Measures to Detect Word Substitution in Intercepted Communication

SW. Fong[1], D.B. Skillicorn[1], D. Roussinov[2]

[1] School of Computing,
Queen's University
[2] W.P. Carey School of Business,
Arizona State University

Abstract. Those who want to conceal the content of their communications can do so by replacing words that might trigger attention by other words or locutions that seem more ordinary. We address the problem of discovering such substitutions when the original and substitute words have the same natural frequency. We construct a number of measures, all of which search for local discontinuities in properties such as string and bag-of-words frequency. Each of these measures individually is a weak detector. However, we show that combining them produces a detector that is reasonably effective.

1 Motivation

Terrorists and criminals must be aware of the possibility of interception whenever they communicate by phone or email. In particular, terrorists must be aware of systems such as Echelon [3] that examine a very large number of messages and select some for further analysis based on a watchlist of significant words.

Given that it may not be possible to evade some examination of their messages, terrorists and criminals have two defensive strategies: encryption and obfuscation. The problems with encryption are that it draws immediate attention to messages and so permits at least meta-analysis; and it may be that there are backdoors to commonly available encryption methods. Obfuscation tries to hide messages in the background of the vast number of other messages, replacing words that might trigger attention by other innocent-sounding words or locutions. For example, al Qaeda, for a time, used the word 'wedding' to mean 'attack'.

When a word is replaced by a word of substantially different natural frequency, Skillicorn [10] showed that a different kind of potentially detectable signature is created. This is because most collections of messages represent an agglomeration of conversations, and conversations are always about something. Rare topics only appear in rare conversations. When a word substitution occurs, a rare topic begins to appear more frequently than it 'should'. An increase in both the frequency difference between original and substituted words and the frequency of messages that contain the substitution both increase the detectability of the presence of a substitution.

However, substitution by a word of approximately similar frequency is possible, given either a predefined codebook or access to a frequency-ranked word list (on the internet perhaps). In this paper, we address the detection of messages in which a word has been replaced by a word of similar frequency.

Consider the sentence "the attack will be tomorrow". Using the al Qaeda substitution, we get "the wedding will be tomorrow" which is designedly a natural-sounding sentence. However, 'attack' is the 1072nd most common English word according to the site www.wordcount.org/main.php, while 'wedding' is the 2912th most common, so the substantial frequency difference might make this substitution detectable using the approach described above. On the other hand, if the word 'attack' is replaced by the word 'complex' which has similar frequency, than any human will be able to detect that the sentence "the complex will be tomorrow" is extremely unusual. However, detecting this kind of substitution automatically using software has not been attempted, except in a very preliminary way [4].

The contribution of this paper is to show that a number of techniques using only syntactic properties such as word frequencies can detect such word substitutions, although only weakly. Most techniques are either good at detecting substitutions with a high false positive rate, or have a low false positive rate but do not detect substitutions well. However, combining the best of these techniques produces a detector whose detection rate, on an individual sentence basis, is close to 82% with a false positive rate of only 20%.

2 Related Work

The problem of detecting a word that is somehow out of context occurs in a number of settings. For example, speech recognition algorithms model the expected next word, and back up to a different interpretation when the next word becomes sufficiently unlikely [1]. This problem differs from the problem addressed here because of the strong left context that is used to decide on how unlikely the next word is, and the limited amount of resources that can be applied to detection because of the near-realtime performance requirement.

Detecting words out of context can also be used to detect (and correct) misspellings [5]. This problem differs from the problem addressed here because the misspelled words are nonsense, and often nonsense predictably transformed from the correctly spelled word, for example by letter reversal.

Detecting words out of context has also been applied to the problem of spam detection. For example, SpamAssassin uses rules that will detect words such as 'V!agra'. The problem is similar to detecting misspellings, except that the transformations have properties that preserve certain visual qualities rather than reflecting lexical formation errors. Lee and Ng [7] detect word-level manipulations typical of spam using Hidden Markov Models. As part of this work, they address the question of whether an email contains examples of obfuscation at all. They expected this to be simpler than the problem they set out to address – recovering the text that had been obfuscated – but remark that detecting obfuscation at

all is 'surprisingly difficult' [7, Section 5] and achieve prediction accuracies of around 70% using word-level features.

