Word replacement recognition using Sentence Oddity, K gram and NGD

Santosh Bhosale[1], Vidya Dhamdhere[2]

[1] G.H.Raisoni College of engineering and Management

[2]G.H.Raisoni College of engineering and Management

ABSTRACT

There are so many vigilance system which works on recognition of replaced word from sentence, for example in antiterrorism organizations like ATS. While sending messages incendiary changes the word which may cause to set alarm. For example, "we are ready for the blast tonight" can be transform into "we are ready for the complex tonight". This type of transformations or replacements can be happen and human being can easily recognized it. But for large documents or set of documents, emails, chat messages it is not possible to detect such transformations. To solve this issue we can make this process automatic with the help of frequencies of each word. For example in above sentence "complex" doesn't make any sense. So frequency of complex is less as compare to blast. We define three measures to detect the transformation or replacement of the word, Sentence Oddity (SO), K gram and Normalized Google Distance (NGD). With the help of these three algorithms we show that after combination of the three we can get 90% positive results. We also developed the watchlist which contains different words which may be replaced. After detecting the replaced word we are matching detected word with words from the watchlist. For this matching we used cosine similarity.

Keywords: NGD, Sentence Oddity, K gram, cosine similarity.

1. INTRODUCTION

There is a vast development in communication media, especially in India, in last fifteen years. This includes use of telephones, mobile phones, internet, email etc. This facility is proved beneficial for the illicit acts in terrorisms and crimes too. It includes sending text messages via email or SMS to the group members either using fake identification or by hacking/stealing the device or network link. Emails containing sensitive text can be separated by scanning every email for occurrence of sensitive words and then processing it using another level of data mining algorithms. However, illicit groups started substituting the sensitive word in the email by a normal word in order to hide the meaning of the sentence so that it can be interpreted as a normal mail. Such type of obfuscation also is seen in the bribe cases where both parties communicate in public. Human intervention can detect such substitutions with the help of contextual information and general sense. However, automatic detection of such obfuscated messages is quite difficult. At the same time, it is not possible to manually scan every message. Terrorists are aware that their communication can be intercepted by smart systems, and they tried to hide their message or content of the message as far as possible. Criminals also face similar issues since their communications may be intercepted by law enforcement.

One way to conceal content is to encrypt the messages, but this strategy has a number of drawbacks. Another strategy to conceal the content of messages is to replace significant words with other words or locations that are judged less likely to attract attention. For example, it is known that Echelon scans for a list of significant words or phrases, and terrorists would presumably wish not to use these words in their messages. The difficulty is that, while it is clear that some words must be on these lists.

Apart from email communication, terrorist groups are using websites to publish objectionable material for example, publishing detailed procedure to manufacture bomb. However, in order to hide the actual meaning of the published material, the data uploaded on the website is obfuscated such that it looks normal to the users. As substituted words are selected without logic in word selection and they are selected such that the substituted message looks like normal. This paper discusses the approaches to identify such suspects which can then be processes to next level Data Mining algorithms for further analysis. The approaches present here are based on Search Engine hit count. First approach is based on search count of k-gram of the sentences and second is based on Normal Google Distance (NGD), the algorithm presented by Google Research Lab.

2. RELATED WORK

A standard model for many natural language problems is to assume a language-generation model that describes how sentences in English are generated, and an alteration model that describes how such sentences are changed in the problem domain being considered. The probability of a given sentence w being generated is given by some probability

P(W). The alternation model changes w to some new sentence y with probability P(Y/W). The task is to estimate w given y . In the problem we address, the alteration model is the replacement of some set of words with other words of similar frequency. We are interested not in predicting the original sentence (which would be extremely difficult) but in detecting when P(W) differs significantly from P(y). Some early results have already appeared .An easier variant, the problem of detecting a substituted word with substantially different frequency from the word it replaces was addressed by Skillicorn. This work considered, not individual sentences, but large collections of messages.

