Augmenting Lexicons Automatically: Clustering Semantically Related Adjectives

Kathleen McKeown
Vasileios Hatzivassiloglou

Department of Computer Science
450 Computer Science Building
Columbia University
New York, N.Y. 10027

ABSTRACT

Our work focuses on identifying various types of lexical data in large corpora through statistical analysis. In this paper, we present a method for grouping adjectives according to their meaning, as a step towards the automatic identification of adjectival scales. We describe how our system exploits two sources of linguistic knowledge in a corpus to compute a measure of similarity between two adjectives, using statistical techniques and a clustering algorithm for grouping. We evaluate the significance of the results produced by our system for a sample set of adjectives.

1. INTRODUCTION

A linguistic scale is a set of words, of the same grammatical category, which can be ordered by their semantic strength or degree of informativeness [1]. For example, ''lukewarm,'' ''warm'', ''hot'' fall along a single adjectival scale since they indicate a variation in the intensity of temperature of the modified noun. Linguistic properties of scales derive both from conventional logical entailment on the linear ordering of their elements and from Gricean scalar implicature [1]. Despite these properties and their potential usefulness in both understanding and generating natural language text, dictionary entries are largely incomplete for adjectives in this regard. Yet, if systems are to use the information encoded in adjectival scales for generation or interpretation (e.g. for selecting an adjective with a particular degree of semantic strength, or for handling negation), they must have access to the sets of words comprising a scale.

While linguists have presented various tests for accepting or rejecting a particular scalar relationship between any two adjectives (e.g., [2], [3]), the common problem with these methods is that they are designed to be applied by a human who incorporates the two adjectives in specific sentential frames (e.g. ''X is *warm*, even *hot*'') and assesses the semantic validity of the resulting sentences. Such tests cannot be used computationally to identify scales in a domain, since the specific sentences do not occur frequently enough in a corpus to produce an adequate description of the adjectival scales in the domain [4]. As scales vary across domains, the task of compiling such information is compounded.

In this paper we describe a technique for automatically grouping adjectives according to their meaning based on a given text corpus, so that all adjectives placed in one group describe different values of the same property. Our method is based on statistical techniques, augmented with linguistic information derived from the corpus, and is completely domain independent. It demonstrates how high-level semantic knowledge can be computed from large amounts of low-level knowledge (essentially plain text, part-of-speech rules, and optionally syntactic relations). While our current system does not distinguish between scalar and non-scalar adjectives, it is a first step in the automatic identification of adjectival scales, since the scales can be subsequently ordered and the non-scalar adjectives filtered on the basis of independent tests, done in part automatically and in part by hand in a post-editing phase. The result is a semi-automated system for the compilation of adjectival scales.

In the following sections, we first describe our algorithm in detail, present the results obtained, and finally provide a formal evaluation of the results.

2. ALGORITHM

Our algorithm is based on two sources of linguistic data: data that help establish that two adjectives are related, and data that indicate that two adjectives are unrelated. We extract adjective-noun pairs that occur in a modification relation in order to identify the distribution of nouns an adjective modifies and, ultimately, determine which adjectives it is related to. This is based on the expectation that adjectives describing the same property tend to modify the same set of nouns. For example, temperature is normally defined for physical objects and we can expect to find that adjectives conveying different values of temperature will all modify physical objects. Therefore, our algorithm finds the distribution of nouns that each adjective modifies and categorizes adjectives as similar if they have similar distributions.

Second, we use adjective-adjective pairs occurring as premodifiers within the same NP as a strong indication that the two adjectives do not belong in the same group. There are three cases:

1. If both adjectives modify the head noun and the two adjectives are antithetical, the NP

would be self-contradictory, as in the scalar sequence *hot cold* or the non-scalar *red black*.

2. For non-antithetical scalar adjectives which both modify the head noun, the NP would violate the Gricean maxim of Manner [1] since the same information is conveyed by the strongest of the two adjectives (e.g. *hot warm*).

3. Finally, if one adjective modifies the other, the modifying adjective has to qualify the modified one in a different dimension. For example, in *light blue shirt, blue* is a value of the property color, while *light* indicates the shade[*].

