2011 10th International Conference on Machine Learning and Applications

Combining Corpus-Based Features for Selecting Best Natural Language Sentences

Foaad Khosmood
Department of Computer Science
California Polytechnic State University
San Luis Obispo, California, USA
foaad@calpoly.edu

Robert Levinson
Department of Computer Science
University of California at Santa Cruz
Santa Cruz, California, USA
levinson@soe.ucsc.edu

***Abstract*— Automated paraphrasing of natural language text has many interesting applications from aiding in better translations to generating better and more appropriate style language. In this paper, we are concerned with the problem of picking the best English sentence out of a set of machine generated paraphrase sentences, each designed to express the same content as a human generated original. We present a system of scoring sentences based on examples in large corpora. Specifically, we use the Microsoft Web N-Gram service and the text of the Brown Corpus to extract features from all candidate sentences and compare them against each other. We consider three feature combination methods: A handcrafted decision tree, linear regression and linear power-set regression. We find that while each method has particular strengths, the linear power set regression performs best against our human-evaluated test data.**

Computational Natural Langauge Processing, Paraphrasing Computational Linguistics, Linear regression, Linear power-set regression

I. INTRODUCTION

Automated paraphrasing, or re-stating of the same content with the same basic meaning in different language parameters, can be used to improve machine translation output, automatically summarize text or personalize and transform writing styles. Invariably some paraphrasing techniques[1][2][3][4][5] perform relatively better under certain circumstances and particular text inputs. One method of increasing the quality of the paraphrase is to generate many variants and select the best sentence based on some scoring criteria.

In this paper, we consider the question "How do we pick the best sentence?" The question is important for several reasons. First, finding a method to pick the best of many choices allows us to combine the power of many paraphrasing techniques. Second, while many paraphrased sentences may be technically correct in both meaning and grammar, they are nevertheless undesirable. For example "I locomote to school with a two wheeled mechanical device," is technically a valid transformation for "I bike to school." However, there are very few circumstances that would require the use of the former over the latter. If substituted during a real transformation exercise, the longer sentence will generally add to the awkwardness of the text, perhaps to a distracting level. Thus in addition to avoiding possible ungrammatical responses, we must somehow also avoid the "unnatural" or "awkward" ones as much as possible.

II. FEATURES OF A "GOOD" SENTENCE

We use two categories of metrics that we can derive from a general sentence that we could later use to evaluate it. These are 1) Internet search hit rates and 2) Large English corpus word co-occurrences.

The idea behind Internet search hit rates is simple. If one group of consecutive words have higher hit rates on the Internet compared to another, then that group must be more commonly used and therefore likely to be more natural sounding. For example, if we compare "I saw an owl" to "I saw a owl", the former will have more hit rates and thus we can avoid a grammatical mistake by simply doing searches on the World Wide Web.

The Microsoft Web N-Gram service[6] provides a great opportunity to evaluate sentences using search engine hit rates. The service provides Microsoft BING search engine-crawled statistics reflecting the state of the World Wide Web as of April 10, 2010. The service accepts input in form of a sequence of 2, 3, 4 or 5 words. It then returns the log of the joint probability of those words occurring in that sequence on the Web. We concentrate on word bigrams and trigrams in main text of Web pages. This is mostly due to the fact that smaller sentences cannot be evaluated using the 4 or 5-gram models. The service ignores punctuation marks and word capitalization.

For both bigram and trigram models, we can derive an average, a minimum and a maximum value per sentence. The average is calculated by taking the average return value of every n-gram present in the sentence. The lowest n-gram log-probability encountered would be the minimum and the highest the maximum. We use both minimum and average for our purposes and ignore the maximum. As the owl sentence demonstrates above, often the problem is one misplaced word or combination. Thus a low minimum n-gram value would serve to isolate that misplaced construction (i.e. "a owl") and is very useful. However, the maximum, which simply reflects the most popular n-gram in the sentence, does nothing to indicate potential problems elsewhere in the sentence.

