A Corpus-based Approach to Automatic Compound Extraction

Keh-Yih Su
Dept. of Electrical Engineering
National Tsing-Hua University
Hsinchu, Taiwan 30043, R.O.C.
kysu@bdc.com.tw

Ming-Wen Wu
Behavior Design Corporation
No. 28, 2F, R&D Road II
Science-Based Industrial Park
Hsinchu, Taiwan 30077, R.O.C.
mingwen@bdc.com.tw

Jing-Shin Chang
Dept. of Electrical Engineering
National Tsing-Hua University
Hsinchu, Taiwan 30043, R.O.C.
shin@hera.ee.nthu.edu.tw

Abstract

An automatic compound retrieval method is proposed to extract compounds within a text message. It uses n-gram mutual information, relative frequency count and parts of speech as the features for compound extraction. The problem is modeled as a two-class classification problem based on the distributional characteristics of n-gram tokens in the compound and the non-compound clusters. The recall and precision using the proposed approach are 96.2% and 48.2% for bigram compounds and 96.6% and 39.6% for trigram compounds for a testing corpus of 49,314 words. A significant cutdown in processing time has been observed.

Introduction

In technical manuals, technical compounds [Levi 1978] are very common. Therefore, the quality of their translations greatly affects the performance of a machine translation system. If a compound is not in the dictionary, it would be translated incorrectly in many cases; the reason is: many compounds are not compositional, which means that the translation of a compound is not the composite of the respective translations of the individual words [Chen and Su 1988]. For example, the translation of 'green house' into Chinese is not the composite of the Chinese translations of 'green' and 'house'. Under such circumstances, the number of parsing ambiguities will also increase due to the large number of possible parts of speech combinations for the individual words. It will then reduce the accuracy rate in disambiguation and also increase translation time.

In practical operations, a computer-translated manual is usually concurrently processed by several posteditors; thus, to maintain the consistency of translated terminologies among different posteditors is very important, because terminological consistency is a major advantage of machine translation over human translation. If all the terminologies can be entered into the dictionary before translation, the consistency can be automatically maintained, the translation quality can be greatly improved, and lots of postediting time and consistency maintenance cost can be saved.

Since compounds are rather productive and new compounds are created from day to day, it is impossible to exhaustively store all compounds in a dictionary. Also, it is too costly and time-consuming to inspect the manual by people for the compound candidates and update the dictionary beforehand. Therefore, it is important that the compounds be found and entered into the dictionary before translation without much human effort; an automatic and quantitative tool for extracting compounds from the text is thus seriously required.

Several compound extracting approaches have been proposed in the literature [Bourigault 1992, Calzolari and Bindi 1990]. Traditional rule-based systems are to encode some sets of rules to extract likely compounds from the text. However, a lot of compounds obtained with such approaches may not be desirable since they are not assigned objective preferences. Thus, it is not clear how likely one candidate is considered a compound. In LEXTER, for example, a text corpus is analyzed and parsed to produce a list of likely terminological units to be validated by an expert [Bourigault 1992]. While it allows the test to be done very quickly due to the use of simple analysis and parsing rules, instead of complete syntactic analysis, it does not suggest quantitatively to what extent a unit is considered a terminology and how often such a unit is used in real text. It might therefore extract many inappropriate terminologies with high false alarm. In another statistical approach by [Calzolari and Bindi 1990], the association ratio of a word pair and the dispersion of the second word are used to decide if it is a fixed phrase (a compound). The drawback is that it does not take the number of occurrences of the word pair into account; therefore, it is not

known if the word pair is commonly or rarely used. Since there is no performance evaluation reported in both frameworks, it is not clear how well they work.

A previous framework by [Wu and Su 1993] shows that the mutual information measure and the relative frequency information are discriminative for extracting highly associated and frequently encountered n-gram as compound. However, many non-compound n-grams, like 'is a', which have high mutual information and high relative frequency of occurrence are also recognized as compounds. Such n-grams can be rejected if syntactic constraints are applied. In this paper, we thus incorporate parts of speech of the words as a third feature for compound extraction. An automatic compound retrieval method combining the joint features of n-gram mutual information, relative frequency count and parts of speech is proposed. A likelihood ratio test method, designed for a two-class classification task, is used to check whether an n-gram is a compound. Those n-grams that pass the test are then listed in the order of significance for the lexicographers to build these entries into the dictionary. It is found that, by incorporating parts of speech information, both the recall and precision for compound extraction is improved. The simulation result shows that the proposed approach works well. A significant cutdown of the postediting time has been observed when using this tool in an MT system, and the translation quality is greatly improved.