The use of linguistic data, in addition to statistical measures, is a unique property of our work and significantly improves the accuracy of our results. One other published model for grouping semantically related words [5], is based on a statistical model of bigrams and trigrams and produces word groups using no linguistic knowledge, but no evaluation of the results is performed.

Our method works in three stages. First, we extract linguistic data from the parsed corpus in the form of syntactically related word pairs; in the second stage, we compute a measure of similarity between any two adjectives based on the information gathered in stage one; and in the last stage, we cluster the adjectives into groups according to the similarity measure, so that adjectives with a high degree of similarity fall in the same cluster (and, consequently, adjectives with a low degree of similarity fall in different clusters).

2.1. Stage One: Extracting Word Pairs

During the first stage, the system extracts adjective-noun and adjective-adjective pairs from the corpus. To determine the syntactic category of each word, and identify the NP boundaries and the syntactic relations between each word, we used the Fidditch parser [6][**]. For each NP, we then determine its **minimal NP**, that part of an NP consisting of the head noun and its adjectival pre-modifiers. We match a set of regular expressions, consisting of syntactic categories and representing the different forms a minimal NP can take, against the NPs. From the minimal NP, we produce the different pairs of adjectives and nouns.

The resulting adjective-adjective and adjective-noun pairs are filtered by a morphology component, which removes pairs that contain erroneous information (such as mistyped words, proper names, and closed-class words which may be mistakenly classified as adjectives (e.g. possessive pronouns)). This component also reduces the number of different pairs without losing information by transforming words to an equivalent, base form (e.g. plural nouns are converted to singular) so that the expected and actual frequencies of each pair are higher. Stage one then produces as output a simple list of adjective-adjective pairs that occurred within the same minimal NP and a table with the observed frequencies of every adjective-noun combination. Each row in the table contains the frequencies of modified nouns for a given adjective.

2.2. Stage Two: Computing Similarities Between Adjectives

This stage processes the output of stage one, producing a measure of similarity for each possible pair of adjectives. The adjective-noun frequency table is processed first; for each possible pair in the table we compare the two distributions of nouns.

We use a robust non-parametric method to compute the similarity between the modified noun distributions for any two adjectives, namely Kendall's τ coefficient [7] for two random variables with paired observations. In our case, the two random variables are the two adjectives we are comparing, and each paired observation is their frequency of co-occurrence with a given noun. Kendall's τ coefficient compares the two variables by repeatedly comparing two pairs of their corresponding observations. Formally, if (X_i,Y_i) and (X_j,Y_j) are two pairs of observations for the adjectives X and Y on the nouns *i* and *j* respectively, we call these pairs **concordant** if $X_i>X_j$ and $Y_i>Y_j$ or if $X_i<X_j$ and $Y_i<Y_j$; otherwise these pairs are **discordant**[***]. If the distributions for the two adjectives are similar, we expect a large number of concordances, and a small number of discordances.

Kendall's τ is defined as

$$\tau = p_c - p_d$$

where p_c and p_d are the probabilities of observing a concordance or discordance respectively. τ ranges from -1 to +1, with +1 indicating complete concordance, -1 complete discordance, and 0 no correlation between X and Y.

An unbiased estimator of τ is the statistic

$$T = \frac{C-Q}{\binom{n}{2}}$$

where *n* is the number of paired observations in the sample and *C* and *Q* are the numbers of observed concordances and discordances respectively [8]. We compute *T* for each pair of adjectives, adjusting for possible ties in the values

[*]Note that sequences such as *blue-green* are usually hyphenated and thus better considered as a compound.

[**]We thank Diane Litman and Donald Hindle for providing us with access to the parser at AT&T Bell Labs.

[***]We discard pairs of observations where $X_i=X_j$ or $Y_i=Y_j$.

of each variable. We determine concordances and discordances by sorting the pairs of observations (noun frequencies) on one of the variables (adjectives), and computing how many of the $\binom{n}{2}$ pairs of paired observations agree or disagree with the expected order on the other adjective. We normalize the result to the range 0 to 1 using a simple linear transformation.

