Detecting Learner Errors in the Choice of Content Words Using Compositional Distributional Semantics

Ekaterina Kochmar
Computer Laboratory
University of Cambridge
ek358@cl.cam.ac.uk

Ted Briscoe
Computer Laboratory
University of Cambridge
ejb@cl.cam.ac.uk

Abstract

We describe a novel approach to error detection in adjective–noun combinations. We present and release a new dataset of annotated errors where the examples are extracted from learner texts and annotated with error types. We show how compositional distributional semantic approaches can be applied to discriminate between correct and incorrect word combinations from learner data. Finally, we show how the output of the compositional distributional semantic models can be used as features in a classifier yielding good precision and accuracy.

1 Introduction

The task of error detection and correction (henceforth, EDC) in non-native writing in English has been a focus of research in recent years. However, usually research in this area focuses on EDC in the use of function words, such as articles or prepositions (Leacock et al., 2010; Dale et al., 2012), while much less attention has been paid to errors in the choice of content words.

Errors in function words are some of the most common error types in learner writing (Dalgish, 1985; Leacock et al., 2010), so it is important for any EDC system to be able to deal with such errors. Certain properties of these errors facilitate their detection and correction. As function words belong to closed classes, the set of possible corrections is limited by the size of the function word set. Since errors in function words are systematic and highly recurrent, in practice, each article or preposition has an even smaller number of appropriate alternatives. We illustrate this point with the following examples on (1) article and (2) preposition errors:

(1) I am *0*/a* student.

(2) Last October, I came *in*/to* Tokyo.

In (1) an EDC system would consider $\{$*a, an, the*$\}$ as possible corrections for the missing article. To correct the preposition *in* in (2), an EDC system would consider the most frequent prepositions $\{$*on, from, for, of, about, to, at, with, by*$\}$, among which *at* or *to* would have a higher chance to be appropriate corrections as these are most often confused with *in*. Confusion sets can be learnt from learner texts, and probabilities can be set up according to the distribution of the confusions (Rozovskaya and Roth, 2011).

EDC is usually cast as a multi-class classification task, with the number of classes equal to the number of target corrections. Detection and correction can occur simultaneously: an error is detected when an EDC system suggests using a word different from the one originally used by the learner, and the suggested word can be used as a correction. Each occurrence of a function word is represented with a feature vector, where features are derived from the surrounding context. This is usually highly informative for function words: for example, a context of *I am* and *student* or a similar noun requires the use of an indefinite article, while the only correct preposition to relate a verb of movement like *come* to a locative like *Tokyo* is *to*.

In this work, however, we focus on errors in the choice of content words, which have received much less attention in spite of being the third most frequent error type in learner writing (Leacock et al., 2010). Errors in content words are more challenging than errors in function words, since the number of possible

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1740–1751, Dublin, Ireland, August 23-29 2014.

confusions and corrections cannot be reduced to a finite set. For example, consider incorrect choice of adjectives in the following sentences extracted from learner data:

(3) A *big*/great* damage has been made to the environment.

(4) I have tried a rock'n'roll dance and a *classic*/classical* dance already.

The confusion in (3) is caused by semantic similarity of the adjectives *big* and *great*, while in (4) it is due to similarity in form between *classic* and *classical*. It is much harder to cast the EDC in content words as multi-class classification, unless we consider the full set of English adjectives as possible classes. The surrounding contexts are much sparser and less informative, and in addition to that, often contain further errors. In this work, we address error detection and focus on adjective-noun combinations (ANs), which are representative of the more general task of EDC in content word combinations and are a frequent error type in learner text.

We have created a dataset of ANs, where the combinations are extracted from learner texts and manually error-coded using a novel annotation scheme. This scheme is motivated by observations about typical learner confusions in the choice of adjectives and nouns – for example, semantically-related or form-related confusions. Since errors in content words are related to semantics, we derive semantically-motivated features through models of compositional distributional semantics and use these features for error detection. We treat error detection as a binary classification task, following the usual convention in EDC.

The original contributions of this paper are that we:

- present and release an error-annotated AN dataset extracted from learner data;
- show how compositional distributional semantic models can be applied to detect semantic anomalies in this dataset;
- demonstrate that the output of these models can be used to derive features for error detection in AN combinations.

2 Previous work

2.1 Error Detection in Content Words

Previous work on EDC for content words has either focused on correction alone assuming that errors are already detected (Liu et al., 2009; Dahlmeier and Ng, 2011), or has reformulated the task as *writing improvement* (Shei and Pain, 2000; Wible et al., 2003; Chang et al., 2008; Futagi et al., 2008; Park et al., 2008; Yi et al., 2008; Östling and Knutsson, 2009).