The existence of identical substitutions in different messages was shown to be detectable, via the correlations that were created among them using matrix decompositions. Speech recognition uses an alteration model in which text is converted to an analog waveform. Predicting the original sentence w is done using the left context of the current word and a statistical model of word co-occurrences. Such algorithms are heavily dependent on left-to right processing, backing up to a different interpretation when the next word becomes sufficiently unlikely.

Speech recognition differs from the problem addressed here because it is limited to the left context, whereas we are able to access both left and right contextual information. Detecting misspellings uses an alteration model that incorporates common keystroke errors, themselves derived from visual, aural, and grammatical error patterns. Spam detection is closer to our problem in the sense that the alteration model assumes human-directed transformations with the intent to evade detection by software. For example, Spam Assassin 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 detect word-level manipulations typical of spam using Hidden Markov Models. They addressed the question of whether an e-mail contains examples of obfuscation by word substitution, expecting this to be simpler than recovering the text that had been replaced. They remark that detecting substitution at all is “surprisingly difficult” and achieve prediction accuracies of around 70 percent 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 specific contexts was considered. Using Google (or other Internet search engines with large coverage) to check for spelling and grammatical errors has been suggested in the academic literature. Indeed, since substitutions frequently result in incorrect grammatically or semantically formed phrases, detecting such errors may also detect substitutions. For example, the erroneous use of a word in the phrase “had ice-cream for desert” means that it occurs on the Web only 44 times, according to Google. The correct phrase “had ice-cream for dessert” occurs 316 times.

3 PROBLEM DEFINITION

Content of the communications can be concealed by replacing words that might trigger attention by other words that seem more ordinary. We address the problem of discovering such substitutions.

A problem of word substitution in the text can be solved by using similarity search importance as terrorist groups are using substitutions for conveying their messages to their counter parts via email. As the substituted words are normal word, it is difficult to automatically recognize it.

Idea behind it is to search Large-scale database to find out sensitive words substitutes. These substitutes can be easily noticed in the sentence with the help of other words in the sentence. Substituted words can be determined by finding out their relative frequencies with other words in the sentence with the help of database. Most closely matching word of relatively high frequency can be thought of as substitute.

4 MEASURES

4.1 Sentence Oddity (SO)

This measure considers a sentence as a whole, and the relationship between the entire sentence, and the sentence with a particular word of interest deleted. As noted above, we can only get useful frequency estimates by treating sentences as bags of words, that is, we generate search engine queries with a list of the words in the sentence, rather than treating the entire sentence as a quoted string. SO is based on the observation that removing a contextually appropriate word from a sentence should not substantially change the frequency of the resulting bag of words in comparison to the frequency of the entire sentence, since the contextually appropriate word co-occurs frequently with the other words in the sentence. On the other hand, removing a contextually inappropriate word might be expected to produce a large increase in frequency of the remaining bag of words because it would only rarelyco-occur with the other words. Hence, we define the SO of asentence with respect to a particular target word as,

$$SO = \frac{Frequency\ of\ sentence\ removing\ target\ word}{Frequency\ of\ complete\ sentence}$$

SO should be large for a sentence in which a word has been substituted. The frequency, at Google, of our example sentence with “attack” removed is 5.78M, whereas the frequency of the entire sentence is 2.42M, so the sentence oddity of the example sentence is 2.4. For the sentence with substitution, the frequency of the entire sentence is 1.63M, so the sentence oddity is 3.5. As expected, the sentence oddity of the sentence containing the substitution is significantly larger than that of the original sentence.