After the similarities have been computed for any pair of adjectives, we utilize the knowledge offered by the observed adjective-adjective pairs; we know that the adjectives which appear in any such pair cannot be part of the same group, so we set their similarity to 0, overriding the similarity produced by τ.

2.3. Stage Three: Clustering The Adjectives

In stage three we first convert the similarities to dissimilarities and then apply a non-hierarchical clustering algorithm. Such algorithms are in general stronger than hierarchical methods [9]. The number of clusters produced is an input parameter. We define dissimilarity as (1 - similarity), with the additional provision that pairs of adjectives with similarity 0 are given a higher dissimilarity value than 1. This ensures that these adjectives will never be placed in the same cluster; recall that they were determined to be definitively dissimilar based on linguistic data.

The algorithm uses the exchange method [10] since the more commonly used K-means method [9] is not applicable; the K-means method, like all centroid methods, requires the measure *d* between the clustered objects to be a distance; this means, among other conditions, that for any three objects *x*, *y*, and *z* the triangle inequality applies. However, this inequality does not necessarily hold for our dissimilarity measure. If the adjectives *x* and *y* were observed in the same minimal NP, their dissimilarity is quite large. If neither *z* and *x* nor *z* and *y* were found in the same minimal NP, then it is quite possible that the sum of their dissimilarities could be less than the dissimilarity between *x* and *y*.

The algorithm tries to produce a partition of the set of adjectives in such a way that adjectives with high dissimilarities are placed in different clusters. This is accomplished by minimizing an **objective function** Φ which scores a partition $\mathcal{P}$. The objective function we use is

$$\Phi(\mathcal{P}) = \sum_{C \in \mathcal{P}} [\frac{1}{|C|} \sum_{x,y \in C} d(x,y)]$$

The algorithm starts by producing a random partition of the adjectives, computing its Φ value and then computing for each adjective the improvement in Φ for every cluster where it can be moved; if there is at least one move for an adjective that leads to an overall improvement of Φ, then the adjective is moved to the cluster that yields the best improvement and the next adjective is considered. This procedure is repeated until no more moves lead to an improvement of Φ.

This is a hill-climbing method and therefore is guaranteed to converge, but it may lead to a local minimum of Φ, inferior to the global minimum that corresponds to the optimal solution. To alleviate this problem, the partitioning algorithm is called repeatedly with different random starting partitions and the best solution in these runs is kept. It should be noted that the problem of computing the optimal solution is NP-complete, as a generalization of the basic NP-complete clustering problem [11].

antitrust	new
big	old
economic	political
financial	potential
foreign	real
global	serious
international	severe
legal	staggering
little	technical
major	unexpected
mechanical	

Figure 1: Adjectives to be grouped.

3. RESULTS

We tested our system on a 8.2 million word corpus of stock market reports from the AP news wire[****]. A subset of 21 of the adjectives in the corpus (Figure 1) was selected for practical reasons (mainly for keeping the evaluation task tractable). We selected adjectives that have one modified noun in common (*problem*) to ensure some semantic relatedness, and we included only adjectives that occurred frequently so that our similarity measure would be meaningful.

The partition produced by the system for 9 clusters appears in Figure 2. Since the number of clusters is not determined by the system, we present the partition with a similar number of clusters as humans used for the same set of adjectives (the average number of clusters in the human-made models was 8.56).

Before presenting a formal evaluation of the results, we note that this partition contains interesting data. First, the results contain two clusters of gradable adjectives which fall in the same scale. Groups 5 and 8 contain adjectives that indicate the size, or scope, of a problem; by augmenting the system with tests to identify when an adjective is gradable, we could separate out these two groups from other potential scales, and perhaps consider combining them. Second, groups 1 and 6 clearly identify separate sets of non-gradable, non-scalar adjectives; the former group contains adjectives that describe the geographical scope of the problem, while the latter contains adjectives that

[****]We thank Karen Kukich and Frank Smadja for providing us access to the corpus.

	Answer should be Yes	Answer should be No
The system says Yes	a	b
The system says No	c	d

Table 1: Contingency table model for evaluation.