In the first case, the task is reduced to the search for the most suitable correction among the alternatives typically composed of synonyms, homophones or L1-related paraphrases (Dahlmeier and Ng, 2011), while the more challenging error detection step is omitted. In the second case, error detection is integrated into suggestion of alternatives and their comparison to the originally used word combination according to some metric of collocational strength. Such approaches aim to improve the fluency of non-native texts by correcting erroneous idioms or collocations, where low frequency or low collocational strength clearly signifies an error.

These approaches might be useful for correcting collocations, but they are less suitable for error detection in free word combinations. As they compare original word combinations to their alternatives using corpus statistics, they are not applicable to unseen word combinations, while learner texts contain many previously unseen combinations, not all of which are errors. Moreover, some word combinations may be correct even though less fluent than some of their alternatives. For example, *appropriate concern*, though it is correct, would have lower collocational strength than its alternative *proper concern*, and would, according to this approach, be tagged as an error. From the educational point of view, tagging an acceptable combination as an error is misleading for language learners and should be avoided.

We implement a baseline model inspired by such comparison-based approaches and demonstrate that it cannot be usefully applied to error detection in content word combinations. Then we present an approach that is capable of dealing with unseen data and does not rely on direct corpus-based comparison.

2.2 Semantic Anomaly Detection

Learner errors in content words often result from a semantic mismatch between the chosen words. A similar problem of semantic anomaly detection in content word combinations has been addressed with compositional distributional semantic models.

These models are based on distributional representations for words which are then composed to derive phrase representations. They rely on the assumption that a word meaning can be approximated by its distribution across its contexts of use. Words are represented as vectors in a high-dimensional space with each dimension encoding a word's co-occurrence with one of its contextual elements. Distributional models are less suitable for representing content word combinations directly since these will be very sparse and will often remain unattested even in an extremely large corpus.

A promising solution is provided by compositional distributional semantic models, which combine distributional vectors for the component words using some function over such vectors. Compositional distributional semantic representations have been previously used to detect semantic anomaly in AN combinations (Vecchi et al., 2011). Vecchi *et al.* have applied the *additive* and *multiplicative* models of Mitchell and Lapata (2008) and *adjective-specific linear maps* of Baroni and Zamparelli (2010) to a set of corpus-unattested ANs. They show that there is a distinguishable difference in the compositional semantic representations for the semantically acceptable and anomalous combinations, suggesting that compositional distributional models can be used to detect semantic anomaly without relying directly on corpus statistics.

Kochmar and Briscoe (2013) have applied the same models of semantic composition to distinguish between correct and incorrect ANs extracted from learner texts. Their results support the assumption that there is a distinguishable difference between the composite vectors for the correct and incorrect ANs, but they did not address the question of how to integrate these semantic models into an error detection system.

Recent work by Lazaridou *et al.* (2013) has shown that measures used for quantifying the degree of semantic anomaly in phrases derived from their compositional distributional semantic representations can be used as features by a classifier to help resolve syntactic ambiguities.

Our goals are to test, using a new and larger AN dataset, whether semantic models can distinguish between correct and incorrect AN combinations, which cannot be dealt with using simpler error detection approaches, and to implement an error detection system using these semantically-based features.

3 Data Annotation

We present and release a dataset of AN combinations which, on the one hand, exemplify the typical errors committed by language learners in the choice of content words within such combinations, and, on the other hand, are challenging for an EDC system.

For that, we examined the publicly available CLC-FCE dataset (Yannakoudakis et al., 2011), used the error annotation (Nicholls, 2003), and analysed the typical errors in AN combinations committed by language learners. We have compiled a list of 61 adjectives that are most problematic for learners.

Most typically, learners confuse semantically related words: for example, they are unable to distinguish between synonyms, near-synonyms or co-hyponyms and choose an appropriate one from the set. Our list of adjectives contains some frequent ones that are confused with each other due to their similarity in meaning. For example, the adjectives within the set {*big, large, great*} are frequently confused with each other as in:

(5) *big*/large* quantity

(6) *big*/great* importance

Another common source of error related to the high-frequency adjectives involves using them instead of more specific ones: in such cases, learners are unable to distinguish between the more specific terms and they choose the most frequent adjective, usually encompassing a variety of related meanings, to represent the whole class of similar words. For example, adjectives *big* and *large* encompass a variety of meanings including those of *high*, *wide* or *broad*. As learners often lack intuitions about which of these more specific adjectives should be chosen, they use the ones with more general meaning. This results in errors like:

(7) *big*/long* history

(8) *bigger*/wider* variety

(9) *greatest*/highest* revenue

(10) *large*/broad* knowledge

The reverse of this – an incorrect selection of a more specific term instead of the more general one – also leads to learner errors.