978-0-7695-4607-0/11 $26.00 © 2011 IEEE
DOI 10.1109/ICMLA.2011.170

Table 1. Microsoft N-Gram based sentence features

Model	Value	Method of calculation
word bigrams / webpage body	average	average of all roving window bigrams in the sentence
word bigrams / webpage body	minimum	minimum bigram probability encountered
word trigrams / webpage body	average	average of all roving window trigrams in the sentence
word trigrams / webpage body	minimum	minimum trigram probability encountered

In a similar fashion to Web N-Grams, we use the Brown Corpus [7] to get more features from sentences that we can correlate to their appropriateness. The corpus is large and topically diverse. Sentences that appear in it can be assumed to be correct and natural sounding. The idea here is to use a word co-occurrence vector [8] with sentence-length windows size. Co-occurrence of words within the same sentence in the Brown Corpus is considered evidence of relatedness of those words. Thus if a sentence we are examining has the same co-occurring words, we can take that as evidence for its correctness and naturalness. For example, given the following two sentences:

- *I deposited my check at our friendly neighborhood bank.*
- *I deposited my dog at our friendly neighborhood bank.*

We should be able to eliminate the second and choose the first. It is likely that word combinations of "check" and "bank" appear somewhere in the same sentence in the Brown Corpus, but "dog" and "bank" probably much less often if ever. We also note that the N-Gram data would not help make the decision here, as the two words are too far apart for the bigram and trigram data to be used to evaluate them.

We can derive similar statistics using this co-occurrence concept. These are outlined in Table 2.

Table 2. Brown Corpus Features

Value	Elaboration and Calculation
BC-avg	Brown co-occurrence average: The concurrence hit rate (number of sentences in Brown that contain both words) for every combination of 2 words in the sentence.
BC-bin-avg	Brown co-occurrence binary average: Same as above, but instead of actual hit rates a 0 or 1 is returned. A 1 means at least one sentence contains the word combination.
BC-min-drop0	Brown co-occurrence minimum (above 0): Returns the lowest non-Zero hit rate from any combination of words in the sentence.
BC-sum	Brown co-occurrence sum: The sum of all hit rates of all word pairs in the sentence.
BC-max	Brown co-occurrence maximum: The biggest numerical hit rate among all the word pairs.

III. Evaluation of transforms using 50 random English sentences

Fifty sentences from the Tatoeba [9] English language corpus are evaluated by six paraphrase generating transforms [10]. Each transform is a specialized method of paraphrasing a given input sentence and may produce none, one or multiple paraphrase responses. The resulting 381 transformed responses are evaluated by three human judges in two separate sessions. The scoring criteria for the evaluations are as follows.

Table 3. Paraphrase evaluation coding scheme

Code	Criteria
3	Sentence is modified, preserves original meaning and original level of grammaticality.
2	Sentence is unmodified, or barely modified with very minor changes, or modified and the result is "acceptable" but not great.
1	Sentence is modified and the result is not acceptable.
0	No response, error or incomprehensible response

IV. Correlating the statistical features of sentences to human evaluations

We derive correlations between average normalized sentence features and human evaluations (and also among the features themselves.) Pearson correlation is used as the standard in statistical correlation. However, we also derive the Spearman correlations where the main difference is that relative values between ranked items are discounted.

Human evaluations are produced in two formats. First is average of all three evaluators (user-avg) and second is a "voted" score which is quantized to be at values of 0, 1, 2 or 3, normalized. In case of a tie, the average was used in the voted evaluation (user-vote) as well. Nine statistical measures described in tables 1 and 2 are correlated in Figure 1 below.

	2g-min	2g-avg	3g-min	3g-avg	bc-avg	bc-bin-avg	bc-sum	bc-min-0	bic-max	user-avg
2g-avg	0.6928									
3g-min	0.7714	0.5135								
3g-avg	0.5702	0.6499	0.7546							
bc-avg	0.0075	0.3717	-0.0944	0.1737						
bc-bin-avg	-0.0944	0.17	-0.2398	0.0122	0.5893					
bc-sum	-0.0335	0.3384	-0.1454	0.1422	0.9788	0.635				
bic-min-0	0.0769	0.3883	0.2269	0.2438	0.358	0.0586	0.2783			
bc-max	-0.001	0.3417	-0.1101	0.1582	0.9884	0.5904	0.9818	0.3308		
user-avg	0.1138	0.1781	0.0841	0.1649	0.1247	0.1414	0.136	0.0827	0.0119	
user-vote	0.1556	*0.2375*	0.1179	*0.1925*	*0.1762*	*0.1760*	0.1825	0.1375	*0.1710*	0.939

Figure 1. Spearman correlations between features and user evaluations

The strongest observable correlations are those in the Spearman table with the single strongest feature being the

bigram averages from Microsoft Web N-Gram data. We italicize five of the nine sentence features as exhibiting the strongest correlations to user scores. These are: 2-gram and 3-gram averages from Microsoft N-Grams and Brown correlation average, binary average and maximum from the Brown data.