1. foreign global international
2. old
3. potential
4. new real unexpected
5. little staggering
6. economic financial mechanical political technical
7. antitrust
8. big major serious severe
9. legal

Figure 2: Partition found for 9 clusters.

specify the nature of the problem. It is interesting to note here that the expected number of adjectives per cluster is $\frac{21}{9} \approx 2.33$, and the clustering algorithm employed discourages long groups; nevertheless, the evidence for the adjectives in group 6 is strong enough to allow the creation of a group with more than twice the expected number of members. Finally, note that even in group 4 which is the weakest group produced, there is a positive semantic correlation between the adjectives *new* and *unexpected*. To summarize, the system seems to be able to identify many of the existent semantic relationships among the adjectives, while its mistakes are limited to creating singleton groups containing adjectives that are related to other adjectives in the test set (e.g., missing the semantic associations between *new-old* and *potential-real*) and ‘‘recognizing’’ a non-significant relationship between *real* and *new-unexpected* in group 4.

We produced good results with relatively little data; the accuracy of the results can be improved if a larger, homogeneous corpus is used to provide the raw data. Furthermore, some of the associations between adjectives that the system reports appear to be more stable than others, e.g. when we vary the number of clusters in the partition. We have noticed that adjectives with a higher degree of semantic content (e.g. *international* or *severe*) appear to form more stable associations than relatively semantically empty adjectives (e.g. *little* or *real*). This observation can be used to actually filter out the adjectives which are too general to be meaningfully clustered in groups.

4. EVALUATION

To evaluate the performance of our system we compared its output to a model solution for the problem designed by humans. Nine human judges were presented with the set of adjectives to be partitioned, a description of the domain, and a simple example. They were told that clusters should not overlap but they could select any number of clusters.

For our scoring mechanism, we converted the comparison of two partitions to a series of yes-no questions, each of which has a correct answer (as dictated by the model) and an answer assigned by the system. For each pair of adjectives, we asked if they fell in the same cluster (‘‘yes’’) or not (‘‘no’’). Since human judges did not always agree, we used fractional values for the correctness of each answer instead of 0 (‘‘incorrect’’) and 1 (‘‘correct’’). We used multiple human models for the same set of adjectives and defined the correctness of each answer as the relative frequency of the association between the two adjectives among the human models. We then sum these correctness values; in the case of perfect agreement between the models, or of only one model, the measures reduce to their original definition.

Then, the contingency table model [12], widely used in Information Retrieval, is applicable. Referring to the classification of the yes-no answers in Table 1, the following measures are defined :

- Recall $= \frac{a}{a+c} \cdot 100\%$
- Precision $= \frac{a}{a+b} \cdot 100\%$
- Fallout $= \frac{b}{b+d} \cdot 100\%$

In other words, recall is the percentage of correct ‘‘yes’’ answers that the system found among the model ‘‘yes’’ answers, precision is the percentage of correct ‘‘yes’’ answers among the total of ‘‘yes’’ answers that the system reported, and fallout is the percentage of incorrect ‘‘yes’’ answers relative to the total number of ‘‘no’’ answers[*****]. We also compute a combined measure for recall and precision, the F-measure [13], which always takes a value between the values of recall and precision, and is higher when recall and precision are closer; it is defined as

[*****] Another measure used in information retrieval, **overgeneration**, is in our case always equal to (100 - precision)%.

	Recall	Precision	Fallout	F-measure (β=1)
7 clusters	50.78%	43.56%	7.48%	46.89%
8 clusters	37.31%	38.10%	6.89%	37.70%
9 clusters	49.74%	46.38%	6.54%	48.00%
10 clusters	35.23%	41.98%	5.54%	38.31%

Table 2: Evaluation results.

$$F = \frac{(\beta^2+1) \times \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}}$$

where β is the weight of recall relative to precision; we use β=1.0, which corresponds to equal weighting of the two measures.

The results of applying our evaluation method to the system output (Figure 2) are shown in Table 2, which also includes the scores obtained for several other sub-optimal choices of the number of clusters. We have made these observations related to the evaluation mechanism :

1. Recall is inversely related to fallout and precision. Decreasing the number of clusters generally increases the recall and fallout and simultaneously decreases precision.