4.2 K Gram

The difficulties of using the frequencies of exact strings containing the word of interest are illustrated by looking at the frequencies of substrings of our example sentences. These are illustrated in Table 1.Frequencies overall are lower for the fragments of the sentence that contains the substitution, but we would not, in practice, know the frequencies of the original sentence to compare them with. The frequencies of exact strings are often so low that they are difficult to work with. K - grams are measures of frequency for strings of limited length. We define the left k-gram of a word to be the string that begins with the word and extends left, up to and including the first nonstop word. Similarly, the right k-gram of a word is the string that begins with the word and extends constitutes a stop word might vary with application domain; we use the stop word list from Word net 2.1 in our work. In our ordinary example sentence, the left k-gram of “attack” is “expect that the attack”(f=50), and the right k-gram is “attack will happen” (f=9,260). In the sentence with a substitution, the left k-gram of “campaign” is “expect that the campaign” (f=77), and the right k-gram is “campaign will happen” (f=132). We expect that, in general, k-grams will be smaller for sentences containing a substitution, although in the example, this is only true for the right k-gram. Left and right k grams capture significantly different information about the structure of sentences, which is not surprising given the linear way in which English is understood.

The k-gram of a substituted word is the string containing that word and its context up to and including the first non-stopword to its left, and the first non-stopword to its right. Left k gram is starting from considered noun towards left till the start of the sentence is reached and right k gram is starting from considered noun to rightwards till the end of sentence is reached.

Table 1: Margin specifications

Sr. No.	Words from sentence	Frequency
1.	We except that the attack will happen tonight	F=2.42 M
2.	We except that the operation will happen tonight	F=1.31 M
3.	We except that the campaign will happen tonight	F=1.63 M
4.	We except that the race will happen tonight	F==1.97 M

While comparing left k gram of original sentence and left k gram of word substituted sentence of each sentence, we observed that frequencies for word substituted sentence is greater than original sentence for a set but it is exactly opposite for right k gram.

4.3 Normalized Google Distance

NGD (Normalized Google Distance) is an approximation of NID(Normalized Information Distance). It is a semantic relativity measure derived from the number of hits returned by Google search engine for a given set of words. NGD value is between 0 and 1, value 0 indicates closely related words and value 1 indicates loosely related words. Normalized Google distance between two search terms x and y is

$$NGD(x,y) = \frac{\max\{\log f(x), \log (y)\} - \log\ f(x,y)}{\log M - \min\{\log f(x), \log f(y)\}}$$

where M is total number of web pages searches by Google, f(x) and f(y) are the number of hits for searched terms x and y respectively and f(x, y) is the number of web pages on which both x and y occurs. NGD can be calculated for a pair of words. Being it gives relative distance between a pair of words, NGD can be used to detect the substitution of words problem. It is obvious that NGD calculated for the original word and its every adjacent word in a sentence except a stop word should be less than the substituted word and its adjacent words in the sentence.

We made a survey of two measures NGD and K gram frequency for detection of substitution of word in test dataset. For experimentation of above mentioned, we used Google and Yahoo! Search engines search count. Google is considered to be the most used and effective search engine. Google uses 'Trust Rank' algorithm to create a personalization vector in Google matrix that decreases the harmful effect of link spamming. So here we decided to consider Google as search engine for testing the data. Behavior of Google search engine is peculiar to the use of punctuations used along with the search term. Use of double quotes to the keyword results in different hit count than that we get with no quotes. This in turn is different than that we get when conjunction 'AND' is used. In our experimentations, we considered all possible ways of giving keywords. For many test cases, use of quotes revealed in hit count of zero only, hence such observations were not considered.

The normalized Google distance is a theory of similarity between words and phrases, based on information distance and Kolmogorov complexity by using the world-wide-web as database, with its page counts derived from a search engine such as Google. This unsupervised method regards the word sense disambiguation as a process of searching minimum normalized Google distance between n-gram and the translation or synonym of the target word, based on the supposition that one sense per n-gram. Our System is tested on Multilingual Chinese-English Lexical Sample task in Semeval-2007. Experimental result shows that our method outperforms the best competing system. NGD is tested for set of two words and we tried to quantify the strength of relationship between these words. In order to find combined frequency of terms x and y, i.e. f(x, y), to calculate NGD is taken on various basis. NGD values ranges from 0 to 1. If NGD of words is 0, we can conclude that there is strong relationship between these two words and if it is 1 then these words are not related. Consider test data given in Table 1 and Table 2. Search count of these terms is calculated by using 'space' between words for combined frequency of x and y in Google search engine. Figure 1 shows the result for NGD calculated for related strings such as "Mahatma Gandhi", "United States" etc.

