* (vs. our 81.35%). This is only a suggestive comparison, because the datasets were not of the same size: by the time of the workshop we hope to have a more rigorous baseline to report. Clearly, unless afflicted by data sparseness, the raw word n-gram method will be very hard to beat, since it will be based on frequently encountered examples of correct usage. It is therefore encouraging that our method appears to be of roughly comparable accuracy even though we are using no actual word features at all, but only more abstract ones as described earlier. An obvious next step, if this result holds up to further scrutiny, is to experiment with combinations of both types of information.

6.2 Related work

Although, as noted above, there is much research being carried out on prepositions at the moment, to the best of our knowledge there is no work which takes an approach similar to ours in the task of preposition choice and error correction, i.e. one that aims to automate the process of context construction rather than relying on manually constructed grammars or other resources such as dictionaries (cf. TPP). Furthermore, much current research seems to have as its primary aim a semantic and functional descrip-

[5] A third, omitting it altogether, will be accounted for in future work.

tion of prepositions. While we agree this is a key aspect of preposition use, and indeed hope at a later stage of our research to derive some insights into this behaviour from our data, at present we are focusing on the more general task of predicting a preposition given a context, regardless of semantic function.

With regard to related work, as already mentioned, there is no direct comparison we can make in terms of learning preposition use by a similar method. One useful benchmark could be results obtained by others on a task similar to ours, i.e. error detection, especially in the language of non-native speakers. In this case the challenge is finding work which is roughly comparable: there are a myriad of variables in this field, from the characteristics of the learner (age, L1, education...) to the approach used to the types of errors considered. With this in mind, all we can do is mention some work which we feel is closest in spirit to our approach, but stress that the figures are for reference only, and cannot be compared directly to ours.

Chodorow and Leacock (2000) try to identify errors on the basis of context, as we do here, and more specifically a ± 2 word window around the word of interest, from which they consider function words and POS tags. Mutual information is used to determine more or less likely sequences of words, so that less likely sequences suggest the presence of an error. Unlike ours, their work focuses on content words rather than function words; they report a precision of 78% and a recall of 20%. Our precision is comparable to this, and our recall is much higher, which is an important factor in error detection: a user is likely to lose trust in a system which cannot spot his/her errors very often[6]. Izumi et al. (2004) work with a corpus of English spoken by Japanese students; they attempt to identify errors using various contextual features and maximum entropy based-methods. They report results for omission errors (precision 75.7%, recall 45.67%) and for replacement errors (P 31.17%, R 8%). With the caveat that we are not working with spoken language, which presents several other challenges, we note that in our task the errors, akin to replacement errors, are detected with much more success. Finally we can note the work done by Eeg-Olofsson and Knutsson (2003) on preposition errors in L2 Swedish. Their system uses manually crafted rules, unlike ours, and its performance is reported as achieving a recall of 25%. On the basis of this brief and by no means exhaustive overview of the field, we claim that our results in the error detection task are competitive, and we are working on fine-tuning various parameters to improve them further.

7 Conclusion

We have presented an automated approach to learning associations between sentence contexts and prepositions which does not depend on manually crafted grammars and achieves a success rate of up to 84.5%. This model was tested on a small set of texts with artificially created preposition errors, and was found to be successful at detecting between 76% and 81% of errors. Ongoing work is focusing on how to further improve performance taking into consideration both the parameters of the voted perceptron algorithm and the feature set of the vectors.

Acknowledgements

We wish to thank Stephen Clark for stimulating discussions and the anonymous reviewers for their helpful comments. Rachele De Felice is supported by an AHRC scholarship.

References

Ted Briscoe, John Carroll, and Rebecca Watson. 2006. The second release of the RASP system. In *COLING/ACL-06 Demo Session*, Sydney, Australia.

Martin Chodorow and Claudia Leacock. 2000. An unsupervised method for detecting grammatical errors. In *NAACL-00*, Seattle, Washington.

Stephen Clark and James Curran. To appear. Wide-coverage Efficient Statistical Parsing with CCG and Log-linear Models.

Jens Eeg-Olofsson and Ola Knutsson. 2003. Automatic grammar checking for second language learners - the use of prepositions. In *Nodalida-03*, Reykjavik, Iceland.

Yoav Freund and Robert E. Schapire. 1999 Large margin classification using the perceptron algorithm. *Machine Learning* 37:277-296

Emi Izumi, Kiyotaka Uchimoto, and Hitoshi Isahara. 2004 SST speech corpus of Japanese learners' English and automatic detection of learners' errors. *ICAME* 28:31-48

Ken Litkowski and Orin Hargraves. 2005. The Preposition Project. In *Second ACL-SIGSEM Prepositions Workshop*, Colchester, UK.

Guido Minnen, John Carroll, and Darren Pearce. 2001 Applied Morphological Processing of English. *Natural Language Engineering* 7(3):207-223

[6] Although of course precision is a key measure: it is not helpful for the user to be exposed to false alarms.