A Two Cluster Classification Model for Compound Extraction

The first step to extract compounds is to find the candidate list for compounds. According to our experience in machine translation, most compounds are of length 2 or 3. Hence, only bigrams and trigrams compounds are of interest to us. The corpus is first processed by a morphological analyzer to normalize every word into its stem form, instead of surface form, to reduce the number of possible alternatives. Then, the corpus is scanned from left to right with the window sizes 2 and 3. The lists of bigrams and trigrams thus acquired then form the lists of compound candidates of interest. Since the part of speech pattern for the n-grams (n=2 or 3) is used as a compound extraction feature, the text is tagged by a discrimination oriented probabilistic lexical tagger [Lin *et al.* 1992].

The n-gram candidates are associated with a number of features so that they can be judged as being compound or non-compound. In particular, we use the *mutual information* among the words in an n-gram, the *relative frequency count* of the n-gram, and the *part of speech* patterns associated with the word n-grams for the extraction task. Such features form an 'observation vector' $\vec{x}$ (to be described later) in the feature space for an input n-gram. Given the input features, we can model the compound extraction problem as a two-class classification problem, in which an n-gram is either classified as a compound or a non-compound, using a *likelihood ratio* λ for decision making:

$$\lambda = \frac{P(\vec{x}|M_c) \times P(M_c)}{P(\vec{x}|M_{nc}) \times P(M_{nc})}$$

where M_c stands for the event that 'the n-gram is produced by a compound model', M_{nc} stands for the alternative event that 'the n-gram is produced by a non-compound model', and $\vec{x}$ is the observation associated with the n-gram consisting of the joint features of mutual information, relative frequency and part of speech patterns. The test is a kind of *likelihood ratio test* commonly used in statistics [Papoulis 1990]. If $\lambda > 1$, it is more likely that the n-gram belongs to the compound cluster. Otherwise, it is assigned to the non-compound cluster. Alternatively, we could use the *logarithmic likelihood ratio* $\ln \lambda$ for testing: if $\ln \lambda > 0$, the n-gram is considered a compound; it is, otherwise, considered a non-compound.

Features for Compound Retrieval

The statistics of mutual information among the words in the n-grams, the relative frequency count for each n-gram and the transition probabilities of the parts of speech of the words are adopted as the discriminative features for classification as described in the following subsections.

Mutual Information Mutual information is a measure of word association. It compares the probability of a group of words to occur together (joint probability) to their probabilities of occurring independently. The bigram mutual information is known as [Church and Hanks 1990]:

$$I(x;y) \equiv \log_2 \frac{P(x,y)}{P(x) \times P(y)}$$

where x and y are two words in the corpus, and $I(x;y)$ is the mutual information of these two words (in this order). The mutual information of a trigram is defined as [Su *et al.* 1991]:

$$I(x;y;z) \equiv \log_2 \frac{P_D(x,y,z)}{P_I(x,y,z)}$$

where $P_D(x,y,z) \equiv P(x,y,z)$ is the probability for x, y and z to occur jointly (**D**ependently), and $P_I(x,y,z)$ is the probability for x, y and z to occur by chance (**I**ndependently), i.e., $P_I(x,y,z) \equiv P(x) \times P(y) \times P(z) + P(x) \times P(y,z) + P(x,y) \times P(z)$.

In general, $I(\cdot) >> 0$ implies that the words in the n-gram are strongly associated. Otherwise, their appearance may be simply by chance.

Relative Frequency Count The relative frequency count for the i^{th} n-gram is defined as:

$$r_i = \frac{f_i}{K}$$

where f_i is the total number of occurrences of the i^{th} n-gram in the corpus, and K is the average number of occurrence of all the entries. In other words, f_i is normalized with respect to K to get the relative frequency. Intuitively, a frequently encountered word n-gram is more likely to be a compound than a rarely used n-gram. Furthermore, it may not worth the cost of entering the compound into the dictionary if it occurs very few times. The relative frequency count is therefore used as a feature for compound extraction.