2. We have found fallout to be a better measure overall than precision, since, in addition to its decision-theoretic advantages [12], it appears to be more consistent across evaluations of partitions with different numbers of clusters. This has also been reported by other researchers in different evaluation problems [14].

3. For comparison, we evaluated each human model against all the other models, using the above evaluation method; the results ranged from 38 to 72% for recall, 1 to 12% for fallout, 38 to 81% for precision, and, covering a remarkably short range, 49 to 59% for the F-measure, indicating that the performance of the system is not far behind human performance.

Finally, before interpreting the scores produced by our evaluation module, we need to understand how they vary as the partition gets better or worse, and what are the limits of their values. Because of the multiple models used, perfect scores are not attainable. Also, because each pair of adjectives in a cluster is considered an observed association, the relationship between the number of associations produced by a cluster and the number of adjectives in the cluster is not linear (a cluster with *k* adjectives will produce $\binom{k}{2} = O(k^2)$ associations). This leads to lower values of recall, since moving a single adjective out of a cluster with *k* elements in the model will cause the system to miss *k*-1 associations. In general, defining a scoring mechanism that compares one partition to another is a hard problem.

To quantify these observations, we performed a Monte Carlo analysis [15] for the evaluation metrics, by repeatedly creating random partitions of the sample adjectives and evaluating the results. Then we estimated a (smoothed) probability density function for each metric from the resulting histograms; part of the results obtained are shown in Figure 3 for F-measure and fallout using 9 clusters. We observed that the system's performance (indicated by a square in the diagrams) was significantly better than what we would expect under the null hypothesis of random performance; the probability of getting a better partition than the system's is extremely small for all metrics (no occurrence in 20,000 trials) except for fallout, for which a random system may be better 4.9% of the time. The estimated density functions also show that the metrics are severely constrained by the structure imposed by the clustering as they tend to peak at some point and then fall rapidly.

5. CONCLUSIONS AND FUTURE WORK

We have described a system for extracting groups of semantically related adjectives from large text corpora. Our evaluation reveals that it has significantly high performance levels, comparable to human models. Its results can be filtered to produce scalar adjectives that are applicable in any given domain.

Eventually, we plan to use the system output to augment adjective entries in a lexicon and test the augmented lexicon in an application such as language generation. In addition, we have identified many directions for improving the quality of our output:

- Investigating non-linear methods for converting similarities to dissimilarities.
- Experimenting with different evaluation models, preferably ones based on the goodness of each cluster and not of each association.
- Developing methods for automatically selecting the desired number of clusters for the produced partition. Although this is a particularly hard problem, a steepest-descent method based on the tangent of the objective function may offer a solution.
- Investigating additional sources of linguistic

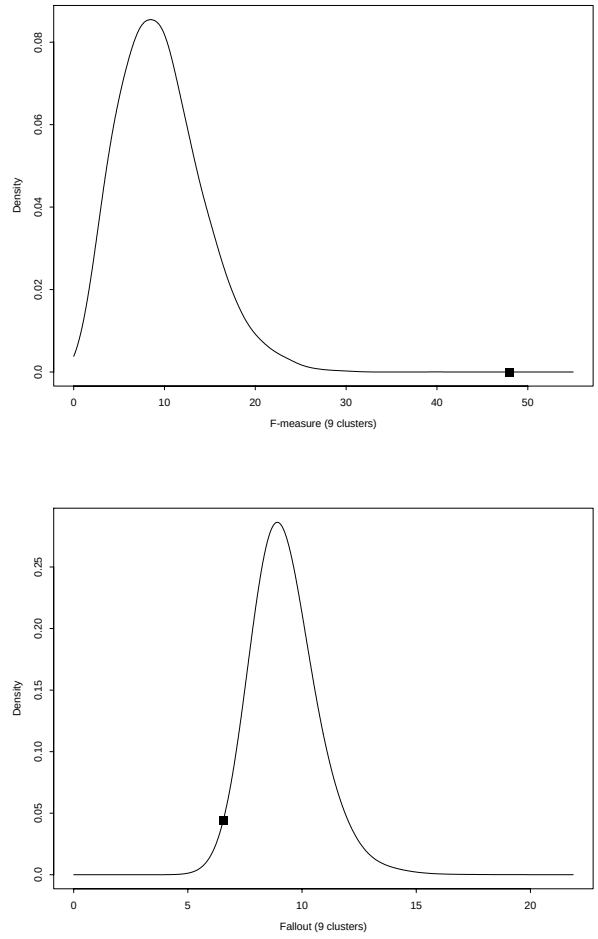

Figure 3: Estimated probability densities for F-measure and fallout with 9 clusters.