Form-related confusions represent another typical source of learner errors, and we have included pairs of adjectives such as *classic* and *classical*, *economic* and *economical* and the like in our dataset:

(11) *classic*/classical* dance

(12) *economical*/economic* crisis

Using this set of 61 adjectives, we have extracted AN combinations from the Cambridge Learner Corpus (CLC),[1] a large corpus of texts produced by English language learners, sitting Cambridge Assessment's examinations.[2] We have focused on AN combinations previously unseen in a native English corpus, as we hypothesise that they would have a higher chance of containing an error. Such combinations are more challenging for EDC algorithms since:

- these ANs cannot be effectively handled with simple comparison-based approaches like the ones overviewed in section 2.1;
- language learners are creative in their writing, so there is a substantial number of such previously unseen combinations;
- as no corpus could cover all possible acceptable content word combinations in language, the fact that these combinations are not seen in the corpus cannot be used as definitive evidence of incorrectness.

To summarise, it is important for an EDC algorithm to handle such combinations, but their absence in a native corpus of English makes it impossible to rely on simpler approaches and suggests that semantic analysis of such combinations would be more effective. In our research, we used the British National Corpus (BNC)[3] to select the corpus-unattested combinations.

We have compiled a set of 798 AN combinations.[4] An annotation scheme has been devised to annotate these examples as correct or incorrect, and for the incorrect combinations, to identify the locus of error (adjective, noun or both) and the type of confusion (incorrect synonym, form-related word, or non-related word). The most appropriate corrections are included in the dataset.

We also distinguish between *out-of-context* (*OOC*) and *in-context* (*IC*) annotation. The motivation behind this distinction is as follows: some combinations may appear to be correct when considered out of their original context of use, because there might be other contexts where the same combination would be appropriate. For example, *classic dance* is annotated as correct out of context because one could imagine using it in a context where it would denote some typical dance like:

(13) They performed a *classic Ceilidh dance*.

However, in practice, the AN *classical dance* is used much more frequently, and *classic dance* is most often errorful in context, as in (4) above.

Some ANs in our dataset are represented with more than one context of use, and in that case the *in-context* annotation can be conditioned on each context, or used to derive the most typical annotation for the AN. Both types of information are useful, as EDC systems which make use of the surrounding context should rely on the annotation in each particular context of use and, for example, be able to detect

[1] `http://www.cup.cam.ac.uk/gb/elt/catalogue/subject/custom/item3646603/Cambridge-International-Corpus-Cambridge-Learner-Corpus/`

[2] `http://www.cambridgeenglish.org`

[3] `http://www.natcorp.ox.ac.uk/`

[4] This dataset is released and publicly-available at `http://www.ilexir.com/`

Type	Cor.	Incor.	LB	UB
OOC	633	165	0.7932	0.8650
IC	394	404	0.5063	0.7467

Table 1: Distribution of correct (cor.) and incorrect (incor.) ANs in the dataset.

that *classic dance* is correct in one specific context, while in others it is incorrect. EDC systems that do not make use of the context can simply rely on the most frequent *in-context* annotation and detect that *classic dance* is typically an error in learner writing.

To create the two-level annotation, the annotators were first presented with an AN combination and asked to tag each word as correct or incorrect depending on whether they can think of some appropriate contexts of use for it. Next, the same combination was presented in its context of use from the CLC and the annotators were asked to annotate it with respect to its context.

The dataset was primarily annotated by a professional linguist. To ensure that the annotation scheme is clear and efficient, the dataset was split into 100 and 698 ANs, and the 100 ANs were first annotated by the same professional annotator and three other annotators. We have measured the inter-annotator agreement for the two levels of annotation using the mean values for the observed agreement within each pair of annotators, as well as mean Cohen's *kappa* value (Cohen, 1960). In Table 1 we report the mean inter-annotator agreement for the correct versus incorrect combinations at the two annotation levels, which represents the upper bound (*UB*) in our experiments. We have obtained the mean *kappa* values of 0.65 and 0.49 at the two levels of annotation, which are interpreted as substantial and medium agreement between annotators and confirm that the annotation scheme is clear.[5] Table 1 presents the distribution of ANs and the majority class baseline which we further use as a lower bound (*LB*).

4 Semantic Models for Error Detection

We replicate the semantic approaches, which have previously shown promising results in detecting semantic anomaly and content word errors (Vecchi et al., 2011; Kochmar and Briscoe, 2013), and test their performance on our dataset of corpus-unattested correct and incorrect AN combinations.

4.1 Experimental Setting

We use the *additive* (*add*) and *multiplicative* (*mult*) models of Mitchell and Lapata (2008), and the *adjective-specific linear maps* (*alm*) of Baroni and Zamparelli (2010).