V. COMBINING FEATURES FOR AN OVERALL PREDICTION ALGORITHM

We experiment with three distinct methods of combining the top sentence level features (as derived in the previous section). Two of the methods are regression based and one is a decision tree based on correlation performance. To make it easier to discuss these features in this section and the next, we will designate a function to stand for each of them.

Table 4. Sentence quality feature function designation

Sentence features	**Function**
Average word bigrams from Microsoft Web N-Grams	$F_1(S)$
Average word trigrams from Microsoft Web N-Grams	$F_2(S)$
Brown Correlations average word pair hits	$F_3(S)$
Brown Correlations binary average word pair hits	$F_4(S)$
Brown Correlations maximum word pair hits	$F_5(S)$

A. F1 formula: a decision-tree with tie breaking

This method uses the Spearman correlation data to determine which of the feature values are most important and uses that information to determine of which sentence should be scored higher. In case of a tie (which are frequent given that many sentences are very similar), the next highest feature value will be examined.

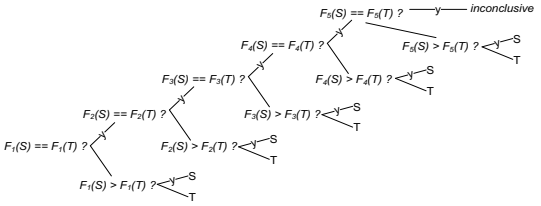

Figure 2. F1 "decision tree tie break" pair wise sentence comparison algorithm between sentences (S) and (T)

B. Linear regression among the five top features

The linear regression method sets up the following formula for each of the human-scored training sentences and attempts to derive the coefficients that cause the outcome to be closest to the human scored value.

$$LR(S) = A_1F_1(S) + A_2F_2(S) + A_3F_3(S) + A_4F_4(S) + A_5F_5(S) + A_6$$

Figure 3. Linear regression formula

Where F_1 to F_5 denote the five chosen features from Table 4 and A is the coefficient matrix. We calculated the following values for the coefficient matrix.

Table 5. Linear regression coefficient values

A_1	A_2	A_3	A_4	A_5	A_6
-.0447	.111	-.531	.0719	.401	-.00051

C. Linear regression on the power set the five top features

We apply linear regression on the power set of the original 5 features, calling this Linear Power Set regression (LPS). The terms represent every combination of the five original terms multiplied together. For the five terms this produces 31 different multiplicative combinations. With the additional linear offset term, this makes 32 coefficients that we derive using the training data.

$$\begin{aligned}LPSR(S) = {} & A_1F_1(S) + A_2F_2(S) + A_3F_3(S) + A_4F_4(S) + A_5F_5(S) + A_6F_1(S)F_2(S) + A_7F_1(S)F_3(S) \\ & + A_8F_1(S)F_4(S) + A_9F_1(S)F_5(S) + A_{10}F_2(S)F_3(S) + A_{11}F_2(S)F_4(S) + A_{12}F_2(S)F_5(S) \\ & + A_{13}F_3(S)F_4(S) + A_{14}F_3(S)F_5(S) + A_{15}F_4(S)F_5(S) + A_{16}F_1(S)F_2(S)F_3(S) \\ & + A_{17}F_1(S)F_2(S)F_4(S) + A_{18}F_1(S)F_2(S)F_5(S) + A_{19}F_1(S)F_3(S)F_4(S) + A_{20}F_1(S)F_3(S)F_5(S) \\ & + A_{21}F_1(S)F_4(S)F_5(S) + A_{22}F_2(S)F_3(S)F_4(S) + A_{23}F_2(S)F_3(S)F_5(S) + A_{24}F_2(S)F_4(S)F_5(S) \\ & + A_{25}F_3(S)F_4(S)F_5(S) + A_{26}F_1(S)F_2(S)F_3(S)F_4(S) + A_{27}F_1(S)F_2(S)F_3(S)F_5(S)V \\ & + A_{28}F_1(S)F_2(S)F_4(S)F_5(S) + A_{29}F_1(S)F_3(S)F_4(S)F_5(S) + A_{30}F_2(S)F_3(S)F_4(S)F_5(S) \\ & + A_{31}F_1(S)F_2(S)F_3(S)F_4(S)F_5(S) + A_{32}\end{aligned}$$

Figure 4. Linear regression on the power set of the features

VI. EVALUATION OF COMBINATION METHODS

We conduct an additional study between the three feature combination methods. We first apply them back to the original sentences to see which one would have produced the best results. For each original sentence a set of paraphrases are generated and scored by human judges. The same set of paraphrases is evaluated using the three methods below. We record if the highest scoring paraphrase for each method corresponds with a human score of 3, if such a score is available in the set. The number of times where each method successfully picks the best sentence in the set are displayed below in Table 6.