5 MEASURES

In our experimentation, we got result zero for both original sentence and substituted sentence. This implied that use of SO leads to no conclusion regarding the relationship between two sentences. Hence we tried an approach of k gram to detect word substitution. Here we divided the sentence into two parts, since it is not very usual to find k gram for whole sentence directly. We calculated left k gram and right k gram starting from a noun in the sentence. Left k gram is starting from considered noun towards left till the start of the sentence is reached and right k gram is starting from considered noun to rightwards till the end of sentence is reached. Considering original and substituted sentence for testing data, we got following results for left k gram and right k gram for original and substituted sentences. List of sentences and associated results used to test k-gram method is given below in the Table 3. Also the behavior of the right and left k-grams is given graphically in the following figures:

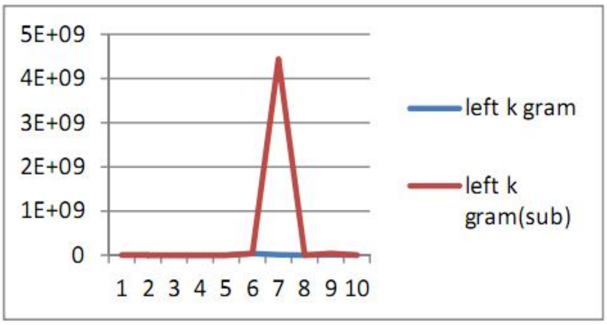

Figure1 Left k gram for original and substituted sentences

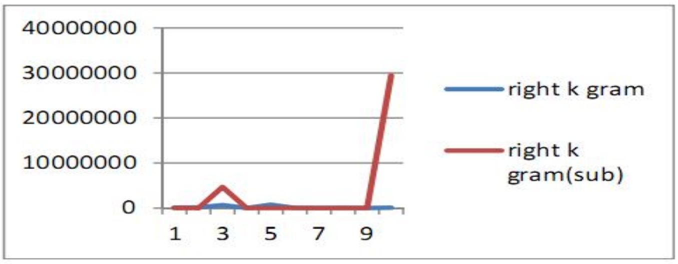

Figure 2 Right k gram for original and substituted sentence

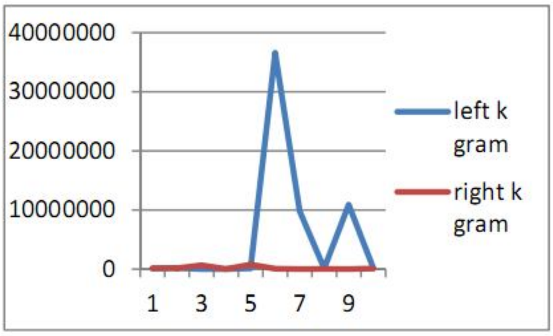

Figure 3 Left and right k gram for original sentences

Table 1: K-gram calculations for sample sentences

Sr. No.	Original sentences	Left k gram	Left k gram frequency	Left k gram freq for substitution	Right k gram	Right k gram freq	Right k gram freq for substitution
1	you will get bomb at Delhi (chocolate)	"will get bomb"	67400	1400000	"bomb at Delhi"	111000	8
2	we have to do murder in Mumbai (felicitation)	“have to do murder"	159000	0	"murder in Mumbai"	142000	2930
3	ramesh will come to collect explosive material(cotton)	"come to collect explosive"	4020	17500	"explosive material"	650000	4670000
4	we expect that attack will happen tonight(marriage)	"we expect that attack"	15000	19900	"attack will happen tonight"	5250	5
5	give training to attack on city(rain)	"give training to attack"	98700	385	"attack on city”	746000	19400
6	the bomb is in position(flower)	"the bomb"	36600000	39800000	"bomb is in position"	46400	39900
7	burn the train tomorrow(colour)	"burn"	9840000	4.44E+09	"burn the train tomorrow"	1	0
8	spread violence as soon possible(happiness)	"spread violence"	144000	1070000	"violence as soon as possible"	32100	21400
9	our next target will be business center(picture)	"next target"	10900000	40900000	"target will be business center"	0	0
10	keep the bomb in the car(bag)	"keep the bomb"	160000	2670000	"bomb in the bag"	93600	29400000