The first two models derive the composite phrase vector through addition and multiplication of the components of the word vectors. These models have a clear mathematical interpretation and require no training. Their principal weakness is that they are symmetric, and fail to represent the difference in grammatical function of the component words. The *alm* model provides a theoretically more appropriate way of representing ANs based on this asymmetry: nouns are represented by their distributional vectors, while attributive adjectives are functions mapping from noun meanings to a composite noun-like vector for the ANs. Adjectives are represented as weight matrices which are learned from corpus-attested examples of noun–AN mappings, and composition is defined by matrix-by-vector multiplication.

We use the experimental setting previously described (Vecchi et al., 2011; Kochmar and Briscoe, 2013) and populate the semantic space with the constituent nouns and adjectives from the test ANs, frequent nouns and adjectives from the BNC and the AN combinations containing these frequent words. We use about $8K$ nouns, $4K$ adjectives and $64K$ ANs following Kochmar and Briscoe (2013). The semantic space is represented by a matrix encoding word co-occurrences, where the rows represent the $76K$ elements mentioned above, and the columns represent a selected set of $10K$ context elements. The $10K$ context elements include the most frequent nouns, adjectives and verbs from the corpus. The word co-occurrence counts are estimated using the BNC. The corpora have been lemmatized, tagged and parsed with the RASP system (Briscoe et al., 2006; Andersen et al., 2008; Yannakoudakis et al., 2011), and all statistics are extracted at the lemma level.

[5] Further details of the annotation experiment are described in the dataset release.

We transform the raw sentence-internal co-occurrence counts into Local Mutual Information scores (Baroni and Zamparelli, 2010; Evert, 2005), and perform dimensionality reduction applying Singular Value Decomposition to the noun and adjective matrix rows, projecting the AN rows onto the same reduced space following Baroni and Zamparelli (2010). The original $76K \times 10K$ matrix is reduced to a $76K \times 300$ matrix. This allows us to perform training and other calculations in the semantic space more efficiently.

The weight coefficients for the *alm* model are estimated with multivariate partial least squares regression using the RPLS package (Mevik and Wehrens, 2007). The weight matrix is learned for each adjective separately.

4.2 Semantic Cues

In previous work (Vecchi et al., 2011; Kochmar and Briscoe, 2013) several semantic measures for detecting semantic anomaly have been introduced. We reimplement these measures (1 to 8), but also test some additional measures (9 to 13) that we hypothesise can also help distinguish between correct and incorrect word combinations:

1. **Vector length (*VLen*)**: vectors for correct and incorrect combinations may differ with respect to their length, and the latter are expected to be shorter;

2. **Cosine to the input noun (*cosN*)**: the distance between the model-generated AN vector and the input noun vector is expected to be greater for the incorrect combinations, as the noun meaning is typically ‘distorted’;

3. **Cosine to the input adjective (*cosA*)**: analogical to *cosN* measure, the adjective meaning might be ‘distorted’ as well, especially as two of the composition functions are symmetric;

4. **Density of the neighbourhood populated by** 10 **nearest neighbours (*dens*)** is calculated as the average distance from the model-generated vector to the 10 nearest neighbours in the original semantic space, and is expected to be higher for the correct ANs;

5. **Density among the** 10 **nearest neighbours (*densAll*)** is a modification of *dens*, which is estimated as an average for the 11 density values calculated for each member within the set consisting of the AN vector and its 10 neighbours;

6. **Ranked density in close proximity (*Rdens*)** relies on the notion of *close proximity*, which is defined as a neighbourhood populated by some very close neighbours (for example, within a distance of $\geq$ 0.8). It is calculated as: $RDens = \sum_{i=1}^{N} rank_i distance_i$ with N being the total number of close neighbours within close proximity, each with its rank and distance;

7. **Number of neighbours within close proximity (*num*)** is used as another measure, and is assumed to be lower for incorrect combinations, which are expected to be more isolated in the semantic space;

8. **Overlap between the** 10 **nearest neighbours and constituent noun/adjective (*OverAN*)** assumes correct ANs should be surrounded by similar words and combinations. It is calculated as the proportion of the 10 nearest neighbours containing the same constituent words as in the tested ANs;

9. **Overlap between the** 10 **nearest neighbours and input noun (*OverN*)** is a variant of the *OverAN* with only the noun considered;

10. **Overlap between the** 10 **nearest neighbours and input adjective (*OverA*)** is a variant of the *OverAN* with only the adjective considered;

11. **Overlap between the** 10 **nearest neighbours for the AN and constituent noun/adjective (*NOverAN*)** assumes that correct ANs and their constituent words should be placed in similar neighbourhoods. It is calculated as the proportion of the common neighbours among the 10 nearest neighbours for the model-generated AN and the constituent words;