Table 6. Comparison of feature combination methods

	F1	**Linear**	**Linear Power Set**
All derivations	13	9	10
Max-1 per group	10	9	10

When considering "Max-1" per group, meaning consider just one response from each transform, F1 and LPS are very competitive and both consistently ahead of Linear.

We evaluate another 25 sentences as and consider the results just between F1 and LPS. Here, we decisively find that LPS is the best method as indicated by the evidence:

390 paraphrases from the 25 sentences are evaluated by humans and run through F1 and LPS. In 18 of the 25 instances, LPS method is able to pick a sentence from the response group that was also rated highest by the human scorers. F1 was only able to accomplish this for 11.5 of the instances.

VII. Conclusion and Future Work

We evaluate three different algorithms for scoring of sentences for the purpose of finding the best, most naturally worded paraphrase of a given sentence. All candidate paraphrases are generated using one of six transforms that we have developed or adapted[10]. Each scoring algorithm uses the same (best) subset of features derived from the Microsoft N-Gram project and co-occurrence data from the Brown Corpus.

We find that LPS regression scores outperform linear regression and F1 decision trees in our experiments.

LPS method on the Web N-Gram and Brown Corpus co-occurrence features can be considered a robust predictor of good sentences, at least in comparison to the other two methods explored. Relying on the theory of language style as a set of conscious choices of the author [11], we can simulate such choices artificially to create paraphrases and use the scoring methods described in this paper to select the most appropriate one.

Future work will explore both different features, and better predictive algorithms suitable for style-based sentence selection.

References

[1] Houda Bouamor, Aurélien Max, Gabriel Illouz, Anne Vilnat, "Web-based validation for contextual targeted paraphrasing," *Proceedings of the ACL Workshop on Monolingual Text-To-Text Generation* (Text-To-Text-2011), Portland, Oregon, USA, 2011.

[2] Sander Wubben, Erwin Marsi, Antal van den Bosch and Emiel Krahmer, "Comparing Phrase-based and Syntax-based Paraphrase Generation," *Proceedings of the ACL Workshop on Monolingual Text-To-Text Generation* (Text-To-Text-2011), Portland, Oregon, USA, 2011.

[3] F. Khosmood and R. Levinson, "Toward automated stylistic transformation of natural language text," Digital Humanities, Washington, D.C. , 2009.

[4] F. Khosmood and R. Levinson. "Automatic Synonym and Phrase Replacement Show Promise for Style Transformation," ICMLA, Washington D.C., 2010.

[5] Advaith Siddharthan, "Complex lexico-syntactic reformulation of sentences using typed dependency representations". In Proceedings of the 6th International Natural Language Generation Conference (INLG 2010), pages 125-133, Dublin, Ireland, 2010.

[6] Micorsoft Web N-Gram service, public Beta program, http://web-ngram.research.microsoft.com/info/ accessed 2011.

[7] W. N. Francis and H. Kucera, *A Standard Corpus of Present-Day Edited American English, for use with Digital Computers*, Department of Linguistics, Brown University, 1964, Providence, Rhode Island, USA.

[8] Ido Dagan, "Contextual Word Similarity", in *Handbook of Natural Language Processing* edited by Dale, Moisl and Somer, Marcel Dekker, Inc. 2000, pp 459-463.

[9] Project Tatoeba English language sentence corpus (about 150,000 sentences). accessed from tatoeba.org, September 2010.

[10] F. Khosmood, Computational Style Processing, Ph.D. dissertation, University of California Santa Cruz, Dec. 2011.

[11] J. Walpole "Style as Option," College Composition and Communication, vol. 31, No. 2, Recent Work in Rhetoric: Discourse Theory, Invention, Arrangement, Style, Audience, (May, 1980), pp. 205-212, 1980.