The task of detecting replacements can be considered as the task of detecting words that are "out of context," which means surrounded by the words with which they typically do not co-occur. The task of detecting typical co-occurrences of words in the specific contexts was considered in [8, 9].

3 Strategies

We wish to detect places where word substitutions have occurred, without any access to direct semantic information. The techniques we use are all based on the intuition that a substitution creates a local 'bump' in the frequencies of substrings or sentences containing the substitute.

An obvious starting point might be the 2-gram (or n-gram) frequencies of adjacent pairs (or n-tuples) of words in the sentence. A 2-gram that contain the substituted word might have lower frequency than expected. There are two problems with this simple idea. First, what *is* the expected frequency, given that we don't know what the original word was? Second, Ferrer i Cancho and Solé [6] have shown that the graph of English word adjacencies has a small world property. In other words, most rare words are surrounded by common words, and the pairwise frequencies of pairs that include rare words do not differ much from the rare word single-word frequencies. This can be seen in the example sentence above: there is nothing unusual about the fragment "the complex will be"; it is not until the word 'tomorrow' is appended that the sentence becomes unusual. It is the pair of non-stopwords (complex, tomorrow) whose frequency is significant, and these may be separated by many stopwords.

We determine frequencies by querying large repositories such as Google. Such repositories implicitly contain information about the frequencies of fragments of text, of bags of words, and of sets of words with stopwords deleted. However, it is not always possible to get this implicit information directly, which forces us to use subtle measures to obtain the scores we want.

We concentrate on nouns, since these represent the most likely targets of substitution, there is more information available about their frequencies than about other parts of speech, and there are fewer variant forms in English than for verbs.

3.1 k-gram frequencies

In measuring a sentence for potential substitutions, we consider each noun in sequence. In our initial work we considered the region surrounding each noun extending to the left until the first non-stopword was encountered, and extending to the right until the first non-stopword was encountered. For example, in the sentence "A nine mile walk is no joke", the region surrounding 'walk' is "mile walk is no joke". However, we discovered that, in real, informal text, these regions are long enough that there are typically *no* instances of them, even at Google.

This is partly because they are long enough to capture author idiosyncrasies, partly because of the grammatical oddities of informal text, and partly because texts in limited domains also tend to use limited vocabulary, such as technical terms which are not well represented in general-purpose repositories.

However, what we call the *left k-gram*, the text from the considered noun leftwards up to and including the first non-stopword; and the *right k-gram*, the text from the considered noun rightwards up to and including the first non-stopword, seem to produce more useful fragments. In the example sentence above, the left k-gram of 'walk' is "mile walk" ($f = 50$ at Google) and the right k-gram is "walk is no joke" ($f = 876{,}000$). Intuitively, each of these fragments considered separately is more natural, and so more likely, than the complete k-gram above ($f = 33$). Surprisingly, the left and right k-grams detect substantially different properties of sentences, presumably because word order is important in English, both to convey meaning and style (observe the different frequencies above).

3.2 Sentence oddity

Sentence oddity measures are designed to measure the frequency of an entire sentence. Because most sentences do not appear verbatim even once in a large text repository, obtaining such frequencies comes at the expense of ignoring the order of the sentence words.

In general, if a word is discarded from a bag of words, the frequency of the smaller bag should be greater than that of the original bag. However, if the bag of words was a sentence with the word order ignored, and the discarded word was meaningful in the context of the sentence, then we might expect that the difference in frequency might be moderate. If the discarded word was not meaningful in the context of the sentence, then the difference in frequency might be much greater. Hence we define sentence oddity as:

$$\text{sentence oddity} = \frac{\text{frequency of bag of words with word discarded}}{\text{frequency of entire bag of words}}$$

The more unusual the discarded word was in the context of its sentence, the greater we expect the sentence oddity to be.