Metric	*add*	*mult*	*alm*
VLen	0.7589	0.7690	0.1676
cosN	0.1621	**0.0248**	**0.0227**
cosA	**0.0029**	0.4782	0.0921
dens	0.6731	0.1182	0.1024
densAll	0.4967	0.1026	0.1176
RDens	0.2786	0.8754	0.1970
num	0.3132	0.4673	0.3765
OverAN	0.8529	0.1622	0.2808
OverA	**0.0151**	0.6377	0.4886
OverN	**0.0138**	0.0764	0.4118
NOverAN	0.3941	0.6730	0.0858
NOverA	**0.0009**	0.3342	0.1575
NOverN	**0.0018**	0.1463	0.1497

Table 2: p values, *out-of-context* annotation

Metric	*add*	*mult*	*alm*
VLen	0.6675	**0.0027**	**0.0111**
cosN	**0.0417**	**0.0070**	0.1845
cosA	**0.00003**	0.1791	0.1442
dens	0.4756	0.7120	0.1278
densAll	0.2262	0.7139	0.5310
RDens	0.8934	0.8664	0.1985
num	0.7077	0.7415	0.4259
OverAN	0.1962	0.8635	0.5669
OverA	**0.00007**	0.7271	0.6229
OverN	**0.0017**	0.9680	0.7733
NOverAN	**0.0227**	0.3473	0.1587
NOverA	**0.000004**	0.3749	0.1576
NOverN	**0.0001**	0.6651	0.2610

Table 3: p values, *in-context* annotation

12. **Overlap between the** 10 **nearest neighbours for the AN and input noun (*NOverN*)** is a variant of the *NOverAN* with only the noun considered;

13. **Overlap between the** 10 **nearest neighbours for the AN and input adjective (*NOverA*)** is a variant of the *NOverAN* with only the adjective considered.

4.3 Results

We evaluate the models and report the results following the procedure that has been used before in Vecchi *et al.* (2011) and Kochmar and Briscoe (2013). For each model and semantic measure, we report the p value denoting statistical significance of the difference between the groups of correct and incorrect ANs. The statistical significance is reported at the $p<0.05$ level, and if a measure applied to the two groups of ANs shows statistically significant difference we interpret that as an ability of this measure to distinguish the correct ANs from the incorrect ones in general. The results for the out-of-context annotation are reported in Table 2, and those for the in-context annotation in Table 3.

The results show that the difference between the vector representations for the correct and incorrect AN combinations can be reliably detected with a number of the proposed measures. Measures which show statistically significant results with at least one model are marked in bold. These results also suggest that the values for the semantic measures can be used to derive discriminative features for a classifier.

5 Error Detection as Classification Task

5.1 Baseline System

We implement a simple comparison-based baseline system inspired by previous work on error detection in content words (see section 2.1). For every AN, we create a set of possible alternatives crossing the confusion set for the adjective with that for the noun, and compare the collocational strength of the original combination with that for each of the alternatives. If an alternative has higher collocational strength than the original combination, the original combination is tagged as an error and the alternative is chosen as a correction. Since semantically related confusions are a rich source of learner errors in content word combinations, we include adjective synonyms in the confusion set for an adjective, and noun synonyms and hyponyms in the confusion set for a noun. All synonyms and hyponyms are retrieved using WordNet 3.0 without word sense disambiguation.

We measure collocational strength using *normalized pointwise mutual information (npmi)* of the adjective a and noun n, which is defined as:

$$npmi(a,n) = \frac{pmi(a,n)}{-log[p(a,n)]} \quad (1) \qquad pmi(a,n) = log\frac{p(a,n)}{p(a)p(n)} \quad (2)$$

All probabilities are estimated from the BNC. This approach performs poorly on the unseen ANs in our dataset, since any alternative AN seen in the BNC would be preferred by this system over the original unseen AN. This ensures that less fluent (in this case, unseen) word combinations are substituted with more fluent (seen) ones. As a result, even though an original AN *important conversation* in our dataset is correct, it is still "corrected" by this system to *serious conversation*. At the same time, some incorrect combinations are not recognised if no appropriate alternative is found (e.g., **high shyness*). It shows that this approach lacks deeper semantic analysis and is also too dependent on the set of alternatives found for a word combination.

We measure *accuracy (acc)* as the proportion of true positives (*TP*) and true negatives (*TN*) to the total number of test items:

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

Accuracy reflects how often an error detection system correctly identifies that an AN is correct or incorrect. We compare the results to the lower and upper bounds set as the majority class distribution and inter-annotator agreement, respectively (see section 3).

With this approach we get quite low accuracy of 0.3897 on the out-of-context annotation since most of the test items are correct out of context (LB=0.7932), and the baseline system overcorrects many of those. Accuracy of the baseline system on the in-context annotation is 0.5147, which is slightly above the lower bound of 0.5063. These results are used as a baseline and included in Table 4.