Using both the mutual information and relative frequency count as the extraction features is desirable since using either of these two features alone cannot provide enough information for compound finding. By using relative frequency count alone, it is likely to choose the n-gram with high relative frequency count but low association (mutual information) among the words comprising the n-gram. For example, if $P(x)$ and $P(y)$ are very large, it may cause a large $P(x, y)$ even though they are not related. However, $P(x, y)/P(x) \times P(y)$ would be small for this case.

On the other hand, by using mutual information alone it may be highly unreliable if $P(x)$ and $P(y)$ are too small. An n-gram may have high mutual information not because the words within it are highly correlated but due to a large estimation error. Actually, the relative frequency count and mutual information supplement each other. A group of words of both high relative frequency and mutual information is most likely to be composed of words which are highly correlated, and very commonly used. Hence, such an n-gram is a preferred compound candidate.

The distribution statistics of the training corpus, excluding those n-grams that appear only once or twice, is shown in Table 1 and 2 (MI: mutual information, RFC: relative frequency count, cc: correlation coefficient, sd: standard deviation). Note that the means of mutual information and relative frequency count of the compound cluster are, in general, larger than those in the non-compound cluster. The only exception is the means of relative frequencies for trigrams. Since almost 86.5% of the non-compound trigrams occur only once or twice, which are not considered in estimation, the average number of occurrence of such trigrams are smaller, and hence a larger relative frequency than the compound cluster, in which only about 30.6% are excluded from consideration.

	no. of tokens	mean of MI	sd of MI	mean of RFC
bigram	862	7.49	3.08	2.43
trigram	245	7.88	2.51	2.92
	sd of RFC	covariance	cc	
bigram	3.18	-0.71	-0.072	
trigram	2.18	-0.41	-0.074	

Table 1: **Distribution statistics of compounds**

	no. of tokens	mean of MI	sd of MI	mean of RFC
bigram	7190	3.11	2.54	2.28
trigram	8057	3.55	2.24	3.14
	sd of RFC	covariance	cc	
bigram	3.50	-0.45	-0.051	
trigram	2.99	-0.33	-0.049	

Table 2: **Distribution statistics of non-compounds**

Note also that mutual information and relative frequency count are almost *uncorrelated* in both clusters since the correlation coefficients are close to 0. Therefore, it is appropriate to take the mutual information measure and relative frequency count as two supplementary features for compound extraction.

Parts of Speech Part of speech is a very important feature for extracting compounds. In most cases, part of speech of compounds has the forms: [noun, noun] or [adjective, noun] (for bigrams) and [noun, noun, noun], [noun, preposition, noun] or [adjective, noun, noun] (for trigrams). Therefore, n-gram entries which violate such syntactic constraints should be filtered out even with high mutual information and relative frequency count. The precision rate of compound extraction will then be greatly improved.

Parameter Estimation and Smoothing

The parameters for the compound model M_c and non-compound model M_{nc} can be evaluated form a training corpus that is tagged with parts of speech and normalized into stem forms. The cor-

pus is divided into two parts, one as the training corpus, and the other as the testing set. The n-grams in the training corpus are further divided into two clusters. The compound cluster comprises the n-grams already in a compound dictionary, and the non-compound cluster consists of the n-grams which are not in the dictionary. However, n-grams that occur only once or twice are excluded from consideration because such n-grams rarely introduce inconsistency and the estimation of their mutual information and relative frequency are highly unreliable.

Since each n-gram may have different part of speech (POS) patterns L_i in a corpus (e.g., $L_i = [n\ n]$ for a bigram) the mutual information and relative frequency counts will be estimated for each of such POS patterns. Furthermore, a particular POS pattern for an n-gram may have several types of contextual POS's surrounding it. For example, a left context of 'adj' category and a right context of 'n' together with the above example POS pattern can form an *extended* POS pattern, such as $\bar{L}_{ij} = [adj\ (n\ n)\ n]$, for the n-gram. By considering all these features, the numerator factor for the *log-likelihood ratio* test is simplified in the following way to make parameter estimation feasible:

$$P(\vec{x}|M_c) \times P(M_c)$$
$$\approx \prod_{i=1}^{n}[P(M_{L_i}, R_{L_i}|M_c) \cdot \prod_{j=1}^{n_i} P(\bar{L}_{ij}|M_c)] \times P(M_c)$$

where n is the number of POS patterns occuring in the text for the n-gram, n_i is the number of *extended* POS patterns corresponding to the i^{th} POS pattern, L_i, $\bar{L}_{ij}$ is the j^{th} extended POS pattern for L_i, and M_{L_i} and R_{L_i} represent the means of the mutual information and relative frequency count, respectively, for n-grams with POS pattern L_i. The denominator factor for the non-compound cluster can be evaluated in the same way.

For simplicity, a subscript c ($/nc$) is used for the parameters of the compound ($/$non-compound) model, e.g., $P(\vec{x}|M_c) \triangleq P_c(\vec{x})$. Assume that M_{L_i} and R_{L_i} are of Gaussian distribution, then the bivariate probability density function $P_c(M_{L_i}, R_{L_i})$ for M_{L_i} and R_{L_i} can be evaluated from their estimated means and standard deviations [Papoulis 1990]. Further simplification on the factor $P_c(\bar{L}_{ij})$ is also possible. Take a bigram for example, and assume that the probability density function depends only on the part of speech pattern of the bigram (C_1, C_2) (in this order), one left context POS C_0 and one right lookahead POS C_3, the above formula can be decomposed as:

$$P(\bar{L}_{ij}|M_c)$$
$$= P_c(C_0, C_1, C_2, C_3)$$
$$\approx P_c(C_3|C_2) \times P_c(C_2|C_1) \times P_c(C_1|C_0) \times P_c(C_0)$$

A similar formulation for trigrams with one left context POS and one right context POS, i.e., $P_c(C_0, C_1, C_2, C_3, C_4)$, can be derived in a similar way.

The n-gram entries with frequency count ≤ 2 are excluded from consideration before estimating parameters, because they introduce little inconsistency problem and may introduce large estimation error. After the distribution statistics of the two clusters are first estimated, we calculate the means and standard deviations of the mutual information and relative frequency counts. The entries with outlier values (outside the range of 3 standard deviations of the mean) are discarded for estimating a robust set of parameters. The factors, like $P_c(C_2|C_1)$, are smoothed by adding a flattening constant 1/2 [Fienberg and Holland 1972] to the frequency counts before the probability is estimated.

Simulation Results

After all the required parameters are estimated, both for the compound and non-compound clusters, each input text is tagged with appropriate parts of speech, and the log-likelihood function $\ln \lambda$ for each word n-gram is evaluated. If it turns out that $\ln \lambda$ is greater than zero, then the n-gram is included in the compound list. The entries in the compound list are later sorted in the descending order of λ for use by the lexicographers.

The training set consists of 12,971 sentences (192,440 words), and the testing set has 3,243 sentences (49,314 words) from computer manuals. There are totally 2,517 distinct bigrams and 1,774 trigrams in the testing set, excluding n-grams which occur less than or equal to twice. The performance of the extraction approach for bigrams and trigrams is shown in Table 3 and 4. The recall and precision for the bigrams are 96.2% and 48.2%, respectively, and they become 96.6% and 39.6% for the trigrams. The high recall rates show that most compounds can be captured to the candidate list with the proposed approach. The precision rates, on the other hand, indicate that a real compound can be found approximately every 2 or 3 entries in the candidate list. The method therefore provides substantial help for updating the dictionary with little human efforts.

Note that the testing set precision of bigrams is a little higher than the training set. This situation is unusual; it is due to the deletion of the low frequency n-grams from consideration. For instance, the number of compounds in the testing set occupies only a very small portion (about 2.8%) after low frequency bigrams are deleted from consideration. The recall for the testing set is therefore higher than for the training set.