3.3 Semantic oddity

If a word is a substitution, then we expect that word not to fit into the context well. If the substituted word is, in turn, replaced by a related word, the frequency of the resulting sentence will change, and this change will reflect something about how unusual the original substitution was. This requires a way to find related words, which is fundamentally a semantic issue, but there are sources of such words, for example Wordnet.

The hypernym of a noun is the word immediately above it in the ordinary ontology of meanings; for example, the hypernym of 'car' is 'motor vehicle'. We had expected that, when a normal word is replaced by its hypernym, the

frequency of the resulting sentence would stay the same or increase; while when a substituted word is replaced by its hypernym the frequency of the resulting sentence would decrease.

This turns out to be exactly wrong – the actual behavior is the other way around, and considerably more subtle. The hypernym of a word has its own hypernym, and original word also has a hyponym, a more specialized word, so that there are a chain of hypernyms and hyponyms passing through any given noun. The place on this chain that best represents the entire chain is called the *class word*. An example of a chain is (from the bottom): "broodmare, mare, horse, equine, odd-toed ungulate, hoofed mammal, mammal, vertebrate". Here the class word is 'horse'. What happens when a word is replaced by its hypernym depends on where in such a chain the word appears. If the word is below the class word, then the hypernym is probably more common, and the frequency of the new sentence greater; if the word is above the class word, then the hypernym is probably more technical and less common, and the frequency of the new sentence is smaller. For example, the hypernym of 'rabbit' is the biological term 'leporid', which is unlikely to be used in ordinary sentences.

In fact, the chain of hypernyms for many words exhibits an oscillating structure, moving from technical terms to common terms and then back to technical terms, and so on. For example, a chain containing 'attack" is (from the bottom): "foray, incursion, attack, operation, activity, act, event" in which 'attack' and 'act' are simpler words than the others. Another chain is "comprehension, understanding, knowing, higher cognitive process, process, cognition", in which 'understanding', 'knowing, and 'process' are ordinary words while the other words in the chain are more technical.

In ordinary informal text, the nouns in use are likely to be close to the appropriate class words – using non-class words tends to sound pompous. Substitution by a hypernym is likely to produce a more technical sentence, with a lower frequency. If the noun under consideration is already a substitution, however, it is less likely to be a simple word. Substitution by a hypernym may produce a less technical sentence with a greater frequency. The chain containing 'complex' is: "hybrid, complex, whole, concept, idea, mental object". In our example sentence, "the complex is tomorrow", replacement produces "the whole is tomorrow" which is a much more common bag of words.

We define the *hypernym oddity* to be:

$$\text{hypernym oddity} \;=\; f_H - f$$

where f is the frequency of a sentence, regarded as a bag of words; and f_H is the frequency of a bag of words in which the noun under consideration has been replaced by its hypernym. We expect this measure to be close to zero or negative for ordinary sentences, but positive for sentences that contain a substitution.

These three strategies, looking for frequencies of exact substrings of the sentence under consideration, looking for changes in frequency between the entire sentence and the sentence without the word under consideration, and looking for changes in frequency when the word under consideration is replaced by its

hypernym (or other related words) can all suggest when a substitution has occurred. In the next section, we describe the exact measures we have used in our experiments.

4 Techniques

4.1 Usable frequency data

In order to be able to measure the frequencies of sentences, sentence fragments, and bags of words, we must use data about some repository of text. The choice of repository makes a great deal of difference, since the better the match between the repository and the style of text in which substitutions may have occurred, the more accurate the prediction of substitutions will be. It is well known, for example, that perplexity, which measures a one-sided 2-gram frequency, is considerably reduced in sets of documents from a particular domain.