Type	Accuracy	Baseline	LB	UB
OOC	**0.8113** ± 0.0149	0.3897	0.7932	0.8650
IC	**0.6535** ± 0.0189	0.5147	0.5063	0.7467

Table 4: *Decision Tree* classification results

Type	P (correct)	P (incorrect)
OOC	0.8193	0.7500
IC	0.6241	0.6850

Table 5: Classification precision

5.2 Classification

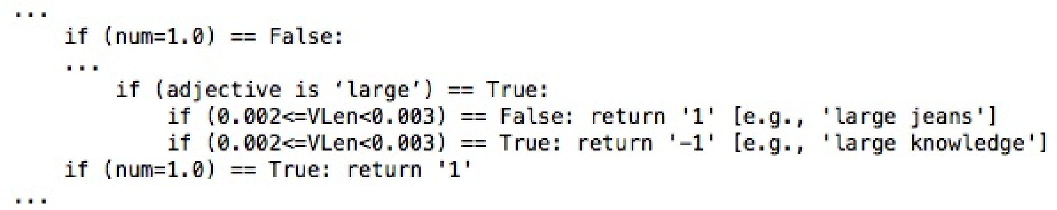
We implement a supervised classifier which uses output of the semantic models as features. We have tested a number of classifier models but the best results so far have been obtained with the *Decision Tree* classifier using $NLTK$ (Bird et al., 2009). We assume that this classifier effectively learns the inter-dependencies between the features within the small feature set that we use in our experiments. We use feature binning where the whole range of feature values is divided into 10 bins according to the distribution of values for each feature. This feature representation technique combined with the classifier helps generalise over feature values, reducing feature space dimensionality. The order of the feature application to the data is determined by the classifier on the basis of the information gain for the features and their values.

We apply 5-fold cross-validation and report average accuracy over the folds. The 798 ANs are split into 5 subsets with 80% in each of the splits used for training and 20% for testing. We keep the AN error rate in the training and test sets, as well as for each adjective, approximately the same across the splits to avoid any bias. Error detection is cast as a binary classification task. The output of the semantic models is used to derive numeric features for the classifier. Most values are in the range of $[0, 1]$, and we apply normalisation to *VLen*, *RDens* and *num* which originally have a different range.

The full feature set contains 14 features, with 13 features derived from the semantic measures, and 1 feature representing adjective identity. We hypothesise that introduction of this feature might help classifier learn that, for example, an AN containing an adjective *classic* has a higher chance of being incorrect, as most of the ANs with this adjective in the learner data are incorrect and involve confusions with *classical*. We also hypothesise that it facilitates learning correlations between the adjective and other feature values: it might be the case that ANs with an adjective *adj*$_1$, on the average, have higher *cosN* values than ANs with an adjective *adj*$_2$. This feature helps the classifier establish such dependencies between the adjective and the values of the semantic measures. For instance, in our data ANs with the adjective *true* have significantly higher cosine between AN vectors and vectors for their constituent nouns than ANs with the adjective *false*: this is in accordance with an intuition that, for example, *true happiness* is more similar to *happiness* than *false happiness* is.

The best results in our experiments have been obtained with the *mult* model. We have performed ablation tests incrementally removing features that did not improve classifier performance in order to find an optimal feature set. The best-performing feature set we found for the *mult* model on the out-of-context annotation uses *adjective*, *cosN* and *RDens* features, while for the in-context annotation the best-performing feature set found uses a combination of features including *adjective*, *VLen*, *densAll*, *NOverA*, *NOverN*, *RDens* and *num* features.

We note that the sets of best performing features in the classification experiments do not coincide with the semantic measures that showed the highest statistically significant difference (Tables 2 and 3). We conclude that although the p values reported in Tables 2 and 3 show that some semantic measures can distinguish one group of ANs from another on the basis of the statistically significant difference between the means of the two groups, when the measures are used as features for a classifier the results depend on how these features interact with each other as well as on their individual discriminativeness across the test dataset. For example, Figure 1 illustrates a small part of the decision tree constructed using the best performing feature set on the in-context annotation:



```
...
    if (num=1.0) == False:
    ...
        if (adjective is ‘large’) == True:
            if (0.002<=VLen<0.003) == False: return '1' [e.g., 'large jeans']
            if (0.002<=VLen<0.003) == True: return '-1' [e.g., 'large knowledge']
    if (num=1.0) == True: return '1'
...
```

Figure 1: *Decision Tree* classifier pseudocode.

Figure 1 shows how interaction of feature values for *num* and *VLen* in combination with the adjective identity feature can help classify the two ANs containing adjective *large* as correct (`1`) or incorrect (`-1`).