To make better trade-off between the precision rate and recall, we could adjust the threshold for $\ln \lambda$. For instance, when $\ln \lambda = -4$ is used for separating the two clusters, the recall will be raised with a lower precision. On the contrary, by raising the threshold for $\ln \lambda$ to positive numbers, the precision will be raised at the cost of a smaller recall.

	training set	testing set
recall rate (%)	97.7	96.2
precision rate (%)	44.5	48.2

Table 3: **Performance for bigrams**

	training set	testing set
recall rate (%)	97.6	96.6
precision rate (%)	40.2	39.6

Table 4: **Performance for trigrams**

Table 5 shows the first five bigrams and trigrams with the largest λ for the testing set. Among them, all five bigrams and four out of five trigrams are plausible compounds.

bigram	trigram
dialog box	Word User's guide
mail label	Microsoft Word User's
main document	Template option button
data file	new document base
File menu	File Name box

Table 5: **The first five bigrams and trigrams with the largest λ for the testing set.**

Concluding Remarks

In machine translation systems, information of the source compounds should be available before any translation process can begin. However, since compounds are very productive, new compounds are created from day to day. It is obviously impossible to build a dictionary to contain all compounds. To guarantee correct parsing and translation, new compounds must be extracted from the input text and entered into the dictionary. However, it is too costly and time-consuming for the human to inspect the entire text to find the compounds. Therefore, an automatic method to extract compounds from the input text is required.

The method proposed in this paper uses mutual information, relative frequency count and part of speech as the features for discriminating compounds and non-compounds. The compound extraction problem is formulated as a two cluster classification problem in which an n-gram is assigned to one of those two clusters using the likelihood test method. With this method, the time for updating missing compounds can be greatly reduced, and the consistency between different posteditors can be maintained automatically. The testing set performance for the bigram compounds is 96.2% recall rate and 48.2% precision rate. For trigrams, the recall and precision are 96.6% and 39.6%, respectively.

References

[Bourigault 1992] D. Bourigault, 1992. "Surface Grammar Analysis for the Extraction of Terminological Noun Phrases," In *Proceedings of COLING-92*, vol. 4, pp. 977–981, 14th International Conference on Computational Linguistics, Nantes, France, Aug. 23–28, 1992.

[Calzolari and Bindi 1990] N. Calzolari and R. Bindi, 1990. "Acquisition of Lexical Information from a Large Textual Italian Corpus," In *Proceedings of COLING-90*, vol. 3, pp. 54–59, 13th International Conference on Computational Linguistics, Helsinki, Finland, Aug. 20–25, 1990.

[Chen and Su 1988] S.-C. Chen and K.-Y. Su, 1988. "The Processing of English Compound and Complex Words in an English-Chinese Machine Translation System," In *Proceedings of ROCLING I*, Nantou, Taiwan, pp. 87–98, Oct. 21–23, 1988.

[Church and Hanks 1990] K. W. Church and P. Hanks, 1990. "Word Association Norms, Mutual Information, and Lexicography," *Computational Linguistics*, pp. 22–29, vol. 16, Mar. 1990.

[Fienberg and Holland 1972] S. E. Fienberg and P. W. Holland, 1972. "On the Choice of Flattening Constants for Estimating Multinominal Probabilities," *Journal of Multivariate Analysis*, vol. 2, pp. 127-134, 1972.

[Levi 1978] J.-N. Levi, 1978 *The Syntax and Semantics of Complex Nominals*, Academic Press, Inc., New York, NY, USA, 1978.

[Lin *et al.* 1992] Y.-C. Lin, T.-H. Chiang and K.-Y. Su, 1992. "Discrimination Oriented Probabilistic Tagging," In *Proceedings of ROCLING V*, Taipei, Taiwan, pp. 85–96, Sep. 18–20, 1992.

[Papoulis 1990] A. Papoulis, 1990. *Probability & Statistics*, Prentice Hall, Inc., Englewood Cliffs, NJ, USA, 1990.

[Su *et al.* 1991] K.-Y. Su, Y.-L. Hsu and C. Saillard, 1991. "Constructing a Phrase Structure

Grammar by Incorporating Linguistic Knowledge and Statistical Log-Likelihood Ratio," In *Proceedings of ROCLING IV*, Kenting, Taiwan, pp. 257–275, Aug. 18–20, 1991.

[Wu and Su 1993] Ming-Wen Wu and Keh-Yih Su, 1993. "Corpus-based Automatic Compound Extraction with Mutual Information and Relative Frequency Count", In *Proceedings of ROCLING VI*, Nantou, Taiwan, ROC Computational Linguistics Conference VI, pp. 207-216, Sep. 2-4, 1993.