We use Google as the source of frequency data, on the grounds that it indexes a very large number of English documents, and so provides a good picture of frequencies of English text. That said, it is surprising how often an apparently ordinary phrase occurs zero times in Google's document collection.

There are also particular idiosyncrasies of Google's techniques that have some impact on our results. First, the frequencies returned via the Google API and via the Google web interface are substantially different; the API frequency values are used in all programs here. Second, the Google index is updated every 10 days or so, but this is not easily detectable, so frequencies may be counted from different instantiations of the index (large frequencies are rounded so this makes little difference, except for rare strings). Third, the way Google handles stop words is not transparent, and makes it impossible to invoke exactly the searches we might have wished. For example, "chase the dog" occurs 9,580 times whereas "chase dog" occurs 709 times, so quoted string searches clearly do not ignore stopwords. On the other hand, the bag of words search {chase the dog} occurs 6,510,000 times while {chase dog} occurs only 6,490,000 times, which seems counterintuitive. Fourth, the order of words seems to be significant, even in bag-of-word searches. For example, searches for {natural language processing} and {natural processing language} consistently produce different frequencies.

We use the number of pages returned by Google as a surrogate for word frequency. This fails to take into account intraword frequencies within each individual document. It also fails to take into account whether two words appear, say, adjacently or at opposite ends of a given returned document, which we might expect to be relevant information about their relationship. We have experimented with using locality information of this kind, but it does not improve performance.

4.2 Usable semantic data

The only semantic information we use is the hypernyms of nouns being considered. We get this information from Wordnet (wordnet.princeton.edu). In general,

a word can have several hypernyms, so we collect the entire set and use them as described below. For example, the direct hypernyms of 'complex' are 'whole', 'compound', 'feeling', and 'structure', derived from the different meanings of 'complex'.

4.3 Experimental data

In order to evaluate measures to detect substitutions, we need sets of reasonable sentences to use as data. Standard grammatical sentences, for example from news articles, do not make good test data because the kinds of sentences intercepted from email and (even more so) from speech will not necessarily be complete or formal grammatical sentences.

A large set of emails was made public as the result of the prosecution of the Enron corporation. This set of emails was collected over three and a half years and contains emails from and to a large set of individuals who never imagined that they would be made public. This set of emails is therefore a good surrogate for the kinds of texts that might be collected by systems such as Echelon, and we use it as a source of informal, and so realistic, sentences.

Enron emails contain many strings that are not English words, for example words in other languages, acronyms, and highly technical terms relating to energy. We use the British National Corpus (BNC) [2] to discard strings that do not appear to be English words, and also as our source for the frequencies of English words.

We extracted all strings ending with periods as possible sentences, except when the BNC corpus indicated the possibility of periods as integral parts of words, e.g. 'Mr.'. Sentences with fewer than 5 words or more than 15 words were discarded, leaving a total of 712,662 candidate sentences. A random sample of 3000 sentences were drawn from this set.

We detected the first noun in each sentence, and replaced it with an adjacent word in the BNC frequency ranking for nouns. Sentences for which the selected noun either did not have a hypernym known to Wordnet, or occurred with zero frequency at Google were discarded.

The resulting set of sentences still contained sentences that did not make good test examples because they contained unusual word use (i.e., they were too informal), because they contained typos at the level of words, or because they used technical vocabulary for which Google frequencies were too low ($f < 10$) to be useful. To remove such sentences, we computed the sentence oddity for each original sentence and for the sentence derived from it by substitution. This measure should increase when a substitution is inserted; when it did not, we discarded the pair of sentences, since this means that the original sentence was *more* unusual that the one containing the substitution. This reduced the available set of sentences by approximately a further 25%. Of course, this means that the set of sentences is biased towards successful detection using sentence oddity, so the further results using this measure are included for interest only.

Our test set is therefore a set of 1108 sentences from the Enron corpus, and a set of 1108 sentences derived from them by substituting a word of equal

frequency. The original set of sentences is useful because it lets us measure the false positive rate of the various measures. Also using a set in which the only difference is the occurrence of a substitution guarantees that performance differences do not arise from other features of the sentences.