knowledge, such as the use of conjunctions and adverb-adjective pairs.

- Augmenting the system with tests particular to scalar adjectives; for example, exploiting gradability, checking whether two adjectives are antonymous (essentially developing tests in the opposite direction of the work by Justeson and Katz [16]), or comparing the relative semantic strength of two adjectives.

ACKNOWLEDGEMENTS

This work was supported jointly by DARPA and ONR under contract N00014-89-J-1782, by NSF GER-90-24069, and by New York State Center for Advanced Technology Contract NYSSTF-CAT(91)-053.

REFERENCES

1. Levinson, S.C., *Pragmatics*, Cambridge University Press, Cambridge, England, 1983.

2. Horn, L., ‘‘A Presuppositional Analysis of *Only* and *Even*’’, *Papers from the Fifth Regional Meeting*, Chicago Linguistics Society, 1969, pp. 98-107.

3. Bolinger, D., *Neutrality, Norm, and Bias*, Indiana University Linguistics Club, Bloomington, IN, 1977.

4. Smadja, F., *Retrieving Collocational Knowledge from Textual Corpora. An Application: Language Generation*, PhD dissertation, Department of Computer Science, Columbia University, 1991.

5. Brown P., Della Pietra V., deSouza P., Lai J., and Mercer R., ‘‘Class-based n-gram Models of Natural Language’’, *Computational Linguistics*, Vol. 18:4, 1992, pp. 467-479.

6. Hindle, D. M., ‘‘Acquiring Disambiguation Rules from Text’’, *Proceedings of the 27th meeting of the Association for Computational Linguistics*, Vancouver, B.C., 1989, pp. 118-125.

7. Kendall, M.G., ‘‘A New Measure of Rank Correlation’’, *Biometrika*, Vol. 30, 1938, pp. 81-93.

8. Wayne, D.W., *Applied Nonparametric Statistics (2nd edition)*, PWS-KENT Publishing Company, Boston, The Duxbury Advanced Series in Statistics and Decision Sciences, 1990.

9. Kaufman, L. and Rousseeuw, P.J., *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, New York, Wiley Series in Probability and Mathematical Statistics, 1990.

10. Spath, Helmuth, *Cluster Dissection and Analysis : Theory, FORTRAN Programs, Examples*, Ellis Horwood, Chichester, West Sussex, England, Ellis Horwood Series in Computers and their Applications, 1985.

11. Brucker, P., ‘‘On the complexity of clustering problems’’, in *Optimierung und Operations Research*, Henn, R., Korte, B., and Oletti, W., eds., Springer, Berlin, Lecture Notes in Economics and Mathematical Systems, 1978.

12. Swets, J.A., ‘‘Effectiveness of Information Retrieval Methods’’, *American Documentation*, Vol. 20, January 1969, pp. 72-89.

13. Van Rijsbergen, C.J., *Information Retrieval (2nd edition)*, Butterwoths, London, 1979.

14. Lewis, D. and Tong, R., ‘‘Text Filtering in MUC-3 and MUC-4’’, *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, DARPA Software and Intelligent Systems Technology Office, 1992, pp. 51-66.

15. Rubinstein, R.Y., *Simulation and the Monte Carlo method*, Wiley, New York, Wiley Series in Probability and Mathematical Statistics, 1981.

16. Justeson, J.S. and Katz, S.M., ‘‘Co-occurences of Antonymous Adjectives and Their Contexts’’, *Computational Linguistics*, Vol. 17:1, 1991, pp. 1-19.