In Table 4 we report results for the *out-of-context* (*OOC*) and *in-context* (*IC*) annotation. The accuracy is reported with its mean ± standard deviation over the 5 data splits. We compare the *Decision Tree* classifier results to those obtained with the baseline system, as well as to the lower and upper bounds set as before (see section 3). The results show that a classifier that uses output of the semantic models as features outperforms the comparison-based baseline system by a large margin.

6 Discussion

In the previous section, we showed that a classifier that uses output of the semantic models as features outperforms the comparison-based baseline system and shows good accuracy. In this section, we analyse the classifier's performance in more detail.

We note that, from an educational point of view, it is important for an EDC system to have high precision. For example, it has been shown that grammatical error detection systems with high precision maximize learning effect, and that systems with high precision but lower recall are more useful in language learning than systems with high recall and lower precision (Nagata and Nakatani, 2010). This suggests that learners might be misled and confused if they are frequently notified by a system that something is an error when it is not.

Since precision is measured as the proportion of true positives (*TP*) to the sum of true positives and false positives (*FP*):

$$P = \frac{TP}{TP + FP} \quad (4)$$

an EDC system that achieves precision less than 0.5 is, in fact, misleading for language learners: for example, precision of less than 0.5 on the class of errors means that the system misidentifies correct use as an error more frequently than it correctly detects an error.

Our classifier achieves good precision values with respect to both out-of-context and in-context annotations, on correct and incorrect examples. Precision (P) values are reported in Table 5. As precision figures are higher than 0.5 in each case, it shows that the implemented error detection system would, on balance, help guide a learner to text regions in need of reformulation.

With respect to the out-of-context annotation, the error detection system has good precision and recall on correct examples ($P = 0.8193$, $R = 0.9762$). Precision on the incorrect examples is also high ($P = 0.7500$). This is a very encouraging result, suggesting the system would rarely misidentify an originally correct AN combination as an error.

For the in-context annotation, both precision and recall on correct and incorrect examples are quite high: $P = 0.6241$ and $R = 0.7169$ on the correct examples, and $P = 0.6850$ and $R = 0.5849$ on the incorrect examples.

Error analysis on the classifier's output shows that the majority of the incorrect examples misclassified as correct (*missed errors*) contain semantically-related confusions. It appears that the classifier relying on semantically-motivated features misses a number of cases where the original AN and its correction are semantically similar: for example, it misses the errors in *big*/great anger*, *biggest*/greatest painter* and *small*/short speech*. Since the ANs in these pairs are semantically similar, the features based on their semantic representations might not be discriminative enough. In contrast, the classifier is more effective in detecting errors in cases where the original AN and its correction are only similar in form, or not related to each other.

7 Conclusion

We have presented and released a dataset of learner errors in ANs, which has been extracted from learner texts and annotated with error types and corrections. The dataset contains examples not seen in a native corpus of English, and error annotation shows that a substantial number of such examples are correct. Error detection in this dataset is a challenging task, since absence of the ANs in a corpus of English cannot be used as definitive evidence of incorrectness. We have implemented a simple baseline system inspired by previous work on improving content word combinations and shown that such a system would not be effective for error detection in our dataset.

We have cast error detection as a binary classification task and implemented a supervised classifier that uses semantically-motivated features. The features are derived from the compositional distributional semantic representations of the AN combinations. We use a number of semantic measures that describe and distinguish between semantic representations for correct and incorrect combinations. We have introduced new semantic measures in addition to the ones used in previous work and show that they can be effectively applied to this task.

The best results in our experiments are obtained with a *Decision Tree* classifier, and we show that the resulting error detection system can identify errors with high precision and accuracy. We aim to extend this system to perform error correction on ANs, as well as error detection and correction on other types of content word combinations.

Acknowledgments

We are grateful to Cambridge English Language Assessment and Cambridge University Press for supporting this research and for granting us access to the CLC for research purposes. We would like to thank Øistein Andersen for providing us with the annotation tool, Diane Nicholls for undertaking the bulk of the annotation work, and Helen Yannakoudakis and the anonymous reviewers for their valuable comments.

References

Øistein Andersen, Julien Nioche, Ted Briscoe and John Carroll 2008. *The BNC parsed with RASP4UIMA*. In Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC), pp. 865–869.

Marco Baroni and Roberto Zamparelli 2010. *Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space*. In Proceedings of the EMNLP-2010, pp. 1183–1193.

Steven Bird, Ewan Klein, and Edward Loper 2009. *Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

Ted Briscoe, John Carroll and Rebecca Watson 2006. *The Second Release of the RASP System*. In Proceedings of the COLING/ACL-2006 Interactive Presentation Sessions, pp. 59–68.