For each measure defined below, we train a decision tree on the measured values for original sentences and sentences containing substitutions to learn the best boundary between the two classes. For all of these measures, there is considerable overlap between the measured values for the two classes (that is, there are many examples on the wrong side of the boundary), reflecting the complex possibilities for informal English sentences. It is therefore not surprising that the error rates of each individual measure are quite high.

4.4 Experiments

We applied the measures described previously to the sentence set.

For the family of k-gram measures, we compute the left k-gram frequency, the right k-gram frequency, and the average of these two measures.

There are often several hypernyms for a given word. We had observed, in previous work [4], that trying to choose a single hypernym could lead to poor results. We compute the hypernym oddities for all of the possible hypernyms of the noun under consideration, and compute: the *minimum* hypernym oddity over all hypernyms, the *maximum* hypernym oddity over all hypernyms, and the *average* hypernym oddity over all hypernyms.

5 Results

Even though we have used sentence oddity to select the set of sentences used as data, it is still reasonable to see how well this measure separates original and substituted sentences. The decision tree trained on both sets of sentences choose the boundary sentence oddity > 2.5 to predict sentences with substitutions. In other words, removing a substituted word from a sentence typically makes the frequency of the remaining bag of words more the double.

Figure 1 summarizes the performance of the various measures on the sentence dataset. In general, each of these techniques makes errors on different sentences, and so combining measures produces better results than using each measure alone. This is clear for the k-gram measures: the average k-gram measure has a much lower false positive rate than either of its two components; but the right k-gram detects substitutions very strongly. Notice that the left k-gram measure detects substitutions only weakly – this suggests that adapting techniques from speech recognition is not likely to work well for this problem. The three hypernym measures also show divergent properties: the minimum hypernym measure does not detect sentences with substitutions well, but has a low false positive rate. The boundaries for all of these measures were determined automatically using a decision tree, but it is clear that there is some scope for altering these boundaries to get better substitution detection at the expense of higher false positive rates

Measure	Detection Rate (%)	False Positive Rate (%)	Boundary score
Sentence oddity	71	20	2.5
Left k-gram	51	28	461
Right k-gram	89	48	722
Average k-gram	51	13	418
Minimum hypernym	40	15	10
Maximum hypernym	68	31	10
Average hypernym	59	22	0
Combined	82	20	see Figure 2

Fig. 1. Detection performance results

(and *vice versa*). However, it is not clear how to do this in a principled way, that is other than by trial and error.

A decision tree was trained using all of the measures as attributes. The resulting decision tree is shown in Figure 2. It is clear from the Table that the combined tree uses only the sentence oddity, average and right k-gram measures and minimum hypernym semantic oddity; but that these measures make their errors on different sentences, so that the overall accuracies are higher than those of the component measures. The high false positive rate is a problem, given that ordinary sentences are much more likely in intercepts than sentences with substitutions.

It might be argued that the decision tree above performs well because the training sentences were selected so that this measure behaves appropriately. Figure 3 shows a combined decision tree in which only the k-gram and hypernym measures are used. The performance is only slight worse (accuracy for sentences containing a substitution 78% and false positive rate 22%).

The performance results and the boundaries were computed for smaller sets of sentences and were remarkably stable as the size of the dataset grew.

6 Conclusions

We have tested how word substitutions within textual communication can be detected. Our technique allows us to automatically flag suspicious messages, so that they can be further investigated, either by a more sophisticated data-mining techniques or manually. The task of detecting substitutions is becoming important since terrorists, criminals, spies and other adversarial parties may use substitution in order to avoid being flagged because of the use of certain words (e.g. 'bomb', 'explosives', 'attack', etc.). Our technique extends prior work, which was not able to detect substitutions when a word is replaced by another word

```
SO <= 2.48
|   KGRAM_AVG <= 4271.5
|   |   SO <= 1.27: 0
|   |   SO > 1.27
|   |   |   KGRAM_R <= 623: 1
|   |   |   KGRAM_R > 623
|   |   |   |   Hyp_MIN <= 5000: 0
|   |   |   |   Hyp_MIN > 5000: 1
|   KGRAM_AVG > 4271.5: 0
SO > 2.48
|   KGRAM_R <= 1380: 1
|   KGRAM_R > 1380
|   |   KGRAM_R <= 173000: 0
|   |   KGRAM_R > 173000: 1
```