6 CONCLUSION

This technique allows us to automatically flag suspicious messages, so that they can be further investigated, either by more sophisticated data-mining techniques or manually.

In this technique we will use k-gram and NGD for probable detection of substitution of text. The measures uses search count returned by search engine for the given phrases. While entering keywords we used ‘and’ed strings, used quotations and also searched without quotation. We observed that the result thus obtained proves that k-gram and NGD can be used to detect substitution.

The problem of detecting word substitution has a role to play in settings such as counterterrorism and law enforcement, where large amounts of message traffic may be intercepted in an automated way, and it is desirable to reduce the number of messages to which further analysis must be applied. The existence of simple mechanisms such as watchlists of significant words may actually make the discovery of illicit groups easier, because they must react to the existence of watchlists, whereas innocent groups are either unaware of them or do not alter their messages.

The problem of detecting word substitution has a role to play in settings such as counterterrorism and law enforcement, where large amounts of message traffic may be intercepted in an automated way, and it is desirable to reduce the number of messages to which further analysis must be applied. The existence of simple mechanisms such as watch lists of significant words may actually make the discovery of illicit groups easier, because they must react to the existence of watch lists, whereas innocent groups are either unaware of them or do not alter their messages. If the goal of illicit groups is to evade automated detection, then it is important that the word substitutions should look as normal as possible from a syntactic perspective (whereas if humans were searching for suspicious messages, a much more semantic form of substitution would be required).

References

[1] The use of the internet for Terrorist purposes, Report of United Nations office on drugs and crime, Vienna incollaboration with the United Nation□s Counter Terrorism implementation task force 2004 published by united Nation Newyork Sep 2012.

[2] Mrs. Shilpa Mehta, Dr. U Eranna, Dr. K. Soundararajan, Surveillance Issues for Security over ComputerCommunications and Legal Implications, Proceedings of the World Congress on Engineering 2010 Vol I WCE2010, June 30 - July 2, 2010, London, U.K.

[3] Peter D. Turney, Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL, Proceeding of the 12th Europium Conference on Machine Learning ,pages 491-502,Springer-Verlag, UK,2001

[4] J.Tang, H.Li, Y. Cao, Z.Tang, Email Data Cleaning, Proceedings of KDD, Chicago, USA, (2005).

[5] Appavu alias Balamurugan and Ramasamy , Suspicious E-mail Detection via Decision Tree: A Data MiningApproach, RajaramThiagarajar College of Engineering, Madurai, India

[6] www.wordcount.org/main.php

[7] Gang Wang, Hsinchun Chen and HomaAtabakhsh, Criminal Identity Deception and Deception Detection in LawEnforcement, Group Decision and Negotiation, Mar 2004, Vol. 13 Issue 2, p111-127. 17p.Department ofManagement Information Systems, University of Arizona, 430 McClelland Hall, Tucson, AZ

[8] Szewang Fong, Dmitri Roussinov, And David B. Skillicorn, Detecting Word Substitutions In Text, IEEE Transactions On Knowledge And Data Engineering, Vol. 20, No. 8, August 2008

[9] http://en.wikipedia.org/wiki/Cosine_similarity

AUTHOR

Santosh Bhosale G.H.Raisoni College of engineering and Management

Vidya Dhamdhere G.H.Raisoni College of engineering and Management