Yu-Chia Chang, Jason S. Chang, Hao-Jan Chen and Hsien-Chin Liou 2008. *An automatic collocation writing assistant for Taiwanese EFL learners: A case of corpus-based NLP technology*. Computer Assisted Language Learning, 21(3), pp. 283–299.

Jacob Cohen 1960. *A coefficient of agreement for nominal scales*. Educational and Psychological Measurement, 20(1), pp. 37–46.

Robert Dale, Ilya Anisimoff and George Narroway 2012. *HOO 2012: A Report on the Preposition and Determiner Error Correction Shared Task*. In Proceedings of the 7th Workshop on Innovative Use of NLP for Building Educational Applications, pp. 54–62.

Daniel Dahlmeier and Hwee Tou Ng 2011. *Correcting Semantic Collocation Errors with L1-induced Paraphrases*. In Proceedings of the EMNLP-2011, pp. 107–117.

Gerard M. Dalgish 1985. *Computer-assisted ESL research*. In CALICO Journal, 2(2), pp. 32–37.

Stefan Evert 2005. *The Statistics of Word Cooccurrences*. Dissertation, Stuttgart University.

Yoko Futagi, Paul Deane, Martin Chodorow and Joel Tetreault 2009. *A computational approach to detecting collocation errors in the writing of non-native speakers of English*. Computer Assisted Language Learning, 21(4), pp. 353–367.

Ekaterina Kochmar and Ted Briscoe 2013. *Capturing Anomalies in the Choice of Content Words in Compositional Distributional Semantic Space*. In Proceedings of the Recent Advances in Natural Language Processing (RANLP-2013).

Angeliki Lazaridou, Eva Maria Vecchi and Marco Baroni 2013. *Fish transporters and miracle homes: How compositional distributional semantics can help NP parsing*. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1908–1913.

Claudia Leacock, Martin Chodorow, Michael Gamon and Joel Tetreault 2010. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers.

Anne Li-E Liu, David Wible and Nai-Lung Tsao 2009. *Automated suggestions for miscollocations*. In Proceedings of the 4th Workshop on Innovative Use of NLP for Building Educational Applications, pp. 47–50.

Bjørn-Helge Mevik and Ron Wehrens 2007. *The pls package: Principal component and partial least squares regression in R*. Journal of Statistical Software, 18(2), pp. 1–24.

Jeff Mitchell and Mirella Lapata 2008. *Vector-based models of semantic composition*. In Proceedings of ACL, pp. 236–244.

Jeff Mitchell and Mirella Lapata 2010. *Composition in distributional models of semantics*. Cognitive Science, 34, pp. 1388–1429.

Ryo Nagata and Kazuhide Nakatani 2010. *Evaluating performance of grammatical error detection to maximize learning effect*. In Proceedings of COLING 2010, pp. 894–900.

Diane Nicholls 2003. *The Cambridge Learner Corpus: Error coding and analysis for lexicography and ELT*. In Proceedings of the Corpus Linguistics conference, pp. 572–581.

Robert Östling and Ola Knutsson 2009. *A corpus-based tool for helping writers with Swedish collocations*. In Proceedings of the Workshop on Extracting and Using Constructions in NLP, pp. 28–33.

Taehyun Park, Edward Lank, Pascal Poupart, Michael Terry 2008. *Is the sky pure today? AwkChecker: an assistive tool for detecting and correcting collocation errors*. In Proceedings of the 21st annual ACM symposium on User interface software and technology, pp. 121–130.

Alla Rozovskaya and Dan Roth 2011. *Algorithm Selection and Model Adaptation for ESL Correction Tasks*. InProceeding of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies – Volume 1, pp. 924–933.

Chi-Chiang Shei and Helen Pain 2000. *An ESL Writer's Collocation Aid*. Computer Assisted Language Learning, 13(2), pp. 167–182.

Eva Maria Vecchi, Marco Baroni and Roberto Zamparelli 2011. *(Linear) maps of the impossible: Capturing semantic anomalies in distributional space*. In Proceedings of the DISCO Workshop at ACL-2011, pp. 1–9.

David Wible, Chin-Hwa Kuo, Nai-Lung Tsao, Anne Liu and H.-L. Lin 2003. *Bootstrapping in a language-learning environment*. Journal of Computer Assisted Learning, 19(4), pp. 90–102.

Helen Yannakoudakis, Ted Briscoe and Ben Medlock 2011. *A New Dataset and Method for Automatically Grading ESOL Texts*. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 1, pp. 180–189.

Xing Yi, Jianfeng Gao and William B. Dolan 2008. *A Web-based English Proofing System for English as a Second Language Users*. In Proceedings of the third International Joint Conference on Natural Language Processing (IJCNLP), pp. 619–624.