Fig. 2. Structure of the decision tree, combining measures (0 – normal sentence, 1 – sentence containing a substitution; SO – sentence oddity, KGRAM – k-gram, Hyp – Hypernym)

```
KGRAM_R <= 722
|   Hyp_AVG <= 0
|   |   KGRAM_AVG <= 332: 1
|   |   KGRAM_AVG > 332: 0
|   Hyp_AVG > 0: 1
KGRAM_R > 722
|   KGRAM_L <= 15
|   |   Hyp_AVG <= -151
|   |   |   Hyp_MIN <= -1889000: 1
|   |   |   Hyp_MIN > -1889000: 0
|   |   Hyp_AVG > -151: 1
|   KGRAM_L > 15: 0
```

Fig. 3. Decision tree, without sentence oddity measures (0 – normal sentence, 1 – sentence containing a substitution; KGRAM – k-grams, Hyp – Hypernym)

with similar frequency of use. This is because our approach is grounded in the semantics of word usage rather than in the frequency ranks. We mine the necessary semantic information from World Wide Web through analyzing the frequency of use of specially constructed phrases obtained by transforming sentences from a message or communication. We have been able to demonstrate empirically that such detection is possible and, when several indicators are combined into one model, can be performed with practically useful accuracy.

Since our model is the first formulation of the task of substitution detection through semantic relationships, we were able only to investigate a simple heuristic model. We are leaving for future research the creation of a more fine-grained model (e.g. based on popular language models) and testing with a wider variety

of test sets. It will be also interesting to investigate how the correlation between substitutions can be exploited to increase the accuracy and even to guess what original words were obfuscated.

References

1. J.A. Bilmes and K. Kirchhoff. Factored language models and generalized parallel backoff. In *Proceedings of HLT/NACCL*, 2003.
2. British National Corpus (BNC), 2004. **www.natcorp.ox.ac.uk**.
3. European Parliament Temporary Committee on the ECHELON Interception System. Final report on the existence of a global system for the interception of private and commercial communications (ECHELON interception system), 2001.
4. SW. Fong, D.B. Skillicorn, and D. Roussinov. Detecting word substitution in adversarial communication. In *Workshop on Link Analysis, Counterterrorism and Security at SIAM International Conference on Data Mining*, submitted.
5. A. R. Golding and D. Roth. A Winnow-based approach to context-sensitive spelling correction. *Machine Learning, Special issue on Machine Learning and Natural Language*, 1999.
6. R. Ferrer i Cancho and R.V. Solé. The small world of human language. *Proceedings of the Royal Society of London Series B – Biological Sciences*, pages 2261–2265, 2001.
7. H. Lee and A.Y. Ng. Spam deobfuscation using a Hidden Markov Model. In *Proceedings of the Second Conference on Email and Anti-Spam*, 2005.
8. D. Roussinov and L. Zhao. Automatic discovery of similarity relationships through web mining. *Decision Support Systems*, pages 149–166, 2003.
9. D. Roussinov, L. Zhao, and W. Fan. Mining context specific similarity relationships using the World Wide Web. In *Proceedings of the 2005 Conference on Human Language Technologies*, 2005.
10. D.B. Skillicorn. Beyond keyword filtering for message and conversation detection. In *IEEE International Conference on Intelligence and Security Informatics (ISI2005)*, pages 231–243. Springer-Verlag Lecture Notes in Computer Science LNCS 3495, May 2005.